

Advanced Application Architecture
Umbrella System
Umbrella Programming
Student Workbook

June 2001



Americas Financial Services Division University

This documentation contains trade secrets and confidential information which are proprietary to Computer Sciences Corporation. The use, reproduction, distribution, or disclosure of the documentation, in whole or in part, without the express written permission of Computer Sciences Corporation is prohibited. This documentation is also an unpublished work protected under the copyright laws of the United States of America and other countries. If this documentation becomes published, the following notice shall apply:

Copyright © 2001 Computer Sciences Corporation

All Rights Reserved.

All product names and trademarks are the property of their respective owners.

Preface

This manual provides the course material for this class.

The class is designed to introduce and reinforce the basis for working with the Umbrella System. It serves as an introduction to the facilities of the Process Environment Manager (PEM), the Process Dictionary, and the Application Development Support Systems of the Umbrella. The class maintains a balance of lecture, research exercises, and hands-on development lab exercises.

Who Should Read This Manual

This manual is used by students attending the Umbrella Programming Class.

What You Need To Know To Use This Manual

This manual assumes the reader meets the course prerequisites.

How This Manual Is Organized

The manual consists of numbered chapters to be presented over a ten-day instructional period.

How To Use This Manual

This manual is a comprehensive presentation of topics related to the class. It is to be used as a work book during the class and a reference document after completion of the class.

Related Publications

Umbrella System Features Guide and Reference

Umbrella System Applications Programming Guide

Umbrella System Support Systems Programming Guide

Umbrella System Online Reference Manual

Umbrella System Standards and Procedures Manual

Umbrella System User Report Descriptions Manual

Umbrella System Messages and Codes Manual

Umbrella System Batch Reference Manual



Umbrella Programming

Umbrella System Technical Support Programming Guide

Umbrella System Change Control and Release Control Systems Change Management Guide

Umbrella System Example System Manual



Contents

Course Information	1-1
Prerequisites	1-2
Description	1-2
Objectives	1-3
Agenda	1-6
Umbrella Curriculum	1-11
Class Setup Procedures	1-12
Logon Assignments	1-12
Change Control Number Assignment	1-13
Hogan Systems Overview	2-1
Topics	2-1
Objectives	2-2
Introduction to Hogan Systems	2-3
Hogan Systems	2-3
Tower of Software	2-6
Financial Support Systems (FSS)	2-7
Umbrella Application Development Support Systems	2-8
Change Control/Release Control Systems (CCS/RCS)	2-9
Process Control Data (PCD)	2-9
Function Processing System (FPS)	2-9
Scheduled Processing System (SPS)	2-9
Date Services (DTS)	2-9
Condition Code Processing (CCP)	2-10
Edit and Format System (EFS)	2-10
Document Printing System (DPS)	2-10
Data Gathering System (DGS)	2-10
Process Environment Manager (PEM)	2-11
Physical vs. Logical Environment	2-12
Process Dictionary	2-13
The Big Picture	2-15
The Umbrella	3-1
Purpose	3-1
Topics	3-2
Objectives	3-2
Umbrella Main Menu	3-3
Umbrella Menu Hierarchy	3-4
Process Dictionary Hierarchy	3-5
Process Dictionary Main Menu	3-6
Transactions	3-7
Umbrella Transactions and Commands	3-8
UMB—Transaction	3-8
UMBR—Transaction	3-8
Command, Owner, and CC#	3-8
Umbrella Direct Prompts	3-8
Umbrella Jump Facility	3-8

Umbrella Standard CDMF/PCD Processing PF Key Assignments.....	3-10
System Transactions	3-11
Menu of All Menus—MUAL.....	3-11
Set Defaults—UDFL	3-12
Set Effective Date—UDAT	3-14
Set Company—UCOM.....	3-15
Umbrella Screens Exercise	3-16
Summary	3-18
COBOL Under the Umbrella	4-1
Purpose	4-1
Topic	4-2
Objective	4-3
Standard COBOL Vs. Hogan COBOL	4-4
Example Hogan COBOL Program	4-6
Summary	4-8
Change Control System - CCS	5-1
Purpose	5-1
Topics	5-2
Objectives	5-3
Change Control System	5-4
Online Maintenance	5-5
Change Control Master.....	5-6
Change Control Detail	5-6
Change Control Inquiry/Maintenance	5-6
Change Control BASE Screen	5-7
Change Control PROBLEM Screen—PF10	5-7
Change Control RESOLUTION Screen—PF7	5-8
Change Control DETAIL Screen—PF8	5-8
Change Control DETAIL Screen	5-9
Change Control Processing Flow	5-10
Source Changes	5-10
CDMF Changes	5-10
Managing Change Control	5-10
Change Control Reports	5-11
Report Descriptions	5-11
JCL for Generating the Change Control Reports	5-16
Select, Sort, and Print	5-17
Problem Specifications—Change Control	5-18
Summary	5-20
Data Group Definition	6-1
Purpose	6-1
Topics	6-2
Objectives	6-2
Data Groups	6-3
Data Group Fields.....	6-4
Data Group Action Codes COBOL Copybook P49022D	6-6
Data Group Result Codes COBOL Copybook P49023D.....	6-7
Data Group Field Inquiry/Maintenance	6-9

Display of PEM Internal Element ID	6-10
Display of the Data Group Line Types	6-11
Display of PEM Dictionary Characteristic Lines	6-12
Converted (Pre-CDMF) Data Group Field Definition	6-13
Locating By Data Group Field Sequence Number	6-14
Defining the Data Group	6-16
Step 1—Add the Base Screen	6-16
Step 3—Code the Data Group Fields	6-17
Data Group Field Definition Line Types	6-17
Inserting the Different Line Types	6-18
Inserting COBOL Continuation Lines	6-18
Result: Inserting COBOL Continuation Lines	6-19
Inserting COBOL Data Lines	6-19
Result: Inserting COBOL Data Lines	6-20
Deleting Data Group Definition Lines	6-21
Moving Data Group Definition Lines	6-22
Step 4—DDID Data Elements	6-23
Step 5—Save the Data Group Definition	6-24
Error in Data Group Definition	6-25
Invalid Clause Found	6-26
To Copy a Data Group Definition	6-27
Data Element Definition Screen	6-28
Data Group Definition Exercise	6-29
Problem Specifications—Data Group Definition	6-29
Summary	6-31
 Link and End Activities	7-1
Purpose	7-1
Topics	7-2
Objectives	7-2
Activity Definition	7-3
Activity Definition Header	7-4
Link Activity	7-5
End Activity	7-6
End Activity Definition	7-6
Problem Specifications—Link Activity	7-7
Summary	7-8
 Sequential Database Definitions/Activities	8-1
Purpose	8-1
Topics	8-2
Objectives	8-2
Sequential Data Bases—SDB	8-3
Introduction	8-3
Sequential Data Base Definition	8-4
Sequential Database Activities	8-7
Adding SDB Definition	8-8
Building an SDB File	8-10
Issuing an SDB Activity	8-11
Checking Result Fields	8-12
Setting the DG Action in a Program	8-13
Problem Specifications—Sequential Database	8-14
Sequential Database Exercise I	8-14

Sequential Database Exercise II	8-14
Sequential Database Access Research Exercise	8-15
Summary	8-16
Hierarchical Database Definitions/Activities	9-1
Purpose	9-1
Topics	9-2
Objectives	9-2
Hierarchical Database—HDB	9-3
Data Group Hierarchy.....	9-5
DL/I Access Under PEM	9-6
Hierarchical Database Definition (HDB)	9-7
Hierarchical Database Hierarchy Definition.....	9-8
Hierarchical Database Environment Dependent Data	9-9
Hierarchical Database HIDAM DBD Definition	9-10
How PEM Uses Data Bases	9-11
Data Base Access Management	9-11
Physical Data Base Definitions.....	9-11
Hierarchical VSAM Management	9-12
IMS Data Base Management	9-12
Hierarchical Database Activities	9-13
Rule of Accessibility	9-15
Checking Result Fields.....	9-16
Hierarchical Database Access Research Exercise	9-17
Problem Specifications—HDB Activity	9-19
Summary	9-20
Program Definition	10-1
Topics	10-2
Objectives	10-2
Program Definition	10-3
Program Execution	10-6
Establishing Addressability	10-7
Data Usage	10-7
The User Transaction Control Block	10-8
Activity Execution	10-15
Problem Specification—Program Definition	10-17
Transaction Definition	11-1
Topics	11-2
Objectives	11-2
Transaction Definition	11-3
Umbrella Transaction Definition Activity List.....	11-5
To Display the Individual Activity Definition	11-6
Activity Definition Selected From Transaction Definition	11-7
Environment Dependent Considerations	11-8
Execution Under Different Environments	11-10
PEM/Operating Environment Interfaces	11-11
CICS Interface	11-12
IMS/DC Interface	11-13
IMS/DC Fast Path Interface	11-14

IMS/Batch, IMS/BMP, and IMS/NMD Fast Path Interfaces	11-15
Batch Interface	11-16
Processing Flow	11-17
The Big Picture Revisited	11-18
Problem Specifications—Batch Transaction Definition	11-19
Summary	11-20
 Writing Programs to Run Under PEM	12-1
Purpose	12-1
Topics	12-2
Objectives	12-3
Programming Considerations	12-4
Program Invocation	12-4
Data Addressability	12-5
Activities Issued From a Program	12-6
Working Storage—Static Values	12-7
Hogan's Use of Binary Values	12-8
Halfword and Fullword Fields	12-9
COBOL Coding Conventions When Running Under PEM	12-11
Sample COBOL Program	12-13
Communication Among Programs	12-14
PEM Calling Chain	12-18
Checking Result Fields	12-20
The Precompiler	12-22
Precompiler Messages	12-22
Batch Program Execution	12-25
Problem Specifications—Batch Program	12-26
Suggested Steps for Problem Solution	12-28
Batch Problem Skeleton Program	12-29
Batch Problem Execute JCL	12-63
Summary	12-64
 Date Services System—DTS	13-1
Purpose	13-1
Topics	13-2
Objectives	13-2
Date Services System (DTS)	13-3
Date Formats Supported	13-3
Date Calculations Supported	13-6
Holiday Calculations	13-7
Application Communication With Date Services	13-8
Date Service Actions	13-8
Date Services Online Test Screen	13-11
Field Description/Values	13-12
Date Service Research Exercise	13-14
Date Control Block	13-15
Date Services Actions Copybook T58007D	13-18
Date Services Results Copybook T58008D	13-22
Sample Program Using Date Services	13-25
Problem Specifications—DTS	13-26
Summary	13-27

CDMF Formats and PCD Formats	14-1
Purpose	14-1
Topics	14-2
Objectives	14-3
Process Control Data	14-4
Process Control Data Introduction	14-4
CDMF Concepts and Terminology	14-5
CDMF Components	14-5
Target Data Group	14-6
CDMF/PCD Key.....	14-7
Effective Dating Concepts	14-8
Defining a Format	14-9
Steps to Define a Format	14-9
Format Definition Screen	14-9
Format Custom Processing Parameters	14-19
Accessing the Data	14-20
Commonly Used Actions and Function Keys for CDMF Items	14-21
Defaulting Concepts	14-22
Format Level Defaulting.....	14-22
Format Defaulting Examples	14-23
Item Level Defaulting.....	14-24
Item Level Defaulting Examples	14-25
Application Access to CDMF Formats	14-26
Accessing CDMF—New Technology	14-26
CDMF Application Control Block	14-27
CDMF Action Codes—Copybook U48004D.....	14-28
CDMF Result Codes Copybook U48024D	14-30
Problem Specifications—CDMF Access	14-32
CDMF Access Diagram	14-33
How Old and New Formats Differ	14-34
Items Converted From the PCD System.....	14-34
2005 Format Definition	14-35
Converted PCD Custom Processing Parameters	14-36
47190 Format Definition	14-38
Application Access to PCD Formats	14-39
Accessing Simulated PCD—Converted.....	14-39
Data Group 1452 Copybook I57101D	14-40
PCD Action Codes Copybook I57104D.....	14-41
PCD Result Codes Copybook I57105D	14-42
Problem Specifications—PCD Access	14-45
PCD Diagram	14-46
Problem Specifications—Format Definition (Optional Exercise)	14-47
Summary	14-49
Condition Code Processing—CCP	15-1
Purpose	15-1
Topics	15-2
Objectives	15-3
Condition Code Processing Capabilities	15-4
User-Defined Exceptions	15-4
CDMF Format 1301	15-5
Invoking CCP	15-6
CCP Program Definition.....	15-8

CCP Item Key.....	15-9
CDMF Format 1301	15-11
Format Item Maintenance.....	15-11
Format Definition for CCP.....	15-12
Target Data Group	15-12
Processing Flow	15-13
Fields Set by Application Program.....	15-13
Process Flow.....	15-15
Data Flow Between the TCB and Data Group 48551	15-15
Copybook for Data Group 48551	15-16
Problem Specifications—CCP	15-17
Create CDMF 1301 Entry - Phase I	15-17
Remove Activity 14 from CDMF 1301 Entry - Phase II	15-17
Summary	15-18
 Testing, Debugging, Dumps Under Umbrella	16-1
 Purpose	16-1
Topics	16-2
Objectives	16-3
Testing	16-4
Activity Driver	16-4
Online Operation.....	16-4
Batch Operation	16-5
Testing, Debugging, and Dump Activities	16-6
Dump Activity	16-6
Exception Activity	16-7
Data Base Rollback Request	16-7
Delivered Exception Activities	16-7
Exception Activity	16-8
Exception Activity Vs Dump Activity	16-9
Dumps	16-10
Dumps: Friend or Fiend?	16-10
How to Approach the Abend	16-10
Common Error Messages	16-11
Dump Options	16-13
Option Keywords—Batch.....	16-13
PEM Formatted Dump Options	16-13
Online Dump Options.....	16-14
System Memory Access Retrieval Tool (CICS).....	16-16
System Memory Access Retrieval Tool	16-17
SMART/ABM Tutorial.....	16-17
SMART Commands and General Syntax.....	16-18
SMART PF Keys	16-19
Symbolic Addresses	16-20
SMART Dump	16-24
SMART Header Screen	16-24
SMART Storage Display Screen	16-25
Activity Breakpoint Monitor (ABM)	16-25
Using ABM	16-26
Activity Breakpoint Monitor Display	16-27
PEM Formatted Dumps	16-28
Example of a PEM Formatted Dump	16-29
PEM Formatted Dump Exercises	16-44
Summary	16-45

Mapping Support Facilities	17-1
Purpose	17-1
Topics	17-2
Objectives	17-2
Introduction	17-3
Work Area Activity Definition.....	17-3
Data Communications Activity	17-5
Deblock Service	17-6
Display Service.....	17-7
Map Definition Maintenance	17-8
Steps to Add a Map Definition	17-9
Step 1—Add Map Definition Base Screen.....	17-10
Step 2—Displaying the Design Screen.....	17-11
Step 3—Painting the Screen.....	17-12
.....	17-13
.....	17-13
Step 4—Display the Field Definition Screen.....	17-14
Step 5—Identify the Modifiable Fields	17-21
Using the Locate Command	17-21
Data Group Element Locate Screen	17-22
Using the Element ID	17-22
Using the COBOL Name	17-23
Step 6—Assign Attributes to Data Fields.....	17-24
Step 7—Save the Map Definition.....	17-25
Step 8—GEN the Map Definition	17-26
Step 9 -Assemble the Map	17-27
Batch Map ASSEMBLER	17-28
Map Editor Errors	17-30
Example of Error During Edit Session	17-30
Example of Error During Edit Session	17-31
Example of Error During Map GEN	17-32
The Map Display Feature	17-33
Viewing Options of Map Field Definition Maintenance	17-34
Problem Specifications—Map Definition	17-35
Suggested Steps for Problem Solution	17-36
Map Documentation/Assembly JCL.....	17-37
Summary	17-38
Prompt/Work Transactions	18-1
Purpose	18-2
Topics	18-3
Objectives	18-3
Example Transaction	18-4
Transaction Activity List Screen	18-5
Activity Definition 488301.....	18-6
Map Definition U88201M	18-7
Map Design Screen.....	18-8
Field Definition Screen.....	18-9
Field Definition 2 Screen	18-10
Data Group Definition 488401.....	18-11
Data Group Fields.....	18-12
UEPR Transaction Definition.....	18-13
UEPR Prompt Transaction Display	18-13

Transaction Flow Example	18-15
Online Transaction Flow Exercise Solution	18-17
Problem Specifications—Prompt/Work Transaction	18-18
Sequential Database Access - Phase I	18-19
Random Database Access - Phase II	18-21
Summary	18-22
Executing a Program Online	19-1
Purpose	19-1
Objectives	19-2
Invoking a Program Online	19-2
Problem Specifications—Execute Program Online	19-3
Link to Application Program - Phase I	19-3
Enhance Map/Random Access - Phase II	19-4
Summary	19-5
Function Processing System—FPS	20-1
Purpose	20-1
Topics	20-2
Objectives	20-2
FPS Features	20-3
Basics of FPS	20-4
FPS Environment Flow	20-4
Application Control	20-5
FPS Scenario Schematics	20-6
Information Needed by FPS	20-7
FPS Application Definition Requirements	20-7
Application Programs Running Under FPS	20-8
Application Maps Running Under FPS	20-8
FPS Maintenance Screens	20-9
Application Table Screen	20-13
Menu Processor Control Data Screen	20-14
Conversation Control Screen	20-16
Conversation Condition Code Screen	20-18
FPS and CDMF	20-19
Summary	20-20
FPS Application Requirements	21-1
Purpose	21-1
Topics	21-2
Objectives	21-2
FPS Requirements	21-3
Transaction Definitions	21-4
Application Table Entry	21-5
Control Definition for Main Menu	21-6
FPS Operator Security Control	21-7
Relationship of FPS to Application Programs	21-8
Application Requirements	21-9
Introduction	21-9
Communications Data Group 13589	21-10
Common Referenced Fields - DG13589	21-15

Common Modified Fields - DG13589	21-15
FPS Header Maps	21-16
I53589MA - Standard FPS Header	21-17
Setting of Condition Codes	21-22
The Employee System - Scenario Flow	21-25
Problem Specifications—FPS Inquiry	21-26
Map Definition - Phase I	21-27
Application Program - Phase II	21-28
Summary	21-29
FPS Basic Logic	22-1
Purpose	22-1
Topics	22-2
Objectives	22-2
Conversation Control Screen	22-3
Conversation Condition Code Screen	22-9
Methods of Communication Between Application Programs and FPS	22-14
Status Qualifiers - Program to FPS	22-14
User Environment Code - FPS to Program	22-14
Common Result - Program to Program	22-15
Branching to a Non-FPS Application	22-16
Basic Logic	22-17
Basic Logic Fields	22-17
FPS Logic Flowchart	22-19
FPS Basic Logic Flow UMBR Transaction	22-20
Problem Specifications—FPS Inquiry	22-30
FPS Control Entry - Phase III	22-30
Summary	22-32
FPS Menu Processing	23-1
Purpose	23-1
Topic	23-2
Objectives	23-2
Overview	23-3
The FPS Menu Processor	23-4
Problem Specifications—FPS Menu	23-11
EMPx System Flow	23-12
FPS Branch/Exec Logic	24-1
Purpose	24-1
Topics	24-2
Objectives	24-2
Branch/Exec (Increase Level) Logic	24-3
FPS Logic Flowchart	24-4
FPS Walkthrough--Request for a Data Group	24-9
Action -END- (Decrease Level) Logic	24-25
Walkthrough of END Logic	24-25
Combined EXEC and END Logic	24-32
Branch/Exec and END Logic	24-34
The Employee System - Scenario Flow	24-35
Level Control Exercise I	24-36

Level Control Exercise I Flow	24-37
	24-37
Level Control Exercise I Scenario Worksheet	24-38
Optional Level Control Exercise II.	24-39
Level Control Exercise II Flow	24-40
Phase I	24-41
Optional Exercise II - Phase I Scenario Worksheet	24-43
Phase II	24-44
Exercise II - Phase II Scenario Worksheet	24-46
Optional Phase III	24-47
 Sort Activities	25-1
Purpose	25-1
Topic	25-2
Objectives	25-2
Sort Activity	25-3
Sort Activity Definition - IN(PUT)	25-5
Sort Activity Definition - OUT(PUT)	25-5
Sort Activity Definition - BOTH	25-6
Problem Specifications—SORT Activity	25-7
Problem Specifications—SORT Access	25-8
SORT Activity - Exercise I.	25-8
Batch Skeleton Program - ZUPCxSY.	25-9
Batch Skeleton Program - ZUPCxSZ.	25-25
Batch Execute JCL - ZUPCXXEZ	25-55
Summary	25-56
 Scheduled Processing System - SPS	26-1
Purpose	26-1
Topics	26-2
Objectives	26-3
Scheduled Processing System	26-4
SPS Execution.	26-4
SPS Report Production Phases	26-5
SPS Menu Screen	26-7
Line Definitions (Programs)	26-11
Line Definition Commands	26-12
A—Issue a PEM Activity	26-13
P—Set Print Carriage Control	26-14
S—Execute Shared Format Definition	26-15
E—End Line Definition Processing	26-16
Format Definitions (Subroutines)	26-17
Shared Format Definition Commands	26-18
Format Definition Commands	26-19
CLEAR Instruction	26-20
ADD Instruction	26-21
SUBTRACT Instruction	26-22
MOVE Instruction	26-23
Data String and Print Line Position Notation	26-24
EDIT Instruction	26-25
Default Edit Patterns	26-25
User-Coded Edit Pattern	26-27
SPACE Instruction	26-29

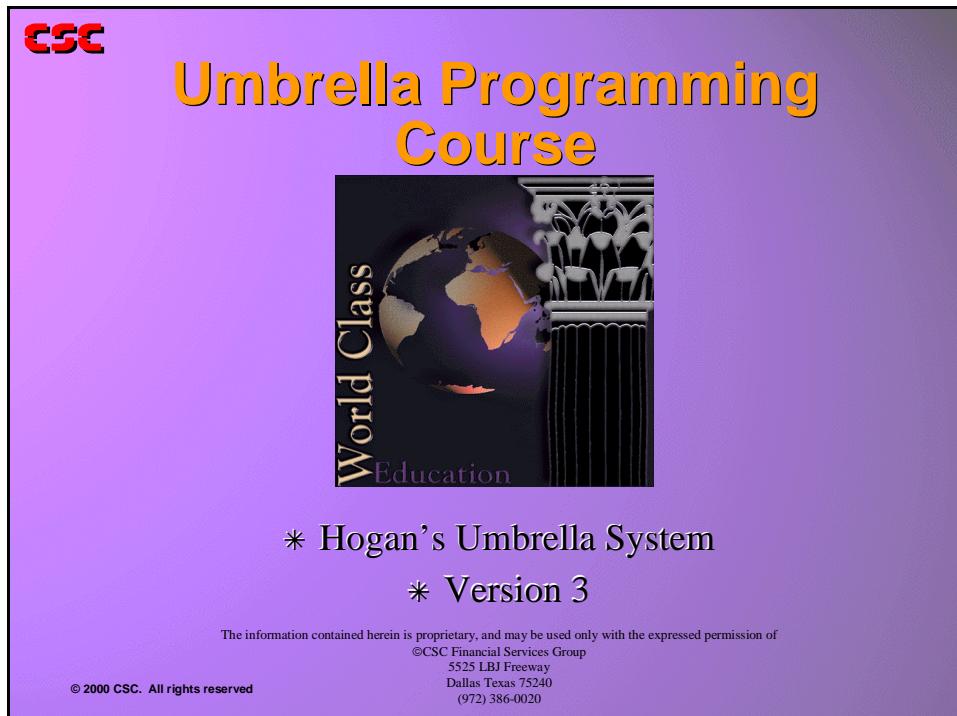
SKIP Instruction	26-29
PRINT Instruction.....	26-30
IF (Compare) Instruction	26-31
GOTO Instruction.....	26-32
END Instruction	26-33
ACTIVITY Instruction	26-34
SPS Program Walk Through	26-35
SPS Code Batch Print	26-40
Online Coding of an SPS Program.....	26-43
GENing the SPS Report Parameters.....	26-57
Functions of the SPS Editor	26-58
Copy Feature.....	26-58
Online Example of Copy	26-58
SPS Error Messages	26-63
SPS Dumps	26-64
Step 1—PEM Exceptions Report	26-64
Step 2—System/User Code	26-65
Step 3—SPS Flower Box	26-66
Step 4—Data Group Storage	26-67
SPS CDMF Structure	26-69
SPS Line Definition—PCD 2300	26-69
SPS Format Definition—PCD 2302.....	26-71
Using SPS PCD 2300 and PCD 2302.....	26-72
Common SPS Errors	26-73
Summary	26-75
 SPS Implosion	27-1
Purpose	27-1
Topics	27-2
Objectives	27-3
Overview of the Implosion Phase	27-4
Link Activity to Trigger an Implosion	27-7
Identifying a Reportable Incident	27-8
PCD 1618	27-9
PCD 15006	27-10
Illustration of the Relationships Among Data Groups in SPS	27-11
Structure of Data Group 3601	27-13
SPS Implosion Considerations	27-15
Example Implosion Program	27-17
Problem Specifications—SPS Implosion	27-20
Testing of Implosion.....	27-21
Reportable Incident.....	27-21
Summary	27-22
 SPS Reports Production	28-1
Topics	28-2
Objectives	28-3
SPS Processing Control Program	28-4
Sort Phase	28-5
Default Sort Activity.....	28-5
SPS Sort Considerations.....	28-6
The Processing Control Entry	28-8
SPS Processing Control Screen	28-9

Example Processing Control Entry	28-11
SPSCTL Beg of Rept Section	28-12
SPSCTL Processing Section	28-14
Explosion Phase	28-15
Example Explosion Program	28-16
Output Phase	28-18
Example Output(Detail) Program	28-19
SPSCTL End of Rept Section	28-21
Problem Specification—SPS Explosion and Output	28-22
Execute JCL for SPS Exercise	28-23
Worksheet for Planning SPS Reports	28-25
Worksheet for Processing Control Entry	28-27
Summary	28-28
SPS Heading Processing	29-1
Topics	29-2
Objectives	29-3
Heading Processing	29-4
Page Overflow	29-4
D Line Definition Command	29-5
Page Overflow Line Definitions	29-6
Example Heading Program	29-7
Generating the Heading	29-8
Special Values of the D Command	29-8
DEC Format Definition Command	29-9
Printing a Specific Number of Lines	29-10
Problem Specifications—SPS Head	29-11
SPS Break Processing	30-1
Topics	30-2
Objectives	30-2
SPS Break Processing	30-4
BREAK Format Definition Command	30-6
Examples of Three Level Breaks	30-7
Example 1	30-7
Example 2	30-7
Example 3	30-8
Break Logic	30-9
Example Break Program	30-10
Printing the Last Set of Totals and Grand Totals	30-12
Problem Specifications—SPS Break	30-13
Summary	30-14
Appendix	A-i
Umbrella System (UMB) Component Identification	A-i
CCP—Condition Code Processing System	A-i
CCS—Change Control System	A-ii
DTS—Date Services System	A-iii
EFS—Edit and Format System	A-iv
FPS—Function Processing System	A-v
PCD—Process Control Data	A-vi

PEM—Processing Environment Manager	A-vii
RCS—Release Control System	A-viii
SPS—Scheduled Processing System	A-ix
UMB—Umbrella	A-x
Budget and Planning System (BPS) Component Identification	A-xi
BPS—Budget and Planning System (BPS)	A-xi
Consumer and Merchant Servicing System (CAMS) Component Identification	A-xii
TCS—Consumer and Merchant Servicing System	A-xii
Credit Risk System (CRS) Component Identification	A-xiii
RAS—Credit Risk System (CRS)	A-xiii
Earnings Analysis System (EAS) Component Identification	A-xiv
EAS—Earnings Analysis System	A-xiv
Financial Information System (FIS) Component Identification	A-xv
FIS—Financial Information System	A-xv
Financial Support System (FSS) Component Identification	A-xvi
CDS—Cycle Data System	A-xvi
CWS—Check Writing System	A-xvii
G/L—General Ledger Interface System	A-xviii
NAF—Name and Address System	A-xix
OTP—Outstanding Transaction Processing and NSF	A-xx
RST—Restraints System	A-xxi
SIM—System Integration Management System	A-xxii
Hogan Education Center Component Identification	A-xxiii
HEC—Hogan Education Center	A-xxiii
Deposits System Component Identification	A-xxiv
DDA—Demand Deposit Application	A-xxv
DEP—Dynamic Environment Processing System	A-xxvi
IDS—Deposits System	A-xxvii
LOC—Line of Credit System	A-xxviii
PPS—Priority Post System	A-xxix
TDA—Time Deposit Application	A-xxx
Loans System Component Identification	A-xxxi
ILP—Loans System	A-xxxi
Management Support System (MSS) Component Identification	A-xxxiii
MSS—Management Support System (MSS)	A-xxxiii
Online Collections System (OCS) Component Identification	A-xxxiv
OCS—Online Collections System	A-xxxiv
Online Delivery System (ODS) Component Identification	A-xxxv
ODS—Online Delivery System	A-xxxv
Platform Automation Support System (PAS) Component Identification	A-xxxvi
PAS—Platform Automation Support System	A-xxxvi
Preferred Client Services System (PCS) Component Identification	A-xxxvii
PCS—Preferred Client Services System	A-xxxvii
Relationships and Profitability System (RPS) Component Identification	A-xxxviii
CIS—Customer Information System	A-xxxviii
RPM—Relationships and Profitability Manager	A-xxxix
SPS Appendix	B-i
Introduction	B-i
Implosion Mechanics Program Characteristics	B-ii
Program 1850 Logic Chain and Key Build	B-vi
New Program Planning	B-viii

Course Information

1



Notes:



Prerequisites

CSC

Prerequisites

- One year IBM Information technology experience
- Working knowledge of COBOL

© 2000 CSC. All rights reserved

Description

CSC

Class Description



- Introduction and reinforcement of the basic concepts required for working with the Umbrella System
- Emphasis on research exercises and development of elementary, hands-on skill

© 2000 CSC. All rights reserved



Objectives

CSC

Objectives



- Describe the *Big Picture* under Umbrella System
- Locate various individual items on the Umbrella System by navigating through the online menus
- Diagram the general use of PEM and explain the use of Process Dictionary
- Explain how change to Hogan products is managed through the CCS.

© 2000 CSC. All rights reserved

CSC

Objectives



- Identify Process Dictionary components, their use, and the major fields in their definitions:
 - Data Groups
 - Data Bases (sequential and hierarchical)
 - Programs
 - Transactions
 - Maps
 - Formats (and PCDs)
 - Activities

© 2000 CSC. All rights reserved



Umbrella Programming

Objectives

CSC

Objectives



Become familiar with:

- Process Dictionary entries required to access a data base
- Types of data groups used in defining a data base
- Distinction between standard COBOL programs and Hogan- oriented COBOL programs.
- Establishment of addressability to data groups
- Authorized and executed activities
- Umbrella *Fullword* and *Halfword* field technology and conventions
- Umbrella's processing flow

© 2000 CSC. All rights reserved

CSC

Objectives



- Create, maintain, and access tables
- Understand CDMF files and their functions
- Process Dictionary items online cross-referencing
- Use cobol program modification, execution, testing, debugging
- Use Date Services calculations

© 2000 CSC. All rights reserved





Objectives



Become familiar with:

- Batch Data Communications and Batch Activity Driver
- Function Processing System transactions, and Non-FPS transactions differences
- Processing flow through an FPS transaction
- Basic instructions of a Scheduled Processing System (SPS) program
- Design objectives and processing phases of SPS

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

Agenda

Agenda

CSC

Day One Agenda



- Hogan Systems Overview
- The Umbrella Basic
- Cobol “Under the Umbrella”
- Controlling Change using CCS -
The Change Control System

© 2000 CSC. All rights reserved

CSC

Day Two Agenda



- Data Group Definitions
- Link and End Activities
- Sequential Database Definitions and Activities
- Hierarchical Database Definitions and Activities

© 2000 CSC. All rights reserved





Day Three Agenda



- Program Definitions
- Transaction Definitions
- Writing Programs Under Pem
- Date Services - DTS

© 2000 CSC. All rights reserved



Day Four Agenda



- Date Services - DTS
- CDMF and PCD Formats
- Condition Code Processing

© 2001 CSC. All rights reserved



Umbrella Programming

Agenda



Day Five Agenda

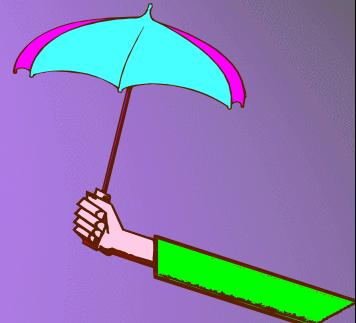


- Testing, Debugging, Dumps
- Map Support Facilities
- Prompt/Work Transactions
- Executing a Program Online

© 2001 CSC. All rights reserved



Day Six Agenda



- Prompt/Work Transactions
- Executing a Program Online
- Function Processing System
- FPS Application Requirements

© 2001 CSC. All rights reserved





Day Seven Agenda



- FPS Basic Logic
- FPS Menu Processing
- FPS Branch/Exec Logic
- Sort Activities

© 2000 CSC. All rights reserved



Day Eight Agenda



- Scheduled Processing System
- SPS Implosion

© 2000 CSC. All rights reserved



Umbrella Programming

Agenda

CSC

Day Nine Agenda



- SPS Reports Production

© 2000 CSC. All rights reserved

CSC

Day Ten Agenda

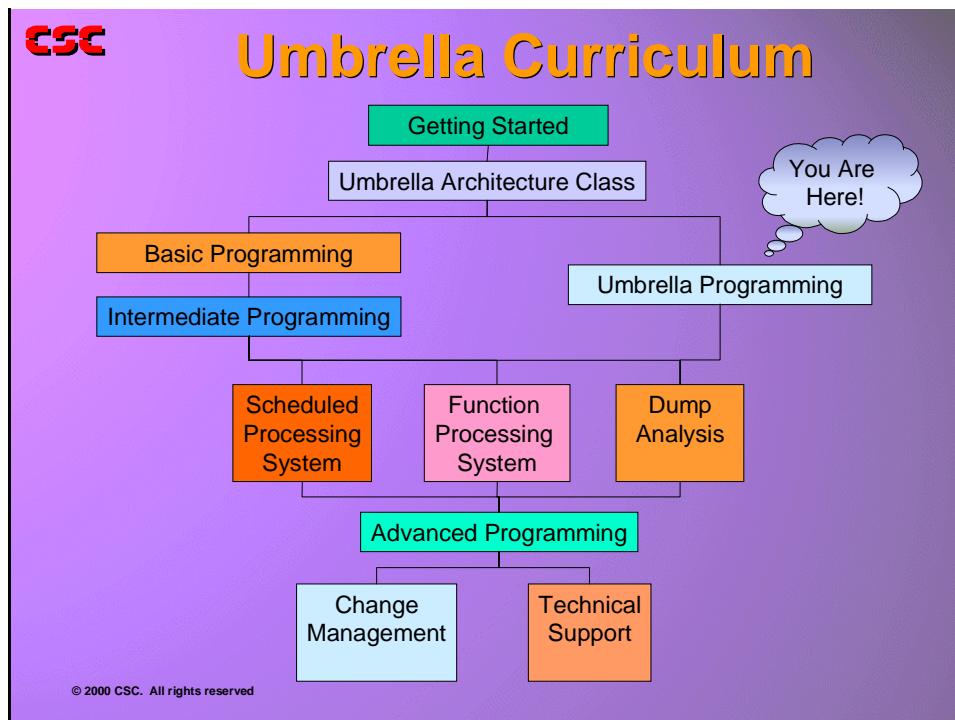


- SPS Heading Processing
- SPS Break Processing

© 2000 CSC. All rights reserved



Umbrella Curriculum



Notes:



Class Setup Procedures



Class Setup Procedures



- Logon assignments
- Group number
- Classroom
- Environment
- User ID/Password
- Change control
number assignment
 - Key:
NNNNxx?TRNG
(Where xx = group
number)

© 2000 CSC. All rights reserved

Logon Assignments

This class creates and modifies CDMF Process and Non-Process Dictionary components based on GROUP NUMBER. Its assignment and use is critical to successful exercise solutions. At HEC, the GROUP NUMBER also is used to construct the student USERID.

GROUP NUMBER: _____

At HEC, the CLASSROOM ID is used to construct the ownership code required for all new CDMF components, and the student USERID.

CLASSROOM ID: _____
ENVIRONMENT: _____

At HEC, the USERID is constructed of the CLASSROOM ID concatenated with the GROUP NUMBER.

USERID: _____



Umbrella Programming

Class Setup Procedures

PASSWORD: _____

When adding new Process and Non-Process Dictionary components, please use effective date 780101 and company DFLT(65535).

Change Control Number Assignment

This class creates and modifies CDMF Process and Non-Process Dictionary components, therefore a valid Change Control Number must be created and/or assigned. At HEC, the CHANGE CONTROL NUMBER is constructed with the GROUP NUMBER as the last two digits; the SITE is TRNG; and the APPLICATION is a constant of ZZ followed by the CLASSROOM ID.

CHANGE CONTROL NUMBER: _____

APPLICATION: _____

In a later chapter, you will create your group's Change Control Number, for use in this class. All new and existing CDMF components assigned to your group should be updated using this number.

Notes:



Umbrella Programming

Class Setup Procedures



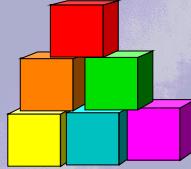
Hogan Systems Overview

2

Topics



Topics



- Hogan History
- Tower of Software
- Financial Support System
- Umbrella System
- Umbrella Subsystems
- Process Environment Manager (PEM)
- Process Dictionary
- The BIG PICTURE under Hogan

© 2000 CSC. All rights reserved

Notes:



Objectives

CSC

Objectives

Become familiar with:

- Hogan history
- Umbrella System and Financial Support System relationship
- Umbrella subsystems and their data processing functions
- Functions performed by PEM and the Process Dictionary

© 2000 CSC. All rights reserved



Notes:



Introduction to Hogan Systems

Hogan Systems



The image is a promotional graphic for CSC Financial Services Group Hogan Systems. It features the CSC logo in red and white at the top left. The main title "CSC Financial Services Group Hogan Systems" is displayed in large, bold, yellow serif font. To the right of the text is a 3D rendering of several colorful buildings in shades of red, green, orange, purple, and blue, set against a light purple background with a subtle cloud pattern. On the left side, there is a list of four bullet points in black text:

- ★ International Company
- ★ Established in 1977
- ★ Integrated Software Products
- Support Services

At the bottom left, there is a small copyright notice: "© 2000 CSC. All rights reserved".

Hogan Systems, is an international company that provides integrated software products and support services to financial institutions worldwide, including money center, regional and community banks, building societies, credit unions and other financial organizations. Hogan Systems, was established in 1977 and is headquartered in Dallas, Texas. Hogan has offices in Camberley, England (London); Melbourne, Australia; Frankfurt, Germany and has several distributor relationships in other countries.

Hogan Systems merged with Continuum in 1996 to form a company prepared to offer clients worldwide, solutions in the insurance, healthcare, and financial applications. Later the same year, Continuum/Hogan merged with Computer Sciences Corporation (CSC), to form an even stronger company to not only provide software, but also a larger support resource base. Even though Hogan has changed its company name to CSC, the Hogan application products will continue to carry the Hogan name.

The technical foundation of Hogan applications is the Umbrella System. It is the Umbrella System architecture that provides the flexibility and high performance of the Hogan applications and allows for the exploiting of new technologies. The architecture provides for the sharing of information across applications, thus increasing productivity. The Umbrella links and controls common data processing



Umbrella Programming

Introduction to Hogan Systems

activities and communications between each of the Hogan modules using common programs, techniques and procedures.



The slide features the CSC logo in red at the top left. The main title is "CSC Financial Services Locations" in large, bold, yellow font. Below the title is a bullet point: "Headquartered In Dallas, Texas." followed by three sub-points: "Camberley, England", "Melbourne, Australia", and "Frankfurt, Germany". To the right of the text are three small illustrations: a man in a cowboy hat and vest, a person in a top hat and coat, and a person in a backpack and gear. The background is a light purple gradient with white clouds.

CSC

CSC Financial Services Locations

■ **Headquartered In Dallas, Texas.**

- Camberley, England
- Melbourne, Australia
- Frankfurt, Germany

© 2000 CSC. All rights reserved

The unique flexibility of Hogan applications allows for creation of new product offerings without enlisting data processing or programming assistance. Users can establish the business parameters as the market needs dictate.

The modular design of Hogan applications allows institutions to implement the information systems that will provide the greatest return in a timely and cost effective manner. Additional applications can be added as they are needed to meet profitability and performance objectives.

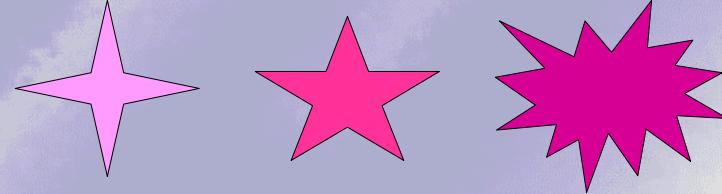
In addition to the advanced products, Hogan offers a full complement of implementation services and software support that includes onsite implementation consulting, management and planning as well as functional and technical project support. A complete education and training curriculum is offered with classes conducted at our training center in Dallas or at our customer locations. Product upgrades and enhancements are made available periodically to customers under a software maintenance agreement. Telephone support is provided 24-hours a day for problem resolution as is an electronic support system.





Company Changes

- Merged with Continuum in 1996
- Merged with Computer Sciences Corporation in 1996
- Company now known as "CSC/FSG"



© 2000 CSC. All rights reserved

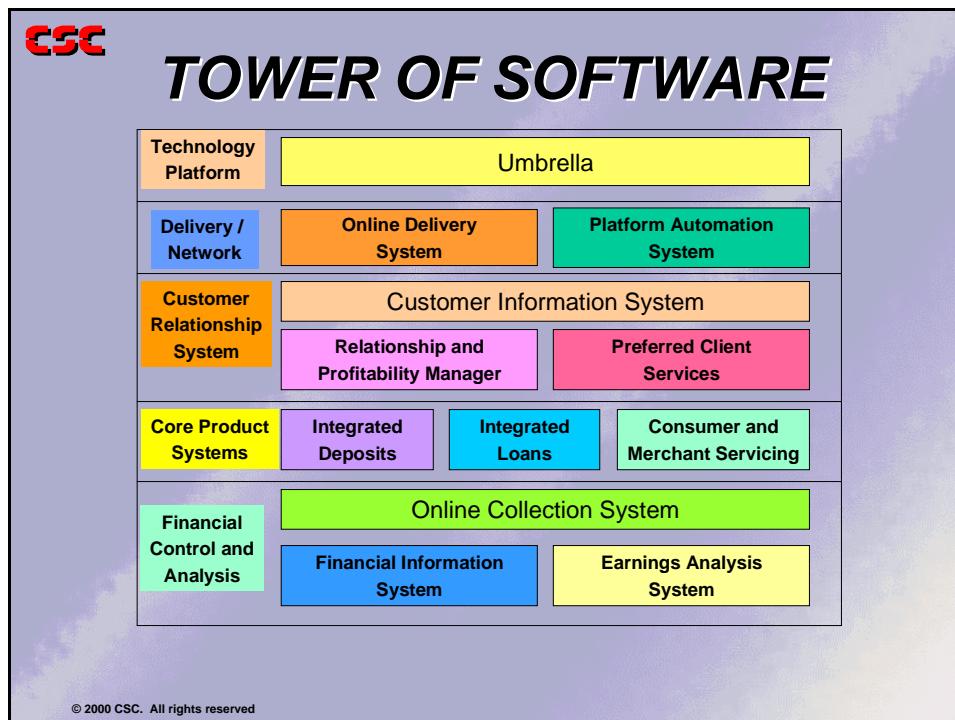
Notes:



Umbrella Programming

Tower of Software

Tower of Software

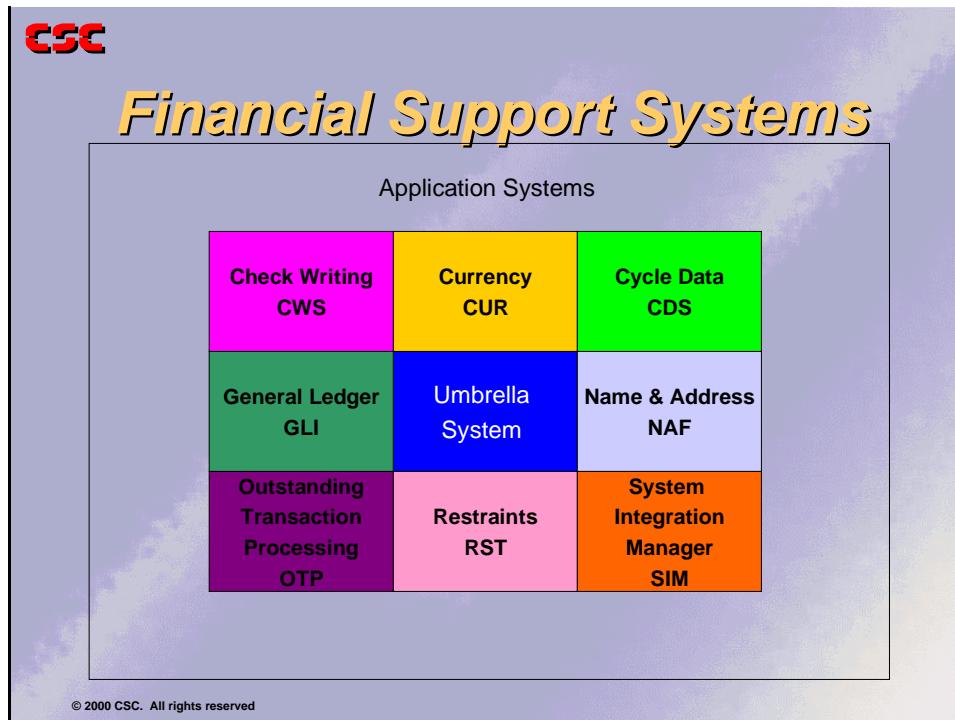


Notes:



Financial Support Systems (FSS)

The purpose of the Financial Support Systems is to provide a common set of modules that can be used by many different applications and support systems. This concept results in one generalized routine rather than having separate ones in each application. The systems within the Financial Support System are delivered with all of the Hogan application systems. A Hogan application system may use one or more systems within this group. Some of the systems were formerly called the software pool.



Notes:

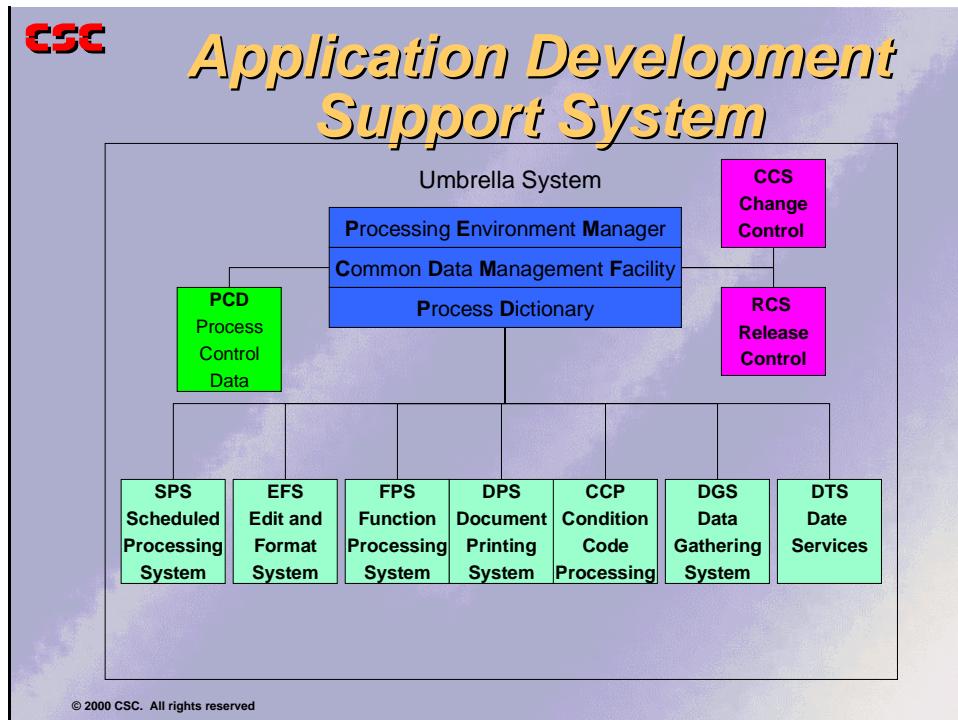


Umbrella Programming

Tower of Software

Umbrella Application Development Support Systems

There are several Umbrella Application Development Support Systems that provide standardized solutions to many common data processing problems. The support systems consist of programs, Process Dictionary entries, and online parameters. Some of the modules are invoked at the discretion of programmers. Others are activated by PEM. These modules standardize system design, improve productivity, promote consistency, and changes may take effect immediately or in the future.



Notes:



The following systems are included in the Applications Development Support Systems:

Change Control/Release Control Systems (CCS/RCS)

Changes are collected by change control number. The change control number must be authorized. It must exist on the change control master file, be open, and belong to the same family as the item being changed.

When changes are made to application systems, the Release Control System (RCS) in conjunction with the Change Control System (CCS) captures the changes. RCS saves a copy of the data as it was before the change and a copy of the data after the change. CCS records the fact that a change to the entity occurred and collects information about the change.

Process Control Data (PCD)

Process Control Data are process-directing variables stored on a data base external to the application. PCDs are tables of information maintained online. Their use builds flexibility into an application system. They save time and effort in design, programming, and testing for maintenance and enhancement. Many changes that would require program modifications under standard coding procedures are done simply by changing online parameters on a PCD.

Function Processing System (FPS)

FPS is a control system for implementation of online systems. It separates display programs from data-handling programs. It enables programmers to concentrate on the online aspects. FPS extends security through operator signons and passwords. It is built around processing levels that chain predefined user functions and, thus, sequences of screens. Up to 10 conversations can be carried on simultaneously with automatic resumption at a point of interruption. The Umbrella development screens are controlled by FPS.

Scheduled Processing System (SPS)

SPS is a report generator with math and logic capabilities. It captures data both online and in batch and constructs data strings. These strings are used to produce reports. Its language is maintained online. Some of the functions performed in SPS are building sort keys, formatting report lines, doing mathematical calculations, defining report sequences, totalling accumulators, and writing summary reports.

Date Services (DTS)

DTS provides for date calculations, manipulation of date formats, and business and holiday processing.



Umbrella Programming

Tower of Software

Condition Code Processing (CCP)

CCP provides a standard approach for identifying and handling the processing exceptions that occur within application systems. Through online parameters, logic flow for the exceptions can be controlled. Messages with explanations are maintained online for easy access by an application.

Edit and Format System (EFS)

EFS is a set of generalized, table-driven routines used for data manipulation. It is primarily used to validate data content and format. It is also capable of translating data, such as, internal to external format and code to description.

Document Printing System (DPS)

DPS provides for the formatting of online documents or reports for applications that run under the Umbrella. It is typically used when the same information needs to be printed on different forms in various locations or when variable data needs to be inserted in standard forms.

Data Gathering System (DGS)

DGS is used for matching and merging batch files. Up to 20 files and six fields can be handled. It allows for match, high, low, end-of-entity, and end-of-file conditions.

Notes:



Process Environment Manager (PEM)

PEM provides the interface or link between the user's logical application program and the physical environment.

PEM deals with:

- The teleprocessing monitor—CICS or IMS/DC
- The physical data structure—VSAM, DL/I, IMS, DB2
- The operating system—OS/MVS, XA, and ESA.

What this means to the manager, designer, or developer concerned with an application is that more time is spent solving the business problem and less time is spent dealing with the environment. Changes in technology are transparent to the application program.

All Hogan software executes under the control of PEM. PEM consists of 70 to 80 ALC CSECTs GENed for each processing environment. Its primary functions are to:

- Interpret the logical environment to the real environment;
- Interact with the operating system, data base management system, and teleprocessing monitor on behalf of the application programs;
- Manage resources for tuning transaction and data base frequencies;
- Obtain information for processing from the parameters stored in the Process Dictionary.

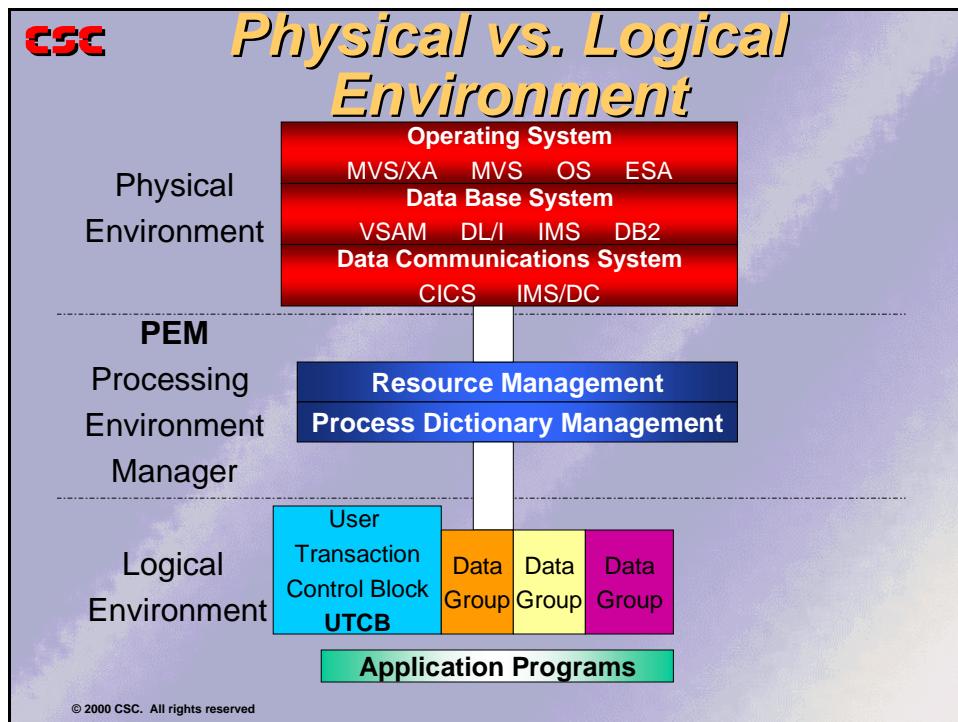
Notes:



Umbrella Programming

Tower of Software

Physical vs. Logical Environment



Notes:



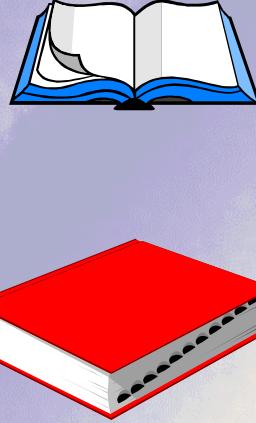
Process Dictionary

CSC

What is the Process Dictionary?

■ A set of interrelated databases containing

- Descriptions of the Logical and Physical Environments
- Application Processing Control Data



© 2000 CSC. All rights reserved

CSC

Process Dictionary?

Process Dictionary Management

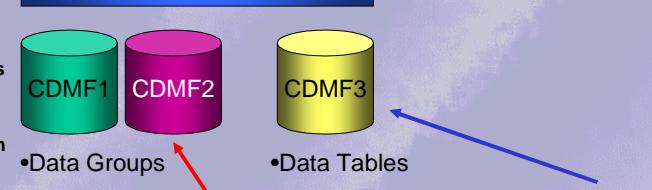
What is it?
A set of interrelated databases containing descriptions of the Logical and Physical environments, and Application Processing Control Data.

CDMF1 **CDMF2** **CDMF3**

•Data Groups
•Activities
•Data Bases
•Programs
•Transactions
•Maps
•Formats (tables)
•+

Non-process Dictionary

Process Dictionary



© 2000 CSC. All rights reserved



Umbrella Programming

Tower of Software

The Process Dictionary brings to the data processing industry a practical means of improving productivity. Program maintenance consumes 20% to over 80% of data processing resources. In addition to the cost factors, no one wants to do only maintenance. Maintenance limits time for new development that can satisfy user needs and improve processing.

The Process Dictionary can significantly reduce program maintenance. It allows the Umbrella user to store data and logic variables external to the application program. Beyond providing this storage medium, it includes an online and menu-driven facility for update and retrieval. Changes are made easily and dynamically.

Effective Dating is supported. It allows the user to define changes that are to go into effect in the future. Changes made using current date as effective date take effect immediately.

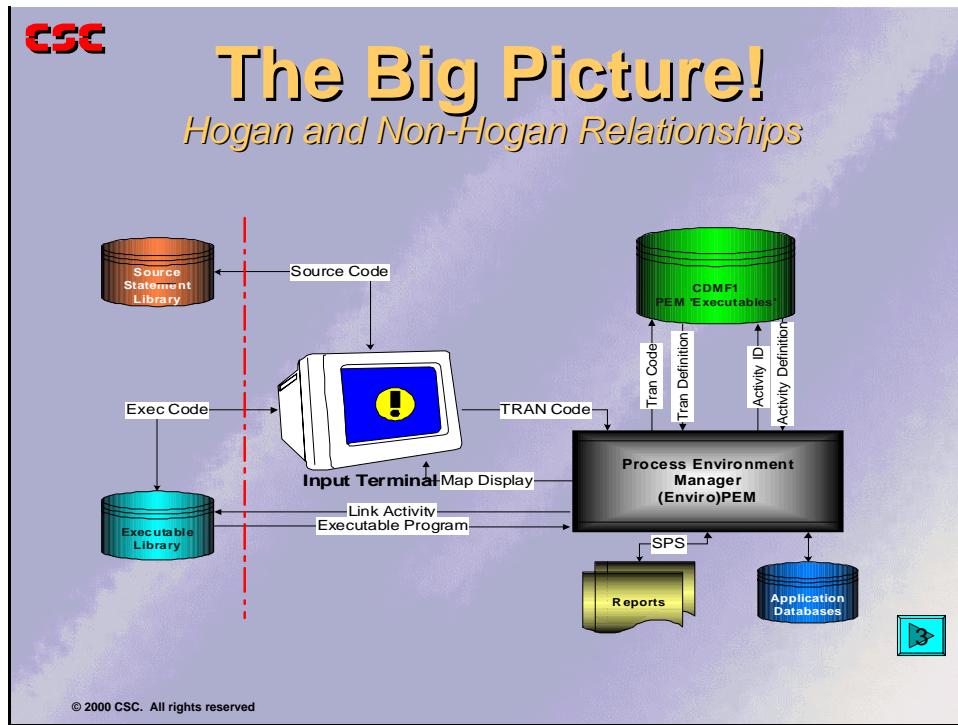
PEM obtains instructions for translating the logical environment to the real environment from the Process Dictionary. Each entry has a symbolic key that uniquely identifies it from other entries of the same type.

The types of definitions found in the Process Dictionary include:

DATA GROUP	A collection of one or more logically related data fields that are defined in a contiguous storage area.
ACTIVITY	A unit of work to be performed and the parameters for that work.
DATA BASE	A description of each logical data base and its structure.
PROGRAM	A description of the application program, the data it uses, and the activities it issues.
TRANSACTION	A list of work requests or activities to be done. A transaction equates somewhat to a job step or an online transaction.
MAP	A screen definition from which BMS or MFS source code is generated.
FORMAT	A description of variables stored on CDMF and accessed by programs to retrieve data for calculations, processing rules, verification, and description.



The Big Picture



Notes:



Umbrella Programming

The Big Picture



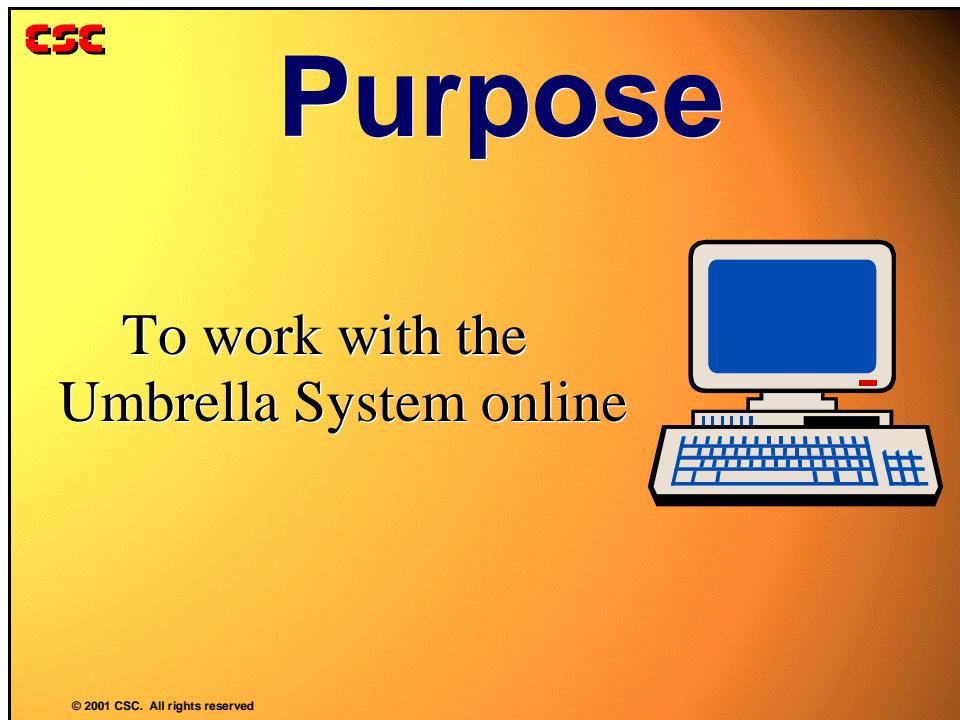
2-16

Hogan Systems Overview

The Umbrella

3

Purpose



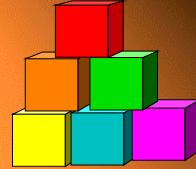
Notes:



Topics

CSC

Topics



- Menus
- Direct prompts
- Secondary search accesses
- Commands
- PF key assignments

© 2000 CSC. All rights reserved

Objectives

CSC

Objectives



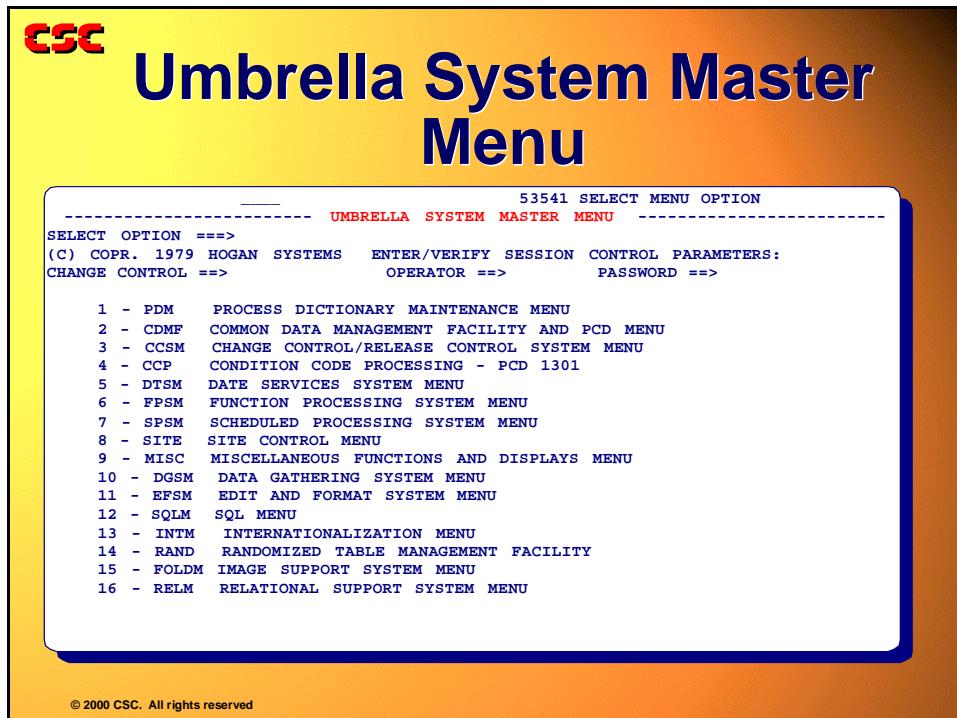
- To locate an individual item on Umbrella
- To use direct prompts
- To understand the term *natural key*
- To list commands and PF keys

© 2000 CSC. All rights reserved



Umbrella Main Menu

To access the Umbrella System enter a U from a cleared screen. This transaction code causes the "Umbrella System Master Menu" to be displayed.



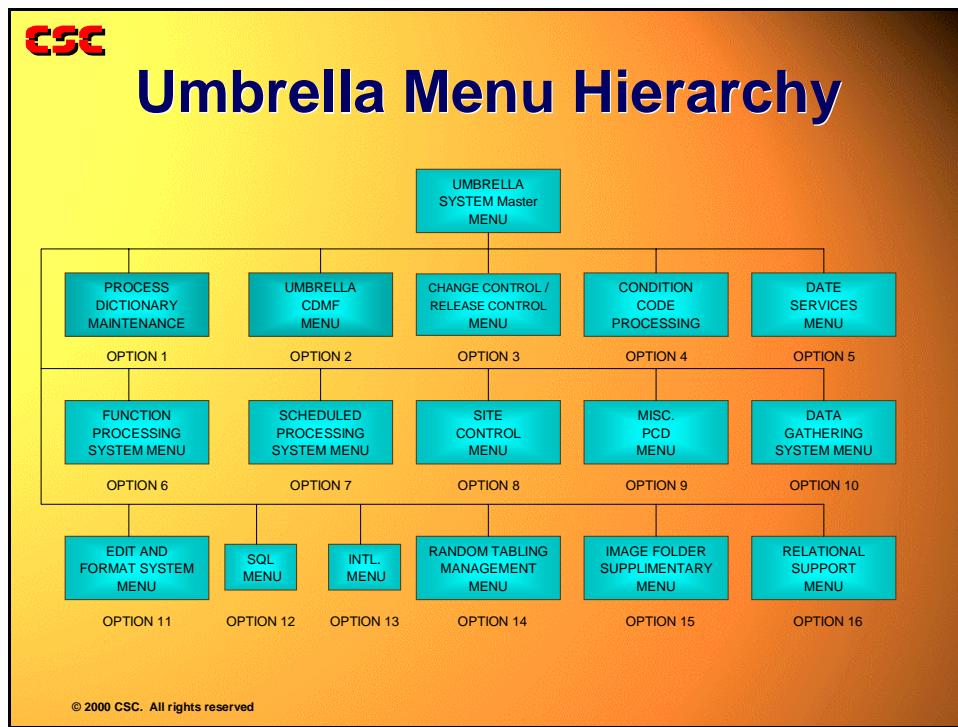
Notes:



Umbrella Programming

Umbrella Menu Hierarchy

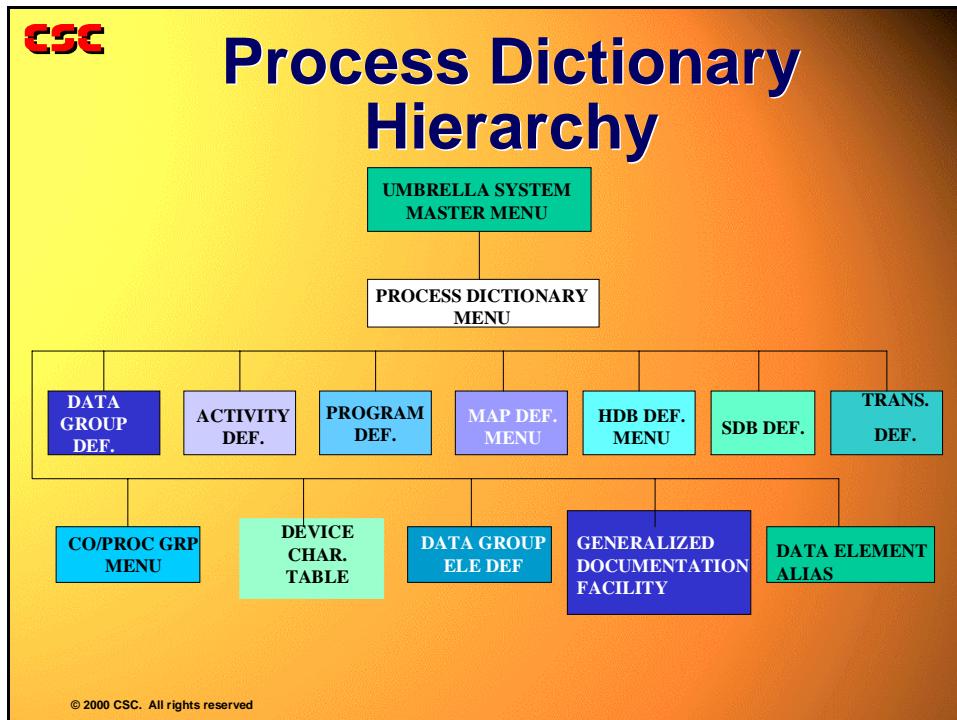
Umbrella Menu Hierarchy



Notes:



Process Dictionary Hierarchy



Notes:

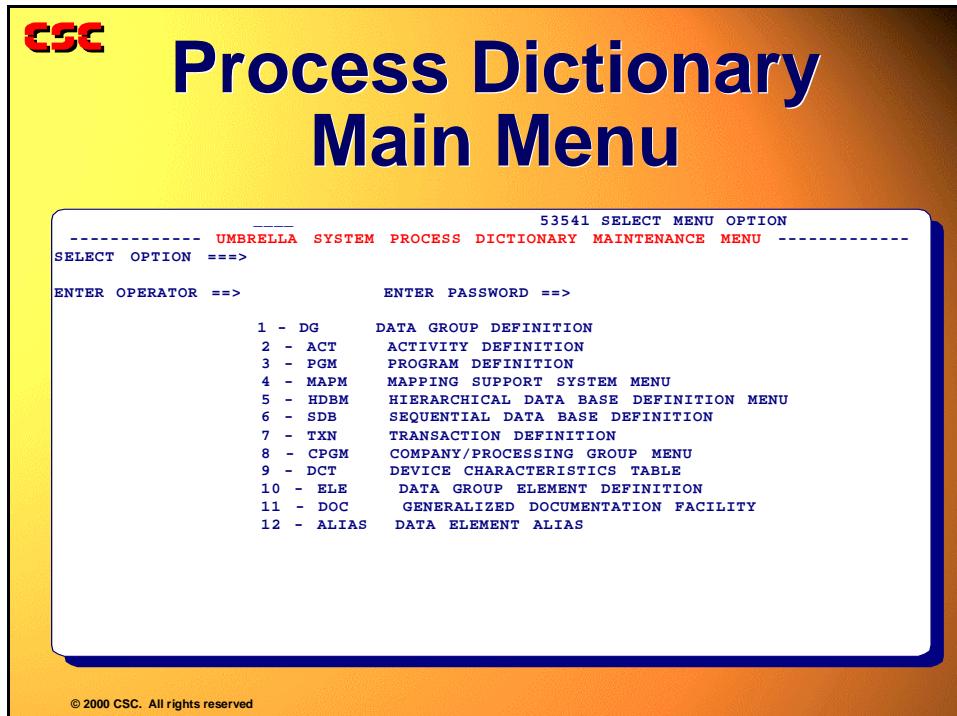


Umbrella Programming

Umbrella Menu Hierarchy

Process Dictionary Main Menu

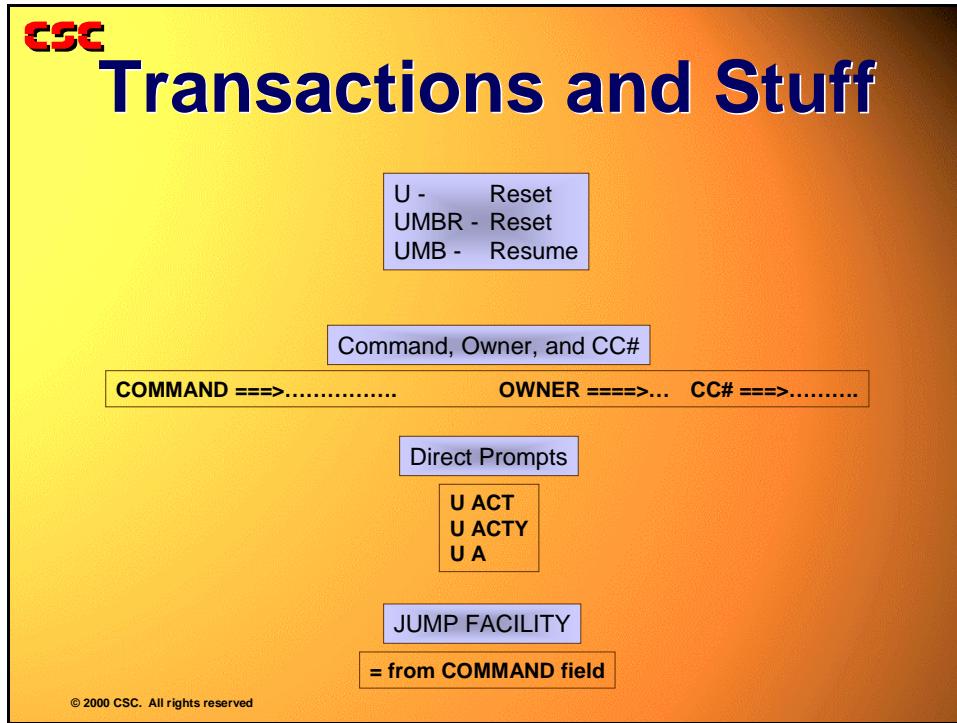
To access the "Umbrella System Process Dictionary Maintenance Menu", select option 1 from the "Umbrella System Master Menu" or enter U 1 from a cleared screen and press ENTER.



Notes:



Transactions



Notes:



Umbrella Transactions and Commands

UMB—Transaction

The UMB transaction, keyed from a cleared screen, will display the last screen of your previous Umbrella session.

UMBR—Transaction

The UMBR transaction will clear out any conversation that may have been in progress from a previous session and display the "Umbrella System Master Menu".

Command, Owner, and CC#

COMMAND ==>	OWNER ==> CC# ==>
-------------------	-----------------------------

These three fields are displayed on all non-menu CDMF and Process Dictionary screens. The COMMAND field is used to control and navigate the Umbrella System. The following pages contain the values that may be entered in this field. The two fields OWNER and CC# are required for use with commands for update, such as add, change, delete.

The instructor will provide owner and change control information later.

Umbrella Direct Prompts

Rather than go through each menu, it is possible to go directly to a particular screen using a direct prompt.

Umbrella Jump Facility

Umbrella Direct Prompts mentioned previously, are most commonly associated with the phrase "From a cleared screen, ENTER ... ". The Umbrella Jump Facility enhances the COMMAND field to be used as a receiver of Direct Prompts, or Jumps. The Jump syntax consists of the prefixing of any Direct Prompt with an equal sign "=".

Notes:





Umbrella Standard Process Dictionary PF Key Assignments

PF1 HELP	PF2 XREF	PF3 PLVL
PF4 CHG	PF5 ADD	PF6 INQ
PF7 VARIOUS	PF8 VARIOUS	PF9 NXT
PF10 VARIOUS	PF11 VARIOUS	PF12 NOT USED

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

Umbrella Transactions and Commands

Umbrella Standard CDMF/PCD Processing PF Key Assignments



CDMF/PCD Processing PF Key Assignment

PF1 HELP	PF2 XREF	PF3 PLVL
PF4 CHG	PF5 ADD	PF6 INQ
PF7 FMT	PF8 NXTE	PF9 NXT
PF10 NOT USED	PF11 NOT USED	PF12 NOT USED

© 2000 CSC. All rights reserved

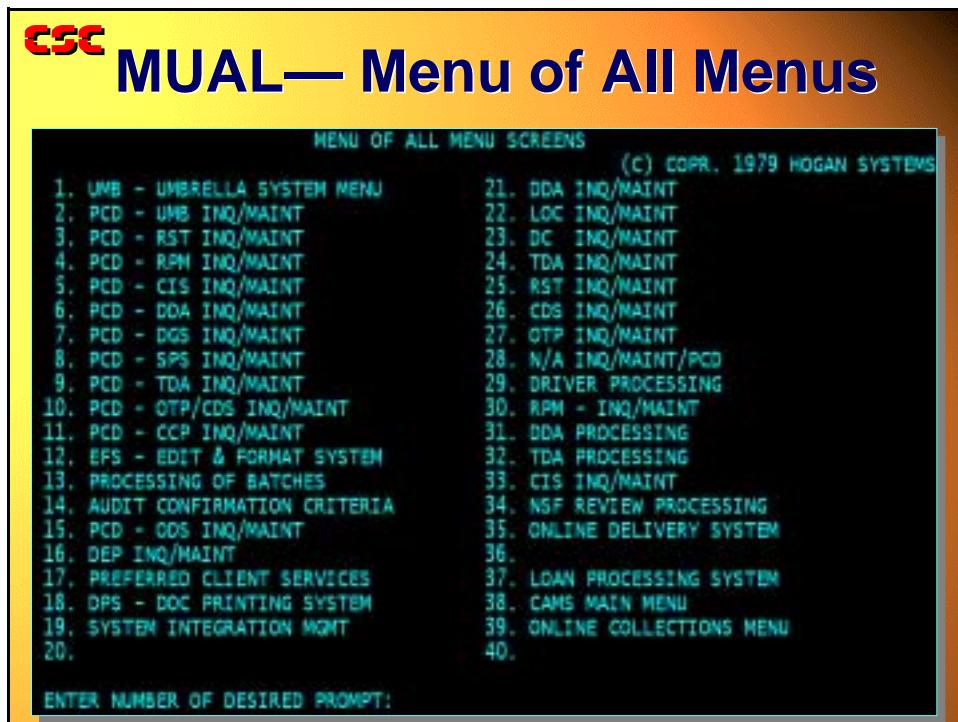
Notes:



System Transactions

Menu of All Menus—MUAL

The MUAL transaction displays the MENU OF ALL MENU SCREENS. From this display any of the Hogan Applications can be accessed.



Notes:



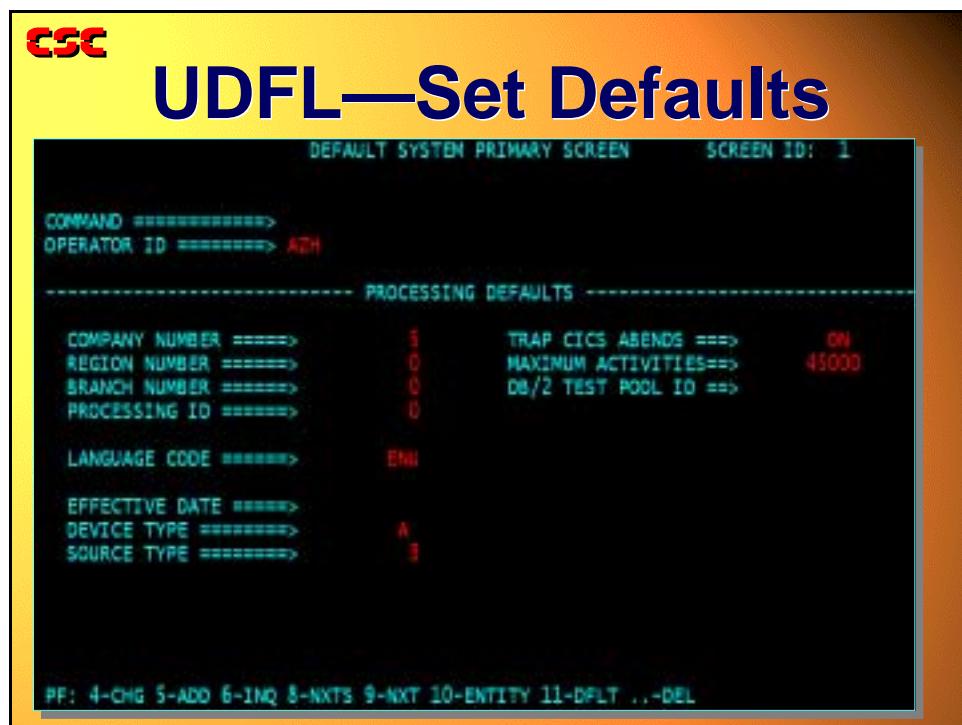
Umbrella Programming

System Transactions

Set Defaults—UDFL

The UDFL transaction replaces all internal and delivered mechanisms for setting default parameters for either a test or production environment. The default parameters are maintained by either Terminal ID or User ID. This transaction should be secured in a production environment. Even though the data is maintained on a CDMF data base, item maintenance is not used to maintain the parameters. Typically, only the date and company are required default parameters in a production environment.

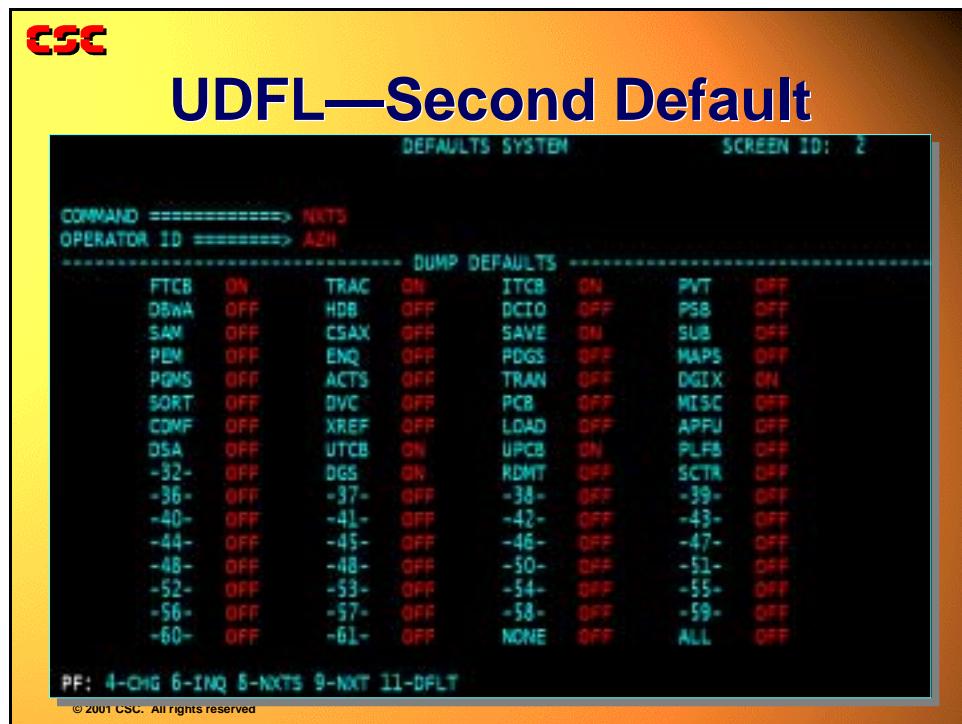
To display the first default screen enter UDFL, from a cleared screen.



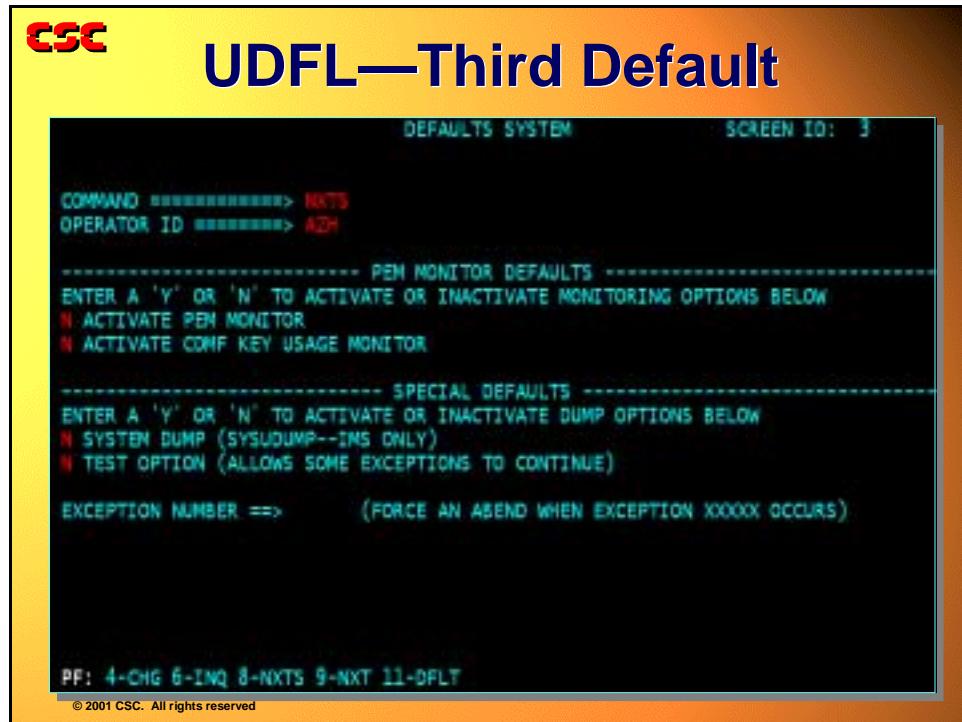
Notes:



To display the second default screen, press PF8.



To display the third default screen, press PF8.



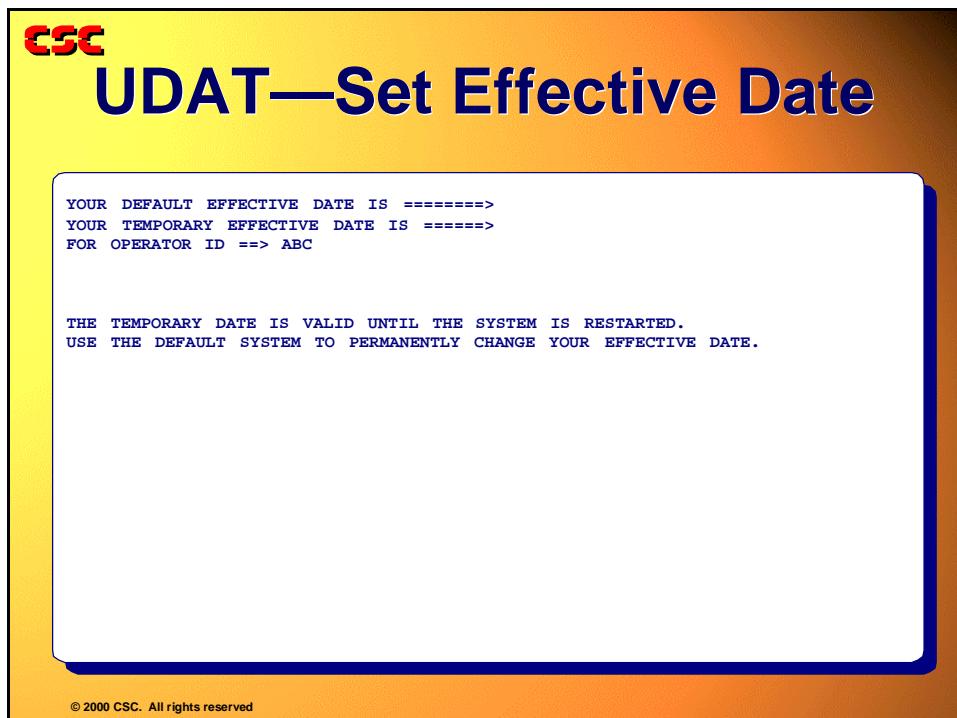
Umbrella Programming

System Transactions

Set Effective Date—UDAT

The UDAT transaction has been created to provide the same capability in an IMS environment as in a CICS environment. Also, in CICS, this transaction is no longer dependent upon the terminal control user area.

To set the effective date from a cleared screen, enter UDAT



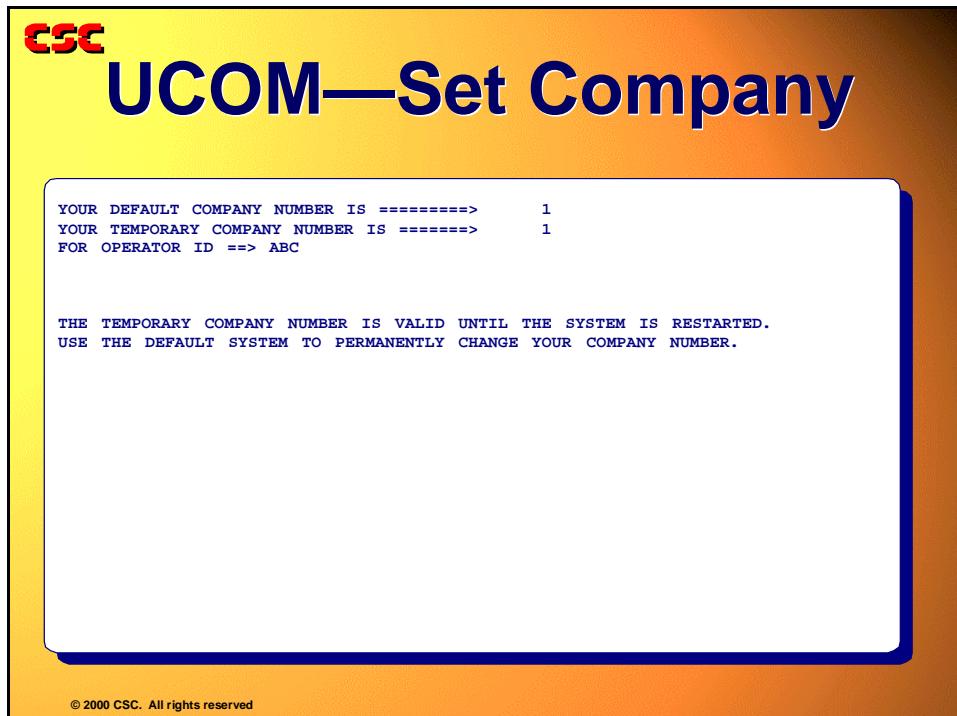
Notes:



Set Company—UCOM

The UCOM transaction has been created to provide the same capability in an IMS environment as in a CICS environment. Also, in CICS, this transaction is no longer dependent upon the terminal control user area.

Set the company from a cleared screen, enter UCOM.



Notes:



Umbrella Programming

Umbrella Screens Exercise



Umbrella Screens Exercise

1. From the Umbrella master menu, option _____ displays the Umbrella date services menu. From a cleared screen _____ will display the same Umbrella date services menu.
2. What is the length of data group 47100? _____.
3. To JUMP from data group definition 47100 to activity definition 15 one must enter:
 - From a cleared screen key JUMP ACTIVITY
 - COMMAND ==> =a 15
 - Press PF5
 - Key =act 15 in the DATA GROUP ID field
4. What happens when u 1 4 1 u48107m is keyed from a cleared screen?
5. What happens when u m u48107m is keyed from a cleared screen?
6. What is the description for program 703?

7. What is the activity type for 16400? _____

8. The primary search for program definition is by:
 - a) program description
 - b) program identification number
 - c) program link name
 - d) program source name
9. Associate a program function key with each of the following Umbrella system functions:

A.	PREVIOUS LEVEL	_____
B.	ADD	_____
C.	CHANGE	_____
D.	DELETE	_____
E.	GEN	_____
F.	INQUIRE	_____
G.	XREF	_____
H.	HELP	_____
I.	NEXT	_____
J.	SAVE	_____
K.	USER ASSIGNED	_____
L.	EDIT	_____
M.	BROWSE	_____
N.	SCROLL FORWARD	_____



Umbrella Programming

Umbrella Screens Exercise

O. SCROLL BACKWARD _____
P. SCROLL LEFT _____
Q. SCROLL RIGHT _____

10. How many data groups are in the hierarchical data base EMP? _____

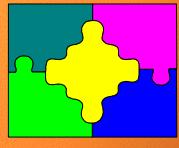
11. Format ID _____ is displayed using the prompt CCP.



Summary



Summary



© 2000 CSC. All rights reserved

- Batch and online updating
- The Umbrella System as menu driven
- Umbrella System access
- Secondary keys for search purposes
- PF key assignments

- Because PEM and the Process Dictionary provide means for accomplishing numerous standard data processing tasks, there are changes to learn for working under Hogan. Many pieces of the batch and online system are stored online and can be updated quickly in a real-time environment.
- The Umbrella System is menu-driven. Items can be located by walking through the menus. The direct prompt to the "Umbrella System Master Menu" is U from a cleared screen.
- Both alphabetic and numeric direct prompts are provided for access to any piece of the Umbrella System.
- Secondary keys exist for search of some Umbrella items, such as data elements.
- Alphabetic commands have corresponding PF key assignments, such as, ADD and PF5. Several are standardized across screens (PF1 through PF6, and PF9) while the others are assigned as needed. PF12 is for user assignment.



COBOL Under the Umbrella

4

Purpose



Purpose

A brief first look at and a comparison between a standard COBOL program and a COBOL program written to execute under PEM

© 2001 CSC. All rights reserved

Notes:



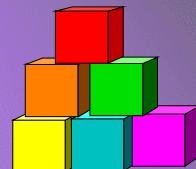
Umbrella Programming

Topic

Topic



Topics



- Standard COBOL
- Hogan COBOL

© 2000 CSC. All rights reserved

Notes:



Objective



Objective



© 2000 CSC. All rights reserved

- To identify the functions performed by PEM that are performed by the program in standard COBOL

Notes:



Standard COBOL Vs. Hogan COBOL

Because so much of the work normally performed in a COBOL program is executed by PEM through activities, the COBOL program written to execute under PEM/Umbrella looks somewhat different from a standard COBOL program. Some of the differences you can expect to encounter are documented below.

```
IDENTIFICATION DIVISION.  
PROGRAM-ID.      Z99999.  
AUTHOR.          HOGAN ASSOCIATES.  
DATE-COMPILED.  
REMARKS. ILLUSTRATE THE DIFFERENCES BETWEEN A STANDARD COBOL PROGRAM,  
AND A COBOL PROGRAM WHICH EXECUTES UNDER THE UMBRELLA.  
ENVIRONMENT DIVISION.  
CONFIGURATION SECTION.  
SOURCE-COMPUTER.   IBM-370.  
OBJECT-COMPUTER.   IBM-370.  
INPUT-OUTPUT SECTION. --  
FILE CONTROL.      |  
...               |  
DATA DIVISION.     |  
FD IN01           |  
...               |  
01 MASTER-RECORD. |  
  05 FIELD-NAME.  |  
    10 FIELD-A     PIC XX.  
    10 FIELD-B     PIC S9(5) COMP-3.  
    o  
    o  
    o  
FD OUT01          |  
...               |  
01 MAST-RECORD.   |  
  05 FIELD-G      PIC X(9).  
  05 FIELD-R      PIC S9(7)V99 COMP-3.  
  o  
  o  
  o  
WORKING-STORAGE SECTION. --  
77 COUNTER         PIC 99  VALUE 0. |  
01 WORK-AREA.       |  
  10 FIELD-P       PIC X.  
  10 FIELD-Q       PIC S9(3)V99 COMP-3.|  
...               --  
LINKAGE SECTION.  --  
01 MASTER-RECORD.  --  
  05 FIELD-NAME.  |  
    10 FIELD-A     PIC XX.  
    10 FIELD-B     PIC S9(5) COMP-3. |  
    ...             |  
01 FILE-TWO-RECORD. --  
  05 FIELD-G      PIC X(9).  
  05 FIELD-R      PIC S9(7)V99 COMP-3.  
  ...             --  
PROCEDURE DIVISION
```

I/O SECTION &
FILE CONTROL
NOT CODED
IN A HOGAN PROGRAM.

ALL HOGAN PROGRAMS
ARE CODED AS REUSABLE.
DATA GROUP DEFINITIONS
ARE CODED IN THE
LINKAGE SECTION.

ONLY STATIC VALUES
ARE CODED IN
WORKING STORAGE

TRANSACTION CONTROL BLOCK
IS ALWAYS THE FIRST 01 IN
LINKAGE.



Umbrella Programming

Standard COBOL Vs. Hogan COBOL

```
USING MASTER-RECORD
FILE-TWO-RECORD.

INIT-IT.          --
    o           | WORK AREA ALLOCATE ACTIVITY
    o           --
GRAND-OPENING.   --
    OPEN INPUT ...      | FILE OPENS ARE HANDLED BY
    OUTPUT ...       | PEM WHEN THE FIRST DATA
                      | BASE READ OR WRITE ACTIVITY
                      | IS REQUESTED.

READ-MASTER.     --
    o           | DATA BASE READ ACTIVITY CAUSES
    o           | PEM TO READ THE DATA BASE
    o           --
EDIT-THE-DATA.  --
    o           | EDIT AND FORMAT SYSTEM CAN
    o           | MEET SOME OF YOUR EDITING
    o           -- REQUIREMENTS.

EDITING o
                      | DATA VALUES USED TO IDENTIFY,
                      | VERIFY, AND CALCULATE CAN BE
                      | STORED AND MAINTAINED ON THE
                      | PROCESS CONTROL DATA SYSTEM.

GET-RATE.
    READ TABLE ....
    o
    o
    --
ERROR-MESSAGE.  --
    o           | CONDITION CODE PROCESSING SYSTEM
    o           --
CALCULATE-ITEM. --
    o           | PEM CANNOT HELP YOU HERE. YOU
    o           | WILL JUST HAVE TO CODE YOUR OWN.
    o           --
VALIDATE-DATE.  --
    o           | DATE SERVICES SYSTEM CONTAINS
    o           | SOME 52 DATE HANDLING FUNCTIONS.
    o           --
PRINT-REPORT.   --
    o           | SCHEDULED PROCESSING SYSTEM (SPS)
    o           | CAN HANDLE ALMOST ALL OF YOUR
    o           -- REPORTING NEEDS.

CLOSE-OUT.       --
    o           | HANDLED BY PEM. HOWEVER COBAL PROGRAMS
    o           -- DESIGNED TO RUN UNDER UMBRELLA MUST INCLUDE
                      THE STANDARD GOBACK AND STOP RUN CLAUSES. THE
                      STOP RUN STATEMENT MUST NEVER BE EXECUTED IN
                      OPEN CODE.

GOBACK.
STOP RUN.
```



Umbrella Programming

Example Hogan COBOL Program

Example Hogan COBOL Program

```
*****00000010
*          IDENTIFICATION DIVISION      *00000020
*****00000030
    SKIP1                                00000040
IDENTIFICATION DIVISION.
PROGRAM-ID.      Z47601.                  00000050
AUTHOR.          HOGAN ASSOCIATES.        00000060
DATE-COMPILED.   .                        00000070
REMARKS.         THIS PROGRAM REPRESENTS A SAMPLE COBOL 00000080
.
.
.

*****00000240
*          DATA DIVISION             *00000250
*****00000260
DATA DIVISION.          00000280
FILE SECTION.          00000290
WORKING-STORAGE SECTION. 00000300
01 FILLER            PIC X(08) VALUE 'Z47601'. 00000310
01 CF-CONSTANTS.      00000330
05 READ-EMP-DB-989613. 00000340
    10 FILLER            PIC S9(08) COMP VALUE +989613. 00000350
.
.
.

*****00000240
*          LINKAGE SECTION       *00000250
*****00000260
LINKAGE SECTION.      00000280
01 TRANSACTION-CONTROL-BLOCK.

.
.

01 EMP-KEY-GROUP.
.
.

*****00000790
*          PROCEDURE DIVISION      *00000800
*****00000810
PROCEDURE DIVISION USING TRANSACTION-CONTROL-BLOCK 00000830
    EMP-KEY-GROUP          00000850
    EMP-INFO-GROUP         00000860
    EMP-JOB-STATUS          00000870
    EMP-CURRENT-PAY        00000880
    EMP-YEAR-TO-DATE-PAY.  00000890
.
.

AB000-MAINLINE SECTION. 00000980
    PERFORM BA000-READ-EMP-DB. 00001000
.
.

GOBACK. STOP RUN.
```



Umbrella Programming

Example Hogan COBOL Program

```
*****00001070
*          S U B R O U T I N E S      *00001080
*****00001090
BA000-READ-EMP-DB SECTION.      00001110

.
.

CA000-CALL-PEM SECTION.      00001320
    CALL 'PEM' USING TRANSACTION-CONTROL-BLOCK.
CA999-EXIT.                  00001340
                                00001360
```

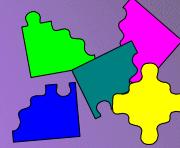
Notes:



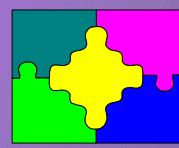
Summary

CSC

Summary



- Most non-logic functions (I/O, System Interfaces) performed in a traditional COBOL program are handled by PEM, either automatically or through the execution of an activity.



© 2000 CSC. All rights reserved

Notes:



Change Control System - CCS

5

Purpose



Purpose

*Explore how changes in
Hogan products are
captured through the
Change Control System and
the types of reports
available*

© 2000 CSC. All rights reserved.

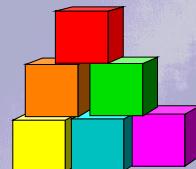
Notes:



Topics



Topics



- Change control number and ownership
- Online maintenance
- Processing flow
- Managing Change Control
- Types of reports
- Sample change control reports
- JCL for generating reports

© 2000 CSC. All rights reserved.

Notes:



Objectives



Objectives



To become familiar with:

- Use of Change Control number with application ownership
- CCS online
- Source and online changes process
- Managing Change Control key issues
- Change Control reports
- JCL parameters for Change Control reports

© 2000 CSC. All rights reserved.

1. To explain the use of Change Control number with application ownership
2. To show which parts of CCS are maintained online
3. To describe how source and online changes are processed
4. To list key issues for managing Change Control
5. To list the general types of reports that can be requested through Change Control
6. To identify the updates made during class from information supplied on sample reports
7. To explain JCL parameters for Change Control reports.

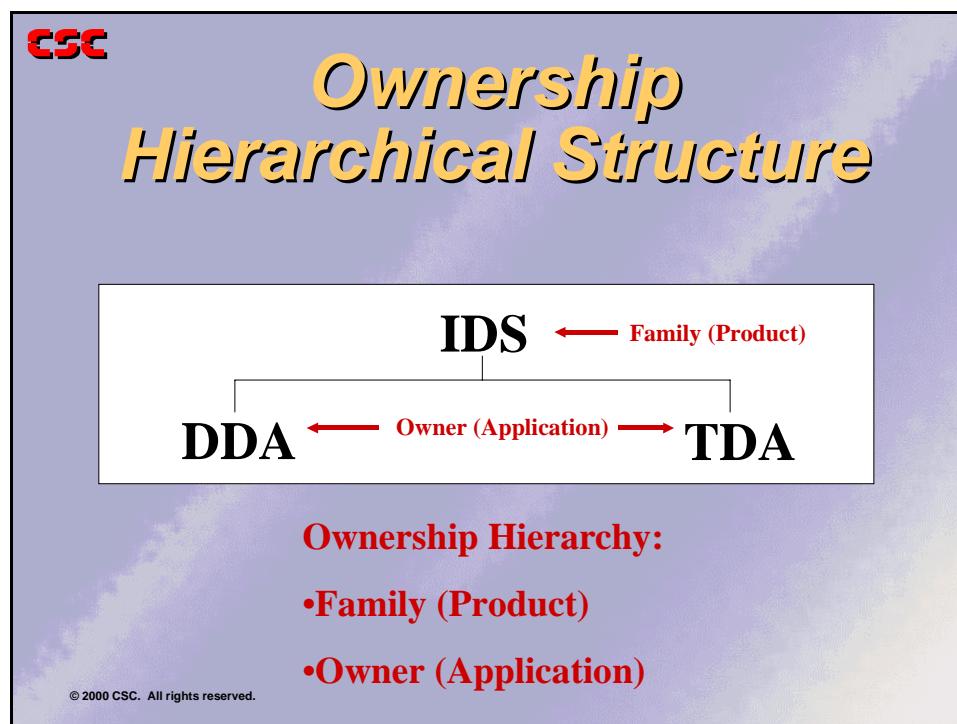


Change Control System

The change control system provides the facility to track and report all changes to the CDMF data bases and source modules.

The key to the system is site and change control number. The change control numbers are assigned to a specific Hogan Application (ownership) when they are set up. The site control record can be flagged to protect the transaction code on all maps and restrict changes to like application ownership.

Applications must be defined to PEM. They belong to families in a hierarchical structure such as:



Change Control Numbers may be assigned at the family level, or at the application level. If issued at the family level (IDS), all items within that family (IDS, TDA and DDA) may be updated. If issued at the application level (DDA), only items within that application (DDA) may be updated.

All source and CDMF data base changes are accomplished by using a change control number. No attempt is made to monitor application data base changes other than those on CDMF.

Online: all screens on the Umbrella System have required fields for Ownership and a Change Control Number.

Source: all programs, copybooks, and tables are maintained on a source manager (LIB, PAN, EDV, IEB) data base. Exits are provided for LIBRARIAN, PANVALET, and ENDEVOR to authorize change control numbers for all changes to source.



Online Maintenance

All Umbrella screens have a place for users to enter a change control number and ownership. When the change control number is added to the system, the ownership may be either a family or an application. The user must specify which by setting a flag in the site control record. Below is a sample screen to illustrate the location of these two fields required for updates.

CSC

Online Maintenance Example

ACTION SUCCESSFUL					
PROGRAM DEFINITION INQUIRY/MAINTENANCE			OWNER ==> UES CC# ==>		
COMMAND ==> INQ			OWNER ==> UES CC# ==>		
PROGRAM ID ==> 488401			EFF DATE ==> 78/01/01		
LINKNAME ==> U88401			PEM TECHNOLOGY ==> FULLWORD		
DESCRIPTION ==> UES QUOTATIONS SYSTEM INQUIRY PGM			STATUS ==> TEST		
HIGH, MED, OR LOW USAGE? ==> LOW			USED ONLINE, BATCH OR BOTH? ==> BOTH		
----- DATA GROUPS USED BY PROGRAM -----					
* PP ---DGID---	* PP ---DGID---	* PP ---DGID---			
1 488401	2 488300	3 488301			
4 488302	5 488303	6 488304			
7 488310	8 488311	9 488001			
10 48551					
----- AUTHORIZED ACTIVITIES -----					
* -ACTIVITY- * -ACTIVITY- * -ACTIVITY- * -ACTIVITY- * -ACTIVITY- * -ACTIVITY-					
1398 48000	488510	488601	488613	488615	
488618	488620				
LAST CHG:DATE 92/06/15 TIME 9.35.40 CC# SRCE UMB130 OPER					
PF: 2-XREF 3-PLVL 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-BYLINK 11-BYID ..-ACTS					

© 2000 CSC. All rights reserved.

Notes:



Umbrella Programming

Change Control System

Change Control Master

- The master file is a keyed VSAM or IMS data base.
- The key is site ID and change control number.
- It contains data concerning the overall scope of the change, such as:
 - Application Ownership
 - Person Assigned
 - Mandays
 - Description
 - Priority.

A master record must exist for a change control number. Associated records to describe the master more completely may exist. These records form a hierarchical data base, CCMSTR.

Change Control Detail

- The Detail file is a keyed VSAM or IMS data base.
- Its key is Site, Change Control Number, and Data ID.
- It contains information about data that has been changed, such as:
 - Name of Person
 - Date
 - Type of Data (such as map or PCD).

CCDETL is the Hierarchical Data Base of the detail records.

Change Control Inquiry/Maintenance

The following pages illustrate some of the Change Control data records as they would appear online.



Change Control BASE Screen

CSC

Change Control BASE Record

```

----- 59021 ACTION COMPLETE
----- CHANGE CONTROL INQUIRY/MAINTENANCE -----
COMMAND ======> INQ SITE ======> HOGN
CHANGE CONTROL # ==> 37195 MOVE TO CC# ==>
DATA ======> M (MASTER,DETAIL,PROBLEM,RESOLUTION,VARIANCE,FIX)
DETAIL KEY: TYPE ==> EFF DATE ==>
FORMAT ======> COMPANY =====>

=====> =====<==
APPLICATION ==> TRD PRIORITY ==> LOW REASON ==> CUSTOM
CLIENT NAME ==> INT HOGN REP=>
CONTACT NAME ==> ROBERTA SMITH PHONE ==> 8386
DESCRIPTION ==> FPS/DPS CLASS UPGRADE FOR 86XX
MATERIALS PROV=> NONE
TARGET DATE ==> 00/00/00 RESPONSIBILITY==> C ESTIMATED TO COMPL ACTUAL
ASSIGNED TO ==> RMS ON=> 86/04/23 MANDAYS => 0.0 0.0 0.0
RESOLUTION ==>
DATE CLOSED ==> 00/00/00 BASE RELEASE ID=> REPORT CODE==>
VARIANCE ==>
FEED INFOSYS? => N DESCRIPTION RECORDS: PROBLEM,RESOLUTION
#DETAIL RECORDS: 269 CDMF/SOURCE UPDATED?: Y REPORTED AGAINST=>
CREATED: 86/04/23 UPDATED: 86/09/29 09.25.45 BY: RMS
PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-RESL 8-DETL 9-NXT 10-PROB 11-VARI ..-DEL

```

© 2000 CSC. All rights reserved.

Change Control PROBLEM Screen—PF10

CSC

Change Control PROBLEM Record

```

----- 59025 ACTION COMPLETE
----- CHANGE CONTROL INQUIRY/MAINTENANCE -----
COMMAND ======> INQ SITE ======> HOGN
CHANGE CONTROL # ==> 37195 MOVE TO CC# ==>
DATA ======> P (MASTER,DETAIL,PROBLEM,RESOLUTION,VARIANCE,FIX)
DETAIL KEY: TYPE ==> EFF DATE ==>
FORMAT ======> COMPANY =====>

=====> =====<==
----- PROBLEM/ENHANCEMENT DESCRIPTION -----
REVISE ONLINE SUBSYSTEMS (FPS/DPS) CLASS FOR UMB 86XX RELEASE

----- PF 10 -----
----- END OF DESCRIPTION -----
PROBLEM DESC: FPS/DPS CLASS UPGRADE FOR 86XX
ASSIGNED TO: RMS DETAIL RECORDS: 269 UPDATED: 86/09/29 CLOSED:
PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-RESL 8-DETL 9-NXT 10-PROB 11-VARI ..-DEL

```

© 2000 CSC. All rights reserved.



Umbrella Programming

Change Control System

Change Control RESOLUTION Screen—PF 7

CSC

Change Control *RESOLUTION Record*

```
59023 ACTION COMPLETE
----- CHANGE CONTROL INQUIRY/MAINTENANCE -----
COMMAND ======> INQ SITE ======> HOGN
CHANGE CONTROL # ==> 37195 MOVE TO CC# ==>
DATA ======> R (MASTER,DETAIL,PROBLEM,RESOLUTION,VARIANCE,FIX)
DETAIL KEY: TYPE ==> EFF DATE ==>
FORMAT ======> COMPANY ==>

=====> RESOLUTION DESCRIPTION <===
REVISE UMBRELLA ON-LINE SUBSYSTEMS CLASS (FPS/DPS) FOR UMB 86XX.
NEW SCREENS - BOTH ON-LINE MAINTENANCE UNDER FPS.
NEW FEATURES
ON-LINE MAINTENANCE OF APPLICATION TABLE--NO LONGER HARD-CODED.
DPS DOCUMENTS CREATED USING MAP MAINTENANCE.

PF 7
```

----- END OF DESCRIPTION -----
PROBLEM DESC: FPS/DPS CLASS UPGRADE FOR 86XX
ASSIGNED TO: RMS DETAIL RECORDS: 269 UPDATED: 86/09/29 CLOSED:
PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-RESL 8-DETL 9-NXT 10-PROB 11-VARI ..-DEL

© 2000 CSC. All rights reserved.

Change Control DETAIL Screen—PF8

Key(00000,47341,00000,00003)

CSC

CCDETL Entry -Type 3

CCP - Key (00000,47341,00000,00003)

```
59022 ACTION COMPLETE
----- CHANGE CONTROL INQUIRY/MAINTENANCE -----
COMMAND ======> NXT SITE ======> HOGN
CHANGE CONTROL # ==> 37195 MOVE TO CC# ==>
DATA ======> D (MASTER,DETAIL,PROBLEM,RESOLUTION,VARIANCE,FIX)
DETAIL KEY: TYPE ==> 3 EFF DATE ==> 78/01/01
FORMAT ======> 1301 COMPANY ==> DFLLT

PCD DATA
=====> COND CODE - 0,47341, 0, 3 <==
TYPE OF CHANGE ==> ADDED
DETAIL TYPES: 1=LIBR,2=DOC,3=PCD,4=EFS,5=MAP,6=ACT 7=DG,8=PGM,9=RELATED CC#
10=TRANS DEF,11=DEV CHAR,12=SEQ FILES,13=HIER DBS, 14=COMP LIST,15=FMT MAINT
*** DETAIL DATA ***
APPLICATION => TRD BY ==> RMS
DESCRIPTION =>
LINKNAME ==> (PGM DEFS ONLY) NUMBER OF UPDATES: 1
*** SOURCE MODULE RELATED DATA ***
MODULE: COPY/RENAME: RESEQ:
LANGUAGE ==> TYPE ==>
*** ENDEVOR INFORMATION ***
TYPE ==> VERSION ==> 0 LEVEL ==> 0 PF 8

TYPE ==>
VERSION ==> 0 LEVEL ==> 0

CREATED: 94/11/14 UPDATED: 00/00/00 00.00.00 BY: BATCH
PF: 3-PLVL 4-CHG 5-ADD 6-INQ 8-IKEY 9-NXT 10-PROB 11-CPY
```

© 2000 CSC. All rights reserved.



Change Control DETAIL Screen

ELE - Key(DP13-ACTION)

CSC

CCDETL Entry - Type 3
ELE - Key (DP13 - ACTION)

59022 ACTION COMPLETE	
----- CHANGE CONTROL INQUIRY/MAINTENANCE -----	
COMMAND =====> NXT	SITE =====> HOGN
CHANGE CONTROL # ==> 37195	MOVE TO CC# ==>
DATA =====> D (MASTER,DETAIL,PROBLEM,RESOLUTION,VARIANCE,FIX)	
DETAIL KEY: TYPE ==> 3	EFF DATE ==> 78/01/01
FORMAT =====> 48333	COMPANY =====> DFLT
PCD DATA	
=====> DG ELE - DP13-ACTION,47013,C,2	<==
TYPE OF CHANGE ==> ADDED	
DETAIL TYPES: 1=LIBR,2=DOC,3=PCD,4=EFS,5=MAP,6=ACT 7=DG,8=PGM,9=RELATED CC#	
10=TRANS DEF,11=DEV CHAR,12=SEQ FILES,13=HIER DBS, 14=COMP LIST,15=FMT MAINT	
*** DETAIL DATA ***	
APPLICATION => TRD	BY ==> RMS
DESCRIPTION =>	
LINKNAME ==>	(PGM DEFS ONLY) NUMBER OF UPDATES: 1
*** SOURCE MODULE RELATED DATA ***	
MODULE: COPY/RENAME:	RESEQ:
LANGUAGE ==>	TYPE ==>
*** ENDEVOR INFORMATION ***	
TYPE ==>	VERSION ==> 0 LEVEL ==> 0
CREATED: 94/11/14 UPDATED: 00/00/00 00.00.00 BY: BATCH	
PF: 3-PLVL 4-CHG 5-ADD 6-INQ 8-IKEY 9-NXT 10-PROB 11-CPY	

© 2000 CSC. All rights reserved.

Notes:



Umbrella Programming

Change Control System

Change Control Processing Flow

Source Changes

Source changes are recorded by the source manager. For sites with LIBRARIAN, PANVALET, and ENDEVOR an exit is used to record the change on CCTRANS when a module is changed.

Nightly, a batch change control job is run that reads the transaction file and updates the detail and master files.

CDMF Changes

CDMF changes are logged as they occur on CCTXNDB. Source updates are logged on CCTRANS. Each night a batch run unloads the logged records, consolidates them and optionally updates CCLOG. Change control applies the consolidated change information to CCMSTR and CCDETL.

If a detail record already exists for the same control number, detail name, and change type, the detail record will be updated to reflect the latest change information. The use of CCS software is required to maintain any Hogan Application System. CCS is the only method of migration for CDMF items.

Managing Change Control

Managing the change control system is essential. The system control needs to be centralized. The person or small group in charge of change control can determine how numbers will be assigned and closed, when to run jobs to update the files, and who has access to CCS.

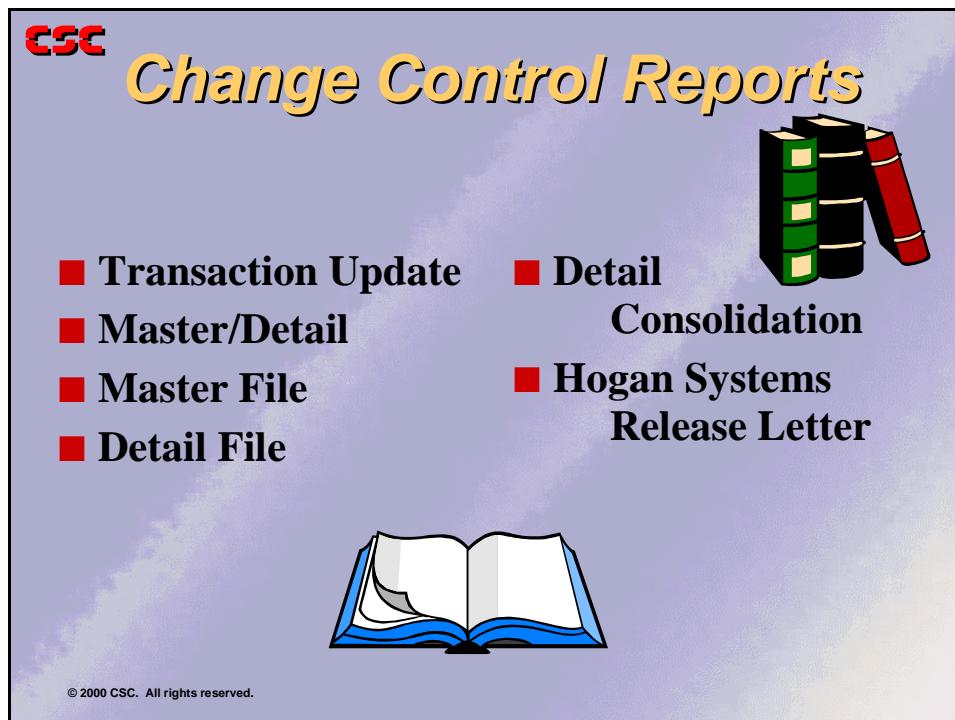
It is generally desirable to limit access to CCS through password and operator restrictions under FPS or via Site Security access to files. Actual assignment of change control numbers can be done manually or automatically at each site's discretion. To prevent the transaction files from filling up, batch jobs should be run nightly to update the master and detail files.

Use of the change control system can expedite custom changes and internal changes. It provides an automated system of migrating the changes from test to production. In addition, its use internally provides reports of user changes to apply to new releases compare to upgrades.

Notes:



Change Control Reports



Report Descriptions

There are many types of reports available through the Change Control System. The multi-purpose reports can be used to document updates, trace corrections, compare new releases with custom coding, review fixes, and manage staff.

The types of reports available through CCS include:

- TRANSACTION UPDATE REPORT

The purpose of this report is to reflect each transaction that is posted to the detail file. A comment indicates how the transaction was posted.

- MASTER/DETAIL FILE LISTING

The purpose of this report is to reflect the contents of the master file and, optionally, the detail records associated with each master.

The report indicates to whom specific control numbers have been assigned, the number of mandays, and the date to be completed. At the end of the report a product matrix can be requested that totals the item counts and mandays for each product. (See example on page 13.)



Umbrella Programming

Change Control Reports

- **MASTER FILE REPORT**

This report contains the change control master records built by the extract program.
The sequential input file can be sorted in any order.

- **DETAIL FILE REPORT**

This report contains the change control detail records built by the extract program.
The sequential input file can be sorted in any order.

- **DETAIL CONSOLIDATION REPORT**

This report prints a one line summary of the activity for each detail name.

- **HOGAN SYSTEMS RELEASE LETTER**

The purpose of this report is to provide the information needed for installations of new releases. It is produced for every release of a Hogan product.

Notes:



Umbrella Programming

Change Control Reports

DATE 88/07/08 TIME 16:13:35	CHANGE CONTROL MASTER FILE LOG	PAGE 00001
CC NUMBER CREATED APP PRTY REASON CODE ASSIGN TARGET ORG EST EST TC SHORT DESCRIPTION OF PROBLEM/ENHANCEMENT /CLIENT NAME /RESP /CLOSED /ACTUAL DAYS /MATERIALS PROVIDED--RESOLUTION /VARIANCE		
1009 84/09/12 TRD HIGH NEW DEVELOPMENT JLB 84/10/01T 1.0 'TOUCH' ALL PARTS OF FPS/DPS CLASS FOR ON SITE CLASS TAPE. ***** RPTCODE- INFOSYS- (OPEN) 1.0 MISC #DTL---87 CDMF/SRCE-		
PROB : IN ORDER TO PULL AN RCS CPN TYPE TAPE ON FPS/DPS, WE NEED TO TOUCH ALL PARTS OF THE 'DEMO' FPS APPLICATION AS WELL AS REQUIRED PD ENTRIES FOR DPS CLASS.		
1020 87/06/18 TRD HIGH NEW DEVELOPMENT JLB 88/01/01T 1.0 CMS CLASS ***** RPTCODE- INFOSYS- (OPEN) 1.0 MISC #DTL---38 CDMF/SRCE-Y		
PROB : BUILD CMS LABS		
1500 86/01/28 INT HIGH NEW DEVELOPMENT JLB 86/01/28A 10.0 ***** RPTCODE- INFOSYS- 86/05/02C 10.0 #DTL--1317 CDMF/SRCE-		
11301 84/08/20 TRD HIGH DEFICIENCY JLB 84/08/23A 0.1 MIRROR CC# FROM HOGAN-UPDATE TRD ITEMS INTERNAL 84/08/24C 10.0 RPTCODE- INFOSYS- #DTL-----4 CDMF/SRCE-		
PROB : FIS OWNERSHIP ON TRD ITEMS, JCL FIXES, LIBR FIXES, ETC		
11 84/12/17 PCS LOW NEW DEVELOPMENT JLB 84/12/17A 100.0 100.0 STANDARD CHANGE CONTROL NUMBER FOR PCS EDUCATION ***** RPTCODE- INFOSYS- (OPEN) 0.0 NONE #DTL--1224 CDMF/SRCE-Y		
PROB : CC# FOR BASIC PCS CLASS WORK		
900 85/10/30 TRD LOW NEW DEVELOPMENT JLB 85/10/29A 2.0 BUILD AUTO/PAGE CONTROL SYSTEM ***** RPTCODE- INFOSYS- 85/11/11C 10.0 #DTL---16 CDMF/SRCE-		
PROB : PROVIDE TXN TO PLACE ALL TERMINMALS LISTED ON A PCD 1694 ITEM IN AUTOPAGE MODE FOR FPS 'TRON' USE AND RETURN TO PAGE MODE FOR STUDENT USE.		
1401 86/01/06 INT LOW NEW DEVELOPMENT JLB 86/01/06A 10.0 ODS DEMO PCD/PCD LINK ***** RPTCODE- INFOSYS- 86/01/07C 10.0 #DTL---91 CDMF/SRCE-		
1402 86/05/05 INT LOW NEW DEVELOPMENT JLB 86/05/05A 10.0 AFC INQ TRAN ***** RPTCODE- INFOSYS- 86/05/07C 10.0 #DTL---43 CDMF/SRCE-		
PROB : ADD A BASIC INQ TXN USING THE AFCFILE AS A PEM DB.		
1403 87/07/07 INT LOW NEW DEVELOPMENT JLB 87/07/01A 000.0 MESSAGE TRAN ***** RPTCODE- INFOSYS- 87/07/07C 10.0 #DTL---46 CDMF/SRCE-Y		
PROB : BUILD MESSAGE TRAN		

DATE 88/07/08 TIME 16:13:35 CHANGE CONTROL MASTER FILE LOG PAGE 00001

ASSIGNED TO=JLB PRIORITY=HIGH		
CC NUMBER CREATED APP PRTY REASON CODE ASSIGN TARGET ORG EST EST TC SHORT DESCRIPTION OF PROBLEM/ENHANCEMENT /CLIENT NAME /RESP /CLOSED /ACTUAL DAYS /MATERIALS PROVIDED--RESOLUTION /VARIANCE		
1009 84/09/12 TRD HIGH NEW DEVELOPMENT JLB 84/10/01T 1.0 'TOUCH' ALL PARTS OF FPS/DPS CLASS FOR ON SITE CLASS TAPE. ***** RPTCODE- INFOSYS- (OPEN) 1.0 MISC #DTL---87 CDMF/SRCE-		
PROB : IN ORDER TO PULL AN RCS CPN TYPE TAPE ON FPS/DPS, WE NEED TO TOUCH ALL PARTS OF THE 'DEMO' FPS APPLICATION AS WELL AS REQUIRED PD ENTRIES FOR DPS CLASS.		
DE UOS53551 DFLT 84/09/12 JLB TRD SOURCE MODULE ADDED 001 DELTA FOR I53551-APPLICATION		
Z47701	DFLT	84/09/12 JLB TRD SOURCE MODULE ADDED 001 STUDENT PGM FOR FPS/DPS
COND CODE - 0,47532, 0,	4 DFLT	1301 84/09/14 JLB TRD PCD DATA UPDATED 001
FPS KEY - DEMDOCS000	DFLT	1690 84/09/12 JLB TRD PCD DATA UPDATED 001
FPS KEY - DEMDOCS050	DFLT	1690 84/09/12 JLB TRD PCD DATA UPDATED 001
FPS KEY - DEMTESTTEMP	DFLT	1690 84/09/12 JLB TRD PCD DATA UPDATED 001
FPS KEY - DEMTEST100	DFLT	1690 84/09/12 JLB TRD PCD DATA UPDATED 001
FPS KEY - DEMTEST110	DFLT	1690 84/09/12 JLB TRD PCD DATA UPDATED 001
DEM	DFLT	1698 86/03/05 JK1 TRD PCD DATA ADDED 001
DEM	DFLT	1698 84/09/12 JLB TRD PCD DATA UPDATED 001



Umbrella Programming

Change Control Reports

SPS LINE - 47532,0,0,100	DFLT	2300	84/09/12	JLB ZZD PCD DATA	UPDATED	002
SPS FORMAT - 47532,0,100,100	DFLT	2302	84/09/12	JLB ZZD PCD DATA	UPDATED	002
SPS FORMAT - 47532,0,100,200	DFLT	2302	84/09/12	JLB PEM PCD DATA	UPDATED	002
SPS FORMAT - 47532,0,100,300	DFLT	2302	84/09/12	JLB PEM PCD DATA	UPDATED	001
SPS FORMAT - 47532,0,100,400	DFLT	2302	84/09/12	JLB PEM PCD DATA	UPDATED	001
SPS FORMAT - 47532,0,100,1000	DFLT	2302	84/09/12	JLB PEM PCD DATA	UPDATED	001
DG ELE - I560#CPS,2760,E,2816	DFLT	48333	84/09/12	JLB DPS PCD DATA	ADDED	001
DG ELE - I560#CPS,2760,E,2816	DFLT	48333	84/09/12	JLB DPS PCD DATA	DELETED	001
DG ELE - I560#PRT,2760,E,3072	DFLT	48333	84/09/12	JLB DPS PCD DATA	ADDED	001
DG ELE - I560#PRT,2760,E,3072	DFLT	48333	84/09/12	JLB DPS PCD DATA	DELETED	001
DG ELE - I560#RPR,2760,E,3328	DFLT	48333	84/09/12	JLB DPS PCD DATA	ADDED	001
DG ELE - I560#RPR,2760,E,3328	DFLT	48333	84/09/12	JLB DPS PCD DATA	DELETED	001
DG ELE - I560ACTN,2760,E,256	DFLT	48333	84/09/12	JLB DPS PCD DATA	ADDED	001
DG ELE - I560ACTN,2760,E,256	DFLT	48333	84/09/12	JLB DPS PCD DATA	DELETED	001
DG ELE - I560BRN,2760,E,1024	DFLT	48333	84/09/12	JLB DPS PCD DATA	ADDED	001
DG ELE - I560BRN,2760,E,1024	DFLT	48333	84/09/12	JLB DPS PCD DATA	DELETED	001
DG ELE - I560COID,2760,E,768	DFLT	48333	84/09/12	JLB DPS PCD DATA	ADDED	001
DG ELE - I560COID,2760,E,768	DFLT	48333	84/09/12	JLB DPS PCD DATA	DELETED	001
DG ELE - I560DOC,2760,E,1280	DFLT	48333	84/09/12	JLB DPS PCD DATA	ADDED	001
DG ELE - I560DOC,2760,E,1280	DFLT	48333	84/09/12	JLB DPS PCD DATA	DELETED	001
DG ELE - I560EFFD,2760,E,1536	DFLT	48333	84/09/12	JLB DPS PCD DATA	ADDED	001
DG ELE - I560EFFD,2760,E,1536	DFLT	48333	84/09/12	JLB DPS PCD DATA	DELETED	001
DG ELE - I560LOOP,2760,E,4352	DFLT	48333	84/09/12	JLB DPS PCD DATA	ADDED	001
DG ELE - I560LOOP,2760,E,4352	DFLT	48333	84/09/12	JLB DPS PCD DATA	DELETED	001
DG ELE - I560MAX#,2760,E,2560	DFLT	48333	84/09/12	JLB DPS PCD DATA	ADDED	001
DG ELE - I560MAX#,2760,E,2560	DFLT	48333	84/09/12	JLB DPS PCD DATA	DELETED	001
DG ELE - I560REP,2760,E,2304	DFLT	48333	84/09/12	JLB DPS PCD DATA	ADDED	001
DG ELE - I560REP,2760,E,2304	DFLT	48333	84/09/12	JLB DPS PCD DATA	DELETED	001
DG ELE - I560RSLT,2760,E,512	DFLT	48333	84/09/12	JLB DPS PCD DATA	ADDED	001
DG ELE - I560RSLT,2760,E,512	DFLT	48333	84/09/12	JLB DPS PCD DATA	DELETED	001
DG ELE - I560SUF,2760,E,2048	DFLT	48333	84/09/12	JLB DPS PCD DATA	ADDED	001
DG ELE - I560SUF,2760,E,2048	DFLT	48333	84/09/12	JLB DPS PCD DATA	DELETED	001
DG ELE - I560TRM,2760,E,1792	DFLT	48333	84/09/12	JLB DPS PCD DATA	ADDED	001
DG ELE - I560TRM,2760,E,1792	DFLT	48333	84/09/12	JLB DPS PCD DATA	DELETED	001
DG ELE - I560UA1,2760,E,7681	DFLT	48333	84/09/12	JLB DPS PCD DATA	ADDED	001

DATE 88/07/08 TIME 16:13:35

CHANGE CONTROL MASTER FILE LOG

PAGE 00001

ASSIGNED TO=JLB PRIORITY=LOW

CC NUMBER /CLIENT NAME	CREATED APP PRTY REASON CODE	ASSIGN TARGET /RESP	ORG EST /CLOSED	EST /ACTUAL	TC /DAYS	SHORT DESCRIPTION OF PROBLEM/ENHANCEMENT /MATERIALS PROVIDED--RESOLUTION /VARIANCE
------------------------	------------------------------	---------------------	-----------------	-------------	----------	--

11 84/12/17 PCS LOW NEW DEVELOPMENT	JLB	84/12/17A 100.0 100.0	STANDARD CHANGE CONTROL NUMBER FOR PCS EDUCATION	(OPEN)	0.0	NONE
RPTCODE- INFOSYS-		#DTL--1224	CDMF/SRCE-Y			

PROB : CC# FOR BASIC PCS CLASS WORK

A64010	DFLT	86/12/11 PSM	SOURCE MODULE UPDATED	001 FIX 40386 - POST PRO8612
INSTAL				
A64059	DFLT	86/12/11 PSM	SOURCE MODULE UPDATED	001 FIX 40339 - POST PRO8612
INSTAL				
A64123	DFLT	86/12/11 PSM	SOURCE MODULE UPDATED	001 FIX 40386 - POST PRO8612
INSTAL				
A64133	DFLT	86/12/11 PSM	SOURCE MODULE UPDATED	001 FIX 40388 - POST PRO8612
INSTAL				
A64157	DFLT	86/12/11 PSM	SOURCE MODULE UPDATED	001 FIX 40442 - POST PRO8612
INSTAL				
E12040	DFLT	86/12/18 PSM	SOURCE MODULE UPDATED	001 FIX 40366 - POST PRO8612
INSTAL				
E12311	DFLT	86/12/18 PSM	SOURCE MODULE UPDATED	001 FIX 40430 - POST PRO8612
INSTAL				
E12470	DFLT	86/12/18 PSM	SOURCE MODULE UPDATED	001 FIX 40456 - POST PRO8612
INSTAL				
E62318	DFLT	86/12/15 PSM	SOURCE MODULE UPDATED	001 FIX 40400 - POST PRO8612
INSTAL				
E62395	DFLT	86/12/15 PSM	SOURCE MODULE UPDATED	001 FIX 40400 - POST PRO8612
INSTAL				
E62700	DFLT	86/12/18 PSM	SOURCE MODULE UPDATED	001 FIX 39506 - POST PRO8612
INSTAL				
H14100	DFLT	86/01/22 JLB	SOURCE MODULE UPDATED	001 FIX 35025
H14110	DFLT	86/01/22 JLB	SOURCE MODULE UPDATED	001 FIX 35026
H14130	DFLT	86/01/22 JLB	SOURCE MODULE UPDATED	002 FIX 34797
H14602	DFLT	86/01/22 JLB	SOURCE MODULE UPDATED	001 FIX 34797
PCACTDATRPM(FFFF)8412	DFLT	38900 85/01/11 KND PCS	PCD DATA	UPDATED 001
DG ELE - PCS70101,140701,E,1059	DFLT	48333 84/12/17 JLB	PCS PCD DATA	ADDED 001
DG ELE - PCS70101,140701,E,1059	DFLT	48333 84/12/17 JLB	PCS PCD DATA	DELETED 001
DG ELE - PCS70102,140701,E,1062	DFLT	48333 84/12/17 JLB	PCS PCD DATA	ADDED 001
DG ELE - PCS70102,140701,E,1062	DFLT	48333 84/12/17 JLB	PCS PCD DATA	DELETED 001
DG ELE - PCS70103,140701,E,1064	DFLT	48333 84/12/17 JLB	PCS PCD DATA	ADDED 001
DG ELE - PCS70103,140701,E,1064	DFLT	48333 84/12/17 JLB	PCS PCD DATA	DELETED 001
DG ELE - PCS70104,140701,E,1065	DFLT	48333 84/12/17 JLB	PCS PCD DATA	ADDED 001
DG ELE - PCS70104,140701,E,1065	DFLT	48333 84/12/17 JLB	PCS PCD DATA	DELETED 001
DG ELE - PCS70105,140701,E,1066	DFLT	48333 84/12/17 JLB	PCS PCD DATA	ADDED 001
DG ELE - PCS70105,140701,E,1066	DFLT	48333 84/12/17 JLB	PCS PCD DATA	DELETED 001
DG ELE - PCS70106,140701,E,1068	DFLT	48333 84/12/17 JLB	PCS PCD DATA	ADDED 001
DG ELE - PCS70106,140701,E,1068	DFLT	48333 84/12/17 JLB	PCS PCD DATA	DELETED 001
DG ELE - PCS70107,140701,E,1073	DFLT	48333 84/12/17 JLB	PCS PCD DATA	ADDED 001
DG ELE - PCS70107,140701,E,1073	DFLT	48333 84/12/17 JLB	PCS PCD DATA	DELETED 001
DG ELE - PCS70108,140701,E,1074	DFLT	48333 84/12/17 JLB	PCS PCD DATA	ADDED 001



Umbrella Programming

Change Control Reports

DG ELE - PCS70108,140701,E,1074	DFLT	48333	84/12/17	JLB	PCS	PCD	DATA	DELETED	001
DG ELE - PCS70109,140701,E,1075	DFLT	48333	84/12/17	JLB	PCS	PCD	DATA	ADDED	001
DG ELE - PCS70109,140701,E,1075	DFLT	48333	84/12/17	JLB	PCS	PCD	DATA	DELETED	001
DG ELE - PCS70110,140701,E,1076	DFLT	48333	84/12/17	JLB	PCS	PCD	DATA	ADDED	001
DG ELE - PCS70110,140701,E,1076	DFLT	48333	84/12/17	JLB	PCS	PCD	DATA	DELETED	001
DG ELE - PCS70111,140701,E,1077	DFLT	48333	84/12/17	JLB	PCS	PCD	DATA	ADDED	001
DG ELE - PCS70111,140701,E,1077	DFLT	48333	84/12/17	JLB	PCS	PCD	DATA	DELETED	001
DG ELE - PCS70111,140701,E,1235	DFLT	48333	84/12/17	JLB	PCS	PCD	DATA	ADDED	001
DG ELE - PCS70111,140701,E,1235	DFLT	48333	84/12/17	JLB	PCS	PCD	DATA	DELETED	001
DG ELE - PCS70112,140701,E,1079	DFLT	48333	84/12/17	JLB	PCS	PCD	DATA	ADDED	001
DG ELE - PCS70112,140701,E,1079	DFLT	48333	84/12/17	JLB	PCS	PCD	DATA	DELETED	001
DG ELE - PCS70113,140701,E,1080	DFLT	48333	84/12/17	JLB	PCS	PCD	DATA	ADDED	001
DG ELE - PCS70113,140701,E,1080	DFLT	48333	84/12/17	JLB	PCS	PCD	DATA	DELETED	001
DG ELE - PCS70114,140701,E,1046	DFLT	48333	84/12/17	JLB	PCS	PCD	DATA	ADDED	001
DG ELE - PCS70114,140701,E,1046	DFLT	48333	84/12/17	JLB	PCS	PCD	DATA	DELETED	001

Notes:



Umbrella Programming

Change Control Reports

JCL for Generating the Change Control Reports

The types of information that can be pulled in a change control report are extensive. Essentially, any field on the master record can be used as a key for generating the report.

The following sample JCL was used to generate the report on a typical class's work. It is a reference for the structure of the JCL and example selection criteria.

```
//JLBCCJOB JOB (HOGN,002,5050),'CCS JOB          ',  
//           MSGCLASS=A,  
//           NOTIFY=JLB,           STD TSO NOTIFY  
//           TIME=(0,50),          ABORT AFTER MM,SS SECONDS  
//           REGION=1500K          USE 1.5 MEG  
//P$$LIB  JCLLIB ORDER=xxxx.PROCLIB  
//*****  
//-----*  
//*-----*  
//JS010  EXEC HGNBPEM  
//CCMSTR  DD DSN=xxxV.CCMSTR,DISP=SHR  
//CCDETL  DD DSN=xxxV.CCDETL,DISP=SHR  
//CCMOUT  DD UNIT=SYSDA,SPACE=(CYL,(5,1),RLSE),DCB=BLKSIZE=4210  
//CCMIN   DD DSN=*.CCMOUT,DISP=(OLD,DELETE),VOL=REF=*.CCMOUT  
//SORTWK01 DD SPACE=(CYL,(2,1)),UNIT=SYSDA  
//SORTWK02 DD SPACE=(CYL,(2,1)),UNIT=SYSDA  
//SORTWK03 DD SPACE=(CYL,(2,1)),UNIT=SYSDA  
//SYSIN   DD *  
1 59 4  
#5400@TRNG@R59040@SELECT@%  
  SELECT ASSIGNED=JLB  
    END  
1 59 4  
#5400@TRNG@R59049@%  
  SORT=ASSIGNED,PRIO,CTL#  
  END  
1 59 4  
#5400@TRNG@R59051@MATRIX@NOBREAK@NODETAIL@@PROB@%  
1 59 4  
#5400@TRNG@R59051@NOMATRIX@BREAK@DETAIL@@PROB@%  
  BREAK=ASSIGNED,PRIO  
  END
```

Notes:



Select, Sort, and Print

In the JCL, there are several files referenced. Both CCMSTR and CCDETL are included. CCMOUT and CCMIN are work files for reading and writing records.

Note that there are three processes in producing the report: selection, sort, and print. The three step process can be done in one job step. Transaction 1 59 4 is issued with control cards for each process.

In the select process illustrated here, the change control records that are to be chosen are those assigned to JLB. Select and assigned are the key words. Selection is a time consuming process as the entire change control master file is scanned.

For the sort, the order will be change control number by priority within signon. The sort work files are specified in the JCL.

There are actually two reports printed. The first one provides a summary of master records and the product matrix. The second one summarizes the detail records and breaks the information by priority within signon.

Please refer to the *Umbrella System Change Control Manual* for complete details and examples.

Notes:



Umbrella Programming

Problem Specifications—Change Control



Problem Specifications—Change Control

The following steps will walk you through the required Umbrella screens to establish your change control numbers.

The change control number available for your use is 9990xx where xx is your group number.

1. From a cleared screen enter the transaction code U.
2. Select option 3 from the "Umbrella System Master Menu".
3. Select option 1, change control maintenance.
4. Add your change control number based on the following:
 - A. COMMAND ==> add
 - B. SITE ==> trng
 - C. CHANGE CONTROL # ==> 9990xx, where xx is your group number.
 - D. DATA ==> m
 - E. APP ==> ZZ?, where ? is the classroom
 - F. PRIORITY ==> ____ Your choice: low, med, or high
 - G. REASON ==> ____ Your choice new development, deficiency, enhancement, custom or wish.
 - H. DESCRIPTION ==> _____ Be creative, but include your group number xx.
 - I. MATERIALS ==> UMB MANUALS and CLASS MANUAL
 - J. TARGET DATE ==> CLASS ENDING DATE (yr/mo/da)
 - K. ASSIGNED TO ==> ?xx, where ? is the classroom and xx is your group number.
 - L. MANDAYS ==> 10.0
 - M. DATE CLOSED==> 00/00/00
 - N. PRESS ENTER



Umbrella Programming

Problem Specifications—Change Control

5. On the problem screen returned to you from Step 4 above, key narrative information and press ENTER. Later in the week when you start using the change control numbers, you may wish to update the comments on this screen.

Later, during this class, Change Control reports will be run to summarize the updates performed by each student group during the week. The reports will illustrate the type of information that can be pulled from the change control system to monitor your system. During the discussion of change control reports, additional details will be provided.

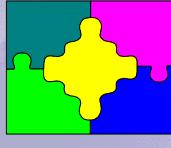
Notes:



Summary



Summary



© 2000 CSC. All rights reserved.

- Change Control System Reports
- JCL structure
- CCS
- Online screens
- Batch jobs
- Central point for managing change control

- There are multiple reports available through the Change Control System.
- The reports can help manage changes, document custom modifications, produce an audit trail, and manage staff.
- The structure of the JCL to provide the reports includes three processes: selection, sort, and print.
- CCS records information about the change.
- CCS records the before and after images that result from a change.
- The changes processed are those to source and CDMF items.
- CCS requires a master item that is keyed by site and change control number. Each change control number is authorized for a specific application and can be used by other applications within the same family.
- Hogan online screens require an ownership and change control number for updates.
- An input exit monitors source changes through one of four source managers (LIB, PAN, EDV, or IEB).



Umbrella Programming

Summary

- CCMSTR data base consists of a master base data group and several optional dependent data groups (problem, resolution, variance, and fix).
- CCDETL houses the detail records.
- There are online screens for maintaining CCS records.
- Batch jobs are run at night to record changes made during the day.
- A central point for managing change control is recommended.

Notes:



Umbrella Programming

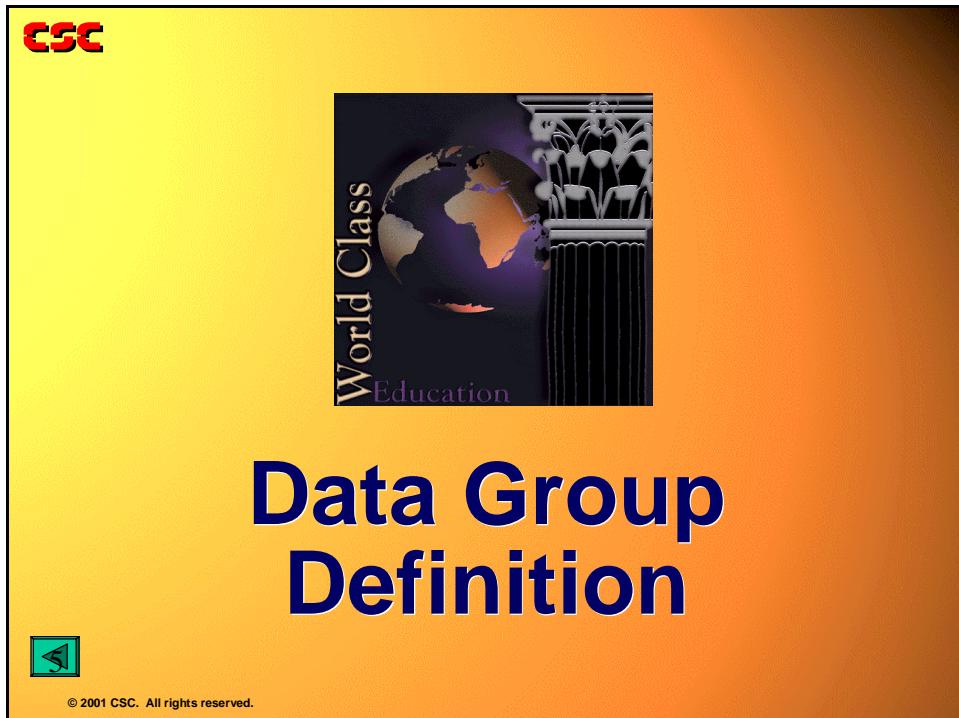
Summary



Data Group Definition

6

Purpose



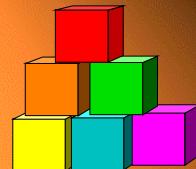
Notes:



Topics

CSC

Topics



- ▶ Data groups in Umbrella scheme of things
- ▶ Data group definitions
- ▶ Adding data groups
- ▶ Data element definitions

© 2000 CSC. All rights reserved.

Objectives

CSC

Objectives



- Identify components of data group definition
- Add a data group definition
- Access a data element definition
- List types of data groups that may occur on a data base definition

© 2000 CSC. All rights reserved.



Data Groups

CSC

Data Groups

- Like a COBOL Copybook
- Defines Contiguous Storage
- Fields make up Data Group

© 2000 CSC. All rights reserved.

The number one rule about data under Hogan is that a field must be part of a data group that is defined to the Process Dictionary. This concept forms the foundation for data handling by PEM.

Data groups can be collections of work fields for totalling, setting flags, or one of an endless variety of possibilities. For example, the print line under Hogan is a data group. Data bases are logically defined through a data group or set of related data groups.

Data group definitions are used to define any memory space that is needed by a program, whether that space is in CPU memory (temporary space) or in file memory (permanent space).

Notes:



Umbrella Programming

Data Groups

Data Group Fields



Data Group Fields

- Work Fields
- Print Line
- Terminal I/O Area
- Part of a Data Base
- Defined to the Process Dictionary
- First two fields
 - Action Field (2 bytes binary)
 - Result Field (2 bytes binary)

© 2000 CSC. All rights reserved.

PEM requires that data, other than constants, be defined as fields within data groups. Each data group is a logical collection of related fields. They can be work areas for user programs, parts of data base records, data fields for terminal I/O, and print lines. Data groups are normally defined on the Process Dictionary at application design time.

Although the definition of the fields within a data group can be determined by the requirements of the application, PEM does place one requirement on the format of a data group. The first four bytes of each data group are reserved for two, two bytes binary fields.

The first of these fields is called ACTION; the second is RESULT. The action field is used to tell PEM what is to be done with this data group and its contents. An example is a request for a read from a data base. The RESULT field is used by PEM to tell the application program the result of the requested ACTION. Example results passed to the application program through this field include X'0000' if the read is DGR-OK or X'0001' if DGR-END-DATA has been reached.



EXAMPLE OF COBOL DATA GROUP FIELDS

```
01  DATA-GROUP-11.  
    02  DG-ACTION-FIELD          PIC XX.  
    02  DG-RESULT-FIELD         PIC XX.  
    02  DG-FIRST-USER-FIELD     PIC ?.  
    ...  
    ...  
    ...
```

In COBAL Data Group definition, the ACTION and RESULT fields are always defined as PIC XX for reasons which will be explained later in the class.

EXAMPLE OF ALC DATA GROUP FIELDS

DATAGP11	DS	0F	
DGACTION	DS	XL2	ACTION FIELD
DGRESULT	DS	XL2	RESULT FIELD
FSTUSER	DS	?	FIRST USER FIELD
...			
...			
...			

To help you use and interpret the action and result fields, data group action and result copybooks are delivered with the Umbrella System. You can include these modules, P49022D for the Data Group Action Codes COBOL Copybook and P49023D for the Data Group Result Codes COBOL Copybook, in your COBOL application programs. Both modules have been placed on the following pages for your reference.

Notes:



Umbrella Programming

Data Groups

Data Group Action Codes COBOL Copybook P49022D

```
*  
*--* START OF P49022D *----* PEM ACTION CODES *-----*  
*  
01  PEM-DATA-GROUP-ACTION-CODES.  
    05  DGA-CODE-VALUES.  
    10  DGA-NO-OP.  
        15  FILLER PIC S9(4) COMP VALUE +0000.  
    10  DGA-READ.  
        15  FILLER PIC S9(4) COMP VALUE +0001.  
    10  DGA-WRITE.  
        15  FILLER PIC S9(4) COMP VALUE +0002.  
    10  DGA-HOLD-P.  
        15  FILLER PIC S9(4) COMP VALUE +0003.  
    10  DGA-ERASE.  
        15  FILLER PIC S9(4) COMP VALUE +0004.  
    10  DGA-READ-KEY-GE.  
        15  FILLER PIC S9(4) COMP VALUE +0005.  
    10  DGA-FORCE-WRITE.  
        15  FILLER PIC S9(4) COMP VALUE +0006.  
    10  DGA-READ-KEY-EQ.  
        15  FILLER PIC S9(4) COMP VALUE +0007.  
    10  DGA-INSERT-FIRST.  
        15  FILLER PIC S9(4) COMP VALUE +0008.  
    10  DGA-READ-LAST-REC.  
        15  FILLER PIC S9(4) COMP VALUE +0009.  
    10  DGA-INSERT-LAST.  
        15  FILLER PIC S9(4) COMP VALUE +0010.  
    10  DGA-READ-FIRST.  
        15  FILLER PIC S9(4) COMP VALUE +0011.  
    10  DGA-INSERT-HERE.  
        15  FILLER PIC S9(4) COMP VALUE +0012.  
    10  DGA-END-REQUEST.  
        15  FILLER PIC S9(4) COMP VALUE +0014.  
    10  DGA-FORCE-NO-OP.  
        15  FILLER PIC S9(4) COMP VALUE +0016.  
    10  DGA-CLOSE.  
        15  FILLER PIC S9(4) COMP VALUE +0036.  
    10  DGA-OPEN-OUTPUT.  
        15  FILLER PIC S9(4) COMP VALUE +0037.  
    10  DGA-OPEN-INPUT.  
        15  FILLER PIC S9(4) COMP VALUE +0038.  
    10  DGA-OPEN-UPDATE.  
        15  FILLER PIC S9(4) COMP VALUE +0039.  
    10  DGA-POINT.  
        15  FILLER PIC S9(4) COMP VALUE +0040.  
    10  DGA-ENABLE-KEY-RANGE.  
        15  FILLER PIC S9(4) COMP VALUE +0041.  
    10  DGA-DISABLE-KEY-RANGE.  
        15  FILLER PIC S9(4) COMP VALUE +0042.  
    10  DGA-ENABLE-MULT-KRDB.  
        15  FILLER PIC S9(4) COMP VALUE +0043.  
*  
*----* END OF P49022D *-----*
```



Data Group Result Codes COBOL Copybook P49023D

```
*--* START OF P49023D *----* PEM RESULT CODES *-----*
*
01  PEM-DATA-GROUP-RESULT-CODES.
05  DGR-CODE-VALUES.
10  DGR-OK.
15  FILLER PIC S9(4) COMP VALUE +0000.
10  DGR-END-DATA.
15  FILLER PIC S9(4) COMP VALUE +0001.
10  DGR-I-O-ERR.
15  FILLER PIC S9(4) COMP VALUE +0002.
10  DGR-DUP-KEY.
15  FILLER PIC S9(4) COMP VALUE +0006.
10  DGR-NO-FIND.
15  FILLER PIC S9(4) COMP VALUE +0007.
10  DGR-SKEY-NO-FIND.
15  FILLER PIC S9(4) COMP VALUE +0008.
10  DGR-KRID-DISABLED.
15  FILLER PIC S9(4) COMP VALUE +0009.
10  DGR-DB-NOT-AVAIL.
15  FILLER PIC S9(4) COMP VALUE +0010.
10  DGR-SQLCA-ERROR.
15  FILLER PIC S9(4) COMP VALUE +0013.
10  DGR-LEN-ERR.
15  FILLER PIC S9(4) COMP VALUE +0014.
*
*----* END OF P49023D *-----*
```

Notes:



Umbrella Programming

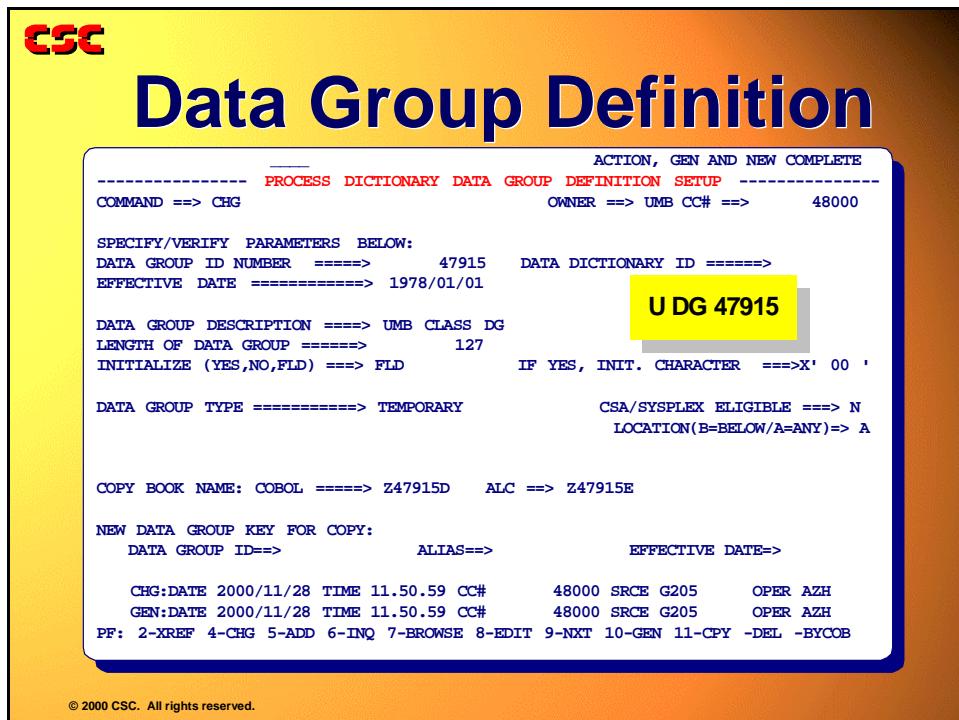
Data Groups

Data Group Definition

In this section we will examine the data group definition screens and interpret the information available on them. Data group 47915 will be used for this walk through.

The "Process Dictionary Data Group Definition Setup" screen shown below can be accessed three ways:

1. Through the "Umbrella System Master Menu" to the "Umbrella System Process Dictionary Maintenance Menu" to "Process Dictionary Data Group Definition Setup" and INQ on the data group ID 47915
2. With the jump facility: =DG 47915
3. From a cleared screen enter U DG 47915 or U 1.1.47915.



To view the field definitions contained within the data group press PF7 or enter **BROWSE** into the COMMAND field.



Data Group Field Inquiry/Maintenance

CSC

Data Group Field Inquiry/Maintenance

```
ACTION COMPLETE
----- DATA GROUP DEFINITION INQUIRY / MAINTENANCE -----
COMMAND INPUT ==> HP                                     CC# ==> 0
DATA GROUP ID ==> 0000047915 EFF DATE==> 78/01/01 COLS: 9 80 MODE ==> BRWS
***** -1-----2-----3-----4-----5-----6-----7-----8
000001 1   W915-ScreenText-DATA-GROUP.
000003 **      THIS DATA GROUP IS DESIGNED AS A SAMPLE FOR
000004 **      DISCUSSION DURING THE PFM CONCEPTS LECTURE
000005 **      IN CONCEPTS AND FACILITIES CLASS.
000006 05 W915-ACTION          PIC XX.
000008 05 W915-RESULT          PIC XX.
000010 05 W915-COMPANY         PIC XX.
000012 05 W915-COMPANY-COMP REDEFINES W915-COMPANY          TQE91501
000013                               PIC S9(4)           COMP.
000015 05 W915-EMPLOYEE-NUMBER    PIC S9(11)          COMP-3.      TQE91502
000017 05 W915-EMPLOYEE-NAME-FIRST  PIC X(015).        TQE91503
000019 05 W915-EMPLOYEE-NAME-LAST  PIC X(015).        TQE91504
000021 05 W915-SEX              PIC X(001).        TQE91505
000023     88 W915-MALE          VALUE 'M'.
000024     88 W915-FEMALE         VALUE 'F'.
000025 05 W915-CURRENT-DEDUCTIONS OCCURS 12 TIMES.      TQE915AA
000026
***** -1-----2-----3-----4-----5-----6-----7-----8
PF: 4-SAVE 5-HP 6-BROWSE 7-SB 8-SF 9-EDIT 10-SL 11-SR ..EID ..LTYP ..CANCEL
```

© 2000 CSC. All rights reserved.

To return to the Base screen enter **PLVL** in the COMMAND field and press PF3.

Notes:



Umbrella Programming

Data Groups

Display of PEM Internal Element ID

To display the PEM internal Element ID enter EID into the COMMAND INPUT field.



PEM Internal Element ID

ACTION COMPLETE

```
----- DATA GROUP DEFINITION INQUIRY / MAINTENANCE -----
- COMMAND INPUT ==> HP CC# ==>
0 DATA GROUP ID ==> 0000047915 EFF DATE==> 78/01/01 COLS: 9 80 MODE ==>
BRWS *****-1-----2-----3-----4-----5-----6-----+
---7-- 000001 E00001 01 W915-SCREEN TEXT-DATA-GROUP.
000003 E00019**** THIS DATA GROUP IS DESIGNED AS A SAMPLE FOR
000004 E00020**** DISCUSSION DURING THE PEM CONCEPTS LECTURE
000005 E00021**** IN CONCEPTS AND FACILITIES CLASS.
000006 E00002 05 W915-ACTION PIC XX.
000008 E00003 05 W915-RESULT PIC XX.
000010 E00004 05 W915-COMPANY PIC XX.
000012 E00005 05 W915-COMPANY-COMP REDEFINES W915-COMPANY
000013 E00005          PIC S9(4)      COMP.
000015 E00006 05 W915-EMPLOYEE-NUMBER PIC S9(11)    COMP-3.
000017 E00007 05 W915-EMPLOYEE-NAME-FIRST PIC X(015).
000019 E00008 05 W915-EMPLOYEE-NAME-LAST PIC X(015).
000021 E00009 05 W915-SEX PIC X(001).
000023 E00017     88 W915-MALE    VALUE 'M'.
000024 E00018     88 W915-FEMALE   VALUE 'F'.
000025 E00010 05 W915-CURRENT-DEDUCTIONS
000026 E00010          OCCURS 12 TIMES.
*****-1-----2-----3-----4-----5-----6-----7-
PF: 4-SAVE 5-HP 6-BROWSE 7-SB 8-SF 9-EDIT 10-SL 11-SR ..-EID ..-LTYP ..-CANCEL
```

EID

Reset - Returns to normal Browse/Edit Screen

© 2001 CSC. All rights reserved.

To return to the original display, enter **RESET** in the COMMAND field and press ENTER.

Notes:



Display of the Data Group Line Types

To display the various Data Group Line Types enter **LTYP** in the COMMAND field and press ENTER.

CSC

Data Group Line Types

ACTION COMPLETE

----- DATA GROUP DEFINITION INQUIRY / MAINTENANCE -----

```
- COMMAND INPUT ==> HP                               CC# ==>
1 DATA GROUP ID ==> 0000047915 EFF DATE==> 78/01/01 COLS: 9 80 MODE ==>
BRWS ***** 1---+---2---+---3---+---4---+---5---+---6---+
----7-- 000001 COBOL> 01 W915-SCREENTEXT-DATA-GROUP.
000003 DATA=>***** THIS DATA GROUP IS DESIGNED AS A SAMPLE FOR
000004 DATA=>***** DISCUSSION DURING THE PEM CONCEPTS LECTURE
000005 DATA=>***** IN CONCEPTS AND FACILITIES CLASS.
000006 COBOL> 05 W915-ACTION          PIC XX.
000008 COBOL> 05 W915-RESULT          PIC XX.
000010 COBOL> 05 W915-COMPANY         PIC XX.
000012 COBOL> 05 W915-COMPANY-COMP REDEFINES W915-COMPANY
000013 CONT=>                           PIC S9(4)      COMP.
000015 COBOL> 05 W915-EMPLOYEE-NUMBER    PIC S9(11)     COMP-3.
000017 COBOL> 05 W915-EMPLOYEE-NAME-FIRST  PIC X(015).
000019 COBOL> 05 W915-EMPLOYEE-NAME-LAST   PIC X(015).
000021 COBOL> 05 W915-SEX              PIC X(001).
000023 DATA=>          88 W915-MALE      VALUE 'M'.
000024 DATA=>          88 W915-FEMALE    VALUE 'F'.
000025 COBOL> 05 W915-CURRENT-DEDUCTIONS
000026 CONT=>          OCCURS 12 TIMES.
***** 1---+---2---+---3---+---4---+---5---+---6---+---7--
PF: 4-SAVE 5-HP 6-BROWSE 7-SB 8-SF 9-EDIT 10-SL 11-SR ..-EID ..-LTYP ..-CANCEL
```

Reset - Returns to normal Browse/Edit Screen

© 2001 CSC. All rights reserved.

To return to the original display enter **RESET** in the COMMAND field and press ENTER.

Notes:



Umbrella Programming

Data Groups

Display of PEM Dictionary Characteristic Lines

To display the PEM Dictionary Characteristic Lines enter SHOW in the COMMAND field and press ENTER.

The screenshot shows a terminal window titled "Display Dictionary Characteristics Lines". The title bar has the CSC logo. The window contains a list of data group definitions in a structured format. A yellow button labeled "SHOW" is overlaid on the right side of the window. At the bottom left, there is a copyright notice: "© 2001 CSC. All rights reserved." The data listed includes fields like COMMAND, DATA GROUP ID, EFF DATE, COLS, and various parameters (LN, OF, TY, FLG, SD) for different data elements (W915-ACTION, W915-RESULT, W915-COMPANY, etc.).

```
ACTION COMPLETE
----- DATA GROUP DEFINITION INQUIRY / MAINTENANCE -----
- COMMAND INPUT ===> HP CC# ==>
0 DATA GROUP ID ===> 0000047915 EFF DATE==> 78/01/01 COLS: 9 80 MODE ==>
BRWS *****-1-----2-----3-----4-----5-----6-----7-----
-----8 00001 1 W915-SCREENTEXT-DATA-GROUP.
00002 C 00001,LN=00127,OF=00000,TY=C,FLG>NN ,SD=SCREENTEXT-DATA-GROUP ,
00003 ** THIS DATA GROUP IS DESIGNED AS A SAMPLE FOR
00004 ** DISCUSSION DURING THE PEM CONCEPTS LECTURE
00005 ** IN CONCEPTS AND FACILITIES CLASS.
00006 05 W915-ACTION PIC XX.
00007 C 00002,LN=00002,OF=00000,TY=B,FLG>NN ,SD=W915-ACTION ,
00008 05 W915-RESULT PIC XX.
00009 C 00003,LN=00002,OF=00002,TY=B,FLG>NN ,SD=W915-RESULT ,
00010 05 W915-COMPANY PIC XX.
00011 C 00004,LN=00002,OF=00004,TY=C,FLG=YY ,SD=W915-COMPANY ,
00012 05 W915-COMPANY-COMP REDEFINES W915-COMPANY
00013 PIC S9(4) COMP.
00014 C 00005,LN=00002,OF=00004,TY=B,FLG=YY ,SD=W915-COMPANY-COMP ,
00015 05 W915-EMPLOYEE-NUMBER PIC S9(11) COMP-3.
00016 C 00006,LN=00006,OF=00006,TY=P,FLG=YY ,SD=W915-EMPLOYEE-NUMBER,
00017 05 W915-EMPLOYEE-NAME-FIRST PIC X(015).
***** -----1-----2-----3-----4-----5-----6-----7-----
PF: 4-SAVE 5-HP 6-BROWSE 7-SB 8-SF 9-EDIT 10-SL 11-SR ..-EID ..-LTYP ..-CANCEL
```

To return to the original display enter **RESET** in the COMMAND field.

This display starts in column 9 only showing the C of DC. PF10 and PF11 can be used to shift the display to the left or right.

The DC lines contain the following information:

LN	Length of field.
OF	Offset of field within the data group.
TY	Field Type. C=Character, P=Packed, B=Binary.
FLG	Four position parameter field. <ul style="list-style-type: none">• Field Initialization/Batch Deblock flag.• Not used.• Not used.• Not used.
SD	20 position short description created from the COBOL name.
Columns 73-80	Data Element name



Converted (Pre-CDMF) Data Group Field Definition

Pre-CDMF did not support COBOL like data group definitions. A pre-CDMF Data Group Fields definition is displayed below.

CSC **Result: Locating by Data Group Field Sequence Number**

ACTION COMPLETE					
----- DATA GROUP DEFINITION INQUIRY / MAINTENANCE -----					
COMMAND INPUT ==> HP					
DATA GROUP ID ==> 0000047915 EFF DATE==> 78/01/01 COLS: 9 80 MODE ==> BRWS					
***** -1-----2-----3-----4-----5-----6-----7-----8					
000015	05	W915-EMPLOYEE-NUMBER	PIC S9(11)	COMP-3.	TQE91502
000017	05	W915-EMPLOYEE-NAME-FIRST	PIC X(015).		TQE91503
000019	05	W915-EMPLOYEE-NAME-LAST	PIC X(015).		TQE91504
000021	05	W915-SEX	PIC X(001).		TQE91505
000023	88	W915-MALE	VALUE 'M'.		
000024	88	W915-FEMALE	VALUE 'F'.		
000025	05	W915-CURRENT-DEDUCTIONS			TQE915AA
000026			OCCURS 12 TIMES.		
000029	10	W915-DED-CODE	PIC X(002).		
000032	10	W915-DED-AMOUNT	PIC S9(7)V99	COMP-3.	
***** E N D O F D A T A G R O U P *****					

L 15

© 2000 CSC. All rights reserved.

Pre-CDMF or Converted Data Group Definitions contain only PEM Dictionary Characteristics lines.

LN	Length of field.
OF	Offset of field within the data group.
TY	Field Type. C=Character, P=Packed, B=Binary.
FLG	Four position parameter field.
	<ul style="list-style-type: none"> • Field Initialization/Batch Deblock flag. • Not used. • Not used. • Not used.
SD	20 position short field description created from the COBOL name.
Columns 73-80	Data Element name



Umbrella Programming

Data Groups

Locating By Data Group Field Sequence Number

The Locate (L) command will position the Data Group Definition Inquiry Maintenance Screen to the desired line/sequence number.

To display line number 15, enter L 15 into the COMMAND INPUT field.

The screenshot shows a yellow-tinted terminal window with the CSC logo at the top left. The title bar reads "Locating by Data Group Field Sequence Number". The main area displays a data group definition with sequence numbers and fields. A yellow box highlights the command input field containing "L nn".

```
ACTION COMPLETE
----- DATA GROUP DEFINITION INQUIRY / MAINTENANCE -----
COMMAND INPUT ==> L 15          CC# ==> 0
DATA GROUP ID ==> 0000047915  EFF DATE==> 78/01/01 COLS: 9 80 MODE ==> BRWS
***** -1---+--2---+--3---+--4---+--5---+--6---+--7---+--8
000001 1 W915-ScreenWidth-DATA-GROUP.
000003 ** THIS DATA GROUP IS DESIGNED AS A SAMPLE FOR
000004 ** DISCUSSION DURING THE PBM CONCEPTS LECTURE
000005 ** IN CONCEPTS AND FACILITIES CLASS.
000006 05 W915-ACTION          PIC XX.
000008 05 W915-RESULT          PIC XX.
000010 05 W915-COMPANY         PIC XX.
000012 05 W915-COMPANY-COMP REDEFINES W915-COMPANY          TQE91501
000013          PIC S9(4)      COMP.
000015 05 W915-EMPLOYEE-NUMBER PIC S9(11)     COMP-3.          TQE91502
000017 05 W915-EMPLOYEE-NAME-FIRST PIC X(015).          TQE91503
000019 05 W915-EMPLOYEE-NAME-LAST PIC X(015).          TQE91504
000021 05 W915-SEX             PIC X(001).          TQE91505
000023          88 W915-MALE    VALUE 'M'.
000024          88 W915-FEMALE   VALUE 'F'.
000025 05 W915-CURRENT-DEDUCTIONS          TQE915AA
000026          OCCURS 12 TIMES.
***** -1---+--2---+--3---+--4---+--5---+--6---+--7---+--8
PF: 4-SAVE 5-HP 6-BROWSE 7-SB 8-SF 9-EDIT 10-SL 11-SR ..-EID ..-LTYP ..-CANCEL
```

© 2000 CSC. All rights reserved.

Notes:



The resulting screen displays line number/sequence number --- 15.

CSC Result: Locating by Data Group Field Sequence Number

ACTION COMPLETE

```
----- DATA GROUP DEFINITION INQUIRY / MAINTENANCE -----
COMMAND INPUT ==> HP                                     ACTION COMPLETE
DATA GROUP ID ==> 0000047915 EFF DATE==> 78/01/01 COLS: 9 80 MODE ==> BRWS
***** -1-----2-----3-----4-----5-----6-----7-----8
000015 05 W915-EMPLOYEE-NUMBER          PIC S9(11)      COMP-3.      TQE91502
000017 05 W915-EMPLOYEE-NAME-FIRST     PIC X(015).    TQE91503
000019 05 W915-EMPLOYEE-NAME-LAST      PIC X(015).    TQE91504
000021 05 W915-SEX                   PIC X(001).    TQE91505
000023   88 W915-MALE                 VALUE 'M'.
000024   88 W915-FEMALE                VALUE 'F'.
000025 05 W915-CURRENT-DEDUCTIONS    OCCURS 12 TIMES.
000026
000029   10 W915-DED-CODE            PIC X(002).
000032   10 W915-DED-AMOUNT          PIC S9(7)V99      COMP-3.
***** END OF DATA GROUP *****
```

L 15

© 2000 CSC. All rights reserved.

Notes:

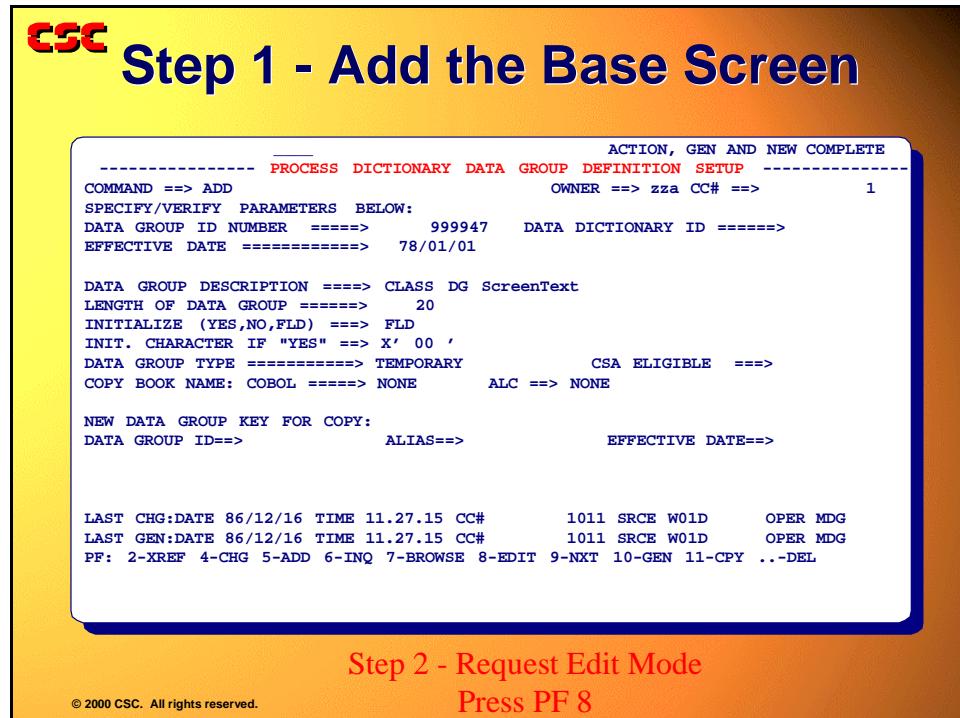


Umbrella Programming

Data Groups

Defining the Data Group

Step 1—Add the Base Screen



Step 2—Request Edit Mode

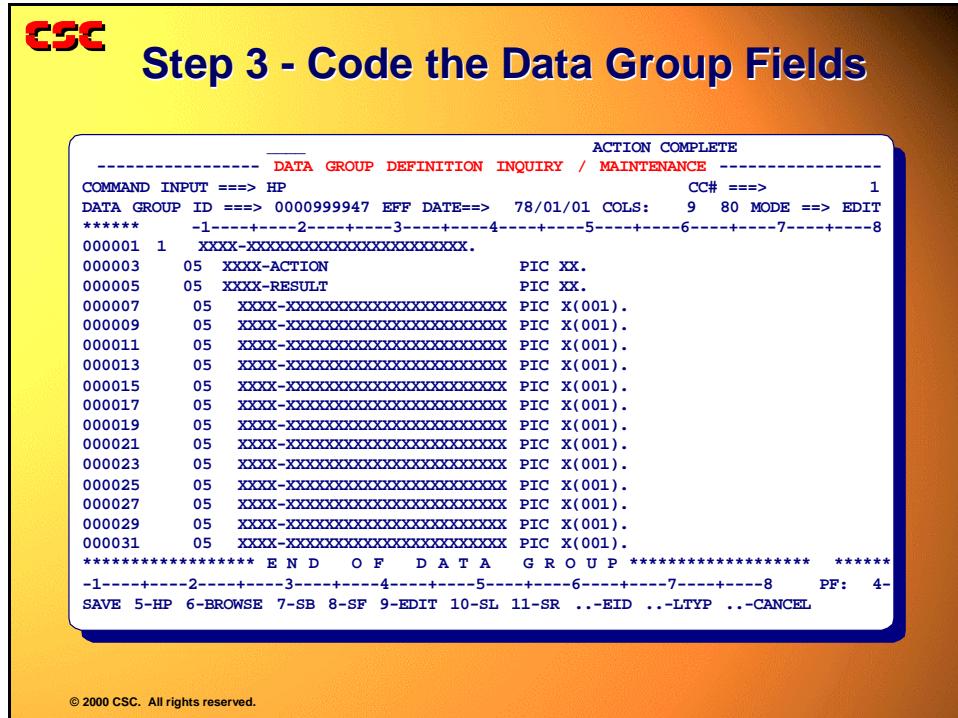
- Press PF8
- OR
- Enter **EDIT** into the COMMAND field.

Notes:



Step 3—Code the Data Group Fields

All lines initially displayed are COBOL Primary Lines. Only COBOL level and field definitions may be coded on this line type. Overkey the Xs to enter the COBOL field definitions. The edit mode is full screen. Terminal keys to delete and insert characters will work on this display. Press ENTER to record your statements.



Data Group Field Definition Line Types

LINES TYPES:

1. COBOL
2. PEM Dictionary

COBOL Line Types:

1. Primary Lines

Contain the level and group or element name.

2. Continuation Lines

Contain the lines for extension of a data group field definition to more than one line.

3. Data Lines

Contain information about data group field definitions, that is, comments and 88 level statements.



Umbrella Programming

Data Groups

Inserting the Different Line Types

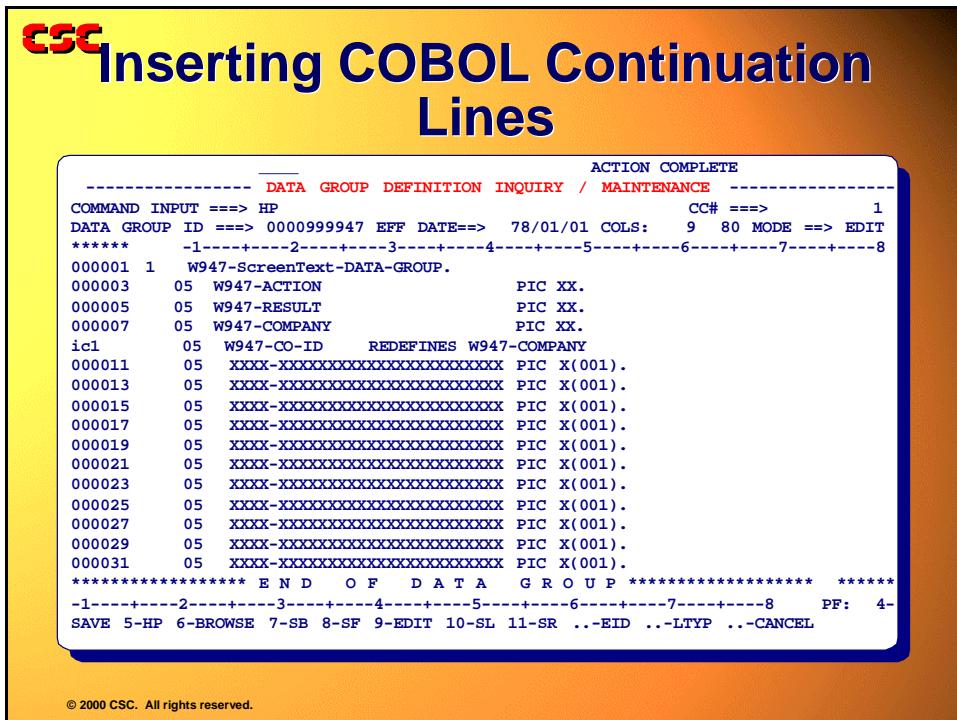
Overkey the sequence line number with one of the following:

In - Insert COBOL Primary Lines
ICn - Insert COBOL Continuation Line
IDn - Insert COBOL Data Lines

(n = The number of lines to be inserted)

The line(s) are inserted after the line containing the command.

Inserting COBOL Continuation Lines



The screenshot shows a yellow header bar with the CSC logo and the title "Inserting COBOL Continuation Lines". Below this is a blue-bordered window titled "ACTION COMPLETE". Inside the window, there is a command input area and a data group definition area. The command input area contains:
COMMAND INPUT ==> HP
DATA GROUP ID ==> 0000999947 EFF DATE==> 78/01/01 COLS: 9 80 MODE ==> EDIT
***** -1---+--2---+--3---+--4---+--5---+--6---+--7---+--8
000001 1 W947-ScreenWidth-DATA-GROUP.
000003 05 W947-ACTION PIC XX.
000005 05 W947-RESULT PIC XX.
000007 05 W947-COMPANY PIC XX.
ic1 05 W947-CO-ID REDEFINES W947-COMPANY
000011 05 XXXX-XXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000013 05 XXXX-XXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000015 05 XXXX-XXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000017 05 XXXX-XXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000019 05 XXXX-XXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000021 05 XXXX-XXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000023 05 XXXX-XXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000025 05 XXXX-XXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000027 05 XXXX-XXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000029 05 XXXX-XXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000031 05 XXXX-XXXXXXXXXXXXXXXXXXXXXX PIC X(001).
***** END OF DATA GROUP *****
-1---+--2---+--3---+--4---+--5---+--6---+--7---+--8 PF: 4-
SAVE 5-HP 6-BROWSE 7-SB 8-SF 9-EDIT 10-SL 11-SR ..-EID ..-LTYP ..-CANCEL

At the bottom left of the window, it says "© 2000 CSC. All rights reserved."

Notes:



Result: Inserting COBOL Continuation Lines

CSC

Result: Inserting COBOL Continuation Lines

```

----- ACTION COMPLETE -----
----- DATA GROUP DEFINITION INQUIRY / MAINTENANCE -----
COMMAND INPUT ==> HP                               CC# ==>      1
DATA GROUP ID ==> 0000999947 EFF DATE==> 78/01/01 COLS:   9  80 MODE ==> EDIT
*****   -1---+---2---+---3---+---4---+---5---+---6---+---7---+---8
000001  1  W947-ScreenWidth-DATA-GROUP.
000003  05  W947-ACTION                         PIC XX.
000005  05  W947-RESULT                          PIC XX.
000007  05  W947-COMPANY                        PIC XX.
000009  05  W947-CO-ID    REDEFINES W947-COMPANY
000011  COBOL CONTINUATION LINE
000012  05  XXXX-XXXXXXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000014  05  XXXX-XXXXXXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000016  05  XXXX-XXXXXXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000018  05  XXXX-XXXXXXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000020  05  XXXX-XXXXXXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000022  05  XXXX-XXXXXXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000024  05  XXXX-XXXXXXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000026  05  XXXX-XXXXXXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000028  05  XXXX-XXXXXXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000030  05  XXXX-XXXXXXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000032  05  XXXX-XXXXXXXXXXXXXXXXXXXXXXXXXX PIC X(001).
*****   -1---+---2---+---3---+---4---+---5---+---6---+---7---+---8
PF: 4-SAVE 5-HP 6-BROWSE 7-SB 8-SF 9-EDIT 10-SL 11-SR ...EID ...LTYP ...CANCEL

```

© 2000 CSC. All rights reserved.

Inserting COBOL Data Lines

CSC

Inserting COBOL Data Lines

```

----- ACTION COMPLETE -----
----- DATA GROUP DEFINITION INQUIRY / MAINTENANCE -----
COMMAND INPUT ==> HP                               CC# ==>      1
DATA GROUP ID ==> 0000999947 EFF DATE==> 78/01/01 COLS:   9  80 MODE ==> EDIT
*****   -1---+---2---+---3---+---4---+---5---+---6---+---7---+---8
id3  1  W947-ScreenWidth-GROUP.
000003  05  W947-ACTION                         PIC XX.
000005  05  W947-RESULT                          PIC XX.
000007  05  W947-COMPANY                        PIC XX.
000009  05  W947-CO-ID    REDEFINES W947-COMPANY
000011  EMPLOYEE-NUMBER                         PIC S9(4) COMP.
000012  05  W947-EMP-NUMBER                     PIC S9(4) COMP-3.
000016  05  W947-EMP-FIRST-NAME                PIC X(15).
000018  05  W947-EMP-LAST-NAME                 PIC X(15).
000020  05  W947-SEX-CODE                       PIC X(001).
000022  05  XXXX-XXXXXXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000024  05  XXXX-XXXXXXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000026  05  XXXX-XXXXXXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000028  05  XXXX-XXXXXXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000030  05  XXXX-XXXXXXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000031  05  XXXX-XXXXXXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000032  05  XXXX-XXXXXXXXXXXXXXXXXXXXXXXXXX PIC X(001).
*****   -1---+---2---+---3---+---4---+---5---+---6---+---7---+---8
PF: 4-SAVE 5-HP 6-BROWSE 7-SB 8-SF 9-EDIT 10-SL 11-SR ...EID ...LTYP ...CANCEL

```

© 2000 CSC. All rights reserved.



Umbrella Programming

Data Groups

Result: Inserting COBOL Data Lines

CSC Result: Inserting COBAL Data Lines

```
ACTION COMPLETE
----- DATA GROUP DEFINITION INQUIRY / MAINTENANCE -----
COMMAND INPUT ===> HP CC# ===> 1
DATA GROUP ID ===> 0000999947 EFF DATE==> 78/01/01 COLS: 9 80 MODE ==> EDIT
***** -1---+--2---+--3---+--4---+--5---+--6---+--7---+--8
000001 1 W947-ScreenText-DATA-GROUP.
000003 **
000004 **
000005 **
000006 05 W947-ACTION PIC XX.
000008 05 W947-RESULT PIC XX.
000010 05 W947-COMPANY PIC XX.
000012 05 W947-CO-ID    REDEFINES W947-COMPANY
000014          PIC S9(4) COMP.
000015 05 W947-EMP-NUMBER PIC S9(4) COMP-3.
000017 05 W947-EMP-FIRST-NAME PIC X(15).
000019 05 W947-EMP-LAST-NAME PIC X(15).
000021 05 W947-SEX-CODE PIC X(001).
000023 05 XXXX-XXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000025 05 XXXX-XXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000027 05 XXXX-XXXXXXXXXXXXXXXXXXXXXX PIC X(001).
000029 05 XXXX-XXXXXXXXXXXXXXXXXXXXXX PIC X(001).
***** -1---+--2---+--3---+--4---+--5---+--6---+--7---+--8
PF: 4-SAVE 5-HP 6-BROWSE 7-SB 8-SF 9-EDIT 10-SL 11-SR ..-EID ..-LTYP ..-CANCEL
```

© 2000 CSC. All rights reserved.

Notes:



Deleting Data Group Definition Lines

CSC

Deleting Data Group Definition Lines

```
ACTION COMPLETE
----- DATA GROUP DEFINITION INQUIRY / MAINTENANCE -----
COMMAND INPUT ==> HP CC# ==> 1
DATA GROUP ID ==> 0000999947 EFP DATE==> 78/01/01 COLS: 9 80 MODE ==> EDIT
***** -1-----2-----3-----4-----5-----6-----7-----8
000001 1 W947-ScreenText-DATA-GROUP.
000003 ** THE TEXT OF YOUR DATA GROUP
000004 ** DESCRIPTION INFORMATION IS
000005 ** ENTERED ON DATA LINES.
000006 05 W947-ACTION PIC XX.
000008 05 W947-RESULT PIC XX.
000010 05 W947-COMPANY PIC XX.
000012 05 W947-CO-ID REDEFINES W947-COMPANY
000014 05 W947-EMP-NUMBER PIC S9(4) COMP.
000015 05 W947-EMP-FIRST-NAME PIC X(15).
000017 05 W947-EMP-LAST-NAME PIC X(15).
000019 05 W947-SEX-CODE PIC X(001).
000021 05 W947-SEX-CODE PIC X(001).
000023 88 W947-MALE VALUE '1'.
000025 88 W947-FEMALE VALUE '2'.
d 05 XXXX-XXXXXXXXXXXXXXXXXXXX PIC X(001).
d 05 XXXX-XXXXXXXXXXXXXXXXXXXX PIC X(001).
***** -1-----2-----3-----4-----5-----6-----7-----8
PF: 4-SAVE 5-HP 6-BROWSE 7-SB 8-SF 9-EDIT 10-SL 11-SR ..EID ..LTYP ..CANCEL
```

© 2000 CSC. All rights reserved.

The D (delete command) on a COBOL primary line will delete all lines related to that COBOL primary line.

The D (delete command) on a data line, continuation line, or dictionary line will only delete that line.

Notes:



Umbrella Programming

Data Groups

Moving Data Group Definition Lines



Moving Data Group Definition Lines

```
ACTION COMPLETE
----- DATA GROUP DEFINITION INQUIRY / MAINTENANCE -----
COMMAND INPUT ==> HP CC# ==> 1
DATA GROUP ID ==> 000099947 EFP DATE==> 78/01/01 COLS: 9 80 MODE ==> EDIT
***** -1---+---2---+---3---+---4---+---5---+---6---+---7---+---8
000001 1 W947-ScreenWidth-DATA-GROUP.
000003 ** THE TEXT OF YOUR DATA GROUP
000004 ** DESCRIPTION INFORMATION IS
000005 ** ENTERED ON DATA LINES.
000006 05 W947-ACTION PIC XX.
000008 05 W947-RESULT PIC XX.
000010 05 W947-COMPANY PIC XX.
000012 05 W947-CO-ID REDEFINES W947-COMPANY
000014 05 W947-EMP-NUMBER PIC S9(4) COMP.
000015 05 W947-EMP-FIRST-NAME PIC S9(4) COMP-3.
000017 05 W947-EMP-LAST-NAME PIC X(15).
000019 05 W947-SEX-CODE PIC X(15).
mm 05 W947-SEX-CODE PIC X(001).
000023 88 W947-MALE VALUE '1'.
mm 88 W947-FEMALE VALUE '2'.
a 05 XXXX-XXXXXXXXXXXXXXXXXXXX PIC X(001).
***** E N D O F D A T A G R O U P *****
***** -1---+---2---+---3---+---4---+---5---+---6---+---7---+---8 PF: 4-
SAVE 5-HP 6-BROWSE 7-SB 8-SF 9-EDIT 10-SL 11-SR ..-EID ..-LTYP ..-CANCEL
```

© 2000 CSC. All rights reserved.

The following line commands are associated with the M (move command).

- | | |
|-----------|---|
| M | Move this line. |
| MM | Block move. Start line and end line must both contain MM. |
| A | After this line. |
| B | Before this line. |

Notes:



Step 4—DDID Data Elements

The ELEMENT NAME is coded in 73-80. This is also called the Data Dictionary Identification (DDID). The name is used to build an element entry in the Process Dictionary. The name is required in coding SPS programs and can be optionally used in map definitions.

CSC

Step 4 - DDID Data Elements

ACTION COMPLETE

```

----- DATA GROUP DEFINITION INQUIRY / MAINTENANCE -----
COMMAND INPUT ==> HP          CC# ==>      1
DATA GROUP ID ==> 0000999947  EFF DATE==> 78/01/01 COLS: 9 80 MODE ==> EDIT
*****      -1----+---2----+---3----+---4----+---5----+---6----+---7----+---8
000001 1   W947-ScreenText-DATA-GROUP.
000003 **    THE TEXT OF YOUR DATA GROUP
000004 **    DESCRIPTION INFORMATION IS
000005 **    ENTERED ON DATA LINES.
000006 05  W947-ACTION          PIC XX.           TQE94701
000008 05  W947-RESULT          PIC XX.           TQE94702
000010 05  W947-COMPANY         PIC XX.           TQE94703
000012 05  W947-CO-ID          REDEFINES W947-COMPANY   TQE94704
000014          PIC S9(4) COMP.
000015 05  W947-EMP-NUMBER     PIC S9(4) COMP-3.   TQE94705
000017 05  W947-EMP-FIRST-NAME PIC X(15).        TQE94706
000019 05  W947-EMP-LAST-NAME  PIC X(15).        TQE94707
000021 05  W947-SEX-CODE       PIC X(001).       TQE94708
000023          88  W947-MALE      VALUE '1'.
000025          88  W947-FEMALE    VALUE '2'.
*****      -1----+---2----+---3----+---4----+---5----+---6----+---7----+---8
PF: 4-SAVE 5-HP 6-BROWSE 7-SB 8-SF 9-EDIT 10-SL 11-SR ..-EID ..-LTYP ..-CANCEL
*****      -1----+---2----+---3----+---4----+---5----+---6----+---7----+---8

```

© 2000 CSC. All rights reserved.

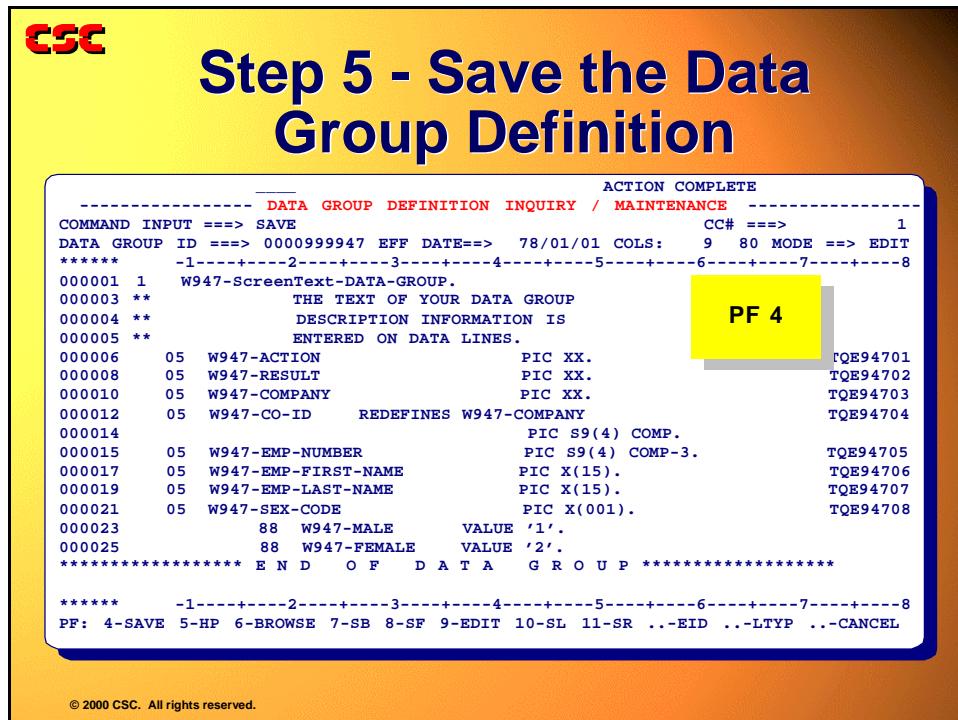
Notes:



Umbrella Programming

Data Groups

Step 5—Save the Data Group Definition

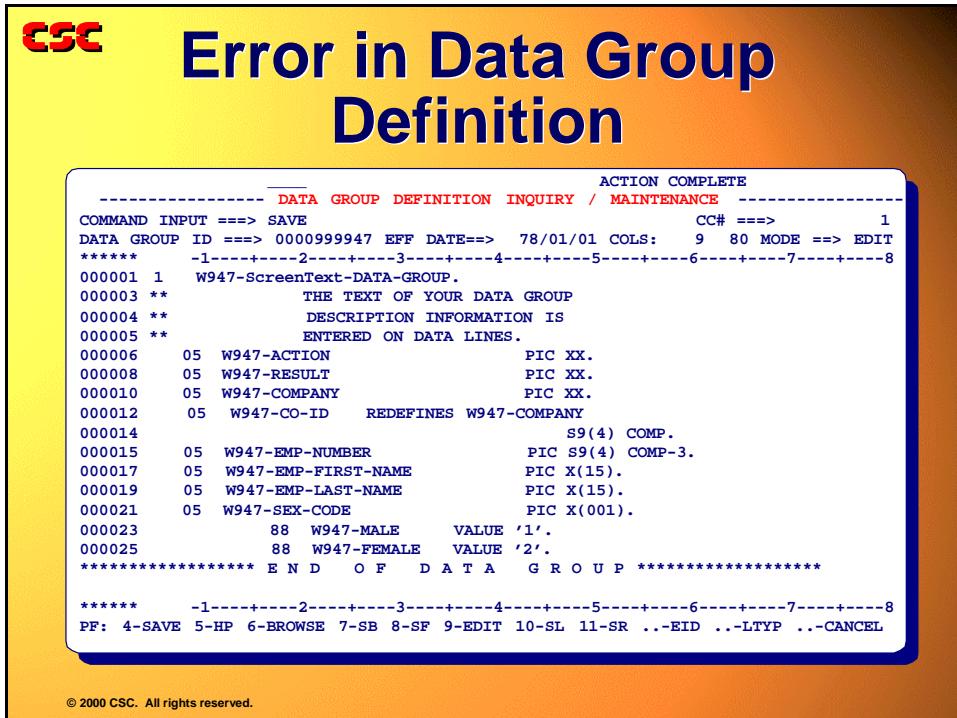


The save command performs the following:

- Recalculates all element characteristics.
- Verifies total Data Group length and corrects if length specified is too short.
- Updates the Element Data Dictionary.
- Replaces the source version of the data group definition on CDMF2.
- Does a GEN to create a new object version for PEM on CDMF1.
- Issues a new-copy function to load the definition into memory.
- Deletes the shadow copy.



Error in Data Group Definition



Notes:

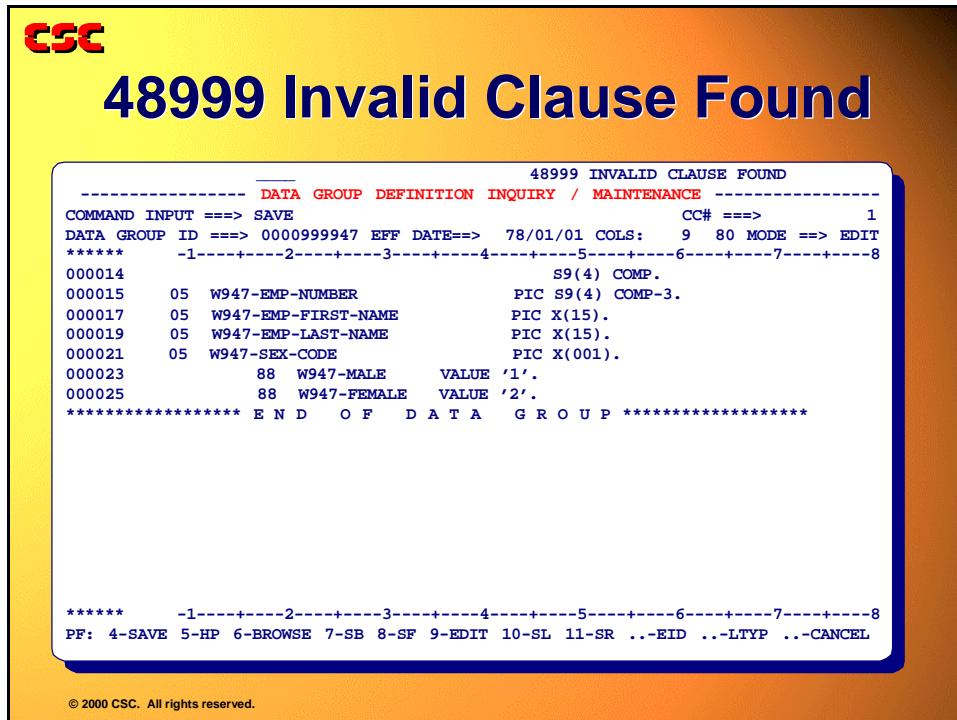


Umbrella Programming

Data Groups

Invalid Clause Found

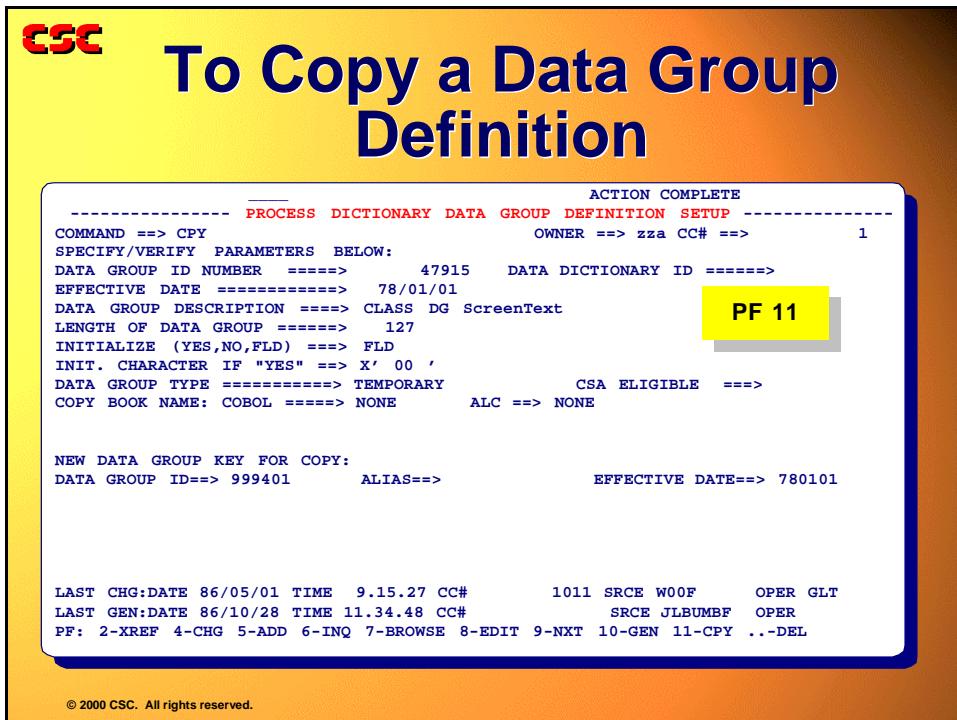
When the error is found the screen is redisplayed on the line in error.



Notes:



To Copy a Data Group Definition



Notes:



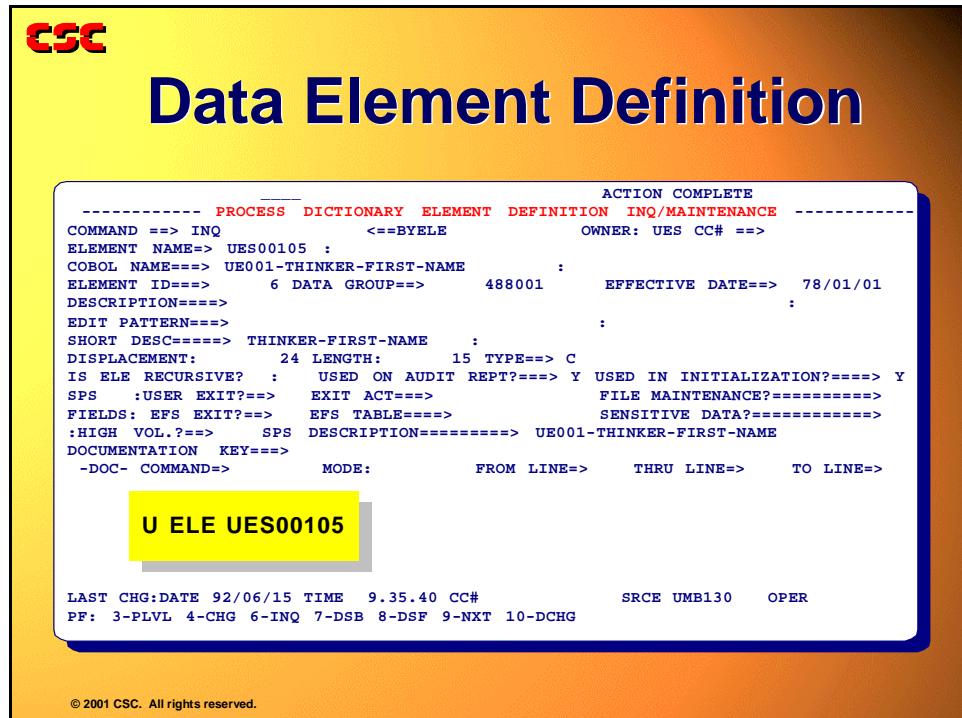
Umbrella Programming

Data Element Definition Screen

Data Element Definition Screen

The Data Element Definition Screen allows inquiry or maintenance on a specific element (field) in a data group. This facility does not provide ADD or DELETE capabilities for Data Element Definitions, nor does it allow CHANGES to the offset and length of the field.

From a cleared screen, the prompt to access the Data Element Definition Screen is U ELExxxxxx, where xxxxxxxx is the optional Data Element Name. This screen may also be accessed from the Umbrella Master Menu by selecting option 1.10.



Notes:



Data Group Definition Exercise



Problem Specifications—Data Group Definition

The data group should conform to the following specifications:

1. Use data group ID 9994xx (where xx is your group number).
2. The Data Dictionary ID should include your group number.
3. The data group description should include your group number.
4. The data group is to have a total length of 100 bytes.
5. The data group should be field initialized.
6. The data group should be a temporary data group.
7. The COBOL Copybook name is Z9994xxD. There are no ALC Copybooks.
8. Use the owner application and change control created in the Change Control Exercise.



Umbrella Programming

Data Element Definition Screen

As you define your data group, complete the data group definition check-list.

1. The 01 level should include your group number.
2. Data group ACTION and RESULT fields.
3. Company ID Number. Define as character and redefine as halfword binary.
4. Employee Number. Define as 9(11) COMP-3.
5. Employee Lastname. Define as 15 characters.
6. Employee Firstname. Define as 15 characters.
7. Department Code. Define as 3 characters.
8. Job Class Code. Define as 2 characters.
9. Job Position Code. Define as 2 characters.
10. Current Earnings. Define as S9(7)V99 COMP-3.
11. Y-T-D Earnings. Define as S9(7)V99 COMP-3.
12. Occurs counter. Define as S9(3) COMP-3.
13. Five (5) Employee salary adjustments. Define as S9(7)V99 COMP-3. You may want to use an OCCURS clause.
14. Type of processing. One byte character field to indicate random or sequential processing. Associate two 88 level statements to identify the type of processing.
15. Filler as needed to bring the length of the data group to 100 bytes.
16. Assign a Data Element Name to all fields except the Action, Result, and the FILLER field. The Data Element Name should follow the naming standard ZUPCxnn, where xx is your group number and nn is a sequence number between 00 and 99.
17. Was the data group saved? _____
18. Is the data group length 100 bytes long? _____



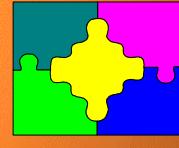
Summary



Summary



- ✓ Data Group definition is used to define any memory space needed
- ✓ Field must be part of data group to be accessed within Hogan system
- ✓ First two fields of data group are ACTION and RESULT



© 2001 CSC. All rights reserved.

Notes:



Umbrella Programming

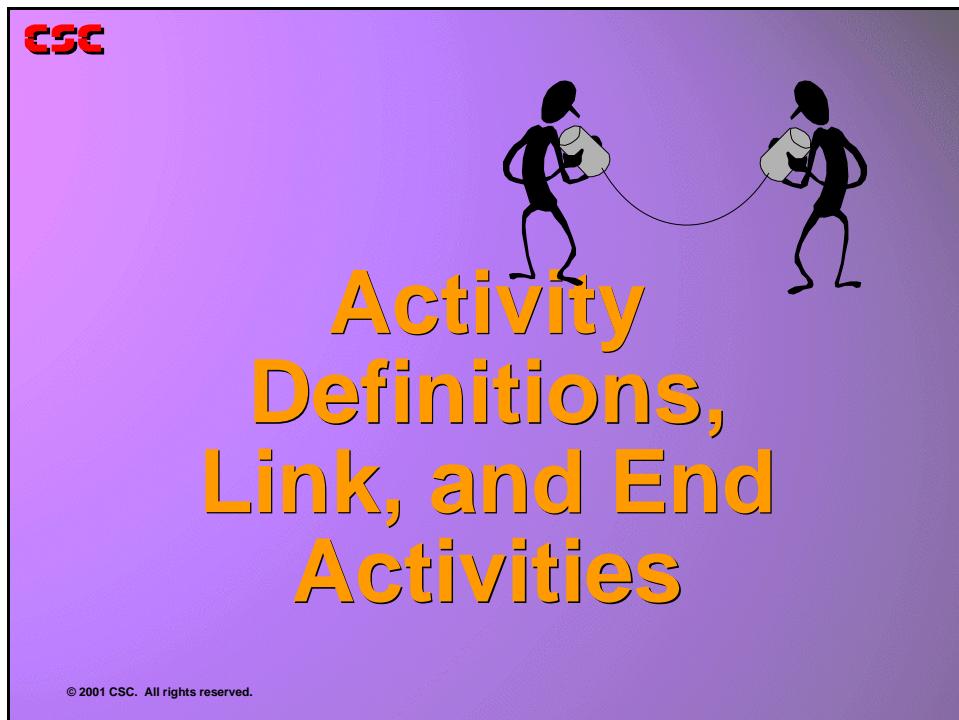
Summary



Link and End Activities

7

Purpose



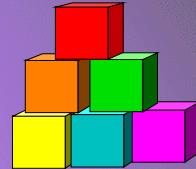
Notes:



Topics

CSC

Topics



- ▶ Activity definition
- ▶ Link activities
- ▶ End activities

© 2000 CSC. All rights reserved.

Objectives

CSC

Objectives



- Define an activity
- Add an activity definition to the Process Dictionary

© 2000 CSC. All rights reserved.



Activity Definition

An Activity is a UNIT OF WORK to be performed. The Activity Definition contains information required to accomplish the work request. There are several types of Activities defined in the Process Dictionary. For example, Work Area activities are used to allocate, initialize, and release Data Groups; Data Base activities are used to read and write Data Base Data Groups; Data Communications activities are used to deblock and display data in an online environment.

An activity request may be made through a Transaction Definition or by a program in a call to PEM. In either case, PEM locates the definition of the activity to be performed in the Process Dictionary and performs the work specified by the activity. Should PEM be unable to locate the Activity Definition, an exception message will be produced and the transaction will abnormally end.

Activity Definitions are categorized by the type of service they perform. Each category has a corresponding service routine within PEM. All of the activity types will be addressed in this class, but not in this chapter.



Activity Definition

Activity = Unit of Work

- ABEND EXIT
- CHECKPOINT
- DATA BASE
 - HIERARCHICAL
 - SEQUENTIAL
- DATA COMMUNICATIONS
- DUMP
- END
- EXCEPTION
- LINK
- SORT
- SQL
- TRACE
- WORK AREA

© 2000 CSC. All rights reserved.



Umbrella Programming

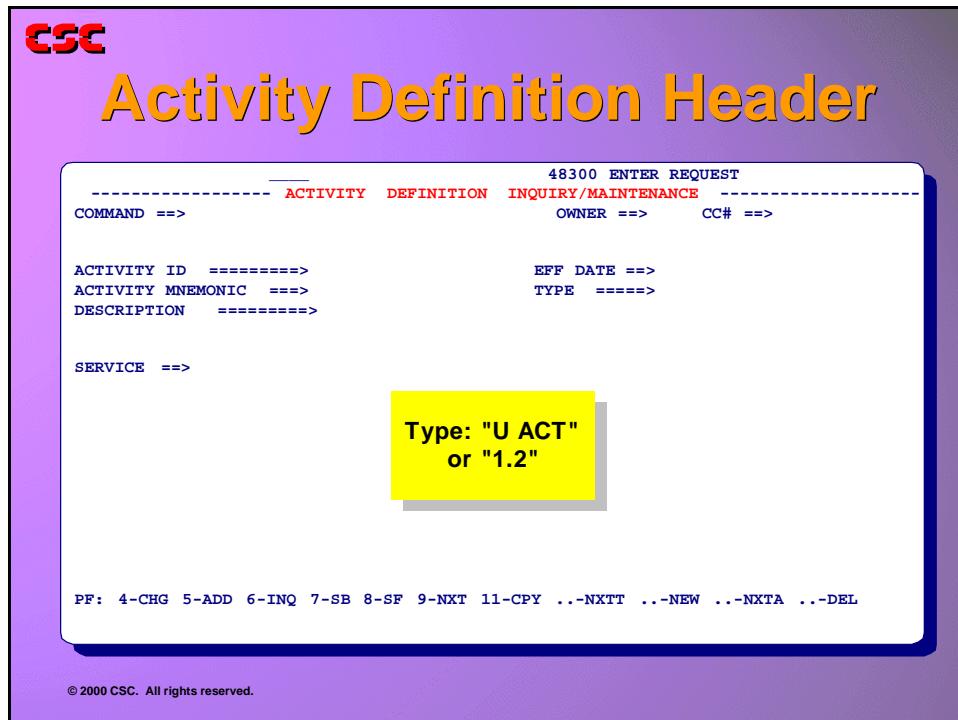
Activity Definition Header

Activity Definition Header

The first few lines of the "Activity Definition Inquiry/Maintenance" screen are common to all types of activities.

To display the Activity Definition header screen:

- Enter **U ACT** from a cleared screen
 - or
- Enter **=1.2** into the COMMAND field of another Process Dictionary screen
 - or
- Enter **2** into the OPTION field of the "Umbrella System Process Dictionary Maintenance Menu".



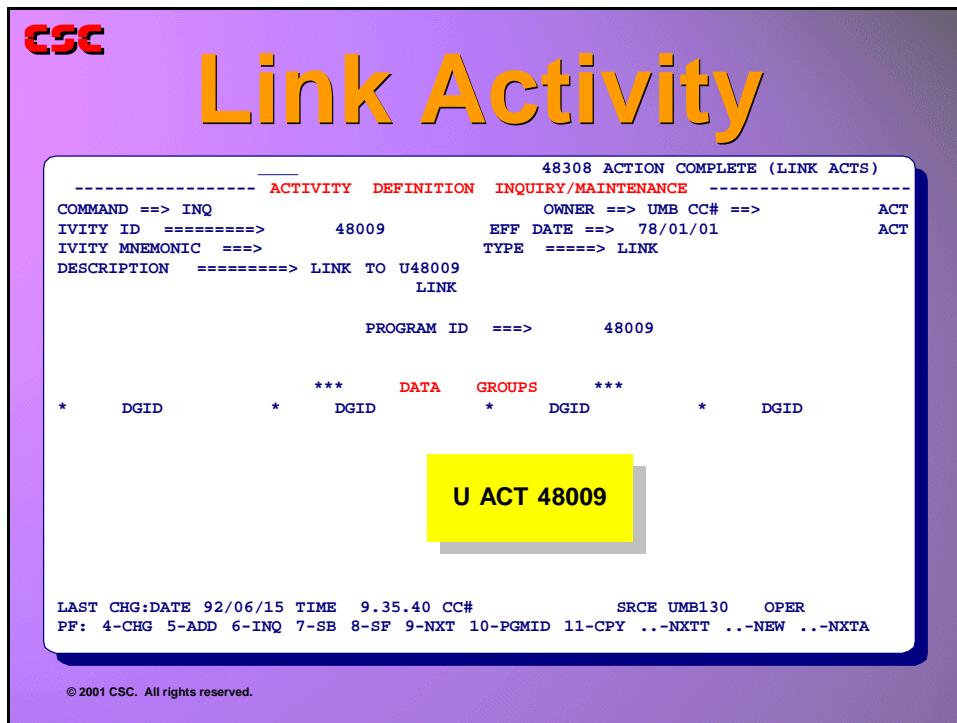
From this screen you can display an Activity Definition by ID or by TYPE.



Link Activity

The Link activity is used to transfer control to a program. It specifies the key to the Program Definition on the Process Dictionary. PEM executes the Link activity by looking up the Program Definition on the Process Dictionary, locating the load module referenced on the Program Definition, establishing addressability to the Data Groups needed by the program, and transferring control to the program.

If the target program of the Link activity has dynamic data groups defined, it is possible to pass addressability to specific Data Groups with a link activity. To accomplish this, a list of data group IDs is placed on the Link Activity Definition. The first data group ID will have its address placed in the first dynamic cell in the target program's parameter list. The second data group ID will be passed in the second dynamic cell, and so on.



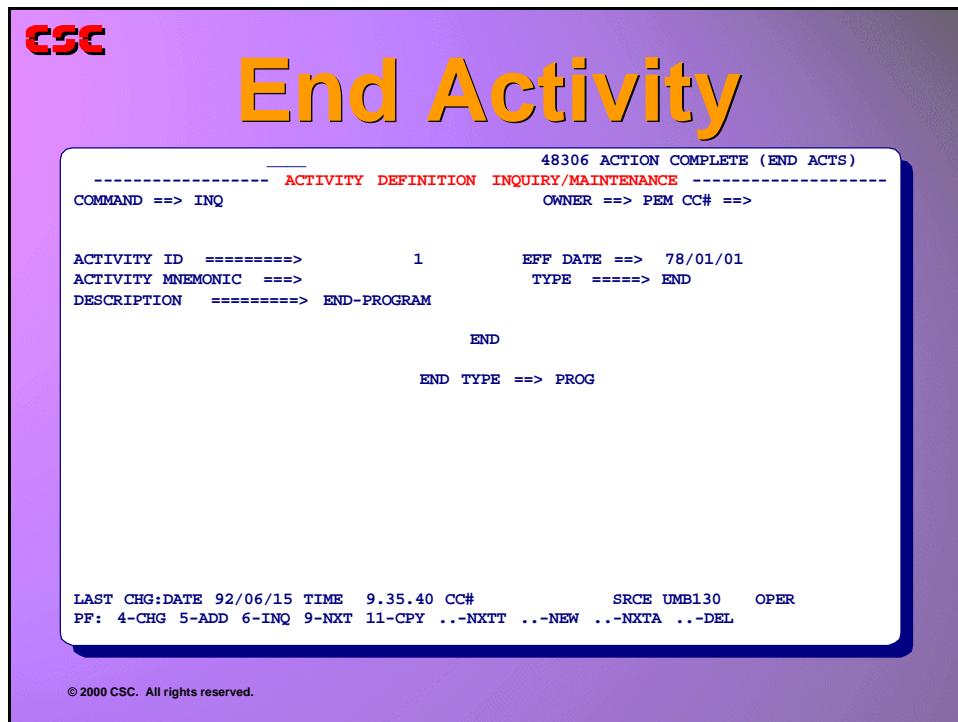
End Activity

The END Activity is used to either end a program or a transaction. All programs must terminate execution by issuing an END Program Activity. In most cases, END Transaction Activities are optional.

Predefined activity IDs for END are:

- ACTIVITY 1—END PROGRAM.
- ACTIVITY 2—END TRANSACTION.

End Activity Definition



Note: End program activities are replaced by COBOL GOBACK clauses under Umbrella 3.



Problem Specifications—Link Activity



- Establish a Link activity for program ID 9994xx, Where xx is your group number. This program, which you will be modifying later in the week, does not use any Dynamically Allocated Data Groups.
- The activity ID should be the same as the program ID.
- Give the activity ID an effective date of 78/01/01.
- The type of activity will be LINK.
- There will be no mnemonic for this activity ID.
- Please include your group number in the description of the activity ID.
- Enter values from the change control number you created.

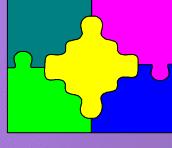
Notes:



Summary



Summary



- Unit of work under Hogan is an activity
- Seven most common activities are:
 - ✓ Work area
 - ✓ Link
 - ✓ Data Base
 - ✓ Data Communication
 - ✓ End activities
 - ✓ Exception
 - ✓ Dump

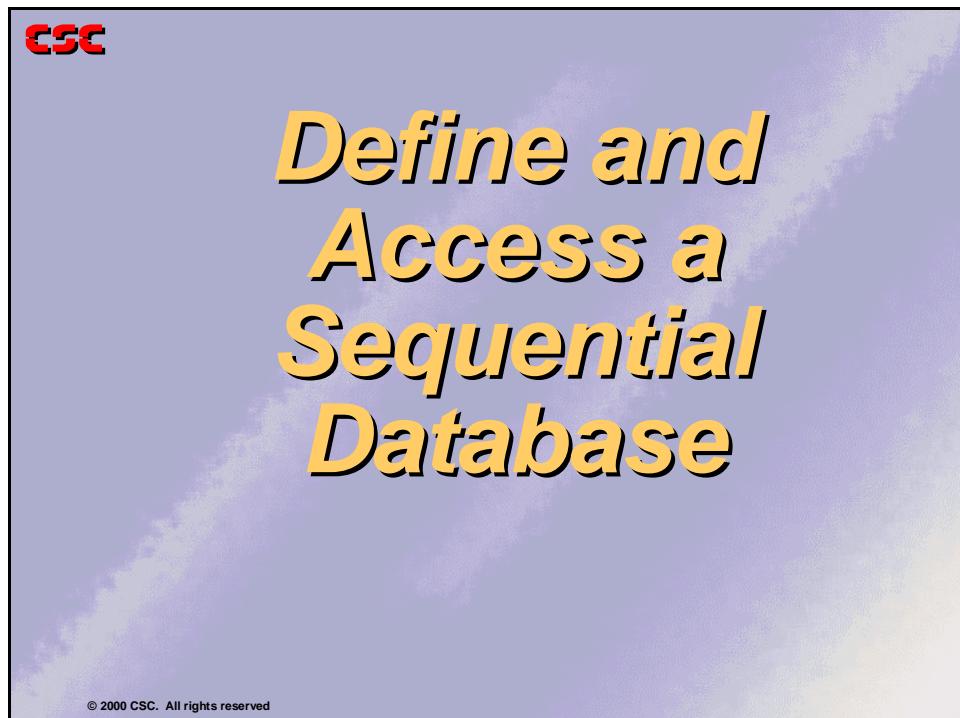
© 2000 CSC. All rights reserved.

Notes:



Sequential Database Definitions/Activities 8

Purpose



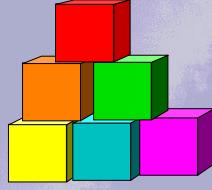
Notes:



Topics

CSC

Topics



- ▶ Purpose of a sequential data base
- ▶ Use of the “Sequential Database Definition” screen
- ▶ Access of an SDB

© 2000 CSC. All rights reserved

Objectives

CSC

Objectives



- Define an SDB under CDMF
- Define an SDB activity under CDMF
- Access an SDB definition under CDMF

© 2000 CSC. All rights reserved



Sequential Data Bases—SDB

Introduction

Sequential data bases are simple sequential files that may be fixed or variable in length. They can contain a variety of different kinds of data, including, but not limited to:

CSC

Sequential Data Bases - SDB

- SYSPRINT
- SPS Reports Data String File
- Change Control Log File
- Application Transaction Files
- Checkpoint Data
- Batch Maintenance Files
- Scrub Files
- Dump/Restore Files
- Etc.

© 2000 CSC. All rights reserved

Sequential files can be stored on disk or tape. They can also reside on unit record devices such as card readers, printers, and fiche. The access methods currently supported for SDB files are BSAM, QSAM, and VSAM/ESDS.

A sequential data base record is comprised of one data group per logical record. This data group is used as an I/O work area for reading and writing the record data. The required action and result fields of the data group are never written to any file.

Notes:



Umbrella Programming

Sequential Data Bases—SDB

Sequential Data Base Definition

All sequential files must be defined using an SDB definition, which provides PEM with the DDNAME and physical characteristics of the data base. The definition also indicates which access method or DBMS to use. HOGAN delivers many SDB definitions that may be used in an application program simply by including the DDNAME of a delivered SDB on a DD statement in the program's execution JCL. If you do not want to use any of the delivered SDB definitions, you can add your own definition.

All SDB files are defined as either input or output files. If the data is to be stored on disk or tape, then two SDB definitions will be needed. One definition will provide write access to the storage device, while the second definition will provide read access. An example of these paired SDB definitions are CCTRANS and CCTRANSO, which are used for the Change Control transaction file. Copies of these definitions are on the next page. Notice that they are identical, except that CCTRANS has an INPUT file type and CCTRANSO has an OUTPUT file type.

Notes:





Sequential Data Base Definition

```

----- ACTION COMPLETE -----
----- SEQUENTIAL DATABASE DEFINITION -----
COMMAND ==> INQ OWNER ==> RCS CC# ==>

DATABASE ID ==> CCTRANS
DESCRIPTION ==> CHANGE CONTROL SYSTEM

FILE TYPE =====> INPUT      (INPUT,OUTPUT)
DEVICE CLASS =====> DISK      (READER,PRINTER,TAPE,DISK)
ACCESS METHOD =====> QSAM      (BSAM,QSAM,VSAM)
RECORD FORMAT =====> FIXED     (FIXED,VARIABLE)
LOGICAL RECORD LENGTH ==> 168    (LRECL <= 32760)
LOG DATABASE ? =====> N       (A/P/N)

DATABASE ATTRIBUTES          CICS VSAM OPTIONS

DDNAME =====> CCTRANS        EMPTY WHEN OPENED =====> N   (Y/N)
RECFM =====>                MASSINSERT WRITES =====> N   (Y/N)
BLKSIZE =====>
TAPE CLOSE COMMAND =====>

LAST CHG:DATE 92/06/15 TIME 9.35.40 CC#           SRCE UMB130  OPER
LAST GEN:DATE 94/11/03 TIME 11.38.02 CC#           SRCE GXB6    OPER
PF: 2-XREF 3-PLVL 4-CHG 5-ADD 6-INQ 9-NXT 10-GEN ..-DEL

```

© 2000 CSC. All rights reserved



SDB CCTRANSO

```

----- ACTION COMPLETE -----
----- SEQUENTIAL DATABASE DEFINITION -----
COMMAND ==> INQ OWNER ==> RCS CC# ==>

DATABASE ID ==> CCTRANSO
DESCRIPTION ==> CCS WORK FILE FOR DAILY BATCH UPDATE

FILE TYPE =====> OUTPUT      (INPUT,OUTPUT)
DEVICE CLASS =====> DISK      (READER,PRINTER,TAPE,DISK)
ACCESS METHOD =====> QSAM      (BSAM,QSAM,VSAM)
RECORD FORMAT =====> FIXED     (FIXED,VARIABLE)
LOGICAL RECORD LENGTH ==> 168    (LRECL <= 32760)
LOG DATABASE ? =====> N       (A/P/N)

DATABASE ATTRIBUTES          CICS VSAM OPTIONS

DDNAME =====> CCTRANSO        EMPTY WHEN OPENED =====> N   (Y/N)
RECFM =====>                MASSINSERT WRITES =====> N   (Y/N)
BLKSIZE =====>
TAPE CLOSE COMMAND =====>

LAST CHG:DATE 92/06/15 TIME 9.35.40 CC#           SRCE UMB130  OPER
LAST GEN:DATE 94/11/03 TIME 11.38.02 CC#           SRCE GXB6    OPER
PF: 2-XREF 3-PLVL 4-CHG 5-ADD 6-INQ 9-NXT 10-GEN ..-DEL

```

© 2000 CSC. All rights reserved



Umbrella Programming

Sequential Data Bases—SDB

The PRINT file is used by SPS. It is used to write records to a printer. Below is a copy of the SDB definition for this file.

From a cleared screen, enter **U SDB** or **U 1.6** or, from the "Umbrella System Process Dictionary Maintenance Menu", select option 6.

The screenshot shows a terminal window titled "SDB Print" with a CSC logo at the top. The window displays the following SDB definition:

```
ACTION COMPLETE
----- SEQUENTIAL DATABASE DEFINITION -----
COMMAND ==> INQ          OWNER ==> PEM CC# ==>
DATABASE ID ==> PRINT    DESCRIPTION ==> SPS'S GENERAL SYOUT FILE (MACHINE CARR CONTROL) :
FILE TYPE ======> OUTPUT      (INPUT,OUTPUT)
DEVICE CLASS ======> PRINTER   (READER,PRINTER,TAPE,DISK)
ACCESS METHOD ======> QSAM     (BSAM,QSAM,VSAM)
RECORD FORMAT ======> FIXED    (FIXED,VARIABLE)
LOGICAL RECORD LENGTH =====> 133  (LRECL <= 32760)
LOG DATABASE ? ======> N      (A/P/N)

DATABASE ATTRIBUTES           CICS VSAM OPTIONS
DDNNAME ======> PRINT        EMPTY WHEN OPENED ======> N      (Y/N)
RECFM ======> FBM           MASSINSERT WRITES ======> N      (Y/N)
BLKSIZE ======>
TAPE CLOSE COMMAND ======>

LAST CHG:DATE 92/06/15 TIME 9.35.40 CC#          SRCE UMB130   OPER
LAST GEN:DATE 93/10/19 TIME 19.26.05 CC#          SRCE CXBINST6 OPER
PF: 2-XREF 3-PLVL 4-CHG 5-ADD 6-INQ 9-NXT 10-GEN ..-DEL

© 2000 CSC. All rights reserved
```

Notes:



Sequential Database Activities

Sequential data base (SDB) activities permit programs running under Umbrella to access and change data stored in data bases defined as SDB files to PEM. An SDB activity provides PEM with the information about data base name, data group ID, use code, and default action.

The data base name is used as a key by PEM to read the CDMF SDB definition, and to point to the appropriate file within the JCL. The data group ID field supplies PEM with the necessary target data group. PEM must have a data group to use as an I/O area when manipulating a record from a sequential file. SDB definitions do not specify a data group. Different SDB activities for the same data base may specify different target data groups.

There are only two valid options for the use code, either I (input) or O (output). PEM supports the following data base actions against a sequential file:

READ, WRITE, CLOSE, OPENI, OPENO, and NOP.

Sequential databases are comprised of one data group per logical record. Random positioning is not allowed. The data group area, following action and result, is used as a work area for reading and writing these records. Action and result fields are not written to the file. When variable-length records are processed, the standard record descriptor word (LLBB) must be defined in the data group definition after the action and result fields. The application program must set the length value of the record before issuing a write access to the sequential database.

Field TCB-OPTIONS-1 (TCB\$OPT0)

This 1-byte field exists in the User TCB extension Area indicating processing requirements to PEM. Field TCB\$OPT0 allows the program to specify which action to perform if an error occurs.

X'02' (TCB\$ODBN)

Setting a value of hexadecimal two specifies to return all Database Unavailable conditions to the application program. This allows the application to attempt a recovery or circumvention if a database cannot be accessed.

The flag must be set before every applicable database activity. The flag is reset by PEM (to X'00') before returning control to the application.

On the next page are two examples of sequential data base activities that write to the PRINT data base. Notice that the two activities reference different target data groups.



Adding SDB Definition



Adding SDB Definition

- Same as any other PD entry
 - Add command
 - Change Control
 - Parameters

© 2000 CSC. All rights reserved

Notes:





Sequential Data Base Activities

```

----- 48313 ACTION COMPLETE (SDB ACTS)
----- ACTIVITY DEFINITION INQUIRY/MAINTENANCE -----
COMMAND ==> INQ OWNER ==> PEM CC# ==>

ACTIVITY ID ======> 114      EFF DATE ==> 78/01/01
ACTIVITY MNEMONIC ==>          TYPE =====> SDB
DESCRIPTION ======> GEN PRT-RPT OUTPUT

----- SEQUENTIAL DATABASE -----
DATA BASE NAME ==> PRINT
DISPOSITION =====> WAIT
DATA GROUP ID ==>          102
DATA GROUP USE ==> O
DG DEFAULT ACT ==> WRITE

LAST CHG:DATE 92/06/15 TIME 9.35.40 CC#           SRCE UMB130   OPER
PF: 4-CHG 5-ADD 6-INQ 7-DGID 9-NXT 10-DBID 11-CPY ..-NXTT ..-NEW ..-NXTA ..-DEL

```

© 2000 CSC. All rights reserved



Activity 201

```

----- 48313 ACTION COMPLETE (SDB ACTS)
----- ACTIVITY DEFINITION INQUIRY/MAINTENANCE -----
COMMAND ==> INQ OWNER ==> PEM CC# ==>

ACTIVITY ID ======> 201      EFF DATE ==> 78/01/01
ACTIVITY MNEMONIC ==>          TYPE =====> SDB
DESCRIPTION ======> PRINT

----- SEQUENTIAL DATABASE -----
DATA BASE NAME ==> PRINT
DISPOSITION =====> WAIT
DATA GROUP ID ==>          6
DATA GROUP USE ==> O
DG DEFAULT ACT ==> WRITE

LAST CHG:DATE 92/06/15 TIME 9.35.40 CC#           SRCE UMB130   OPER
PF: 4-CHG 5-ADD 6-INQ 7-DGID 9-NXT 10-DBID 11-CPY ..-NXTT ..-NEW ..-NXTA ..-DEL

```

© 2000 CSC. All rights reserved



Building an SDB File



Checklist to Build an SDB File

- Define Data Group definition
- Define SDB definition to PEM
- Define SDB activity on Process Dictionary

- OP System requirements: catalog, allocate, initialize file
- Copybook/DSECT
- Execution JCL to reference new data base

© 2000 CSC. All rights reserved

The following steps must be taken to create an SDB file for use in your application program.

1. A data group I/O area must be defined on the Process Dictionary.
2. Define the sequential data base to PEM with an SDB definition.
3. Add SDB activities on Process Dictionary to allow for access against the data base.
4. Catalog, allocate, initialize the file on your operating system.
5. Create a copybook to reflect the data group fields.
6. Modify the execution JCL to reference the new data base.

Notes:



Issuing an SDB Activity



Issuing an SDB Activity

- **Data Group:** contains action and result fields, and field-defined data area
- **SDB Definition:** provides data base physical definition and indicates access method or DBMS
- **SDB Activity Definition:** supplies data base name, data group ID, and default action code

© 2000 CSC. All rights reserved

The SDB activity function is controlled by three CDMF components. They are the data group, the SDB activity, and the sequential data base definition.

The Data Group: Contains the action and result fields, which are not written to the file, and the field-defined data area.

The SDB Definition: Provides the physical definition of the data base and indicates which access method or DBMS.

The SDB Activity Definition: Supplies the data base name as well as the data group ID, its use code, and default action code.

An SDB activity can be issued from both a PEM transaction and an application program. The default action specified in the SDB activity definition is used to manipulate the sequential file. An application program can override the default action by setting the action field in sequential file's target data group with a valid action code.



Checking Result Fields



Checking Result Fields

■ After a Data Base Activity

- Check TCB-Result
- Check Data Group Results Fields

© 2000 CSC. All rights reserved

It is the responsibility of the application program issuing the activity to check the result of the action when control is returned to the program. The appropriate result field(s) should be checked to determine the status of the requested activity.

After a data base activity, check TCB-RESULT. If the value is equal to TCB-OK, the DB activity was successful. A value not equal to TCB-OK indicates that some part of the DB activity failed, such as, end-of-data. You must then check the result field of the data group referenced in the data base activity to determine the result of the requested action.

If the result field in the data group is not equal to DGR-OK, the data group's result field will contain the error code that corresponds to those in the delivered data group result code copybook, P49023D.

Notes:



Setting the DG Action in a Program

CSC

Activity Execution Example

WORKING-STORAGE SECTION.

```
01 SDB-ACTIVITY.  
  05 FILLER          PIC S9(8) COMP VALUE +47410.
```

```
-INC P49022D  DATA GROUP ACTIONS  
-INC P49023D  DATA GROUP RESULTS
```

LINKAGE SECTION.

```
01 TRANSACTION-CONTROL-BLOCK.
```

```
  . . .  
01 USER-DATA-GROUP-1.  
  05 DG1-ACTION      PIC XX.  
  05 DG1-RESULT      PIC XX.  
  05 DG1-FIRST-FIELD PIC ?
```

• • •

© 2000 CSC. All rights reserved

CSC

PROCEDURE DIVISION USING TRANSACTION-CONTROL-BLOCK

```
USER-DATA-GROUP-1
```

```
  . . .  
AA000-HOUSEKEEPING.  
  MOVE DGA-OPEN-OUTPUT TO DG1-ACTION.  
  MOVE SDB-ACTIVITY TO TCB-LONG-ACTIVITY.  
  PERFORM CA000-CALL-PEM.  
  IF TCB-RESULT EQUAL TCB-OK  
    NEXT SENTENCE  
  ELSE  
    PERFORM YY000-ERROR-PROCESSING.  
  . . .
```

```
BA000-WRITE-RECORDS.  
  MOVE DGA-WRITE TO DG1-ACTION.  
  MOVE SDB-ACTIVITY TO TCB-LONG-ACTIVITY.  
  PERFORM CA000-CALL-PEM.  
  IF DG1-RESULT NOT EQUAL DGR-OK  
    PERFORM YY000-ERROR-PROCESSING.  
  . . .
```

© 2000 CSC. All rights reserved



Problem Specifications—Sequential Database



Sequential Database Exercise I

In this exercise you will define to PEM a sequential output file that will contain employee information. Follow these specifications:

1. The Data Base ID will be EMPxxOUT (where xx is your group number)
2. The access method used is QSAM
3. The disk file is fixed length with an LRECL of 96 bytes

To access the new sequential data base, an SDB activity will need to be defined. Follow these specifications:

1. Use activity ID 9995xx (where xx is your group number)
2. Use the data group ID of 9994xx
3. Processing should continue after this activity is issued
4. Data is to be written using this activity.

Use the owner application and change control created in the Change Control Exercise.

Sequential Database Exercise II

In this exercise you will define to PEM a sequential input file that will contain employee information. Follow these specifications:

1. The Data Base ID will be EMPxxIN (where xx is your group number)
2. The access method used is QSAM
3. The disk file is fixed length with an LRECL of 96 bytes

To access the new sequential data base, an SDB activity will need to be defined. Follow these specifications:

1. Use activity ID 9995yy (where yy is your group number+20)
2. Use the data group ID of 9994xx
3. Processing should continue after this activity is issued
4. Data is to be read using this activity.

Use the owner application and change control created in the Change Control Exercise.



Sequential Database Access Research Exercise



1. For each type of SDB, the USE field for the data groups must be specified on the data base activity. List and explain the uses for each type.

SDB ACTIVITY

2. List the ACTION and RESULT codes below.

SDB ACTIVITY	
ACTIONS	RESULTS

Notes:



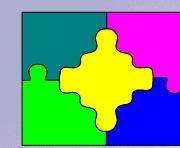
Summary

CSC

Summary



- ✓ Field must be part of data group to be accessed within the Hogan system
- ✓ PEM manages physical access to data bases
- ✓ Application program communicates information to PEM for desired data base access
 - ✓ Data group
 - ✓ Data base
 - ✓ Data base activity definitions



© 2000 CSC. All rights reserved

CSC

Sequential Data Bases Summary

- ✓ One related data group for each sequential data base
- ✓ Sequential data bases used for variety of purposes
 - ✓ Input only
 - ✓ Output only
 - ✓ Only one device level
- ✓ Accessed through activities that specify target data group
 - ✓ Different target data groups in different activities
- ✓ Default action code can be overridden in application program

© 2000 CSC. All rights reserved



Umbrella Programming

Summary



Umbrella Programming

Summary



Hierarchical Database Definitions/Activities 9

Purpose

CSC

Purpose

Define and access an hierarchical database

Overview the interface between PEM and both VSAM and DL/I



© 2001 CSC. All rights reserved.

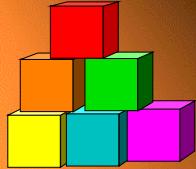
Notes:



Topics

CSC

Topics



- ▶ Hierarchical data base purpose
- ▶ Hierarchical Database Definition screens use
- ▶ HDB access
- ▶ PEM interface for HDB with VSAM
- ▶ PEM interface for HDB with DL/I

© 2000 CSC. All rights reserved.

Objectives

CSC

Objectives



- Access an HDB definition under CDMF
- Define an HDB activity under CDMF
- Diagram the PEM interface for HDB with VSAM
- Diagram the PEM interface for HDB with DL/I

© 2000 CSC. All rights reserved.



Hierarchical Database—HDB

CSC

Hierarchical Data Base - HDB

- Base Data Group
 - Required
 - Contains all keys
- Positional Data Group
 - 1 per key field value
 - Process sequentially
- Associate Data Group
 - Keyed
 - Multiple occurrence
 - Processing randomly
 - Non-keyed
 - Multiple occurrences
 - Processed sequentially/directly

© 2000 CSC. All rights reserved.

Hierarchical Data Bases are made up of one or more data groups that have a hierarchical relationship, that is, a dependent data group may not exist without an occurrence of its parent. Hierarchical Data Bases may be accessed randomly or sequentially and may be stored and retrieved via several Data Base Management Systems (DBMS). Because PEM creates a logical environment for the application programs, The DBMS used is transparent to the application.

Most data bases used by HOGAN applications are hierarchical data bases. The main distinguishing characteristic of HDBs is that they are keyed and must be defined to PEM as hierarchical, even if they contain only a base data group.

When defining a Hierarchical Data Base to PEM, there are several data group types:

1. BASE DATA GROUP

This type is analogous to a root or base segment in a DL/I structure. The Base Data Group requires a key and may be accessed randomly or sequentially.

2. POSITIONAL DATA GROUP

A Positional Data Group is nonkeyed and can occur once for each occurrence of the base.

3. ASSOCIATE DATA GROUP



Umbrella Programming

Hierarchical Database—HDB

Associate data group may be keyed or nonkeyed. There can be 0 (zero) to n occurrences of an associate for each occurrence of its parent.

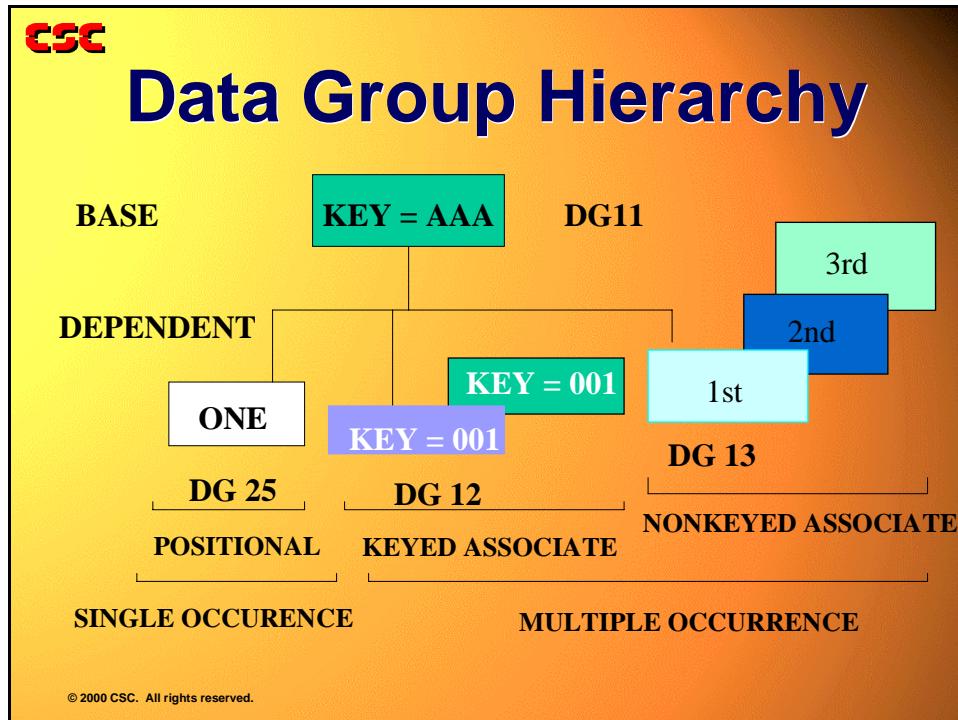
Positional and Associate Data Groups are analogous to dependent segments or physical children in a DL/I structure.

Notes:



Data Group Hierarchy

The following figure illustrates a data group hierarchy.



Notes:

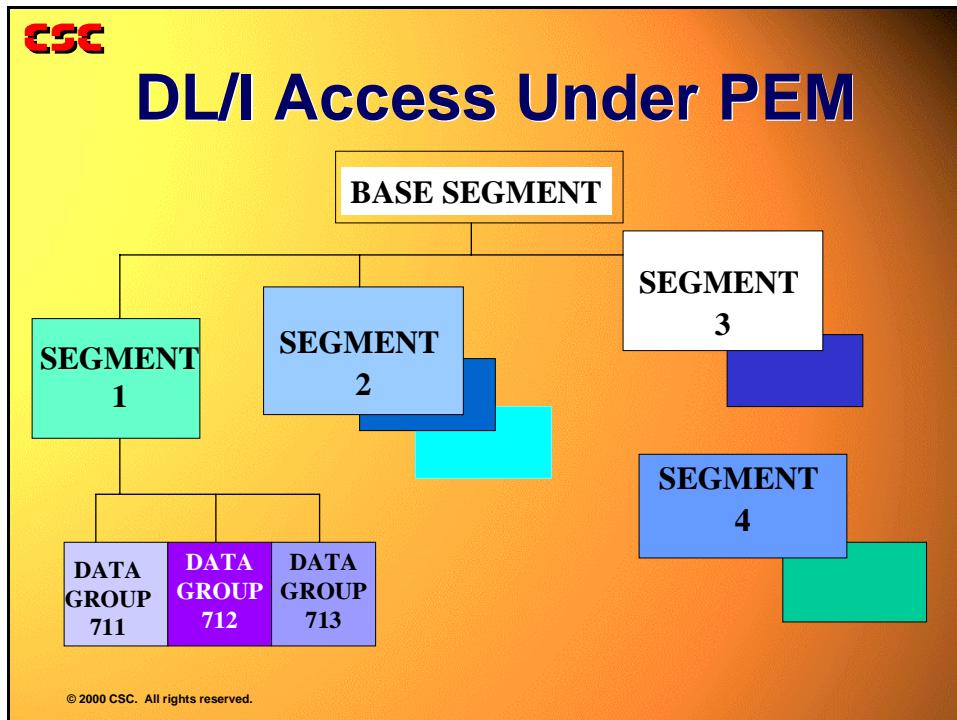


Umbrella Programming

Hierarchical Database—HDB

DL/I Access Under PEM

The following illustrates an example DL/I data base structure and how it could be defined to PEM/Umbrella.

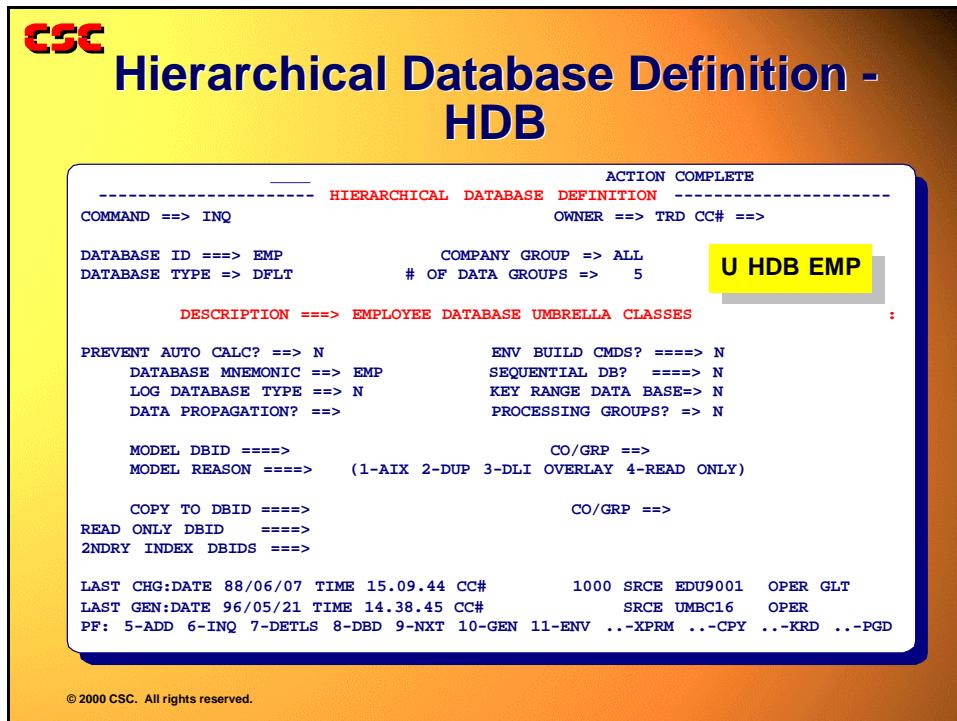


Notes:



Hierarchical Database Definition (HDB)

From a cleared screen, enter U HDB or U 1.5.1 or from the "Umbrella System Process Dictionary Maintenance Menu" select option 5.1 to view the base screen.



To view the Data Base Hierarchy press PF7 or enter DETLS in the COMMAND field.

Notes:



Umbrella Programming

Hierarchical Database Definition (HDB)

Hierarchical Database Hierarchy Definition

This display shows the top to bottom left to right relationship of the Data Groups within the Data Base. The sequence of the Data Groups must be maintained when a Data Base Activity is coded.

The screenshot displays a terminal window titled "Hierarchical Database Hierarchy Definition". The window shows a hierarchical tree of data groups for the "EMP" database. The root node is "47100 BASE". Below it are nodes "47110 POS" and "47130 POS". Further levels down include "47140 POS" and "47120 POS". The columns in the tree structure are: SEQ, LV, DATA-GROUP, TYPE-, VAR, DESCRIPTION, KEY, and LEN. A yellow callout box in the center of the screen contains the text: "PF7 or DETLS Command from HDB Base Screen". At the bottom of the screen, there are function key definitions: PF: 3-PLVL 4-CHG 6-INQ 7-SB 8-SF, and a note: "© 2000 CSC. All rights reserved.".

SEQ	LV	DATA-GROUP	TYPE-	VAR	DESCRIPTION	KEY	LEN
1	1	47100	BASE	FIX	EMP DB KEY		
2	2	47110	POS	FIX	EMP DB INFO	47100	
3	2	47130	POS	FIX	EMP DB INFO	47100	
4	2	47140	POS	FIX	EMP DB INFO	47100	
5	2	47120	POS	FIX	EMP DB INFO	47100	

To view the data group definition or the segment definition, place the appropriate character under the action (*) field:

- * (ACTION)
- ? - Display Data Group Definition.
- S - Display Segment Definition.

Notes:



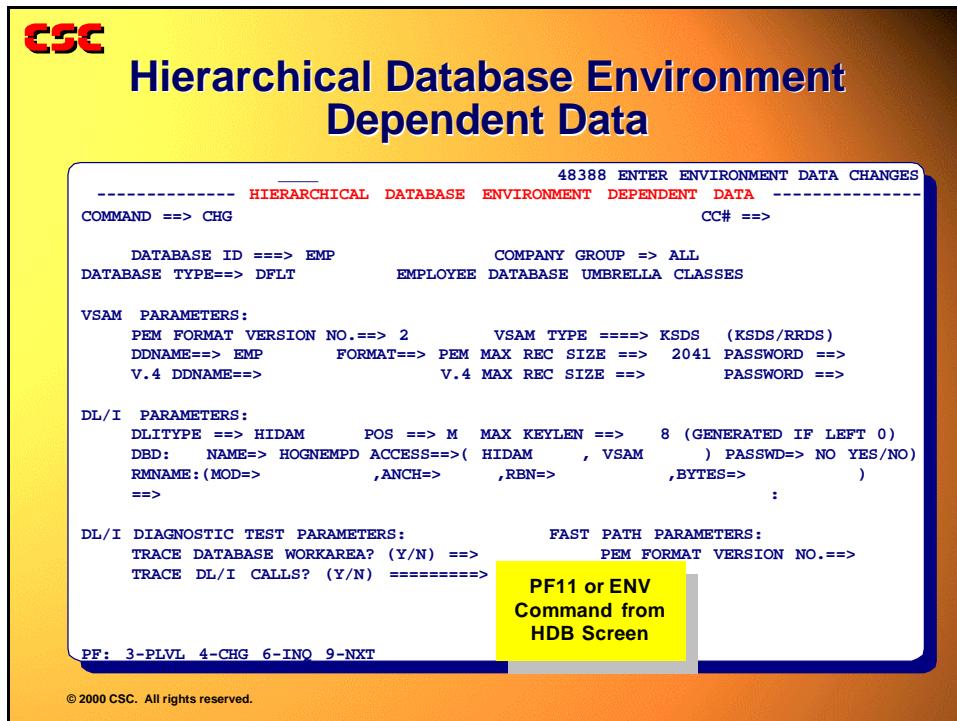
Hierarchical Database Environment Dependent Data

To view the Hierarchical Data Base Environment Dependent data:

- Press PF11
- or
- Enter ENV in the COMMAND field

from the HDB Base Screen.

This display is used to tie the logical environment to the real environment. It is used to establish the Data Base as VSAM or DL/I.



Notes:



Umbrella Programming

Hierarchical Database Definition (HDB)

Hierarchical Database HIDAM DBD Definition

To view the Hierarchical Data Base HIDAM DBD Definition:

- Press PF8
- or
- Enter **DBD** into the COMMAND field

from the HDB Base Screen.

This display is used to define the index for a Hierarchical Index Direct Access Method Data Base (HIDAM). The information is used to create the HIDAM DBD source statements.

CSC Hierarchical Database HIDAM DBD Definition

ACTION COMPLETE
----- HIERARCHICAL DATABASE DL/I HIDAM INDEX DBD DEFINITION -----
COMMAND ==> INQ CC# ==>

DATABASE ID ==> EMP	COMPANY GROUP==> ALL
DBD: NAME==> HOGNEMPX ACCESS==> (INDEX , VSAM) PASSWD==> NO	:
==>	
DATASET: DDL==> EMPX DEVICE==> 3350 MODEL==> OVFLW==>	:
BLOCK==> (,) SIZE==> (,)	:
RECORD==> (,)	:
==>	
SEGMENT: NAME==> INDEX PARENT==> 0 BYTES==> (8 ,)	:
FREQ==> 500 RULES==>	:
==>	
LCHILD: NAME: SEGNAME1==> S0047100 DBDNAME==> HOGNEMPD	:
PTR==> INDEX==> K0047100	:
==>	
FIELD: NAME: FLDNAME1==> EMPIKEY SEQ==> SEQ U/M==> U	
BYTES==> 8 START==> 1 TYPE: X/P/C==> X	
==>	
DBDGEN:	
END:	
PF: 3-PLVL 4-CHG 6-INQ 9-NXT	

PF8 or DBD Command from HDB Base Screen

© 2000 CSC. All rights reserved.

Notes:



How PEM Uses Data Bases

Data Base Access Management

Data bases are defined to the Process Dictionary. The definitions provide not only information about the physical access of the data base but also the logical structure of the data base records. This logical structure is indicated by listing the data groups residing on the record and the relationship among them. A data group defines a contiguous group of logically related elements in a physical record.

A COBOL program accesses the data through work areas or data groups that are in the linkage section of the program. The program instructs PEM to access a data base through an activity. The coded statements required are similar to

```
MOVE activity-name TO TCB-LONG-ACTIVITY.  
CALL 'PEM' USING TRANSACTION-CONTROL-BLOCK.
```

A record key is not required during sequential processing. It is required for random access. The user can set the key fields in the application program or instruct PEM to deblock them.

Physical Data Base Definitions

The data structures supported under PEM include VSAM, DL/I, IMS, and IMS Fast Path. A data base is defined physically under one of these structures.

There is nothing magical about the manner in which the data bases are physically accessed. Normal processing occurs. The magic is that PEM handles the actual access. In order to do so, there are different and additional requirements for the application programmer and data base administrator. They must communicate to PEM information about the access desired.

Notes:



Umbrella Programming

How PEM Uses Data Bases

Hierarchical VSAM Management

PEM provides the VSAM user a new way in which to develop VSAM systems and view data stored in native VSAM structures. PEM allows the user to create a hierarchical structure that is physically stored in a flat file. A variable length VSAM record in effect becomes a root segment with dependent segment(s) of the hierarchy.

At execution time, PEM references a control block referred to as the data base work area (DBWA). The DBWA establishes addressability to data groups within the hierarchical definition. PEM furthermore handles positioning for the application. The application itself simply requests data and remains unconcerned with physical structure.

VSAM efficiency is enhanced by PEM's own data management facilities. When an application instructs PEM to write a data group, the associated data is updated in the output buffer. PEM does not actually write the record until the PEM transaction ends, the user requests an end, or a subsequent read requires a refresh or repositioning.

When a request to read a data group is issued, PEM first attempts to locate the logical record in existing buffers.

IMS Data Base Management

PEM insulates the application program from the physical data structure. Consider IMS as an example. Application programmers do not need to know or understand IMS call structure syntax or definition, maintenance, and utilization of Program Control Blocks or Segment Search Arguments. Data Base Definitions and Program Specification Blocks are still required as PEM maintains a native IMS environment. PEM simply shields the application from the physical data structure.

Notes:



Hierarchical Database Activities

Hierarchical Data Base activities access Hierarchical Data Bases. Hierarchical Data Bases differ from Sequential Data Bases because they can be composed of more than one data group, they can be accessed randomly, and some Data Groups can have a logical dependent relationship to other Data Groups. Please review the chapter on Data Groups and Data Bases for a discussion of the types of Data Groups that can be defined in a Hierarchical Data Base.

A Hierarchical Data Base Activity can access one or all of the Data Groups listed on the Hierarchical Data Base Definition. The Base Data Group must always be accessed. Because of the method used by PEM to retrieve the individual Data Groups on the data base, the order in which the Data Groups are listed on the Data Base Activity must match the order in which they are listed on the Data Base Definition.

Each data group in a Data Base Activity has a USE CODE and a DFLT ACTION. The following types of access may be specified in the use code.

SEQUENTIAL INPUT (USE CODE "I")

Data will be retrieved in a sequential manner. Where possible, the data base system will be told to anticipate a request for the next unit of data and should manage the I/O buffers accordingly. Output actions are not valid for this use code.

SEQUENTIAL OUTPUT (UPDATE) (USE CODE "O")

I/O buffers are to be managed as in the sequential input use code. The difference is both input and output actions may be used.

RANDOM INPUT (USE CODE "R")

Data is retrieved based on the value of the key field in the data group. No anticipatory buffering is done. Sequential actions and output actions are invalid.

RANDOM UPDATE (USE CODE "U")

Similar to random input except that random output actions may also be used.

The Action code may be placed in the Data Group Action Field by the application program, or the Activity Definition may contain DEFAULT Action Codes. These action codes, along with the use code for the data group in the Activity Definition, tell PEM which operations to perform so that the appropriate request may be made to the Data Base Management System.

Notes:



Umbrella Programming

How PEM Uses Data Bases

Not all USE and ACTION CODES are valid for all types of Data Groups. For example, POSITIONAL Data Groups can only be accessed sequentially because they are not keyed. BASE Data Groups, however, can always be accessed randomly because they are always keyed. Because of the different characteristics of the data groups within a data base, you may find that when you create a data base activity, not all of the data groups listed on the activity will have the same use and action codes.

An example of a Hierarchical Data Base Activity with different USE and ACTION Codes follows.

The screenshot shows a terminal window with the CSC logo at the top left. The title is "HDB Activity 47020". The screen displays the following information:

48303 ACTION COMPLETE (HDB ACTS)

----- ACTIVITY DEFINITION INQUIRY/MAINTENANCE -----

COMMAND	OWNER ==> TRD CC# ==>
ACTIVITY ID	===== > 47020
ACTIVITY MNEMONIC	====>
DESCRIPTION	===== > RANDOM READ EMP DB-INPUT ONLY

EFF DATE ==> 78/01/01
TYPE =====> HDB

HIERARCHICAL DB
DATA BASE NAME ==> EMP
DISPOSITION =====> WAIT
2NDRY KEY =====>

*	DGID USE DEFLT	*	DATA GROUPS	*	DGID USE DEFLT
47100 R	KEYEQ	47110 I	READ	47130 I	READ
47140 I	READ	47120 I	READ		

LAST CHG:DATE 88/06/07 TIME 16.03.27 CC# SRCE EDU8610 OPER GLT
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-DBID 11-CPY ..-NXTT ..-NEW ..-NXTA

© 2000 CSC. All rights reserved.

Notes:



Rule of Accessability

CSC

Rule of Accessibility

- Data Groups Definitions
- HDB Definition
- HDB Activity Definition

© 2001 CSC. All rights reserved.

The data base activity is the mechanism to access hierarchical data bases. They are usually issued from an application program.

Actually, there are three types of entries on the Process Dictionary required for a successful access. These are:

DATA GROUPS Contain data, action, and result.

DATA BASE DEFINITIONS Provide the definition of the data base, the access method or DBMS to be used, and the valid data groups and hierarchical dependencies for hierarchical data bases.

DATA BASE ACTIVITIES Supply the ID of the data base as well as a list of data group to be accessed, use codes, and default action codes.

Notes:



Checking Result Fields



Check Result Fields

- After a Data Base Activity
 - Check TCB - Result
 - Check Data Group Result Fields

© 2000 CSC. All rights reserved.

It is the responsibility of the application program issuing the activity to check the result of the action when control is returned to the program. The appropriate result field(s) should be checked to determine the status of the requested activity.

After a data base activity, check TCB-RESULT. If it is equal to TCB-OK, the DB activity was successful. A not equal to TCB-OK indicates that some part of the DB activity failed, such as, end-of-data. You must then check the result field of the data groups referenced in the data base activity to determine the result to the requested action.

If the result field in the base data group or any dependent data group is not equal to DGR-OK, the data group's result field will contain the error code that corresponds to those in the delivered data group result code copybook.

Notes:



Hierarchical Database Access Research Exercise



1. For each type of HDB, the USE field for the data groups must be specified on the data base activity. List and explain the uses for each type.

HDB ACTIVITY

2. List the ACTION and RESULT codes below.

ACTIONS	HDB ACTIVITY	RESULTS
---------	--------------	---------



Umbrella Programming

How PEM Uses Data Bases

3. Go to the Process Dictionary and find the data base definition for the hierarchical data base, EMP.

- How many data groups in this data base?

-
- What is the base data group?

-
- Which field(s) make up the key to this data base?

-
- List the dependent data groups and the type for each.

4. Draw the data group relationship for the EMP Data Base.



Problem Specifications—HDB Activity



You are to add an HDB activity to read the EMP data base.

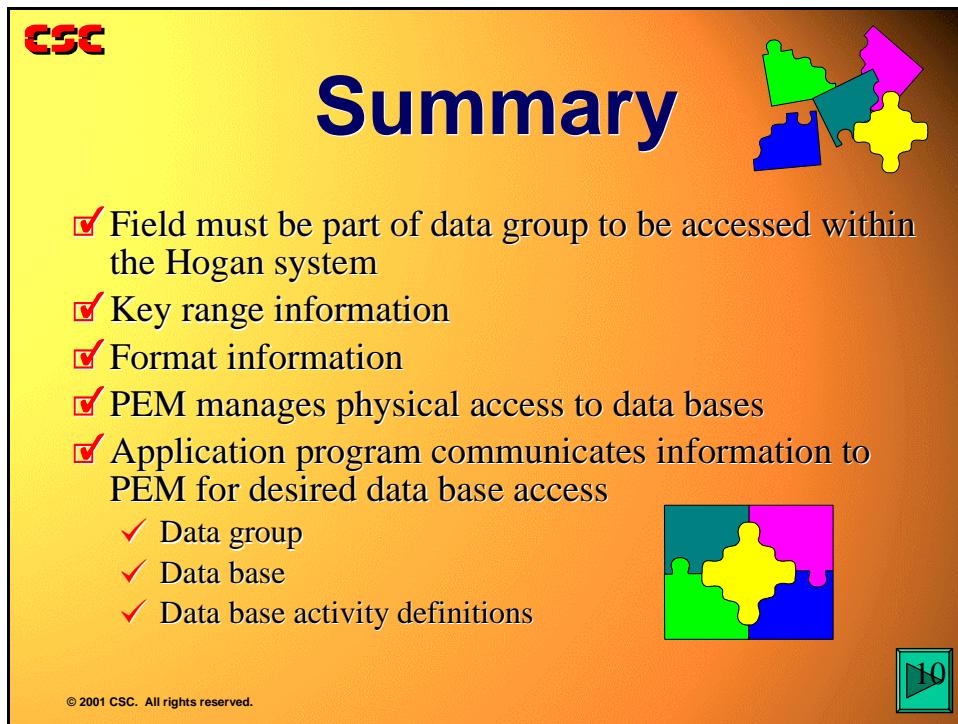
Follow these specifications:

1. Use activity number 9996xx, where xx is your group number.
2. Access is random.
3. All data groups defined in EMP, except 47110, are needed for this read.
4. Processing should continue after this activity is issued.
5. EMP has no secondary key.
6. Use the owner application and change control created in the Change Control Exercise.

Notes:



Summary



The slide features a yellow background with a red CSC logo in the top left corner. In the top right, there's a graphic of four interlocking puzzle pieces in green, blue, yellow, and pink. At the bottom left, a small text reads "© 2001 CSC. All rights reserved." On the bottom right, there's a small icon of a person sitting at a desk with a computer monitor.

Summary

- Field must be part of data group to be accessed within the Hogan system
- Key range information
- Format information
- PEM manages physical access to data bases
- Application program communicates information to PEM for desired data base access
 - Data group
 - Data base
 - Data base activity definitions

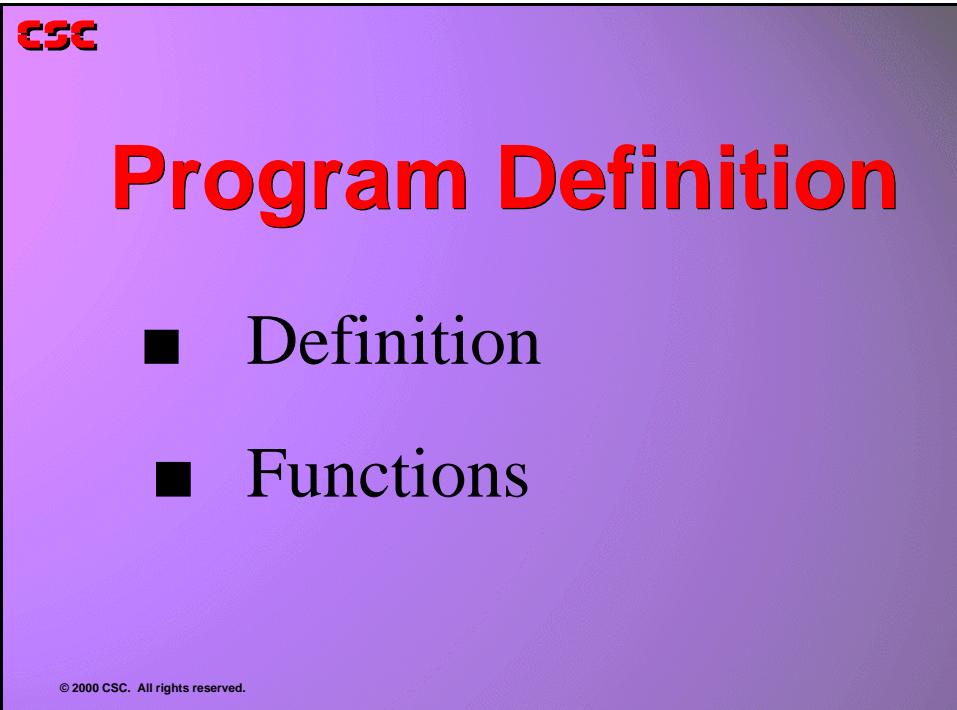
- A field has to be part of a data group for it to be accessed within the Hogan system.
- PEM manages the physical access to data bases.
- PEM provides for a hierarchical view of a VSAM data base.
- The application program communicates to PEM the information needed for a desired data base access. It does so through data group, data base, and data base activity definitions.
- Hierarchical data bases are logically defined by one or more data groups. There must be a base data group that contains the record key.
- An HDB in a VSAM environment can have a PEM format or a FIXED format.
- Information about the DL/I DBD and segments can be obtained from the HDB definition on the Process Dictionary.



Program Definition

10

Purpose



The slide has a purple gradient background. In the top left corner, there is a small red CSC logo. The main title "Program Definition" is centered in large red font. Below it, two black square bullet points list "Definition" and "Functions". At the bottom left, there is a small copyright notice: "© 2000 CSC. All rights reserved."

csc

Program Definition

- Definition
- Functions

© 2000 CSC. All rights reserved.

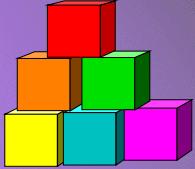
Notes:



Topics

CSC

Topics



- Program definition
- Program execution
- The UTCB
- Activity execution

© 2000 CSC. All rights reserved.

Objectives

CSC

Objectives



- Identify fields on program definition
- Describe how control is passed among program running under PEM
- Describe how addressability to data groups is established
- Name the UTCB and list some important fields
- Describe how activities are authorized and executed
- Define halfword and fullword technology

© 2000 CSC. All rights reserved.



Program Definition

All application programs that run in the PEM logical environment must be defined in the Process Dictionary. In each Program Definition entry the Load Module ID, language, data groups used, and activities used are listed.

To display a Program Definition:

Enter **U PGM** from a cleared screen followed by either the Program ID or Linkname.

```
U PGM 8108  
or  
U PGM I37008
```

It is also possible to jump to a Program Definition. Entering either of the following commands will take you to the Program Definition for program ID 8108.

```
COMMAND ==> =1.3.8108  
or  
COMMAND ==> =PGM 8108  
or  
COMMAND ==> =PGM I37008
```

CSC

Program Definition 1

ACTION SUCCESSFUL

PROGRAM DEFINITION INQUIRY/MAINTENANCE		
COMMAND ==> INQ	OWNER ==> FSS CC# ==>	
PROGRAM ID ==> 8108	EFF DATE ==> 78/01/01	PEM TECHNOLOGY ==> HALFWORD
LINKNAME ==> I37008	LANGUAGE ==> COBOL	SOURCE NAME ==> I37008
DESCRIPTION => SPLIT REPORT DS		STATUS ==> TEST
HIGH, MED, OR LOW USAGE? ==> LOW	USED ONLINE, BATCH OR BOTH? ==> BATCH	

DATA GROUPS USED BY PROGRAM

* PP ---DGID---	* PP ---DGID---	* PP ---DGID---
1 1850	2 13601	3 3601
4 5700	5 3432	6 9206
7 -DYNAMIC-	8 3210	9 8007
10 13210	11 5508	12 8250

AUTHORIZED ACTIVITIES

| * -ACTIVITY- |
|--------------|--------------|--------------|--------------|--------------|--------------|
| 1350 | 5810 | 8044 | 13274 | 13276 | 16700 |
| 16705 | 16715 | 16719 | 16720 | 16726 | 16727 |
| 16728 | 16729 | 16730 | 16732 | 16733 | 16735 |
| 16736 | 16744 | 16747 | 16748 | 16749 | 16757 |

LAST CHG:DATE 92/01/08 TIME 13.19.59 CC# 53909 SRCE F11D OPER KSP
PF: 2-XREF 3-PLVL 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-BYLINK 11-BYID ..-ACTS

© 2000 CSC. All rights reserved.



Umbrella Programming

Program Definition

To scroll forward press PF8 or enter command **SF**.

CSC

Scroll Forward

ACTION SUCCESSFUL

PROGRAM DEFINITION INQUIRY/MAINTENANCE		
COMMAND ==> SF	OWNER ==> FSS CC# ==>	
PROGRAM ID ===> 8108	EFF DATE ==> 78/01/01	PEM TECHNOLOGY => HALFWORD
LINKNAME ===> I37008	LANGUAGE ==> COBOL	SOURCE NAME ==> I37008
DESCRIPTION => SPLIT REPORT DS		STATUS ==> TEST
HIGH, MED, OR LOW USAGE? ==> LOW	USED ONLINE, BATCH OR BOTH? ==> BATCH	

----- DATA GROUPS USED BY PROGRAM -----

* PP ---DGID---	* PP ---DGID---	* PP ---DGID---
12 8250	13 5607	14 8018
15 5941	16 1301	17 9
18 13266	19 8102	20 8180
21 DYNAMIC		

----- AUTHORIZED ACTIVITIES -----

| * -ACTIVITY- |
|--------------|--------------|--------------|--------------|--------------|--------------|
| 16770 16782 | 16783 16784 | 16785 16786 | | | |
| 16787 16788 | 16797 16798 | 16880 16902 | | | |
| 16909 16911 | 16912 16914 | 16915 16917 | | | |
| 16919 16922 | 16925 16926 | 16930 16931 | | | |

LAST CHG:DATE 92/01/08 TIME 13.19.59 CC# 53909 SRCE F11D OPER KSP
PF: 2-XREF 3-PLVL 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-BYLINK 11-BYID ..-ACTS

© 2000 CSC. All rights reserved.

To scroll backward press PF7 or enter command **SB**.

CSC

Scroll Backward

ACTION SUCCESSFUL

PROGRAM DEFINITION INQUIRY/MAINTENANCE		
COMMAND ==> SB	OWNER ==> FSS CC# ==>	
PROGRAM ID ===> 8108	EFF DATE ==> 78/01/01	PEM TECHNOLOGY => HALFWORD
LINKNAME ===> I37008	LANGUAGE ==> COBOL	SOURCE NAME ==> I37008
DESCRIPTION => SPLIT REPORT DS		STATUS ==> TEST
HIGH, MED, OR LOW USAGE? ==> LOW	USED ONLINE, BATCH OR BOTH? ==> BATCH	

----- DATA GROUPS USED BY PROGRAM -----

* PP ---DGID---	* PP ---DGID---	* PP ---DGID---
1 1850	2 13601	3 3601
4 5700	5 3432	6 9206
7 -DYNAMIC-	8 3210	9 8007
10 13210	11 5508	12 8250

----- AUTHORIZED ACTIVITIES -----

| * -ACTIVITY- |
|--------------|--------------|--------------|--------------|--------------|--------------|
| 1350 5810 | 8044 13274 | 13276 16700 | | | |
| 16705 16715 | 16719 16720 | 16726 16727 | | | |
| 16728 16729 | 16730 16732 | 16733 16735 | | | |
| 16736 16744 | 16747 16748 | 16749 16757 | | | |

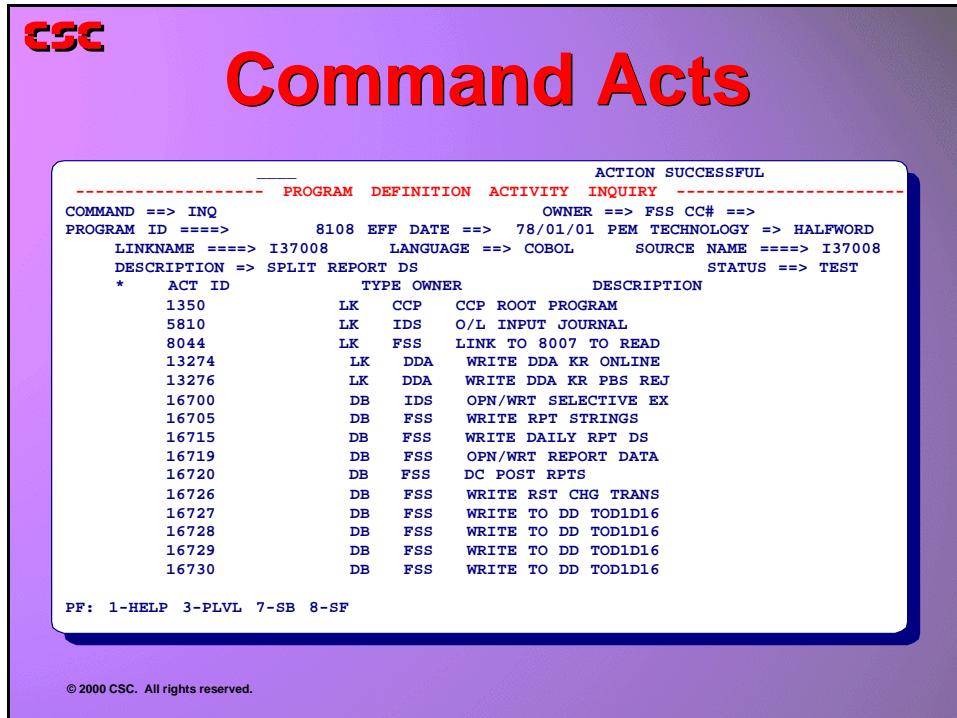
LAST CHG:DATE 92/01/08 TIME 13.19.59 CC# 53909 SRCE F11D OPER KSP
PF: 2-XREF 3-PLVL 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-BYLINK 11-BYID ..-ACTS

© 2000 CSC. All rights reserved.



To display the authorized activities:

- Enter command ACTS.



Notes:



Program Execution



Program Execution

Requires a:

- Link Activity
- Executes as sub-program to PEM
- Data Groups are passed as parameters via standard linkage convention
- Activities must be authorized

© 2000 CSC. All rights reserved.

An application program is executed only as the result of a requesting link activity. This activity may be one in the transaction's definition, or it may be requested by another application program. From whichever source the link activity is issued, the program definition indicates to PEM the load module name containing the program to be executed.

PEM executes application programs as called subprograms. Before an application program is executed, PEM allocates the space for all of the data groups referenced on the Program Definition. The data groups are passed, as parameters, to the application program in the same order as they are defined by the Program Definition. For COBOL programs, each data group corresponds to an 01 level statement in the linkage section. For ALC programs, register one points to a list of fullwords. Each word contains the address of a data group with the high-order bit ON of the last word.

Notes:



Establishing Addressability

Data Usage

All application programs in the Hogan systems are called programs. PEM is the calling program and handles the establishment of addressability to the data. It requires that the order of the data groups correspond in three places:

CSC

Establishing Addressability

- Program Definition
- Linkage Section
- Procedure Division Using Statement
- Rule of Addressability:
 - The order of the Data Groups on the Program Definition **MUST MATCH** the order of DataGroup Copybooks in the Linkage Section, **MUST MATCH** the order of 01 Level Names listed in the Using Statement

© 2000 CSC. All rights reserved.

PEM's TCB is always the first data group for which addressability is established (parameter is passed).

Notes:



Umbrella Programming

The User Transaction Control Block

The User Transaction Control Block

PEM also passes a control block known as the User Transaction Control Block (UTCB). This is always the first parameter passed to the program. The UTCB is used by PEM to communicate to an application program the information pertinent to the current processing. Data such as the Transaction ID, the date and time PEM received the transaction, the source of the transaction, and so on is provided for the application program. The UTCB also contains fields that the application program uses to communicate with PEM.

```
**** START OF P49000D ***** TCB ****
* THIS COPYBOOK NOW DEFINES FOUR DISTINCT DATA GROUPS: *
* 1) D.G. 00001, THE USER TRANSACTION CONTROL BLOCK AND THE CDMF *
* APPLICATION CONTROL BLOCK; *
* 2) D.G. 00012, THE SYSIN INPUT DATA GROUP; *
* 3) D.G. 00013, THE SYSPRINT OUPUT DATA GROUP; AND *
* 4) D.G. 00010, THE SECURITY CONTROL BLOCK, THE TCB USER AREA, *
* THE USER TCB EXTENSION AREA, AND THE INTERNAL TCB EXTENSION *
* AREA (ALC PROGRAMS ONLY). *
***** SKIP1 ****
* DATA GROUP 00001 (USER TRANSACTION CONTROL BLOCK) *
*****
01 TRANSACTION-CONTROL-BLOCK.
 05 FILLER          PIC XXXX.
 05 TCB-TRANS-NO   PIC XXXX.
 05 TCB-CO-ID      PIC XX.
 05 TCB-APPL-ID    PIC XX.
 05 TCB-FUNC-ID    PIC XX.

 05 TCB-SOURCE-TYPE.
    10 TCB-SOURCE-TYPE-N   PIC S9(4) COMP.
      88 TCB-ONLINE        VALUE +3 +5 +6.
      88 TCB-BTCH          VALUE +4.
      88 TCB-APPL-SOURCE   VALUE +2.
      88 TCB-AUTHORIZATIONS VALUE +5 +6.
      88 TCB-MANNED-TELLER VALUE +5.
      88 TCB-UNMANNED-TELLER VALUE +6.

 05 TCB-ACTIVITY.
    10 TCB-ACTIVITY-N    PIC S9(04) COMP.
 05 TCB-RESULT        PIC XX.
 05 TCB-USER-DATA.
    10 FILLER            PIC X(6).
    10 TCB-DATA-GROUP   PIC XX.
    10 TCB-PARM-POS.
      15 TCB-PARM-POS-N  PIC S9(4) COMP.
 05 FILLER            REDEFINES TCB-USER-DATA.
    10 TCB-USER-INFO    PIC XXXX.
    10 TCB-USER-COND    PIC XX.
    10 FILLER            PIC X(4).

 05 FILLER            REDEFINES TCB-USER-DATA.
    10 TCB-USER-CC      PIC XX.
    10 TCB-USER-RESULT  PIC XX.
    10 TCB-USER-ENVMT   PIC XX.
    10 FILLER            PIC X(4).

 05 FILLER            REDEFINES TCB-USER-DATA.
    10 TCB-EOJ-CALL    PIC XXXX.
    10 FILLER            REDEFINES TCB-EOJ-CALL.
      15 TCB-SOT-CALL   PIC XXXX.
    10 FILLER            PIC X(6).
```



Umbrella Programming

The User Transaction Control Block

```

05 TCB-TIME          PIC S9(7)  COMP-3.
05 TCB-SYS-DATE     PIC S9(7)  COMP-3.
05 TCB-SOURCE        PIC X(8).
05 TCB-OPERATOR.
    10 TCB-UMBRELLA-OPERATOR   PIC X(8).
    10 TCB-OPERATOR-FILLER    PIC X(12).
    10 TCB-DXRF             REDEFINES TCB-OPERATOR-FILLER.
        15 TCB-DXRF-ID       PIC X(4).
        15 FILLER            PIC X(8).
    10 TCB-DYN-KEY-RANGE    REDEFINES TCB-OPERATOR-FILLER.
        15 TCB-DKR-ID        PIC X(4).
        15 FILLER            PIC X(8).
    10 TCB-ENQ              REDEFINES TCB-OPERATOR-FILLER.
        15 TCB-ENQ-ID        PIC X(8).
        15 FILLER            PIC X(4).
05 TCB-DESTINATION    PIC X(8).
05 TCB-TERM-DATA.
    10 TCB-PFKEY          PIC X.
        88 TCB-ENTER         VALUE QUOTE.
        88 TCB-PF01          VALUE '1'.
        88 TCB-PF02          VALUE '2'.
        88 TCB-PF03          VALUE '3'.
        88 TCB-PF04          VALUE '4'.
        88 TCB-PF05          VALUE '5'.
        88 TCB-PF06          VALUE '6'.
        88 TCB-PF07          VALUE '7'.
        88 TCB-PF08          VALUE '8'.
        88 TCB-PF09          VALUE '9'.
        88 TCB-PF10          VALUE ':'.
        88 TCB-PF11          VALUE '#'.
        88 TCB-PF12          VALUE '@'.
        88 TCB-PF13          VALUE 'A'.
        88 TCB-PF14          VALUE 'B'.
        88 TCB-PF15          VALUE 'C'.
        88 TCB-PF16          VALUE 'D'.
        88 TCB-PF17          VALUE 'E'.
        88 TCB-PF18          VALUE 'F'.
        88 TCB-PF19          VALUE 'G'.
        88 TCB-PF20          VALUE 'H'.
        88 TCB-PF21          VALUE 'I'.
        88 TCB-PF22          VALUE '>'.
        88 TCB-PF23          VALUE '..'.
        88 TCB-PF24          VALUE '<'.
        88 TCB-PFKEY-NOT-PRESENT  VALUE LOW-VALUE.
    10 FILLER            PIC XX.
05 TCB-GENP-LOG        PIC X(1).
    88 TCB-GENP-NO-LOGGING  VALUE 'N'.
    88 TCB-GENP-LOGGING    VALUE 'Y' ..
                                LOW-VALUE.
05 TCB-EFFECTIVE-DATE  PIC S9(7)  COMP-3.
05 TCB-DEVICE-TYPE-2.
    10 TCB-DEVICE-TYPE    PIC X.
        88 TCB-3270-2        VALUE 'A'.
        88 TCB-BATCH          VALUE 'B'.
        88 TCB-3270-1          VALUE 'C'.
        88 TCB-3604-DS1        VALUE 'D'.
        88 TCB-3604-DS3        VALUE 'E'.
        88 TCB-3604-DS4        VALUE 'F'.
        88 TCB-3600-JP          VALUE 'G'.
        88 TCB-3600-PB          VALUE 'H'.
        88 TCB-3600-LP          VALUE 'I'.
        88 TCB-3270-MOD1-PRINTER  VALUE 'J'.
        88 TCB-3270-MOD2-PRINTER  VALUE 'K'.
        88 TCB-TWX             VALUE 'L'.
        88 TCB-2470-MOD2        VALUE 'M'.

```



Umbrella Programming

The User Transaction Control Block

```
88 TCB-2740-MOD1           VALUE 'N'.
88 TCB-ALIEN-X             VALUE 'X'.
88 TCB-ALIEN-Y             VALUE 'Y'.
88 TCB-ALIEN-Z             VALUE 'Z'.
*
***** RESERVED FOR TCB-DEVICE EXPANSION TO PIC X(2)
    10 FILLER                 PIC X.
05 TCB-LONG-ACTIVITY-N      PIC S9(09) COMP.
05 TCB-LONG-ACTIVITY       REDEFINES TCB-LONG-ACTIVITY-N.
    10 TCB-LONG-ACT-HI        PIC XX.
    10 TCB-LONG-ACT-LO        PIC XX.
05 TCB-LONG-DGID-N         PIC S9(09) COMP.
05 TCB-LONG-DGID          REDEFINES TCB-LONG-DGID-N.
    10 TCB-LONG-DG-HI        PIC XX.
    10 TCB-LONG-DG-LO        PIC XX.
05 TCB-USER-CC-APP         PIC XX.
05 TCB-RESULT-2            PIC XX.
EJECT
*****
* CDMF APPLICATION CONTROL BLOCK *
*****
05 CDMF-CONTROL-BLOCK.
    10 CDMF-ACTION            PIC XX.
    10 CDMF-RESULT            PIC XX.
    10 CDMF-KEY-FIELDS.
        15 CDMF-FORMAT          PIC XXXX.
        15 CDMF-COID             PIC XX.
        15 CDMF-EFF-DATE         PIC S9(7)  COMP-3.
    10 CDMF-EXP-DATE          PIC S9(7)  COMP-3.
    10 CDMF-COID-FOUND        PIC XX.
        88 CDMF-DEFAULT-COID-FOUND  VALUE HIGH-VALUES.
    10 CDMF-EFF-DATE-FOUND    PIC S9(7)  COMP-3.
    10 CDMF-HIGH-USE-FLAG     PIC X.
        88 CDMF-HIGH-USE-ITEM    VALUE 'Y'.
        88 CDMF-NON-PURGEABLE   VALUE 'P'.
*
***** ITEM OWNERSHIP IS ALWAYS RETURNED IN CDMF-OWNER-APPLICATION.
***** OWNERSHIP MAY BE RETRIEVED AND UPDATED FROM DATA GROUP 48007
***** IF THIS FLAG IS SET TO A 'Y'. OWNERSHIP MAY BE UPDATED FROM
***** CDMF-OWNER-APPLICATION IF THIS FLAG IS SET TO A 'C'.
    10 CDMF-OWNER-APP-FLAG    PIC X.
        88 CDMF-OWNER-APP-REQUEST  VALUE 'Y'.
        88 CDMF-OWNER-APP-IN-CTL-BLK  VALUE 'C'.
    10 CDMF-ITEM-LOCATION      PIC X.
        88 CDMF-ITEM-FOUND-IN-TABLE  VALUE 'Y'.
    10 FILLER                  PIC X.
    10 CDMF-CC-NO              PIC X(4).
    10 CDMF-LAST-CHANGE-DATA.
        15 CDMF-LAST-CHANGE-DATE    PIC S9(7)  COMP-3.
        15 CDMF-LAST-CHANGE-TIME    PIC S9(7)  COMP-3.
        15 CDMF-LAST-CHANGE-CC-NO   PIC X(4).
        15 CDMF-LAST-CHANGE-SOURCE  PIC X(8).
        15 CDMF-LAST-CHANGE-OPER    PIC X(8).
    10 CDMF-SECONDARY-KEY-ID    PIC X(4).
    10 CDMF-SUBSTITUTE-DGID    PIC X(4).
*
***** THIS FIELD IS FOR INTERNAL UMBRELLA USE ONLY.
    10 CDMF-RELEASE-CTL-DG-LEN  PIC XX.
*
***** THIS FIELD IS FOR INTERNAL UMBRELLA USE ONLY.
    10 CDMF-RELEASE-CTL-FLAGS   PIC X.
*
***** THE ITEM APPLICATION OWNERSHIP IS ALWAYS RETURNED IN THIS
***** FIELD. THIS FIELD MAY ONLY BE USED IN AN UPDATE WHEN
***** CDMF-OWNER-APP-FLAG IS SET TO A 'C'.
```



Umbrella Programming

The User Transaction Control Block

```

      10 CDMF-OWNER-APPLICATION      PIC X(3).
      10 FILLER                      PIC X(2).
***** END OF DATA GROUP 00001 *****
      EJECT
*****
* DATA GROUP 12 (SYSIN INPUT DATA GROUP) *
*-----*
* NOTE: PROGRAMS THAT REFERENCE DATA GROUP 12 NEED NOT CODE THE *
* DATA GROUP ON THE PROGRAM DEFINITION. INSTEAD, YOU MAY REFER   *
* TO THIS AREA DIRECTLY & ISSUE ACTIVITIES THAT REFERENCE DG 12. *
***** *****
*
***** DATA GROUP CHAIN (DO NOT DESTROY)
      05 FILLER                      PIC X(8).
      05 DATA-GROUP-12.
          10 DG12-ACTION            PIC XX.
          10 DG12-RESULT             PIC XX.
          10 DG12-CARD-IMAGE        PIC X(80).
***** END OF DATA GROUP 00012 *****
      SKIP1
*****
* DATA GROUP 13 (SYSPRINT OUTPUT DATA GROUP) *
*-----*
* NOTE: PROGRAMS THAT REFERENCE DATA GROUP 13 NEED NOT CODE THE *
* DATA GROUP ON THE PROGRAM DEFINITION. INSTEAD, YOU MAY REFER   *
* TO THIS AREA DIRECTLY & ISSUE ACTIVITIES THAT REFERENCE DG 13. *
***** *****
*
***** DATA GROUP CHAIN (DO NOT DESTROY)
      05 FILLER                      PIC X(8).
      05 DATA-GROUP-13.
          10 DG13-ACTION            PIC XX.
          10 DG13-RESULT             PIC XX.
          10 DG13-CONTROL-CHAR       PIC X.
          10 DG13-PRINT-DATA         PIC X(132).
          10 FILLER                  PIC X(7).
***** END OF DATA GROUP 00013 *****
      SKIP1
*****
* DATA GROUP 10 (TCB EXTENSION AREAS) *
*-----*
* THIS DATA GROUP CONTAINS THE SECURITY CONTROL BLOCK, THE TCB    *
* USER AREA (FOR CLIENT USE), THE USER TCB EXTENSION AREA, AND    *
* THE INTERNAL TCB EXTENSION AREA.                                *
* ***** NOTE: DO NOT CODE THIS DATA GROUP ON YOUR PROGRAM        *
* ***** DEFINITION. INSTEAD, YOU SHOULD REFER TO THIS           *
* ***** AREA DIRECTLY SINCE IT IS PART OF THE USER              *
* ***** TRANSACTION CONTROL BLOCK.                               *
***** *****
*
***** DATA GROUP CHAIN (DO NOT DESTROY)
      05 FILLER                      PIC X(8).
      05 DATA-GROUP-10.
      EJECT
*****
* SECURITY CONTROL BLOCK *
*****
      10 SECURITY-CONTROL-BLOCK.
          15 SCB-ACTION            PIC S9(4)  COMP.
          15 SCB-ACTION-X REDEFINES SCB-ACTION
                                         PIC XX.
          15 SCB-RESULT             PIC S9(4)  COMP.
          88 SCB-AUTHORIZATION-VALID VALUE +0.
          88 SCB-AUTHORIZATION-FAILED VALUE +4.
          88 SCB-AUTHORIZATION-ERROR VALUE +8.

```



Umbrella Programming

The User Transaction Control Block

```
88 SCB-EXT-SECURITY-INACTIVE VALUE +12.
15 SCB-RESULT-X REDEFINES SCB-RESULT
      PIC XX.
15 SCB-VIOLATION-ACTION    PIC S9(4)  COMP.
88 SCB-ABEND-TASK          VALUE +0.
88 SCB-RETURN               VALUE +1.
15 SCB-LOGGING-FLAG        PIC X.
88 SCB-LOG-EXCPTNS         VALUE 'Y'.
88 SCB-BYPASS-LOG          VALUE 'N'.
15 SCB-PROCESSING-TYPE     PIC X.
15 SCB-PEM-FLAG1           PIC X.
15 SCB-FUTURE-FLAGS        PIC X(3).
15 SCB-FORMAT-NUMBER       PIC XXXX.
15 SCB-FORMAT-NAME         PIC X(10).

*
***** THIS FIELD IS FOR ALC PROGRAMS ONLY
15 SCB-ADDR-FMT-TARGET-DG  PIC XXXX.

*
15 SCB-TARGET-DG-ID        PIC XXXX.
15 SCB-ITEM-OWNER          PIC X(3).
15 SCB-PREV-OWNER          PIC X(3).
15 SCB-MESSAGE-NO.
20 SCB-MESSAGE-NO-N        PIC S9(4)  COMP.
15 SCB-EXCEPTION-MESSAGE.
20 SCB-RULE-NAME          PIC X(40).
20 FILLER                 PIC X(4).
15 SCB-USER-DATA           PIC X(25).
15 SCB-RESERVED            PIC X(8).
15 FILLER                 PIC X(1).

EJECT
*****
* TCB USER AREA
*-----*
* THIS AREA IS RESERVED FOR CLIENTS AND WILL NEVER BE USED BY *
* HOGAN SYSTEMS.
*****
10 TCB-USER-AREA.
15 FILLER                 PIC X(104).

SKIP1
*****
* USER TCB EXTENSION AREA
*-----*
* THIS AREA IS RESERVED FOR NEW TCB FIELDS TO BE ADDED AND *
* UPDATED BY HOGAN SYSTEMS.
*****
10 TCB-EXTENSION-AREA.
*      CURSOR POSITION AFTER A DEBLOCK; ROW AND COLUMN
15 TCB-ROW                  PIC S9(4)  COMP.
15 TCB-COLUMN                PIC S9(4)  COMP.
15 TCB-BATCH-DISP-OPTION     PIC X.
88 TCB-BATCH-DISP-DUMP       VALUE LOW-VALUES.
*      BATCH DISPLAY TO SYSPRINT IN DUMP FORMAT.
88 TCB-BATCH-DISP-FORMAT    VALUE 'F'.
*      BATCH DISPLAY TO SYSPRINT IN SCREEN FORMAT.
88 TCB-BATCH-DISP-RETURN    VALUE 'R'.
*      BATCH DISPLAY DATA IN DG 47. NOT PRINTED.
15 FILLER                  PIC X.
15 TCB-DYN-TXN-ID           PIC X(008).
15 TCB-OPTIONS-1             PIC X.
88 TCB-RETURN-DATA-UNAVAILABLE VALUE X'01' X'03'.
88 TCB-RETURN-DB-NOT-AVAILABLE VALUE X'02' X'03'.
88 TCB-RETURN-DB-OR-DATA-UNAVA VALUE X'03'.
88 TCB-RETURN-ACT-NOT-DEFINED VALUE X'08'.
15 TCB-OPTIONS-2             PIC X.
15 FILLER                  PIC X(040).
```



Umbrella Programming

The User Transaction Control Block

```

15 TCB-SQL-ACTION.
20 TCB-SQL-ACTION-N      PIC S9(4) COMP.
15 TCB-SQL-RESULT.
20 TCB-SQL-RESULT-N      PIC S9(4) COMP.
15 TCB-SQL-DYNPLAN       PIC X(008).
15 TCB-SQL-CURPLAN       PIC X(008).
15 TCB-SQL-DYN-SUBSID    PIC X(004).
15 TCB-SQL-SUBSID        PIC X(004).
15 TCB-CKPT-COUNT         PIC S9(9) COMP.
15 FILLER                 PIC X(005).
15 TCB-APPC-SERVICE-AREAS.
20 TCB-APPC-DATA-GROUP   PIC X(4).
20 TCB-APPC-SYSTEM-KEY   PIC X(4).
20 TCB-APPC-APPL-KEY     PIC X(8).
20 TCB-APPC-XMIT-IMMED   PIC X(1).
     88 TRANSMIT           VALUE 'Y'.
     88 TRAN-PREPARE-RECEIVE VALUE 'R'.
20 TCB-APPC-XMIT-ERROR   PIC X(1).
*     *88 ISSUE-ERROR      VALUE +1.
*     *88 ISSUE-ERROR-W-DATA VALUE +17.
*     *88 ISSUE-ABEND      VALUE +2.
20 TCB-APPC-MORE-DATA    PIC X(1).
20 TCB-APPC-BYPSS-ERR    PIC X(1).
20 TCB-APPC-RETURN-CDE   PIC X(6).
15 FILLER                 PIC X.
15 TCB-VSAM-RELATIVE-NUMB PIC X(4).
15 TCB-MONETARY-KEY       PIC X(3).          TCB$MON
15 TCB-PRES-CURRENCY-CD   PIC X(3).          TCB$PCUR
15 TCB-LANGUAGE-KEY       PIC X(3).
15 TCB-PACK-COLLECT-NAME  PIC X(18).
15 TCB-DYN-COLLECT-NAME   .
20 TCB-DYN-PLAN-PREF      PIC X(2).
20 TCB-DYN-COMP-GRP       PIC X(8).
20 TCB-DYN-PROC-GRP       PIC X(4).
20 TCB-DYN-KEY-RANGE      PIC X(4).
15 FILLER                 PIC X(8).
15 TCB-DB2-KRID           PIC X(4).
15 TCB-LANG-ENABLED        PIC X(1).
15 TCB-DEFAULT-LANG       PIC X(3).
15 TCB-LANG-ENCODE-FLAG   PIC X(1).
15 TCB-PROC-GROUP          PIC X(4).
15 TCB-PROC-GROUP-BRANCH   PIC 9(5) COMP-3.
15 TCB-PROCESSING-ID       REDEFINES
                           TCB-PROC-GROUP-BRANCH PIC 9(5) COMP-3.
15 TCB-UDFL-LANG          PIC X(3).
15 TCB-SPS-IMplode-LANG   PIC X(3).
15 TCB-OPERATOR-REGION     PIC 9(5) COMP-3.
15 TCB-OPERATOR-BRANCH     PIC 9(5) COMP-3.
15 TCB-DYN-LANG            PIC X(3).
15 TCB-DMAP-ID             PIC X(7).
15 TCB-DB2-PROC-GRP-ID     PIC X(4).
15 TCB-DB2-PROC-GRP-BRANCH PIC 9(5) COMP-3.
15 TCB-DB2-PROCESSING-ID   REDEFINES
                           TCB-DB2-PROC-GRP-BRANCH PIC 9(5) COMP-3.
15 TCB-USER-WORK-AREA       .
20 TCB-DYNAMIC-DG-ADDR     POINTER.
20 TCB-DYNAMIC-DG-ADDRESS   REDEFINES
                           TCB-DYNAMIC-DG-ADDR POINTER.
20 FILLER                 PIC X(8).
15 TCB-HDP-RESERVED        PIC X(10).
15 TCB-DFLT-LANG-ENCODE-BYTE
                           PIC X(01).
15 TCB-DB2-TEST-POOL-ID    PIC X(02).
15 TCB-PRES-CURR-RND-IN    PIC X(01).          TCB$PCRI

```

* DO NOT ROUND PRESENTATION CURRENCY - ADDS AN EXTRA DECIMAL DIGIT



Umbrella Programming

The User Transaction Control Block

```
     88 TCB-DO-NOT-ROUND-PRES-CURR           VALUE 'N'.
* ROUND PRESENTATION CURRENCY
     88 TCB-ROUND-PRES-CURR                 VALUE 'Y'.
15   TCB-RAND-REC-ADDR-NR    PIC S9(9) COMP.
15   TCB-JOB-ID          PIC X(8).
15   TCB-OPERATOR-PROCESSING-ID PIC 9(5) COMP-3.
15   TCB-DFLT-CENTURYWINDOW-CUTOFF PIC S999 COMP-3.
15   TCB-CICS-STARTCODE      PIC XX.
     88 TCB-CICS-START-DPL                VALUE 'D '.
     88 TCB-CICS-START-DPL-SYNC          VALUE 'DS'.
     88 TCB-CICS-START-TD-TRIGGER        VALUE 'QD'.
     88 TCB-CICS-START-CMD              VALUE 'S '.
     88 TCB-CICS-START-CMD-DATA         VALUE 'SD'.
     88 TCB-CICS-START-FEPI             VALUE 'SZ'.
     88 TCB-CICS-START-TERMINAL-INPUT   VALUE 'TD'.
     88 TCB-CICS-START-USER-ATTACH     VALUE 'U '.
15   TCB-DYNAMIC-TCPPIP-ID.
20   TCB-DYNAMIC-TCPPIP-NAME PIC X(8).
20   TCB-DYNAMIC-TCPPIP-LOC  PIC X.
15   TCB-TCPPIP-DATA-LENGTH      PIC S9(9) COMP.
15   TCB-TCPPIP-STATUS          PIC X.
     88 TCB-TCPPIP-CLIENT            VALUE X'80'.
     88 TCB-TCPPIP-CHILD-SERVER     VALUE X'40'.
15   FILLER                      PIC X(647).
15   FILLER                      PIC X(200).

SKIP1
***** END OF DATA GROUP 00010 *****
SKIP1
*
***** END OF P49000D *****
```

Notes:



Activity Execution



Activity Execution

- Activity is a unit of work
- Move Activity ID to TCB
- Call PEM

© 2000 CSC. All rights reserved.

Much of the work of the application program is performed by activities. An important function of the Program Definition is to indicate to PEM which activities the program is authorized to request.

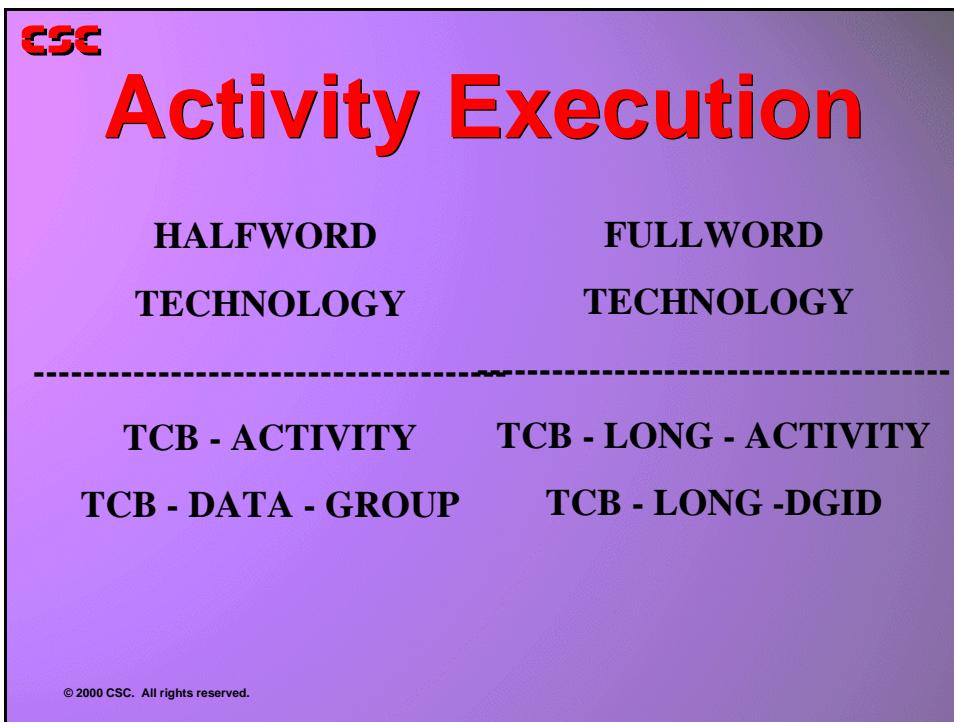
The application program requests a specific activity by placing the number of the activity ID in the User Transaction Control Block prior to calling PEM. PEM uses the activity ID as a key into the Process Dictionary. The Activity Definition tells PEM what must be done to satisfy the request. Following the execution of the activity, control usually returns to the application program, which can resume processing from the point at which the activity was issued.

Program Definitions indicate whether the program uses halfword technology or fullword. Different fields are used in the TCB depending on this indicator as follows:



Umbrella Programming

Activity Execution



The slide has a purple background. In the top left corner is the CSC logo. The title "Activity Execution" is in large red font. Below the title are two rows of text: "HALFWORD TECHNOLOGY" and "FULLWORD TECHNOLOGY". A horizontal dashed line separates this from the bottom section. The bottom section contains four items: "TCB - ACTIVITY", "TCB - LONG - ACTIVITY", "TCB - DATA - GROUP", and "TCB - LONG -DGID". At the very bottom of the slide is a small copyright notice: "© 2000 CSC. All rights reserved."

CSC

Activity Execution

HALFWORD TECHNOLOGY

FULLWORD TECHNOLOGY

TCB - ACTIVITY

TCB - LONG - ACTIVITY

TCB - DATA - GROUP

TCB - LONG -DGID

© 2000 CSC. All rights reserved.

Programs developed before the CDMF version of Umbrella use halfword technology.

While support for halfword programs is to be phased out over future releases, Umbrella System Version 3 Release 0 CST 1 allows existing halfword programs to execute fullword activities. This specifically supports Procedure Division copybooks written to use fullwords and be included in both existing halfword and fullword programs.

Notes:



Problem Specification—Program Definition



Build a Program Definition for program ID 9994xx (where xx is your group number).

1. The program is fullword.
2. The program is coded in COBOL.
3. The linkname is Z9994xx.
4. Enter a description that includes your group number.
5. The program refers to the following Data Groups:

TRANSACTION CONTROL BLOCK	1
DATE CONTROL BLOCK	2000
EMP DATA BASE DATA GROUP	47100
EMP DATA BASE DATA GROUP	47110
EMP DATA BASE DATA GROUP	47130

6. The following activities need to be authorized for use by this program.
 - 9996xx, where xx is your group number.
 - 1900
7. The program's current status is test.
8. The program will be of low usage.
9. The program may be used both in online and batch.
10. Use the owner application and change control created in the Change Control Exercise.

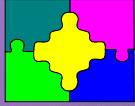
Notes:



Summary



Summary



© 2000 CSC. All rights reserved.

- Information such as linkname, data groups used in the program, and authorized activities must be specified using a program definition in defining programs to PEM.
- Data groups are passed as parameters to the application program based on the program definition.
- The User Transaction Control Block (UTCB) is also passed to the application program. It passes information between the application program and PEM.

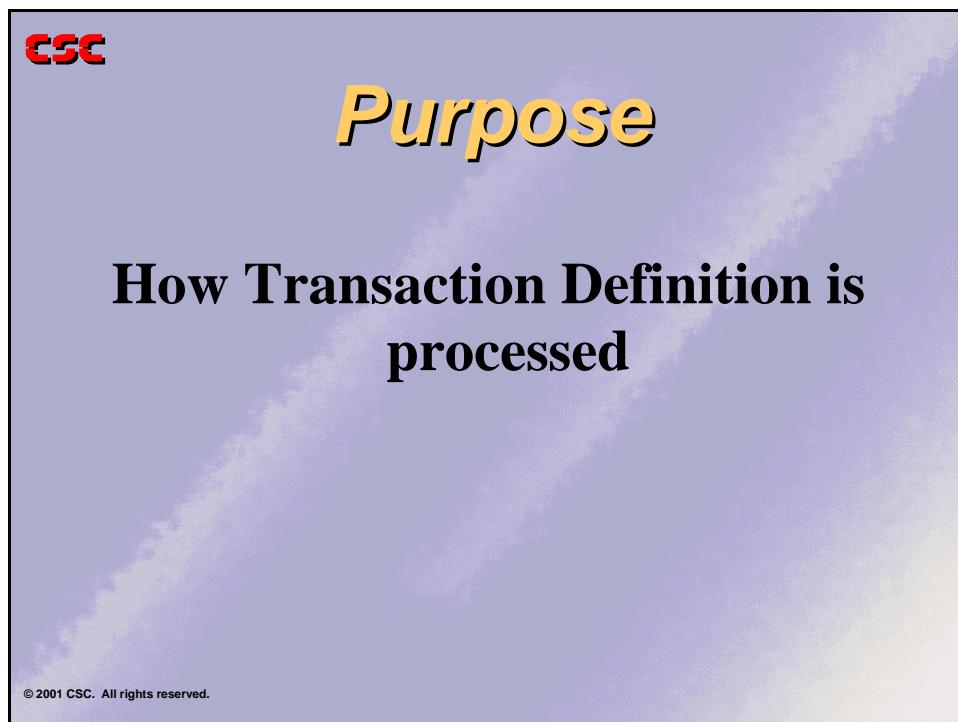
Notes:



Transaction Definition

11

Purpose



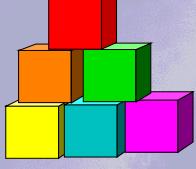
Notes:



Topics

CSC

Topics



- Transaction Definitions
- Operating Environment Interface
- Processing flow within a PEM Transaction

© 2000 CSC. All rights reserved.

Objectives

CSC

Objectives



- List five parts of a key to transaction definition
- Explain how a transaction definition directs application processing
- Explain how a PEM transaction interfaces with the operating environment

© 2000 CSC. All rights reserved.



Transaction Definition

A transaction contains one or more activities that perform a complete unit of work. In data processing, the word transaction usually conjures up thoughts of online work. Typically there is a direct correspondence between an online system's identification of a transaction and PEM's identification of a transaction. In a batch environment, usually there is one PEM transaction definition but thousands of application input data transactions. A batch PEM transaction is analogous to a batch job step or batch execution.

PEM transactions are defined in the Process Dictionary and keyed by application, function, source type, company and effective date. Each Transaction Definition contains a list of Activity Definition IDs. The activities are executed in the order specified on the transaction definition. The transaction definition is the master execution driver for the process. A transaction definition may contain one to many activity IDs. An activity ID may be a link to a program that issues other activities. An end program activity will cause PEM to return control to the transaction definition. After the last activity on the transaction definition is executed, the transaction is ended.

CSC

Transaction Definition 1

----- UMBRELLA TRANSACTION DEFINITION INQUIRY/MAINTENANCE -----					
COMMAND ==>	OWNER ==> ...	CC# ==>
APPLICATION ID ======>
FUNCTION ID ======>
SOURCE ID ======>
COMPANY ID LIST ======>
EFFECTIVE DATE ======>
TRANSACTION CODE ======>
TRANSACTION DESCRIPTION ==>
DL/I PSB NAME ======>
DB2 PLAN NAME ======>
APPC: REMOTE PEM ==> .	CONVERSATION ==>
*	ACTIVITY	*	TRANSACTION ACTIVITIES	*	ACTIVITY
*
.
.
.
LAST CHG:DATE	TIME	CC#	SRCE	OPER
PF:

© 2000 CSC. All rights reserved.



Umbrella Programming

Transaction Definition

To display the following transaction definition:

- Enter **U T 88.8011.4.ALL** from a cleared screen and press ENTER.

CSC

Transaction Definition 2

UMBRELLA TRANSACTION DEFINITION		ACTION COMPLETE INQUIRY/MAINTENANCE
COMMAND ==> INQ		OWNER ==> UES CC# ==>
APPLICATION ID ======>	88	
FUNCTION ID ======>	8011	
SOURCE ID ======>	4	
COMPANY ID LIST ======>	ALL	U T 88.8011.4.ALL
EFFECTIVE DATE ======>	78/01/01	
TRANSACTION CODE ======>		
TRANSACTION DESCRIPTION ==>	UES QUOTATIONS SYSTEM BATCH TRANSACTION	
DL/I PSB NAME ======>	PSBUESA	
DB2 PLAN NAME ======>		
APPC: REMOTE PEM ==>	CONVERSATION ==>	
TRANSACTION ACTIVITIES		
*	ACTIVITY * ACTIVITY * ACTIVITY *	ACTIVITY
	488402	
LAST CHG:DATE 92/06/15 TIME 9:35:40 CC# SRCE UMB130 OPER		
PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-ACTS 11-TRAN ..-DEL ..-NEW		

© 2000 CSC. All rights reserved.

To display the activities associated with this transaction press PF10 or enter ACTS in the COMMAND field.

Notes:



Umbrella Transaction Definition Activity List

csc *Umbrella Transaction Definition Activity List*

UMBRELLA TRANSACTION DEFINITION			ACTION SUCCESSFUL	
			OWNER ==> UES CC# ==>	ACTIVITY LIST MAINT-----
COMMAND ==> INQ	APPL ==> 88	FUNC ==> 8011	SOURCE ==> 4	COMPANY LIST ==> ALL
EFFECTIVE DATE ==> 78/01/01	ONLINE TRANSACTION CODE ==>			
*	ACT ID	TYPE OWNER	DESCRIPTION	
	488402	LK UES	UES QUOTATIONS SYSTE	

PF 10

PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-GEN 11-TRAN ..-DEL

© 2000 CSC. All rights reserved.

To display the activity, enter **S** under the * and press PF6.

Notes:



Umbrella Programming

Transaction Definition

To Display the Individual Activity Definition

csc
Display Individual Activities

ACTION SUCCESSFUL				
----- UMBRELLA TRANSACTION DEFINITION INQUIRY/MAINTENANCE -----				
COMMAND ==>	OWNER ==> UES CC# ==>			
APPLICATION ID ======>	88			
FUNCTION ID ======>	8011			
SOURCE ID ======>	4			
COMPANY ID LIST ======>	ALL			
EFFECTIVE DATE ======>	78/01/01			
TRANSACTION CODE ======>				
TRANSACTION DESCRIPTION ==>	UES QUOTATIONS SYSTEM BATCH TRANSACTION			
DL/I PSB NAME ======>	PSBUESA			
DB2 PLAN NAME ======>				
APPC: REMOTE PEM ==>	CONVERSATION ==>			
TRANSACTION ACTIVITIES				
*	ACTIVITY *	ACTIVITY *	ACTIVITY *	ACTIVITY
S	488402			

LAST CHG:DATE 92/06/15 TIME 9:35:40 CC# SRCE UMB130 OPER
PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-ACTS 11-TRAN ..-DEL ..-NEW

© 2000 CSC. All rights reserved.

To display the activity, enter S under the * and press PF6.

Notes:



Activity Definition Selected From Transaction Definition

CSC

Activity Definition Selected from Transaction Definition

48308 ACTION COMPLETE (LINK ACTS)			
ACTIVITY DEFINITION INQUIRY/MAINTENANCE		OWNER ==> UES CC# ==>	
COMMAND ==> INQ	ACTIVITY ID ======>	488402	EFF DATE ==> 78/01/01
	ACTIVITY MNEMONIC ==>		TYPE =====> LINK
	DESCRIPTION ======>	UES QUOTATIONS SYSTEM BATCH ORDERS	
LINK			
PROGRAM ID ==> 488402			
*	DGID	*	*** DATA GROUPS ***
*	DGID	*	DGID *
DGID			
LAST CHG:DATE 92/06/15 TIME 9.35.40 CC# SRCE UMB130 OPER			
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-PGMID 11-CPY ..-NXTT ..-NEW ..-NXTA			

© 2000 CSC. All rights reserved.

To return to the transaction definition press PF3 or enter **PLVL** in the COMMAND field and press ENTER.

Notes:



Operating Environment Interface

Environment Dependent Considerations



Environment Dependent Considerations

- PEM transactions originally defined and executed in operating environment
- Responsibilities of operating environment are:
 - Start PEM execution
 - Pass information allowing location of transaction definition

© 2000 CSC. All rights reserved.

All PEM transactions must originally be defined and executed in an operating environment. The responsibilities of the operating environment are to start the execution of PEM and to pass enough information to PEM to allow the transaction definition to be located on the Process Dictionary. In many cases a user exit is also necessary to provide fields (Company ID and source type) that are keys to transaction definitions.

The source type field allows different versions of a transaction to be executed from different environments; for example, a transaction executed online might have a display activity. The same transaction executed in batch would not issue display activities. A transaction executed from a manned teller machine would have still a different source type and would require still another transaction definition.

Notes:



Listed below is a summary of how different environments obtain the transaction definition key.



Obtaining the Transaction Definition Key

BATCH

SYSIN card has CO/APP/FUNC. Source type is always 4

CICS/VS, IMS/DC

The Process Dictionary is read with the transaction code as the key to obtain the application code and the function ID. The user exit supplies the company and source type.

© 2000 CSC. All rights reserved.

Notes:



Umbrella Programming

Transaction Definition

Execution Under Different Environments

Application programs written under PEM may run in several operating system environments or be migrated from one to another.

Data bases may be defined to VSAM or IMS.

Teleprocessing monitors may be CICS or IMS DC.

Whatever the environment, the application program can be executed without modification.

This flexibility is possible because Hogan can tailor PEM for the target environment. PEM modules are structured as macros with parameters that are used in conditional assembly statements to govern which environment the assembly will support. This collection of modules is then link-edited into a single load module per environment. Environments that are preassembled and distributed with each release are:



Execution Under Different Environments

BATCHPEM	-	OS/MVS, XA, ESA, VSAM, BATCH
CICSPEM	-	OS/MVS, XA, ESA, VSAM, CICS/VIS
DLIPEM	-	OS/MVS, DL/I, BATCH
BMPPEM	-	OS/MVS, IMS/VS, BMP
IMSPEM	-	OS/MVS, IMS/DB-DC
IMSPEMF	-	OS/MVS, IMS/VS FAST PATH

© 2000 CSC. All rights reserved.

All application programs execute as called subprograms to PEM. They never interface directly with the operating environment. Instead, they make requests of PEM by issuing activities. PEM actually executes the activities, thereby accessing a data base, allocating storage, or performing some other function with the operating environment.



PEM/Operating Environment Interfaces

A unique PEM interface module is generated for each processing environment. PEM is always the first program executed by the processing environment. PEM will, in turn, execute the application programs.

This interface also establishes the PEM Transaction ID. The PEM Transaction ID is made up of the following five numeric fields.

csc

Environment Interfaces *Key Fields*

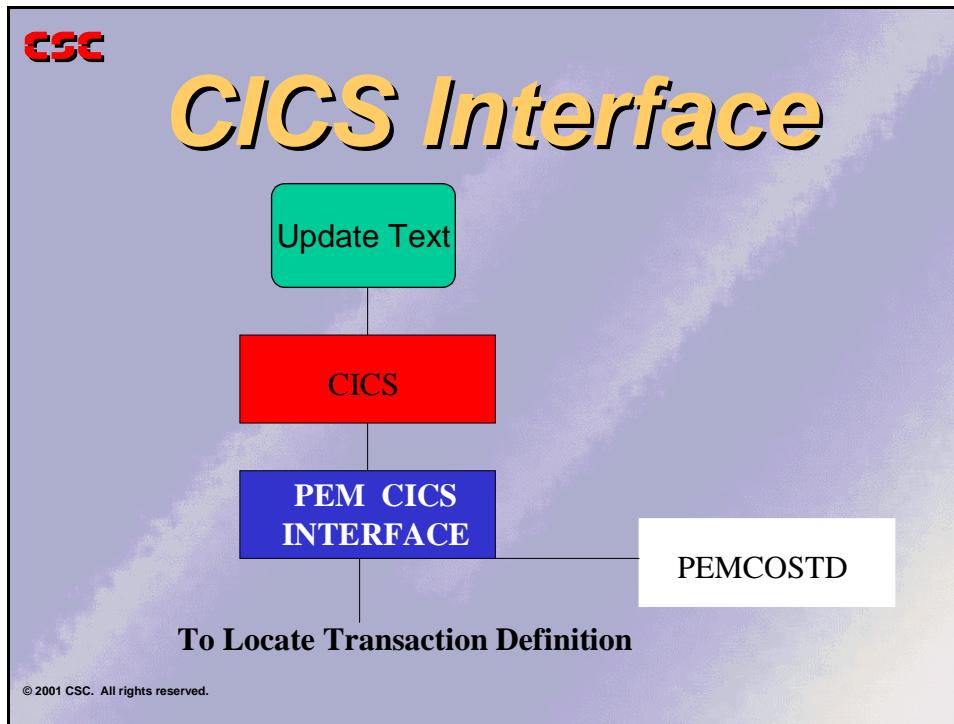
Company Number	range	1 - 65534
65535 = Default		
Application Number	range	1 - 65535
Function Number	range	1 - 65535
Source Type	assigned by PEM	
Effective Date	assigned by user	

© 2000 CSC. All rights reserved.

Notes:



CICS Interface

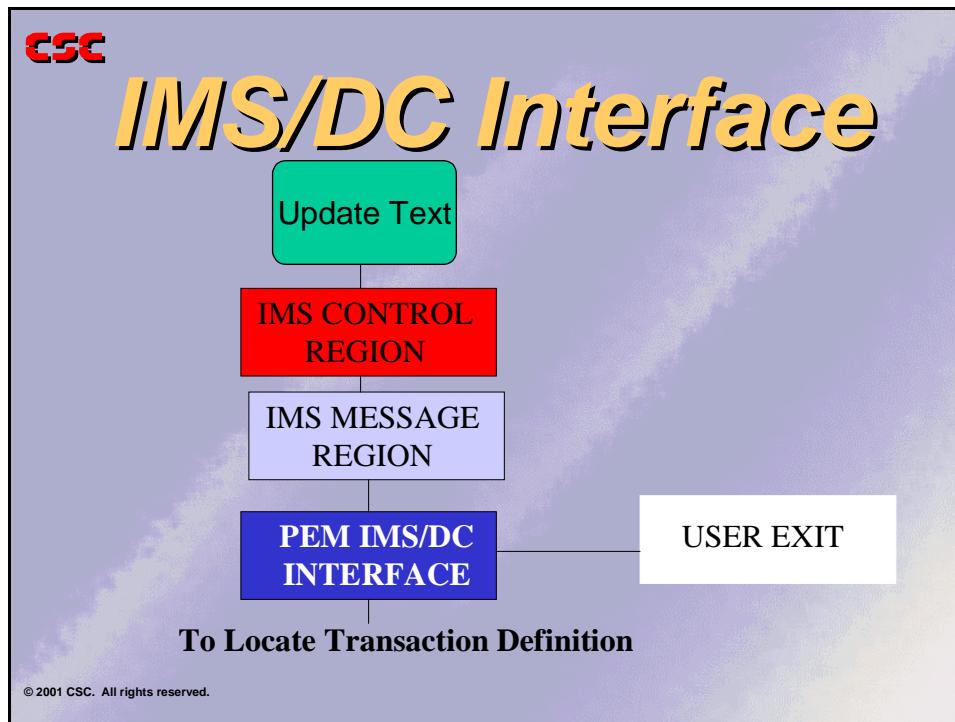


Terminal input is read by CICS. This input must contain a transaction code as prescribed by CICS. The PCT (Program Control Table) must specify the load module name of the PEM CICS interface for all transactions processed by PEM application programs. The PEM CICS interface program cross-references to the PEM application number and function number. The PEM company number and source type are set by a user-prepared exit. The exit program name is *PEMCOSTD*.

Notes:



IMS/DC Interface

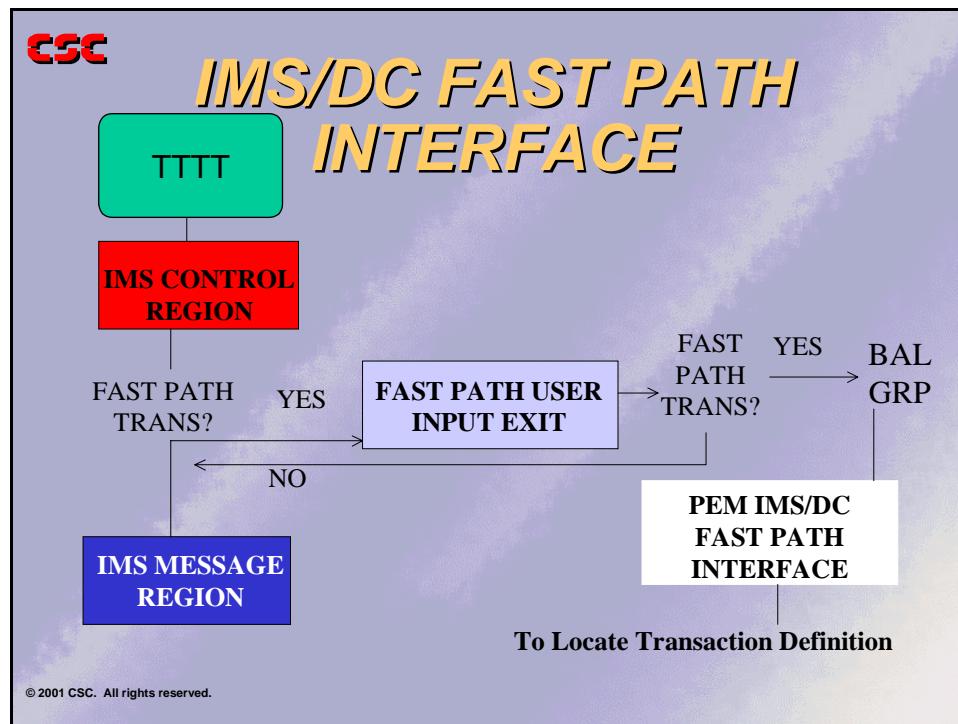


The terminal input is read by the IMS control region and placed on the appropriate message queue. The control region then dispatches the appropriate message processing region. The PSB name for all transactions processed by PEM application programs must be the name of the load module for the PEM IMS/DC interface. Because several PSBs may be required, this module must be link edited with an alias for each PSB. The PEM IMS/DC interface program cross-references to the PEM application number and function number. The PEM company number and source type are set by a user-prepared exit program. The exit program is described in the *Umbrella System Technical Support Programming Guide*. The exit program name is PEMCOSTD.

The PEM IMS/DC interface actually consists of two modules. The first is a simple routine that is to be preloaded by IMS. This is the module whose name must match the PSB name. This module, on the first execution, loads the remainder of PEM and saves the address. On subsequent executions, direct branching to PEM is used. This is to eliminate the overhead associated with initializing PEM.



IMS/DC Fast Path Interface

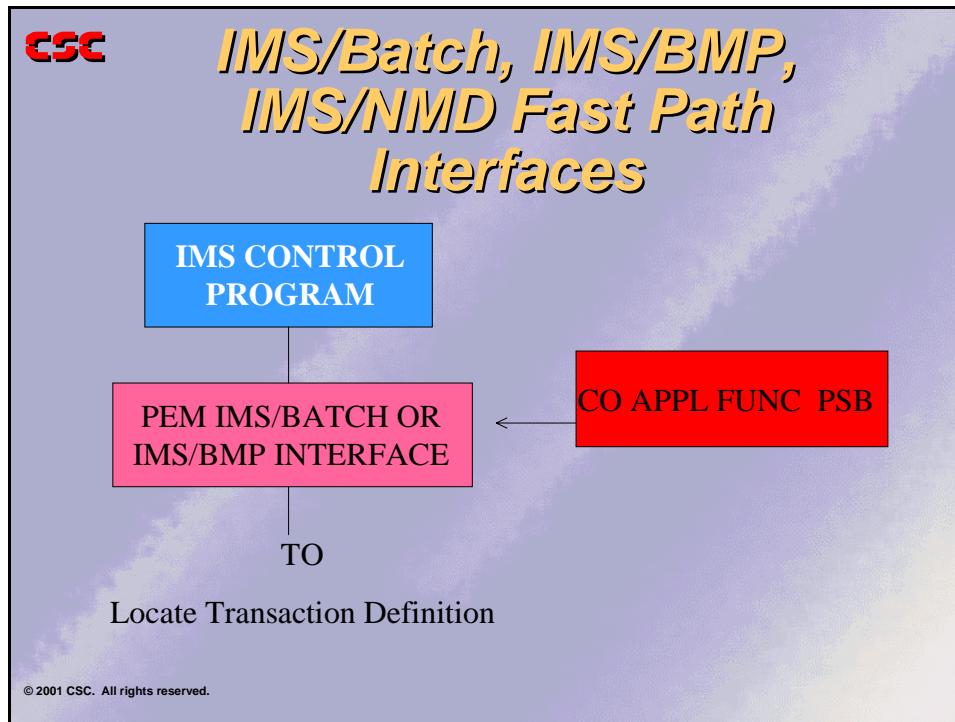


The IMS/VS control region routes all Fast Path exclusive transactions and Fast Path potential transactions to a required Fast Path user input exit. This module then routes the transaction to a Fast Path balancing group, sends it back to IMS/VS for processing, or the exit can cancel the transaction. The Fast Path user input exit is not provided as a part of the PEM Fast Path extension. The Fast Path balancing groups then send the transactions to the PEM IMS/DC interface. Fast Path and non Fast Path transactions use the same PEM interface but in different regions. The PEM Process Dictionary data bases cannot be Fast Path data bases. Therefore, any transaction presented to PEM may run mixed-mode. PEM caches the Process Dictionary definitions as they are retrieved from the data bases. Subsequent occurrences of transactions usually do not require access of the data bases and do not run mixed-mode. PEM also provides a means of caching dictionary definitions before the transaction is presented.

Notes:



IMS/Batch, IMS/BMP, and IMS/NMD Fast Path Interfaces

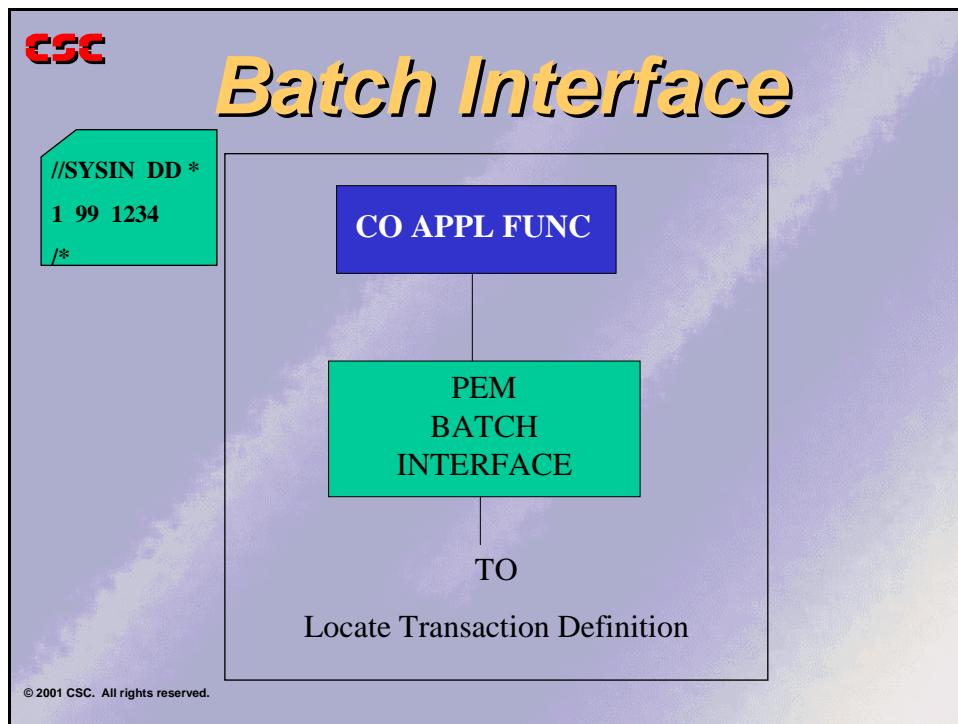


The program ID specified in the PARM= parameter of the execute statement is the Load Module ID for either the PEM IMS/BATCH interface or the PEM IMS/BMP interface. Once IMS gives control to PEM, processing is as described for the PEM batch interface.

Notes:



Batch Interface



The program ID specified in a batch execute statement is the Load Module ID for the PEM batch interface. This program reads SYSIN for a Transaction ID record. This record signifies the start of a PEM transaction. There may be any number of Transaction ID records on SYSIN. SYSIN is read each time a PEM transaction ends. PEM goes to end of job when an end of data condition is reached on SYSIN.

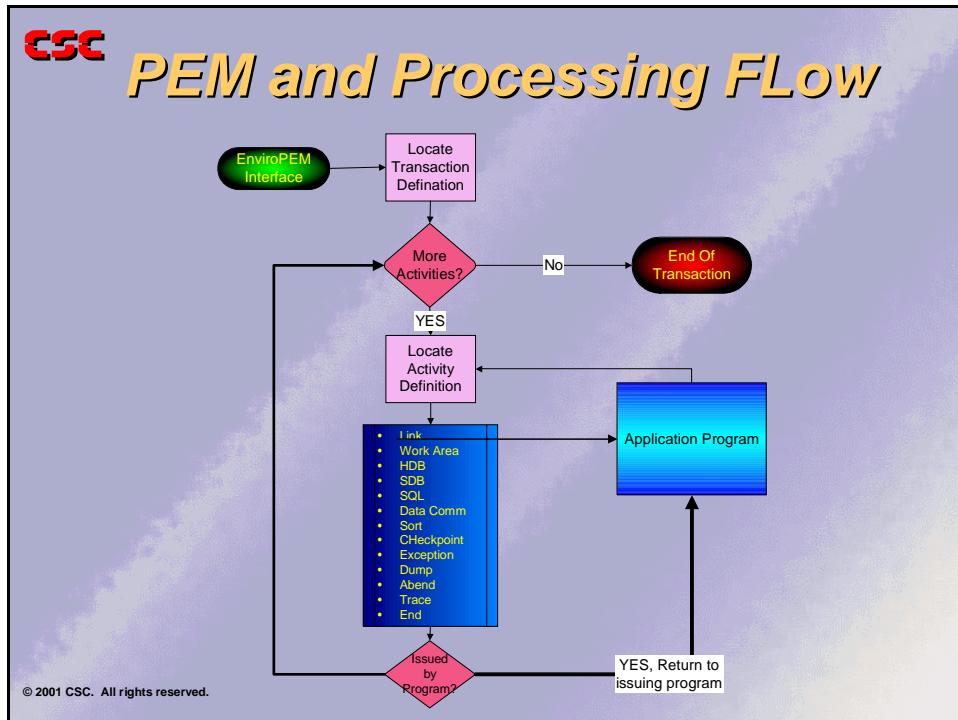
The Transaction ID record must contain three fields. The fields may be entered anywhere from column 1 through column 20. Each field may be delimited by a blank, a comma, or a slash. The first field must contain the company number, the second field must contain the application number, and the third field must contain the function number of a PEM transaction. Because the source type for batch is always 4, PEM supplies this field.

```
//SYSIN DD *  
1 99 1234  
/*
```



Processing Flow

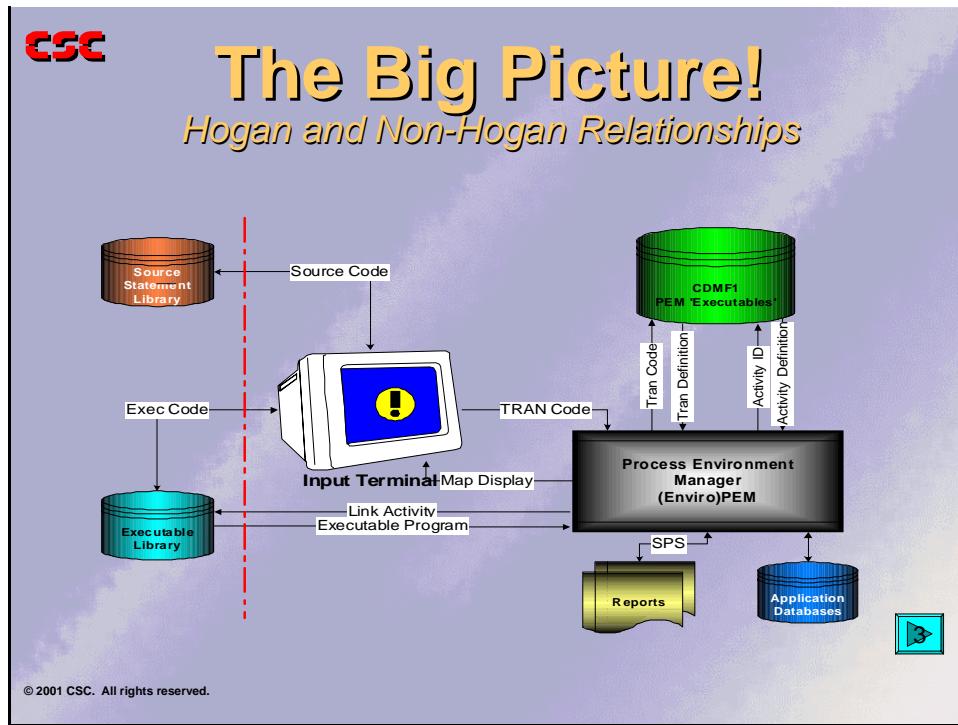
This section describes the processing flow from the time PEM receives a transaction through all of the activities until the processing is completed. Keep in mind that each environment has its own way of obtaining the transaction definition key, but, once this has been done, the flow within the transaction and within the application programs is the same.



Notes:



The Big Picture Revisited



Notes:



Problem Specifications—Batch Transaction Definition



You are to create a batch transaction definition, which will issue a link activity to invoke your program 9994xx.

Key for the batch PEM transaction definition:

APPLICATION ID	99
FUNCTION	99xx where xx is your group number
SOURCE	4
COMPANY	ALL
EFFECTIVE DATE	780101

The Activity Definition 9994xx, where xx is your group number, to link to your program was created in a prior exercise.

Use the owner application and change control created in the Change Control Exercise.

Notes:



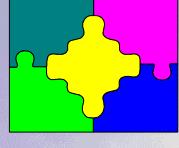
Summary



Summary



- When PEM is executing in batch, the PEM transaction to be processed is identified by a SYSIN DD card.
- When PEM is executing online, the PEM transaction to be processed is identified through the conversion of a CICS or IMS trancode.



© 2000 CSC. All rights reserved.

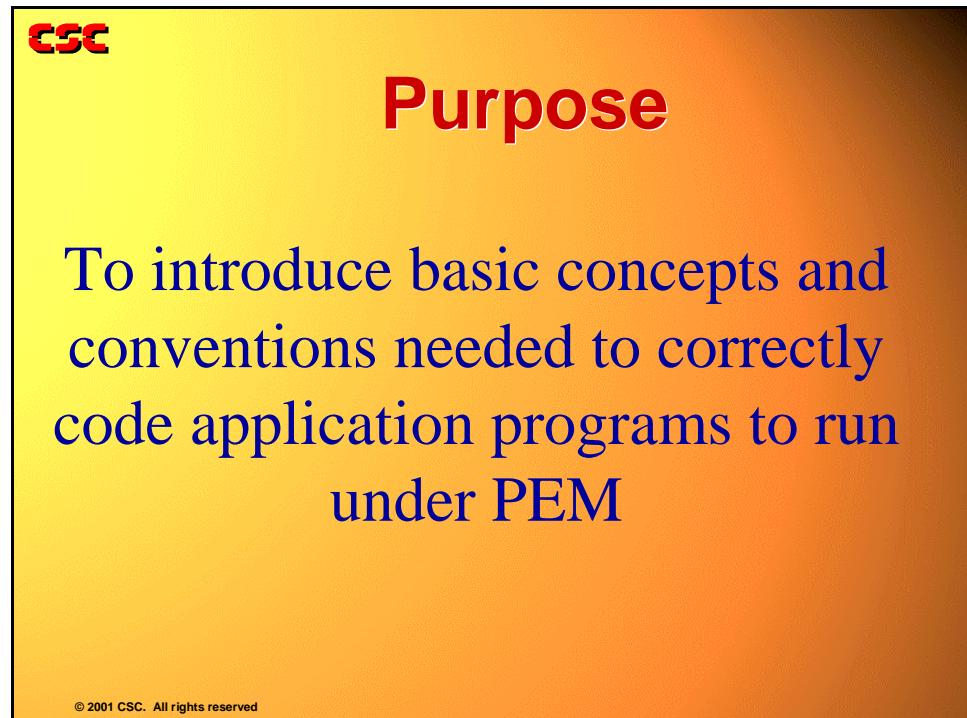
Notes:



Writing Programs to Run Under PEM

12

Purpose



The slide has a yellow-to-orange gradient background. In the top left corner is the CSC logo. The word "Purpose" is centered in large red font. Below it is a large blue text block describing the purpose of the program. At the bottom left is a small copyright notice.

Purpose

To introduce basic concepts and conventions needed to correctly code application programs to run under PEM

© 2001 CSC. All rights reserved

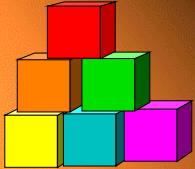
Notes:



Topics



Topics



- ★ Program Linkage under PEM
- ★ Data Usage
- ★ Authorization of Activities
- ★ Communication among programs
- ★ Working Storage - Static Values
- ★ Checking Results
- ★ The Precompiler
- ★ Batch program execution

© 2000 CSC. All rights reserved

Notes:



Objectives

CSC

Objectives



- ▶ Explain the program/subroutine calling chain represented by PEM and other Hogan programs
- ▶ Describe how control is passed among programs running under PEM
- ▶ List key work performed by PEM prior to program execution
- ▶ Define a communication data group
- ▶ Explain the limitations on the types of data in working storage for COBOL programs running under PEM

© 2000 CSC. All rights reserved

CSC

Objectives



- ▶ List examples of halfword and fullword binary fields and linkage conventions
- ▶ List rules for checking result fields after activities have been issued within a program
- ▶ List common precompiler messages and explain
- ▶ Explain how to execute a batch program
- ▶ Modify and execute a simple batch program

© 2000 CSC. All rights reserved



Programming Considerations

Program Invocation

CSC

Program Considerations

- = Program Invocation
 - = Run as subprograms to PEM
 - = Link Activity
- = Data Addressability
 - = Program Definition
 - = Linkage Section
 - = Using Statement

© 2000 CSC. All rights reserved



All programs in the Hogan System run as subroutines to PEM. PEM is the calling program. When writing a program to run under PEM, you are writing a subroutine and always using data areas managed and passed to you by PEM.

Each application program in the Hogan environment is executed as the result of a Link Activity issued either from the Transaction Definition or from another program. Control passes to PEM, which then passes control to the desired program.

Notes:



Data Addressability

Under standard COBOL linkage, conventions exist so that the compiler can maintain addressability to data. When a program calls a subroutine, the order of the data areas on the CALL statement must be the same as the order of the data areas in the USING clause of the procedure division statement of the called program.

As the calling program, PEM follows this standard; however, some modifications have been made in order to improve response and utilize the structure of the Process Dictionary to maintain parameters. The data areas in the COBOL program correspond to data groups on the Process Dictionary. You might think of PEM's formatting the CALL statement for the program based on the parameters from the program's definition on the Process Dictionary.

CSC

Before Executing, PEM:

- ◆ Allocates storage for data groups
- ◆ Establishes or passes addressability of data groups to the program
- ◆ Requires order of data groups correspond to COBOL programs in:
 - Program definition on the Process Dictionary
 - Linkage Section
 - 01 levels listed on the Procedure Division using statement

* 01 Assumed

© 2001 CSC. All rights reserved

Before the called program begins executing, PEM does the following:

1. Allocates storage for data groups to be used by the program as listed on the program definition. Data groups that have already been allocated for programs executed before this one are not allocated again. The current program is simply given addressability to the data groups.
2. Establishes or passes addressability of the data groups to the program. For COBOL programs, this requires an 01 level statement in the linkage section for each data group to be used. For ALC programs, register one points to a list of fullwords. Each contains the address of a data group.
3. Because PEM handles the establishment of addressability to the data, it



Umbrella Programming

Programming Considerations

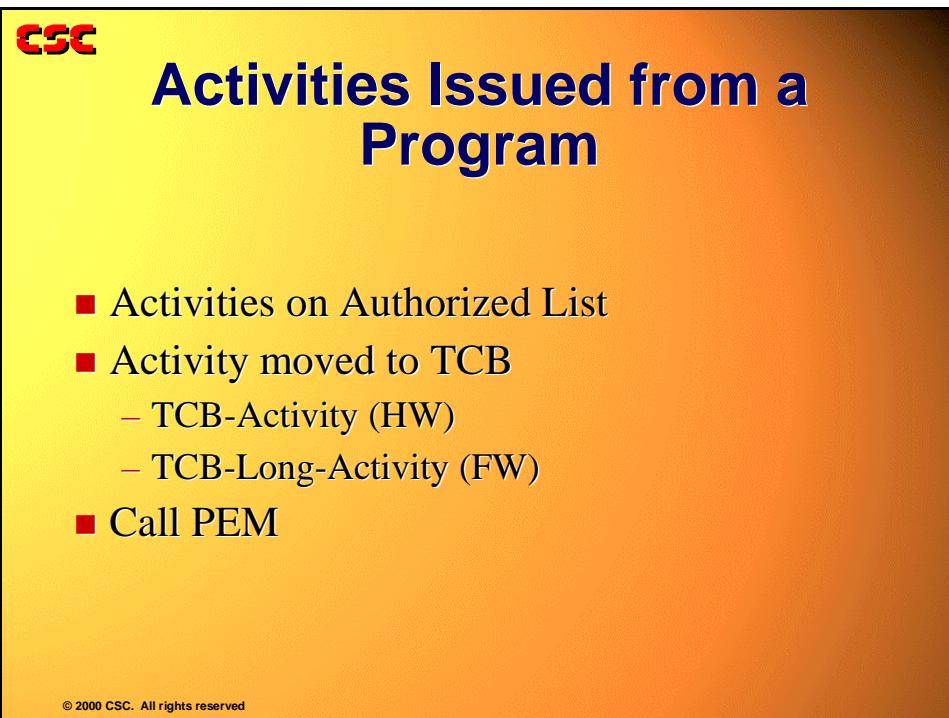
requires that the order of the data groups correspond in three places in COBOL programs:

- Program definition on the Process Dictionary
- Linkage Section
- 01 levels listed on the Procedure Division using statement.

PEM's TCB is always the first data group for which addressability is established.

**Note: THE TCB IS NEVER LISTED ON A PROGRAM DEFINITION.
ADDRESSABILITY WILL BE PROVIDED BY DEFAULT.**

Activities Issued From a Program



The slide features the CSC logo in red at the top left. The main title "Activities Issued from a Program" is centered in large blue text. Below the title is a bulleted list of activities:

- Activities on Authorized List
- Activity moved to TCB
 - TCB-Activity (HW)
 - TCB-Long-Activity (FW)
- Call PEM

At the bottom left of the slide, there is a small copyright notice: "© 2000 CSC. All rights reserved".

PEM establishes the list of activities the program is authorized to issue by reading the program definition form CDMF1. The program may or may not use all of the activities during any one execution.

The program cannot, however, issue an activity request if it is not on the program definition with the exception of activity IDs 100 or less. These activities are the PEM common activities and COBOL copybook P49003D is delivered with these activity numbers predefined. All programs are authorized to issue activities 1 through 100 without listing them on the program definition.



Working Storage—Static Values

CSC

Working Storage - Static Values

- Hogan's use of binary values
 - Client-reserved ranges
 - 50,000 - 100,000
 - 50 million - 100 million
- Halfword and fullword fields
 - Activity ID
 - Format ID
 - Condition Codes
 - Action Copybook
 - Result Copybook

© 2000 CSC. All rights reserved

In some CICS environments, which can have multiple transactions processing simultaneously, only one copy of working storage is generated. In this environment, the working storage data must stay unchanged. Because Hogan programs must run in all environments, they are delivered with only constants defined in working storage.

Many of the codes and values that will remain constant throughout processing are defined in delivered copybooks. The data elements contain static values for such fields as actions and results. The use of the standardized data element names in Hogan subroutines assists programmers in becoming familiar with the function of programs rapidly.

An action field tag name from the data group action copybook will be moved into data group action field during processing. The result of the requested action will be moved to the data group result field by PEM. The data group result copybook contains the tag names and values used to determine the result of the action.

If the program uses CDMF processing and/or Date Services, the corresponding copybooks for action and result codes will be included in the working storage of the program. The action tag name will be moved into the application control block prior to the call to PEM and the result will be tested against the results copybook.



Umbrella Programming

Programming Considerations

Hogan's Use of Binary Values

Many of the entries on the Process Dictionary and other CDMF records are keyed by binary values. In order to centralize the constant values used within a program for such items as activity numbers, they are defined in working storage.

In the release of Umbrella with CDMF, many of the keys are fullword in length (4 bytes). In releases prior to CDMF, these fields were halfword rather than fullword values. They were thus limited to the value 65,535 or less.

With the change to fullword values, the numbers reserved for customer use have been increased from 50,000 through 65,535 to two ranges:

- 50,000 through 100,000 AND
- 50,000,000 through 100,000,000.

COBOL has difficulty handling binary values. Hogan designers chose to use the binary format for its advantage of storing the largest possible number in a small storage area.

While some key fields such as TCB-CO-ID are defined to PEM as binary fields in data groups, these fields are defined as character in the copybooks included in the linkage section of a program. Programmers have to examine the data group definition on the Process Dictionary to identify the fields as binary.

To set the values for these keys, constants are defined centrally in Working Storage. These constants are defined as binary fields in which the values are set. They are redefined in character format for moves into the key fields.

Notes:



Halfword and Fullword Fields

Some key fields are halfword. These include TCB-CO-ID, TCB-USER-CC, PCD-ID, and TCB-RESULT.

With the CDMF release of Umbrella, two key fullword (four bytes) fields were added to the TCB. They are TCB-LONG-ACTIVITY and TCB-LONG-DGID. Prior to CDMF, activity numbers were set in TCB-ACTIVITY, and data groups were set in TCB-DATA-GROUP. These fields are halfword. They are, thus, limited to 65535 or less. The following illustrates a Hogan method for defining constant binary values in a program.



Redefinition Example

WORKING-STORAGE SECTION.

```
01 BINARY-VALUES
  05 HALFWORD-BINARY-N.
    10 FILLER          PIC S9(8) COMP VALUE +47190.
  05 HALFWORD-BINARY-X REDEFINES HALFWORD-BINARY-N.
    10 FILLER          PIC XX.
    10 PCD-ID-47190   PIC XX.
  05 FULLWORD-BINARY-N.
    10 FILLER          PIC S9(8) COMP VALUE +70000.
  05 FULLWORD-BINARY-X REDEFINES FULLWORD-BINARY-N.
    10 LK-ACTIVITY-70000  PIC XXXX.
```

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

Programming Considerations

Another method used for defining fullword constant binary values in Hogan programs is also used.



Redefinition Example

WORKING STORAGE SECTION

```
01 BINARY.  
 05 LK-ACTIVITY-70000.  
    10 FILLER      PIC S9(8) COMP VALUE +70000.
```

© 2000 CSC. All rights reserved

Note: S9(4) is the largest specification for a HALFWORD. The maximum value is +9999. S9(8) is the largest specification for a FULLWORD. The maximum value is +99999999.

While support for halfword programs is to be phased out over future releases, Umbrella System Version 3 Release 0 CST 1 allows existing halfword programs to execute fullword activities. This specifically supports Procedure Division copybooks written to use fullwords and be included in both existing halfword and fullword programs.

Notes:

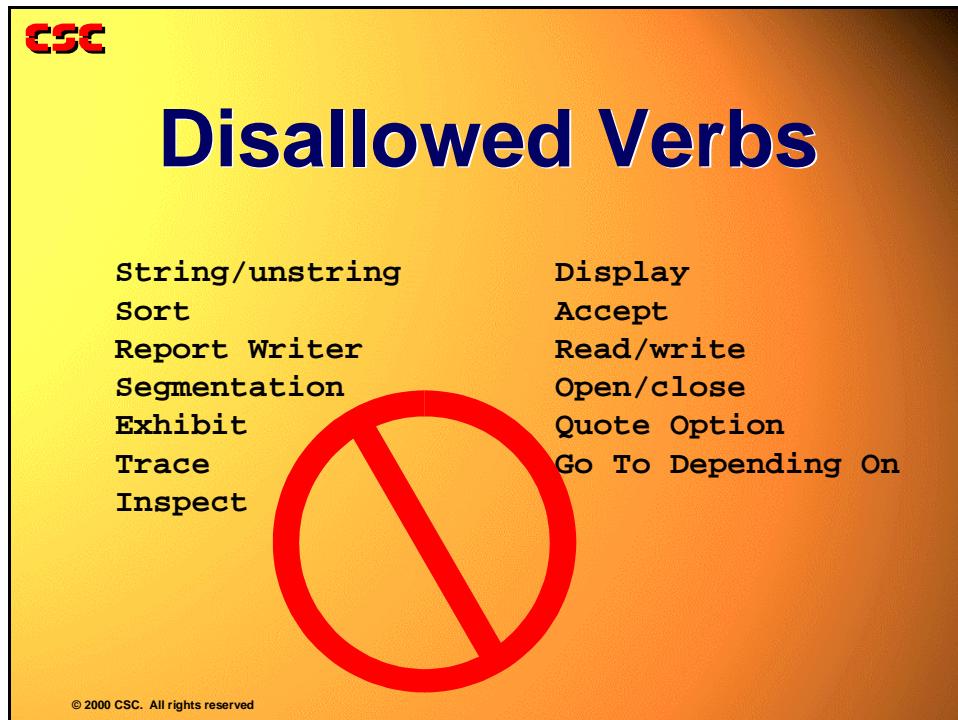


COBOL Coding Conventions When Running Under PEM

In general, the restrictions on COBOL programs are the same that apply to COBOL programs that run under CICS.

For the benefit of those users who do not have access to the IBM *CICS Application Programmer Reference Manual*, which itemizes these restrictions, a summary follows.

The following verbs are disallowed by IBM CICS:



Notes:





Disallowed Features

- Special features requiring an OS-GETMAN -
“current date”
- Options requiring use of Operating System Services
 - COUNT, ENDJOB, FLOW, DYNAM, STATE, STOP RUN*,
SYMDUMP, SYST, TEST, TCBSYSDATE
- Environment and Data Division entries associated
with Data Management
- File Section of Data Division

* Must be coded, but never executed.

© 2001 CSC. All rights reserved



Notes!!

- During the execution of a PEM transaction, data in the working storage section of the program should not be changed. Use a data group in linkage section for data that needs to be modified at execution time.
- In the case of CICS multi-tasking, different executions of a PEM transaction may be in progress concurrently within the same copy of a COBOL program.
- It is recommended that you use the Hogan delivered compile procedures.

© 2001 CSC. All rights reserved



Sample COBOL Program

```
IDENTIFICATION DIVISION.  
...  
ENVIRONMENT DIVISION.  
...  
DATA DIVISION.  
WORKING-STORAGE SECTION.  
...  
01 BINARY-FULLWORD.  
    05 LINK-ACTIVITY-12346.  
        10 FILLER          PIC S9(8) COMP    VALUE +12346.  
...  
LINKAGE SECTION.  
01 TRANSACTION-CONTROL-BLOCK.  
    02 ...  
    02 TCB-ACTIVITY      PIC XX.  
    02 TCB-RESULT       PIC XX.  
    02 ...  
    02 TCB-LONG-ACTIVITY   PIC XXXX.  
    02 ...  
01 DATA-GROUP-11.  
    02 DG11-ACTION-FIELD   PIC XX.  
    02 DG11-RESULT-FIELD   PIC XX.  
    02 DG11-FIELD-1        PIC ?.  
    02 ...  
01 DATA-GROUP-21.  
    02 DG21-ACTION-FIELD   PIC XX.  
    02 DG21-RESULT-FIELD   PIC XX.  
    02 ...  
01 DATA-GROUP-31.  
...  
PROCEDURE DIVISION USING TRANSACTION-CONTROL-BLOCK  
    DATA-GROUP-11  
    DATA-GROUP-21  
    DATA-GROUP-31.  
...  
MOVE LINK-ACTIVITY-12346 TO TCB-LONG-ACTIVITY.  
CALL 'PEM' USING TRANSACTION-CONTROL-BLOCK.  
...
```

Notes:

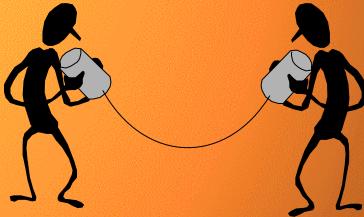


Communication Among Programs

CSC

Communication Among Programs

- Data Group
 - Application Control Block (ACB)
 - Application Data Group
 - Work Data Group
 - The TCB



© 2000 CSC. All rights reserved

PEM manages both program and data storage and executes application programs as subprograms. Information or data is communicated among programs in the task by interrogating the fields of data groups referenced in common. The most important concept to understand is that all programs within a task share a single copy of an allocated datagroup.

Note: All programs executed within a task SHARE one copy of an allocated datagroup.

When a program is executed within a task, the datagroups referenced by the Program Definition are checked by PEM against the task's Allocated Datagroup Table. If the datagroup has already been allocated to the task, the address of the datagroup is passed to the executed program in the form of a calling parameter list, constructed by PEM. If one or more of the datagroups have not been allocated to the task, PEM will allocate the datagroup(s), save them in the allocated datagroup storage area, and pass their address(es) to the executed program.

Note: PEM will allocate any UNALLOCATED data group(s) referenced by a program within a task.

Thus, before an application program is executed within a task, PEM allocates any unallocated data groups referenced on the Program Definition. Any other program within that same task requiring addressability to an allocated datagroup need only specify the datagroup on its Program Definition.



Umbrella Programming

COBOL Coding Conventions When Running Under PEM

Note: Only datagroups REFERENCED by a program should be specified on its Program Definition.

In addition, many delivered subsystems use communication data groups that are sometimes identified as control blocks. For example, the application control block for Date Services is data group 2000.

The subsystems are typically invoked via standardized link activities. For example, activity 1900 is a delivered link activity to Date Services.

In this manner Hogan-delivered programs or customer-coded variations are able to utilize the services and facilities of the multitude of subsystems and subprograms that make up the Hogan System at any desired point in processing.

The fact that each program is a called subroutine encourages programs to deal with single functions and implies modularity. Ideally, a program that performs one functional task can be called at any point in processing by any program needing that particular task to be performed.

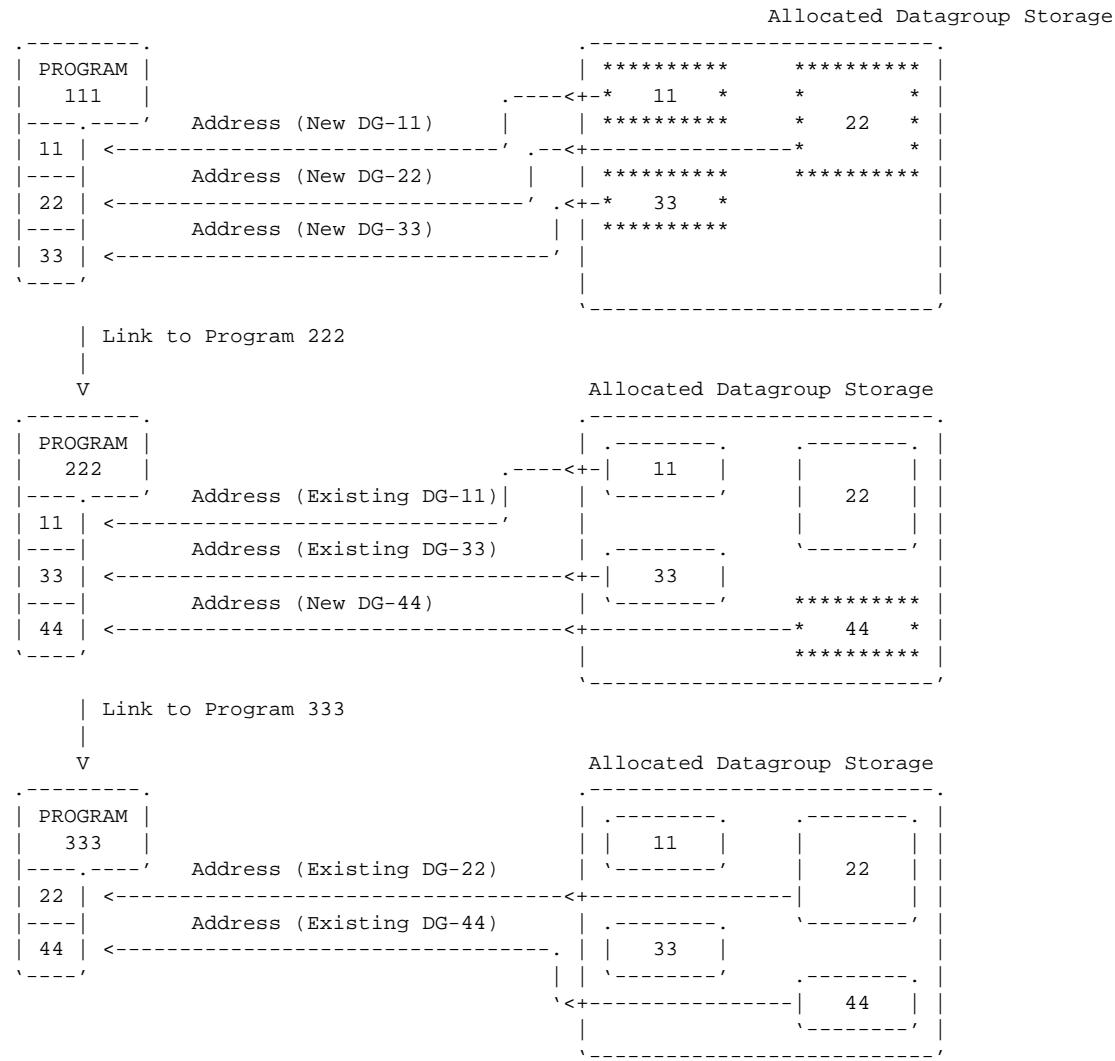
Notes:



Umbrella Programming

COBOL Coding Conventions When Running Under PEM

Program & Datagroup Relationships



This process may be summarized as follows:

- PEM locates Program Definition 111.
- PEM ALLOCATES datagroups 11,22,33.
- PEM invokes program AP111.
- Program AP111 begins executing.
- Program AP111 moves information into fields of datagroups 11,22,33.
- Program AP111 moves link activity 222 into TCB-LONG-ACTIVITY or TCB-ACTIVITY in the TCB.
- Program AP111 calls PEM using TRANSACTION-CONTROL-BLOCK. (01 level of the TCB).



Umbrella Programming

COBOL Coding Conventions When Running Under PEM

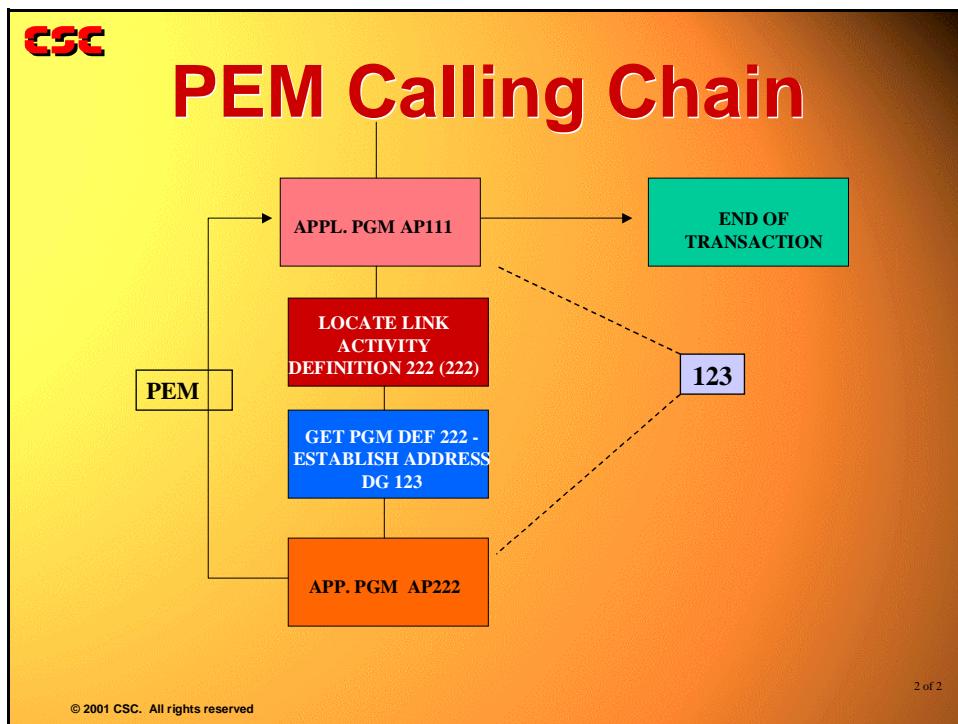
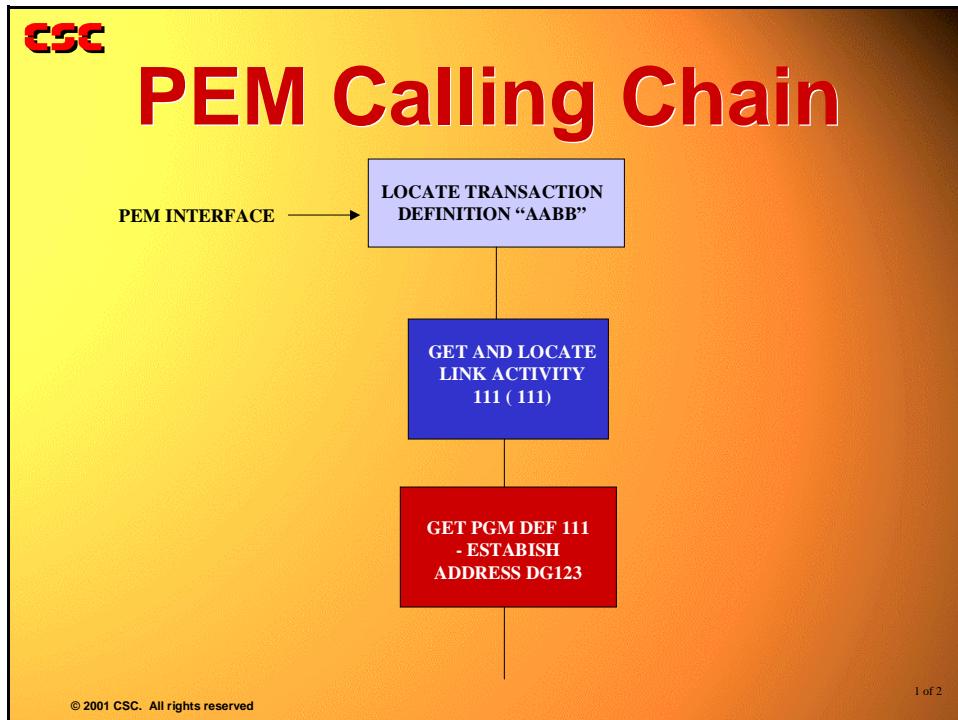
- PEM locates Program Definition 222.
- PEM ADDRESSES datagroups 11,33 and ALLOCATES datagroup 44.
- PEM invokes program AP222.
- Program AP222 begins executing.
- Program AP222 interrogates fields of datagroup 11.
- Program AP222 performs services based on codes or values in fields of datagroup 11.
- Program AP222 moves information into fields of datagroup 44.
- Program AP222 moves link activity 333 into TCB-LONG-ACTIVITY or TCB-ACTIVITY in the TCB.
- Program AP222 calls PEM using TRANSACTION-CONTROL-BLOCK. (01 level of the TCB).
- PEM locates Program Definition 333.
- PEM ADDRESSES datagroups 22,44.
- PEM invokes program AP333.
- Program AP333 begins executing.
- Program AP333 interrogates fields of datagroup 44.
- Program AP333 performs services based on codes or values in fields of datagroup 44.
- Program AP333 completes execution.
- PEM receives control and returns to program AP222.
- Program AP222 interrogates pertinent fields of datagroup 44.
- Program AP222 completes execution.
- PEM receives control and returns to program AP111.
- Program AP111 interrogates pertinent fields of datagroup 11.
- Program AP111 completes execution.
- PEM receives control and executes End-of-Transaction.



Umbrella Programming

COBOL Coding Conventions When Running Under PEM

PEM Calling Chain



Umbrella Programming

COBOL Coding Conventions When Running Under PEM

This process may be summarized as follows:

- Assume program AP111 is invoked by a link activity 111 from a Transaction Definition AABB.
- PEM locates Program Definition 111.
- PEM allocates datagroup 123.
- PEM invokes program AP111.
- Program AP111 begins executing.
- Program AP111 moves information into fields of datagroup 123.
- Program AP111 moves link activity 222 into TCB-LONG-ACTIVITY or TCB-ACTIVITY in the TCB.
- Program AP111 calls PEM using TRANSACTION-CONTROL-BLOCK. (01 level of the TCB).
- PEM locates Program Definition 222.
- PEM addresses datagroup 123.
- PEM invokes program AP222.
- Program AP222 begins executing.
- Program AP222 interrogates fields of datagroup 123.
- Program AP222 performs services based on codes or values in fields of datagroup 123.
- Program AP222 completes execution.
- PEM receives control and returns to program AP111.
- Program AP111 interrogates pertinent fields of datagroup 123.
- Program AP111 continues processing until EOP.



Checking Result Fields



Checking Results Fields



- ✓ After a data base activity
 - ✓ Check TCB-RESULT
 - ✓ Check Data Group RESULT fields
- ✓ After a link activity
 - ✓ NEVER check TCB-RESULT.
- ✓ After a link to a subsystem
 - ✓ Check the RESULT field of ACB

© 2001 CSC. All rights reserved



Checking Results Fields



- ✓ After communication with an application program
 - ✓ Check TCB-USER-CC (user field)
- ✓ After other types of activities
 - ✓ Check TCB-RESULT

© 2001 CSC. All rights reserved



Umbrella Programming

COBOL Coding Conventions When Running Under PEM

It is the responsibility of the application program issuing the activity to check the result of the action when control is returned to the program. The appropriate result field(s) should be checked to determine the status of the requested activity.

Summarized below are three key rules to follow in checking results:

1. After a data base activity, check TCB-RESULT. If the value is equal to TCB-OK, the DB activity was successful. A value not equal to TCB-OK indicates that some part of the DB activity failed, such as, end-of-data. You must then check the result field of the data groups referenced in the data base activity to determine the result of the requested action.

If the result field in the base data group or any dependent data group is not equal to DGR-OK, the data group's result field will contain the error code that corresponds to those in the delivered data group result code copybook, P49023D.

2. After a link activity, NEVER check TCB-RESULT. After a link to a subsystem, check the result field of the data group used to communicate information to and from that facility or program instead. This type of data group is known as the Application Control Block (data group 2000 is an Application Control Block for Date Services). The result code in the ACB will reflect whether the service was performed as expected.

Other fields may also need to be checked to determine if the correct information was passed back (such as, effective dates). Check these result fields in place of checking TCB-RESULT, which is not reset by PEM after a link activity is issued.

Communication between and among application programs may be achieved using TCB-USER-CC, which contains a user-supplied condition code.

3. After other types of activities, check TCB-RESULT. If it is equal to TCB-OK, PEM was able to complete the work of the activity. If not (and you did not abend), you should perform an error routine.

Notes:



Umbrella Programming

The Precompiler

The Precompiler

The purpose of the precompiler is to help reduce the number of PEM and COBOL program discrepancies that cause 0C4 protection exceptions, 0C7 data exceptions, and other abends. The precompiler compares the linkage section and using statement to the corresponding data group definition. If the length of the 01 level statement in the COBOL program is different from the length of the data group definition, it is noted in an error message.

Precompiler Messages

All messages issued by the precompiler are listed below in alphabetical sequence, based on the first word in the message.

NNNN BYTES RESERVED FOR DYNAMICALLY ALLOCATED DATA GROUP(S) XXX...XX

The indicated number (nnnnn) of bytes has been allocated for the dynamically allocated data group(s) indicated by the named 01-level (xxx...xx). Dynamically allocating a data group larger than this could cause BLL cells to be updated incorrectly and produce unpredictable results.

CONDITION CODE 4 HAS BEEN SET DUE TO ERRORS FOUND

Due to a size difference error, source library system open failure, or 01-level sequence error, the condition code has been set to four (4).

CONTINUE CARD ERROR& - SKIPPING TO NEXT 01-LEVEL

An unexpected continue card was found. This is probably a sequence error or coding error in the source program. All cards between the card shown on the next line of the message and the next 01-level are not processed.

DATA GROUP dddd (bbbbbbbb) LENGTH (llll) IS NOT THE LENGTH (nnnnn) OF xxx...xx

Where:

ddd is the data group ID

bbbbbbbb is the copybook from the data group definition

llll is the length of the indicated data group

nnnnn is the length of the named 01-level

xxx...xx is the 01-level corresponding to the named DGID.

A size difference between the size of a data group definition and the size of the named 01-level has been found. (If the 01-level is the object of a redefines, the redefining 01-level may be longer and cause this condition.) If the error indicator (**** ERROR) precedes this message, the named 01-level is longer than the indicated data group



definition; it is this condition that is one cause of system 0C4 (protection exception) abends.

dddd DATA GROUP NOT FOUND ON DATA GROUP DEFINITION READ

The indicated data group definition (ddddd) was not found on the Process Dictionary.

EXPECTED 01-LEVEL NOT FOUND - SKIPPING TO NEXT 01-LEVEL

A new 01-level was expected but was not found. All cards between the card shown on the next line of the message and the next 01- level are not processed. The probable cause of this message is a missing 01-level in a copybook or a comment card without an asterisk (*) in column 7.

FIRST 01-LEVEL NAME IN USING STATEMENT MUST BE TRANSACTION-CONTROL-BLOCK

The first 01-level name in the USING statement is required by PEM to be the TCB.

ILLEGAL CHARACTER WITHIN PARENTHESES - SKIPPING TO NEXT CLAUSE

A nonnumeric character was found within the parentheses of a picture clause on the card shown on the next line of the message. Processing continues with the next valid clause in the sentence.

INVALID CLAUSE FOUND - SKIPPING TILL VALID CLAUSE FOUND

The data definition shown on the next line of the message contains an invalid clause. Processing continues with the next valid clause found.

LINKAGE SECTION MISSING - SKIPPING PROGRAM

The LINKAGE section was not found. Processing is discontinued.

NO CORRESPONDING DATA GROUP ID FOR xxx...xx

No corresponding data group definition identification number was found for the named 01-level (xxx...xx).

PICTURE CLAUSE INCOMPLETE - SKIPPING TO NEXT CLAUSE

End-of-sentence was found after the PICTURE clause but before the picture description.

PROCESSING HAS BEEN COMPLETED

All precompiler processing for this program has been completed.

PROGRAM DEFINITION NOT FOUND

The Process Dictionary did not contain a Program Definition for the ID supplied to the precompiler.

PROGRAM-ID CARD NOT FOUND

The PROGRAM-ID paragraph was not found. Processing is discontinued.



Umbrella Programming

The Precompiler

SOURCE LIBRARY SYSTEM OPEN ROUTINE FAILED - NO ADDITIONAL PROCESSING POSSIBLE.

The OPEN for the indicated source library system failed. Processing is discontinued.

SOURCE LIBRARY SYSTEM INCLUDE STATEMENT FOUND - SKIPPING TO NEXT CARD

An INCLUDE statement for the source library system was found and is displayed on the next line of the message. It is ignored.

UNEXPECTED END-OF-DATA FOUND - CANNOT CONTINUE

An end-of-data condition was found while processing the LINKAGE section. Processing is discontinued.

USAGE IS DISPLAY-ST HANDLED AS USAGE IS DISPLAY

A Sterling USAGE PICTURE description is handled as if it were USAGE DISPLAY.

01-LEVEL NOT FOUND IN LINKAGE SECTION FOR xxx...xx

The data name indicated (xxx...xx) was found in the USING statement, but was not found on an 01-level in the LINKAGE section. It is assumed that a COPY/INCLUDE was omitted in the source program, and processing continues.

01-LEVELS OUT OF SEQUENCE BEGINNING WITH xxx...xx

When dynamically-allocated data groups are used, the sequence of 01-levels in the linkage section must be the same as the sequence of 01-level data names in the USING statement. This message indicates that they are not in sequence beginning with the 01-level named (xxx...xx).

01-LEVEL xxx...xx SOURCE sssss...sss

Is the second line of some messages where:

xxx...xx Is the name of the 01-level for this card

sssss...sss Is the source card containing the condition causing the message

Notes:



Batch Program Execution

Executing a program that runs under PEM is not difficult. You must:

- Define a transaction with a source type of 4 with an activity that will link to your program.

In your JCL:

- Include an EXEC statement that invokes PEM
- Include DD statements for the data bases needed by your program (notice that DD statements are not needed for the CDMF data bases; they are included in the procedure specified in the EXEC statement.)
- Specify the key to the PEM transaction you want to execute in SYSIN.

Some sample JCL for batch program execution follows.

```
//          .....
//JS010    EXEC HGNBPEM
//EMP      DD DSN=xxxV.EMP,DISP=SHR,
//          DCB=BUFNO=10
//SYSIN    DD *
1 99 9913
//*
//
```

Notes:



Umbrella Programming

Problem Specifications—Batch Program

Problem Specifications—Batch Program

This lab will involve modifying and executing a program. A skeleton program is provided as the base of the program so that some setup code has been done for you.

In this problem, modifications are required to the program to read a data base.

Many of the necessary constants for processing have been predefined in the skeleton program. In addition, the coding to invoke the Scheduled Processing System for printing a detail report of the employee data base has been included.

Each group will expand its own version of a skeleton program. Each group will be provided JCL to execute the program.

Insert logic into the skeleton program to read sequentially all employee records that are contained in the employee information data base (DBID=EMP).

The data groups to be accessed from each employee record are 47100, 47110, 47120, 47130, and 47140. These data groups contain data elements that will be written into a report. The data elements and their COBOL data names are:



	FROM DATA GROUP	COBOL NAME
COMPANY	47100	EMP-CO-ID
EMPLOYEE ID	47100	EMP-KEY-ID
NAME (FIRST)	47110	EMP-F-NAME
NAME (LAST)	47110	EMP-L-NAME
DEPARTMENT	47120	JOB-STAT-DEPT
JOB CLASS CODE	47120	JOB-STAT-CLASS
POSITION CODE	47120	JOB-STAT-POSITION
CURRENT EARNINGS	47130	EMP-C-EARN-TOT
YEAR-TO-DATE EARNINGS	47140	EMP-Y-EARN-TOT

© 2000 CSC. All rights reserved

It is not necessary to provide logic to print the report. The code in paragraph PR000-PRINT-REPORT will invoke the services of SPS. SPS will locate the required data elements, move them to the print line, and print the report.



Umbrella Programming

Problem Specifications—Batch Program

The JCL required to execute the program in batch has been provided. The specific batch transaction, you created in a previous exercise, needs to be added to the JCL so that your batch transaction will be invoked by PEM.

Notes:



Umbrella Programming

Problem Specifications—Batch Program

Suggested Steps for Problem Solution

1. It will be necessary to add and/or modify entries to the Process Dictionary as follows:

- The Program Definition for the program Z9994xx, where xx is your group number, was coded in a prior exercise, and will need to be modified to meet the needs of this exercise.

Remember, the data groups on the program definition must be in the same order as they appear in the linkage section of the skeleton program Z9994xx, where xx is your group number. Be sure the program definition authorizes all activities that can be issued by the program. A list of predefined activities can be found among the binary fullword constants located in working storage.

A hard copy of the skeleton program is provided in this section.

- The Activity Definition 9994xx, where xx is your group number, to link to your program was created in a prior exercise.
 - The data base activity 9996xx, where xx is your group number, to read the EMP data base was created in a prior exercise. It should be reviewed and modified for this exercise.
 - Use the owner application and change control created in the Change Control Exercise.
2. Insert COBOL coding to your group's copy of the skeleton program to read the EMP Data Base.
 3. Each group will need to insert the following into the execution JCL after the SYSIN statement.

```
//SYSIN DD *  
1 99 99xx      where xx is your group number
```

Notes:



Batch Problem Skeleton Program

```

MODULE NAME Z9994XX

//ZUP{J}SC JOB (HOGN,{B},BEF),'PGM Z9994\\\'',MSGCLASS=9,
//                      TIME=(00,04),REGION=4M,NOTIFY=&SYSUID
//*
//P$$S$LIB JCLLIB ORDER=( {TL}.PROCLIB)
//*
//***** ****
//JS010 EXEC HOGNBC2B,
//          MASTER=' {L}.HEC.MASTER',
//          SYSMOD=' {L}.TESTLIB',
//          VSAM=' {V}'
//*-----
//* THIS PROC :
//*          EXECUTES LIBRARIAN
//*          VALIDATES PROGRAM VIA HOGAN PRECOMPILER
//*          COMPILES PROGRAM
//*          LINKS INTO TESTLIB (NAME CARD IS GENERATED)
//*-----
//SYSIN DD *
-PID 9994\\\
-OPT EXEC,TEMP
-SEL HECDUMMY <===== DO NOT CHANGE THIS STATEMENT
-REP ALL,NOAUDIT
***** ****
*           I D E N T I F I C A T I O N   D I V I S I O N
***** ****
      SKIP1
      IDENTIFICATION DIVISION.
      PROGRAM-ID.    Z9994\\|.
      AUTHOR.        HOGAN SYSTEMS INC.
      DATE-COMPILED.
*REMARKS.      SKELETON PROGRAM FOR THE
*              UMBRELLA PROGRAMMERS CLASS
*              LAB EXERCISES.
      ENVIRONMENT DIVISION.
***** ****
*           E N V I R O N M E N T   D I V I S I O N
***** ****
      SKIP1
      CONFIGURATION SECTION.
      SOURCE-COMPUTER.   IBM-370.
      OBJECT-COMPUTER.   IBM-370.
      EJECT
***** ****
*           D A T A   D I V I S I O N
***** ****
      DATA DIVISION.
      SKIP3
      WORKING-STORAGE SECTION.
*-----
*   HOGAN LINKS ALL COBOL PROGRAMS AS REENTRANT. ONLY STATIC
*   VALUES ARE DEFINED IN WORKING STORAGE.
*-----
      77 CC-PHASE-NAME          PIC X(8)    VALUE 'Z9994\\\''.
      77 CC-GROUP-ID            PIC XX     VALUE '\\\''.
      SKIP3
*-----
*   THE FOLLOWING AREA CAN BE USED TO DEFINE PROGRAM CONSTANTS.
*   PEM USES BINARY FORMATED VALUES FOR ACTIVITY IDS, CONDITION
*   CODES, FORMAT IDS, ACTION CODES, ETC...
*   THERE ARE MANY DELIVERED COPYBOOK CONTAINING THESE BINARY

```



Umbrella Programming

Problem Specifications—Batch Program

```
* VALUES. THE ONES NEEDED FOR THIS CLASS HAVE BEEN INCLUDED.  
*-----  
01 BINARY-VALUES.  
05 FULLWORD-BINARY.  
    10 FILLER          PIC S9(8) COMP VALUE +01013.  
    10 FILLER          PIC S9(8) COMP VALUE +01398.  
    10 FILLER          PIC S9(8) COMP VALUE +01900.  
    10 FILLER          PIC S9(8) COMP VALUE +47921.  
    10 FILLER          PIC S9(8) COMP VALUE +47922.  
    10 FILLER          PIC S9(8) COMP VALUE +48000.  
    10 FILLER          PIC S9(8) COMP VALUE +0.  
    10 FILLER          PIC S9(8) COMP VALUE +0.  
    10 FILLER          PIC S9(8) COMP VALUE +0.  
05 FILLER REDEFINES FULLWORD-BINARY.  
    10 PCD-LK-ACTIVITY-1013  PIC XXXX.  
    10 CCP-LK-ACTIVITY-1398  PIC XXXX.  
    10 DTS-LK-ACTIVITY-1900  PIC XXXX.  
    10 SPS-LK-ACTIVITY-47921 PIC XXXX.  
    10 SPS-LK-ACTIVITY-47922 PIC XXXX.  
    10 CDMF-LK-ACTIVITY-48000 PIC XXXX.  
    10 FIRST-FULLWORD      PIC XXXX.  
    10 SECOND-FULLWORD     PIC XXXX.  
    10 THIRD-FULLWORD      PIC XXXX.  
05 HALFWORD-BINARY.  
    10 FILLER          PIC S9(8) COMP VALUE +99\\\|.  
    10 FILLER          PIC S9(8) COMP VALUE +0.  
    10 FILLER          PIC S9(8) COMP VALUE +0.  
    10 FILLER          PIC S9(8) COMP VALUE +0.  
05 FILLER REDEFINES HALFWORD-BINARY.  
    10 FILLER          PIC XX.  
    10 COND-CODE-99\\\|  PIC XX.  
    10 FILLER          PIC XX.  
    10 FIRST-HALFWORD   PIC XX.  
    10 FILLER          PIC XX.  
    10 SECOND-HALFWORD  PIC XX.  
    10 FILLER          PIC XX.  
    10 THIRD-HALFWORD   PIC XX.  
EJECT  
*-----  
* THE FOLLOWING COPYBOOK P49002D CONTAINS THE VARIOUS RESULT  
* CODE VALUES THAT MAY BE PLACED IN THE TCB RESULT FIELD  
* (TCB-RESULT). THE TCB IS DEFINED BY COPYBOOK P49000D.  
*-----  
000100***** START OF P49002D ***** TCB RESULT CONSTANTS ***  
000200*  
000300 01 TCB-RESULT-CONSTANTS.  
000400 05 TCB-RESULTS.  
000500 10 TCB-OK.  
000600           15 FILLER PIC S9(4) COMP VALUE +0000  
000700 10 TCB-ERR.  
000800           15 FILLER PIC S9(4) COMP VALUE +0001  
000900 10 TCB-NO-ACT.  
001000           15 FILLER PIC S9(4) COMP VALUE +0002  
001100 10 TCB-NOT-AUTH.  
001200           15 FILLER PIC S9(4) COMP VALUE +0003  
001300 10 TCB-FAIL.  
001400           15 FILLER PIC S9(4) COMP VALUE +0004  
001500 10 TCB-FULL.  
001600           15 FILLER PIC S9(4) COMP VALUE +0005  
001700 10 TCB-ABEND-EXIT.  
001800           15 FILLER PIC S9(4) COMP VALUE +0006  
001900 10 TCB-ROLLBACK.  
002000           15 FILLER PIC S9(4) COMP VALUE +0007  
002100 10 TCB-DATA-BASE-FULL.  
002200           15 FILLER PIC S9(4) COMP VALUE +0008
```



Umbrella Programming

Problem Specifications—Batch Program

```

002300      10 TCB-DB-NOT-AVAILABLE.
002400          15 FILLER PIC S9(4) COMP VALUE +0009
002500      10 TCB-OK-CHECKPOINT.
002600          15 FILLER PIC S9(4) COMP VALUE +0010
002700      10 TCB-FAIL-CHECKPOINT.
002800          15 FILLER PIC S9(4) COMP VALUE +0011
002900      10 TCB-ABEND-TRANS.
003000          15 FILLER PIC S9(4) COMP VALUE +0012
003100      10 TCB-DEFERRED.
003200          15 FILLER PIC S9(4) COMP VALUE +0020
003300      10 TCB-RANDOMIZER-ERROR.
003400          15 FILLER PIC S9(4) COMP VALUE +0021
003500      10 TCB-DATA-NOT-AVAILABLE.
003600          15 FILLER PIC S9(4) COMP VALUE +0022
003700      10 TCB-DATA-FROM-GET-ONLY-DB.
003800          15 FILLER PIC S9(4) COMP VALUE +0030
003900*
004000***** END OF P49002D ****
EJECT
*-----
* THE FOLLOWING COPYBOOK U48004D DEFINES THE POSSIBLE ACTION
* CODES FOR CDMF/PCD PROCESSING.
* THE REQUESTED ACTION MUST BE PLACED INTO CDMF-ACTION FIELD
* IN THE APPPLIACTION CONTROL BLOCK PRIOR TO ISSUING THE
* LINK ACTIVITY 48000 FOR CDMF/PCD PROCESSING.
* THE ACB IS PART OF THE TCB. THE DEFINITION IS IN COPYBOOK
* P49000D INCLUDED IN THE LINKAGE SECTION OF THIS PROGRAM.
*-----
000100*** START OF U48004D *-----
000200* CDMF ACTION LIST FOR 'CDMF CONTROL BLOCK' AND MISCELLANEOUS
000300* CONSTANTS FOR THE 'CDMF CROSS REFERENCE CONTROL BLOCK'.
000400*-----
000500 01 CDMF-ACTION-LIST.
000600    05 CDMF-CODE-VALUES.
000700    10 CDMF-NO-OP.
000800          15 FILLER PIC S9(4) COMP VALUE +000
000900    10 CDMF-ADD.
001000          15 FILLER PIC S9(4) COMP VALUE +000
001100    10 CDMF-ADD-DUMMY.
001200          15 FILLER PIC S9(4) COMP VALUE +000
001300    10 CDMF-REPLACE.
001400          15 FILLER PIC S9(4) COMP VALUE +000
001500    10 CDMF-CHANGE.
001600          15 FILLER PIC S9(4) COMP VALUE +000
001700    10 CDMF-DELETE.
001800          15 FILLER PIC S9(4) COMP VALUE +000
001900    10 CDMF-INQ.
002000          15 FILLER PIC S9(4) COMP VALUE +000
002100    10 CDMF-NXT.
002200          15 FILLER PIC S9(4) COMP VALUE +000
002300    10 CDMF-NXTE.
002400          15 FILLER PIC S9(4) COMP VALUE +000
002500    10 CDMF-KGE.
002600          15 FILLER PIC S9(4) COMP VALUE +001
002700    SKIP1
002800*-----
002900*           ADVANCED CDMF RETRIEVAL ACTION CODES
003000*-----
003100    10 CDMF-INQ-FROM-TABLE.
003200          15 FILLER PIC S9(4) COMP VALUE +000
003300    10 CDMF-NXT-FROM-TABLE.
003400          15 FILLER PIC S9(4) COMP VALUE +000
003500    10 CDMF-NXTE-FROM-TABLE.
003600          15 FILLER PIC S9(4) COMP VALUE +001
003700    10 CDMF-KGE-FROM-TABLE.

```



Umbrella Programming

Problem Specifications—Batch Program

```
003800                      15 FILLER PIC S9(4) COMP VALUE +001
003900      10 CDMF-INQ-FROM-DATABASE.
004000                      15 FILLER PIC S9(4) COMP VALUE +001
004100      10 CDMF-NXT-FROM-DATABASE.
004200                      15 FILLER PIC S9(4) COMP VALUE +001
004300      10 CDMF-NXTE-FROM-DATABASE.
004400                      15 FILLER PIC S9(4) COMP VALUE +001
004500      10 CDMF-KGE-FROM-DATABASE.
004600                      15 FILLER PIC S9(4) COMP VALUE +001
004700      SKIP1
004800*-----*
004900*      ADVANCED CDMF LOGGING ACTION CODES
005000*-----*
005100      10 CDMF-LOG-BEFORE-CHANGE.
005200                      15 FILLER PIC S9(4) COMP VALUE +001
005300      10 CDMF-LOG-BEFORE-DELETE.
005400                      15 FILLER PIC S9(4) COMP VALUE +001
005500      10 CDMF-LOG-AFTER-ADD.
005600                      15 FILLER PIC S9(4) COMP VALUE +001
005700      10 CDMF-LOG-AFTER-CHANGE.
005800                      15 FILLER PIC S9(4) COMP VALUE +002
005900      SKIP1
006000*-----*
006100* CONSTANTS FOR THE 'CDMF CROSS REFERENCE CONTROL BLOCK', DATA
006200* GROUP 48008, COPYBOOK U48008D
006300* NOTE: THE '66' LEVEL IS USED TO KEEP WORKING STORAGE TO A
006400*     MINIMUM. THE 'RENAMES' CLAUSE HERE MEANS THAT THE ENTRY HAS
006500*     THE SAME EXACT VALUES AS THE ENTRY IT REDEFINES (RENAMES).
006600*-----*
006700*
006800**** VALUES FOR U008-SOURCE-TYPE
006900      10 XREF-SOURCE-TYPE-HDB      PIC X    VALUE HIGH-VALUES.
007000*
007100**** VALUES FOR U008-ACTION (ACTION ON SOURCE ENTRY W/IN HOST)
007200      66 XREF-INQ-SOURCE-ENTRY      RENAMES CDMF-ADD.
007300      66 XREF-1ST-SOURCE-ENTRY      RENAMES CDMF-ADD-DUMMY.
007400      66 XREF-NXT-SOURCE-ENTRY      RENAMES CDMF-CHANGE.
007500      66 XREF-1ST-ENTRY-NXT-HOST    RENAMES CDMF-DELETE.
007600      66 XREF-KEY-GE-SOURCE-ENTRY    RENAMES CDMF-INQ.
007700      66 XREF-ADD-SOURCE-ENTRY      RENAMES CDMF-NXT.
007800      66 XREF-DEL-SOURCE-ENTRY      RENAMES CDMF-NXTE.
007900      66 XREF-LAST-CALL-DR-PHS2    RENAMES CDMF-INQ-FROM-TABLE
008000*
008100**** VALUES FOR U008-RESULT
008200      66 XREF-ACTION-SUCCESSFUL    RENAMES CDMF-NO-OP.
008300      66 XREF-SOURCE-ENTRY-NOT-FOUND RENAMES CDMF-ADD.
008400      66 XREF-HOST-KEY-NOT-FOUND    RENAMES CDMF-ADD-DUMMY.
008500      66 XREF-END-OF-ENTRIES-FOR-HOST
008600                      RENAMES CDMF-REPLACE.
008700      66 XREF-NO-SUCH-SOURCE-FMT    RENAMES CDMF-DELETE.
008800      66 XREF-NO-SUCH-HOST-FMT      RENAMES CDMF-INQ.
008900      66 XREF-DB-NOT-AVAILABLE      RENAMES CDMF-NXT-FROM-TABLE
009000*
009100**** VALUES FOR U008-FEEDBACK (FEEDBACK FROM SUCCESSFUL UPDATES)
009200      66 XREF-DEL-HOST-NOT-FOUND    RENAMES CDMF-ADD.
009300      66 XREF-DEL-NO-XRFS-ON-HOST    RENAMES CDMF-ADD-DUMMY.
009400      66 XREF-DEL-NO-MATCHING-ENTRY  RENAMES CDMF-REPLACE.
009500      66 XREF-ADD-DUPE-ENTRY        RENAMES CDMF-DELETE.
009600      66 XREF-ADD-NO-EXP-DATA       RENAMES CDMF-INQ.
009700      66 XREF-ADD-DUMMY-ADDED      RENAMES CDMF-NXT.
009800*
009900*---* END OF U48004D *-----*
          EJECT
*-----*
*   THE FOLLOWING COPYBOOK U48024D DEFINES THE POSSIBLE RESULT
```



Umbrella Programming

Problem Specifications—Batch Program

```

*   CODES RETURNED BY CDMF/PCD PROCESSING.
*   THE RESULT FOR THE ACTION IS PLACED INTO CDMF-RESULT FIELD
*   IN THE APPPLIACTION CONTROL BLOCK PRIOR TO RETURNING TO THIS
*   APPLICATION PROGRAM.
*   THE ACB IS PART OF THE TCB.  THE DEFINITION IS IN COPYBOOK
*   P49000D INCLUDED IN THE LINKAGE SECTION OF THIS PROGRAM.
*-----
000100*---* START OF U48024D *----* CDMF RESULT LIST *-----*
000200*
000300 01 CDMF-RESULT-LIST.
000400    05 CDMF-RESULT-CODES.
000500*   CDMF-NO-ERRORS
000600           10 FILLER PIC S9(9) COMP VALUE +0000
000700*   CDMF-DB-NOT-AVAILABLE
000800           10 FILLER PIC S9(9) COMP VALUE +0000
000900*   CDMF-INVALID-ACTION
001000           10 FILLER PIC S9(9) COMP VALUE +4800
001100*   CDMF-ITEM-NOT-FOUND
001200           10 FILLER PIC S9(9) COMP VALUE +4800
001300*   CDMF-FORMAT-NOT-FOUND
001400           10 FILLER PIC S9(9) COMP VALUE +4800
001500*   CDMF-UNABLE-TO-ALLOC-DG
001600           10 FILLER PIC S9(9) COMP VALUE +4800
001700*   CDMF-END-OF-FORMAT
001800           10 FILLER PIC S9(9) COMP VALUE +4800
001900*   CDMF-DUPE-KEY-ON-ADD
002000           10 FILLER PIC S9(9) COMP VALUE +4800
002100*   CDMF-INVALID-EFF-DATE
002200           10 FILLER PIC S9(9) COMP VALUE +4800
002300*   CDMF-INVALID-APPL
002400           10 FILLER PIC S9(9) COMP VALUE +4800
002500*   CDMF-SKEY-NOT-FOUND
002600           10 FILLER PIC S9(9) COMP VALUE +4800
002700*   CDMF-SKEY-READ-ERR
002800           10 FILLER PIC S9(9) COMP VALUE +4801
002900*   CDMF-INVALID-CC-NO
003000           10 FILLER PIC S9(9) COMP VALUE +4801
003100*   CDMF-INVALID-FAMILY
003200           10 FILLER PIC S9(9) COMP VALUE +4801
003300*   CDMF-NO-OWNER-CHANGE
003400           10 FILLER PIC S9(9) COMP VALUE +4801
003500*   CDMF-SECURITY-VIOLATION
003600           10 FILLER PIC S9(9) COMP VALUE +4803
003700*   CDMF-SECURITY-INACTIVE
003800           10 FILLER PIC S9(9) COMP VALUE +4803
003900*   CDMF-DUMMY-REC-FOUND
004000           10 FILLER PIC S9(9) COMP VALUE +4809
004100*   CDMF-CCNR-MISSING
004200           10 FILLER PIC S9(9) COMP VALUE +4837
004300*   CDMF-CCNR-CLOSED
004400           10 FILLER PIC S9(9) COMP VALUE +4837
004500    05 FILLER           REDEFINES CDMF-RESULT-CODES.
004600           10 FILLER PIC XX.
004700    10 CDMF-NO-ERRORS      PIC XX.
004800           10 FILLER PIC XX.
004900    10 CDMF-DB-NOT-AVAILABLE  PIC XX.
005000           10 FILLER PIC XX.
005100    10 CDMF-INVALID-ACTION    PIC XX.
005200           10 FILLER PIC XX.
005300    10 CDMF-ITEM-NOT-FOUND      PIC XX.
005400           10 FILLER PIC XX.
005500    10 CDMF-FORMAT-NOT-FOUND    PIC XX.
005600           10 FILLER PIC XX.
005700    10 CDMF-UNABLE-TO-ALLOC-DG    PIC XX.
005800           10 FILLER PIC XX.

```



Umbrella Programming

Problem Specifications—Batch Program

```
005900      10 CDMF-END-OF-FORMAT      PIC XX.
006000
006100      10 CDMF-DUPE-KEY-ON-ADD    PIC XX.
006200
006300      10 CDMF-INVALID-EFF-DATE   PIC XX.
006400
006500      10 CDMF-INVALID-APPL       PIC XX.
006600
006700      10 CDMF-SKEY-NOT-FOUND     PIC XX.
006800
006900      10 CDMF-SKEY-READ-ERR      PIC XX.
007000
007100      10 CDMF-INVALID-CC-NO      PIC XX.
007200
007300      10 CDMF-INVALID-FAMILY     PIC XX.
007400
007500      10 CDMF-NO-OWNER-CHANGE    PIC XX.
007600
007700      10 CDMF-SECURITY-VIOLATION PIC XX.
007800
007900      10 CDMF-SECURITY-INACTIVE  PIC XX.
008000
008100      10 CDMF-DUMMY-REC-FOUND   PIC XX.
008200
008300      10 CDMF-CCNR-MISSING     PIC XX.
008400
008500      10 CDMF-CCNR-CLOSED      PIC XX.
008600*
008700*----* END OF U48024D *-----  
EJECT  
*-----  
* THE FOLLOWING COPYBOOK P49003D DEFINES SOME OF THE MORE  
* COMMONLY USED PEM ACTIVITIES. ACTIVITIES 1 THROUGH 100  
* ARE AUTOMATICALLY AUTHORIZED FOR USE BY ALL PROGRAMS AND  
* NEED NOT BE SPECIFIED IN THE PROGRAM DEFINITION IN THE  
* PROCESS DICTIONARY.  
*-----  
000100**** START OF P49003D ***** PEM COMMON ACTIVITIES ***  
000200*
000300 01  PEM-COMMON-ACTIVITIES.  
000400 05  PEM-ACTIVITIES.  
000500 10  CA-LONG-PEM-END-PROG.  
000600
000700 10  CA-LONG-PEM-END-TRANS.  
000800
000900 10  CA-LONG-PEM-DYN-DG-ALOC.  
001000
001100 10  CA-LONG-PEM-DYN-DG-REL.  
001200
001300 10  CA-LONG-PEM-DYN-DG-INIT.  
001400
001500 10  CA-LONG-PEM-SYSPRINT-WRITE.  
001600
001700 10  CA-LONG-PEM-NO-OP.  
001800
001900 10  CA-LONG-PEM-TRANS-DUMP-RETURN.  
002000
002100 10  CA-LONG-PEM-TRANS-DUMP-END.  
002200
002300 10  CA-LONG-PEM-DUMP-TCB-TRACE.  
002400
002500 10  CA-LONG-PEM-DUMP-DG-RETURN.  
002600
002700 10  CA-LONG-PEM-DUMP-DG-END.  
002800
```



Umbrella Programming

Problem Specifications—Batch Program

```

002900    10 CA-LONG-PEM-ENABLE-ABEND-EXIT.
003000                                15 FILLER PIC S9(9) COMP VALUE +0020
003100    10 CA-LONG-PEM-DISABLE-ABEND-EXIT.
003200                                15 FILLER PIC S9(9) COMP VALUE +0021
003300    10 CA-LONG-PEM-CHECKPOINT.
003400                                15 FILLER PIC S9(9) COMP VALUE +0028
003500    10 CA-LONG-PEM-DYN-DG-NO-INIT.
003600                                15 FILLER PIC S9(9) COMP VALUE +0029
003700    10 CA-LONG-PEM-ROLLBACK.
003800                                15 FILLER PIC S9(9) COMP VALUE +0030
003900    10 CA-LONG-PEM-DLI-SYNCPOINT.
004000                                15 FILLER PIC S9(9) COMP VALUE +0032
004100    10 CA-LONG-PEM-USERCC-EXCEP.
004200                                15 FILLER PIC S9(9) COMP VALUE +0035
004300    10 CA-LONG-PEM-MSG9-ABEND.
004400                                15 FILLER PIC S9(9) COMP VALUE +0039
004500    10 CA-LONG-PEM-APPC-SYNCPOINT.
004600                                15 FILLER PIC S9(9) COMP VALUE +0056
004700    10 CA-LONG-PEM-DYN-DG-LENGTH.
004800                                15 FILLER PIC S9(9) COMP VALUE +0074
004900    10 CA-LONG-PEM-DYN-PTR-REL.
005000                                15 FILLER PIC S9(9) COMP VALUE +0075
005100    10 CA-LONG-PEM-DYN-PTR-INIT.
005200                                15 FILLER PIC S9(9) COMP VALUE +0076
005201    10 CA-LONG-PEM-DYN-PTR-ANO.
005202                                15 FILLER PIC S9(9) COMP VALUE +0077
005300    10 CA-LONG-PEM-PDG-HAS-CHANGED.
005400                                15 FILLER PIC S9(9) COMP VALUE +0088
005500*
005600    05 FILLER          REDEFINES PEM-ACTIVITIES.
005700                                10 FILLER PIC XX
005800    10 CA-PEM-END-PROG      PIC XX.
005900                                10 FILLER PIC XX
006000    10 CA-PEM-END-TRANS     PIC XX.
006100                                10 FILLER PIC XX
006200    10 CA-PEM-DYN-DG-ALOC     PIC XX.
006300                                10 FILLER PIC XX
006400    10 CA-PEM-DYN-DG-REL      PIC XX.
006500                                10 FILLER PIC XX
006600    10 CA-PEM-DYN-DG-INIT      PIC XX.
006700                                10 FILLER PIC XX
006800    10 CA-PEM-SYSPRINT-WRITE    PIC XX.
006900                                10 FILLER PIC XX
007000    10 CA-PEM-NO-OP          PIC XX.
007100                                10 FILLER PIC XX
007200    10 CA-PEM-TRANS-DUMP-RETURN    PIC XX.
007300                                10 FILLER PIC XX
007400    10 CA-PEM-TRANS-DUMP-END      PIC XX.
007500                                10 FILLER PIC XX
007600    10 CA-PEM-DUMP-TCB-TRACE     PIC XX.
007700                                10 FILLER PIC XX
007800    10 CA-PEM-DUMP-DG-RETURN     PIC XX.
007900                                10 FILLER PIC XX
008000    10 CA-PEM-DUMP-DG-END       PIC XX.
008100                                10 FILLER PIC XX
008200    10 CA-PEM-ENABLE-ABEND-EXIT    PIC XX.
008300                                10 FILLER PIC XX
008400    10 CA-PEM-DISABLE-ABEND-EXIT    PIC XX.
008500                                10 FILLER PIC XX
008600    10 CA-PEM-CHECKPOINT       PIC XX.
008700                                10 FILLER PIC XX
008800    10 CA-PEM-DYN-DG-NO-INIT       PIC XX.
008900                                10 FILLER PIC XX
009000    10 CA-PEM-ROLLBACK         PIC XX.
009100                                10 FILLER PIC XX

```



Umbrella Programming

Problem Specifications—Batch Program

```

009200      10 CA-PEM-DLI-SYNCPOINT          PIC XX.
009300                                10 FILLER PIC XX
009400      10 CA-PEM-USERCC-EXCEP          PIC XX.
009500                                10 FILLER PIC XX
009600      10 CA-PEM-MSG9-ABEND          PIC XX.
009700                                10 FILLER PIC XX
009800      10 CA-PEM-APPC-SYNCPOINT          PIC XX.
009900                                10 FILLER PIC XX
010000      10 CA-PEM-DYN-DG-LENGTH          PIC XX.
010100                                10 FILLER PIC XX
010200      10 CA-PEM-DYN-PTR-REL           PIC XX.
010300                                10 FILLER PIC XX
010400      10 CA-PEM-DYN-PTR-INIT           PIC XX.
010401                                10 FILLER PIC XX
010402      10 CA-PEM-DYN-PTR-ANO            PIC XX.
010500                                10 FILLER PIC XX
010600      10 CA-PEM-PDG-HAS-CHANGED          PIC XX.
010700*
010800***** END OF P49003D *****

EJECT
*-----
*   THE FOLLOWING COPYBOOK P49022D DEFINES THE POSSIBLE VALUES
*   THAT MIGHT BE PLACED INTO A DATA GROUP ACTION FIELD PRIOR
*   TO ISSUING A PEM DATA BASE ACTIVITY.      DGA ACTION
*-----

000100*
000200*---* START OF P49022D *---* PEM ACTION CODES *-----*
000300*
000400 01  PEM-DATA-GROUP-ACTION-CODES.
000500  05  DGA-CODE-VALUES.
000600  10  DGA-NO-OP.
000700                                15  FILLER PIC S9(4) COMP VALUE +0000
000800  10  DGA-READ.                  15  FILLER PIC S9(4) COMP VALUE +0001
000900                                15  FILLER PIC S9(4) COMP VALUE +0002
001000  10  DGA-WRITE.                 15  FILLER PIC S9(4) COMP VALUE +0003
001100                                15  FILLER PIC S9(4) COMP VALUE +0004
001200  10  DGA-HOLD-P.                15  FILLER PIC S9(4) COMP VALUE +0005
001300                                15  FILLER PIC S9(4) COMP VALUE +0006
001400  10  DGA-ERASE.                 15  FILLER PIC S9(4) COMP VALUE +0007
001500                                15  FILLER PIC S9(4) COMP VALUE +0008
001600  10  DGA-READ-KEY-GE.           15  FILLER PIC S9(4) COMP VALUE +0009
001700                                15  FILLER PIC S9(4) COMP VALUE +0010
001800  10  DGA-FORCE-WRITE.          15  FILLER PIC S9(4) COMP VALUE +0011
001900                                15  FILLER PIC S9(4) COMP VALUE +0012
002000  10  DGA-READ-KEY-EQ.          15  FILLER PIC S9(4) COMP VALUE +0013
002100                                15  FILLER PIC S9(4) COMP VALUE +0014
002200  10  DGA-INSERT-FIRST.         15  FILLER PIC S9(4) COMP VALUE +0015
002300                                15  FILLER PIC S9(4) COMP VALUE +0016
002400  10  DGA-READ-LAST-REC.        15  FILLER PIC S9(4) COMP VALUE +0017
002500                                15  FILLER PIC S9(4) COMP VALUE +0018
002600  10  DGA-INSERT-LAST.          15  FILLER PIC S9(4) COMP VALUE +0019
002700                                15  FILLER PIC S9(4) COMP VALUE +0020
002800  10  DGA-READ-FIRST.           15  FILLER PIC S9(4) COMP VALUE +0021
002900                                15  FILLER PIC S9(4) COMP VALUE +0022
003000  10  DGA-INSERT-HERE.          15  FILLER PIC S9(4) COMP VALUE +0023
003100                                15  FILLER PIC S9(4) COMP VALUE +0024
003200  10  DGA-END-REQUEST.         15  FILLER PIC S9(4) COMP VALUE +0025
003300                                15  FILLER PIC S9(4) COMP VALUE +0026
003400  10  DGA-FORCE-NO-OP.         15  FILLER PIC S9(4) COMP VALUE +0027
003500                                15  FILLER PIC S9(4) COMP VALUE +0028
003600  10  DGA-CLOSE.                15  FILLER PIC S9(4) COMP VALUE +0029
003700                                15  FILLER PIC S9(4) COMP VALUE +0030
003800  10  DGA-OPEN-OUTPUT.          15  FILLER PIC S9(4) COMP VALUE +0031
003900                                15  FILLER PIC S9(4) COMP VALUE +0032
004000  10  DGA-OPEN-INPUT.           15  FILLER PIC S9(4) COMP VALUE +0033

```



Umbrella Programming

Problem Specifications—Batch Program

```

004100                               15 FILLER PIC S9(4) COMP VALUE +0038
004200      10 DGA-OPEN-UPDATE.          15 FILLER PIC S9(4) COMP VALUE +0039
004300                               15 FILLER PIC S9(4) COMP VALUE +0040
004400      10 DGA-POINT.              15 FILLER PIC S9(4) COMP VALUE +0041
004500      10 DGA-ENABLE-KEY-RANGE.    15 FILLER PIC S9(4) COMP VALUE +0042
004700      10 DGA-DISABLE-KEY-RANGE.   15 FILLER PIC S9(4) COMP VALUE +0043
004800      10 DGA-ENABLE-MULT-KRDB.    15 FILLER PIC S9(4) COMP VALUE +0044
005000      10 DGA-ENABLE-MULT-KRDB.    15 FILLER PIC S9(4) COMP VALUE +0045
005100                               15 FILLER PIC S9(4) COMP VALUE +0046
005200*
005300*----* END OF P49022D *-----*
      EJECT
*-----
*   THE FOLLOWING COPYBOOK P49023D CONTAINS THE POSSIBLE VALUES
*   THAT MIGHT BE CONTAINED IN A DATA GROUP RESULTS FIELD AFTER
*   A PEM DATA BASE ACTIVITY HAS BEEN ISSUED.      DGR RESULT
*           # # #
*   WHEN CONTROL IS RETURNED TO AN APPLICATION PROGRAM ON
*   COMPLETION OF A DATA BASE ACTIVITY, EACH DATA GROUP INVOLVED
*   IN THE ACTIVITY WILL CONTAIN A RESULT CODE REFLECTING THE
*   RESULT OF THE REQUESTED ACTION.  IF EACH INDIVIDUAL DATA
*   GROUP'S RESULT FIELD IS ZERO, THE TCB RESULT FIELD
*   (TCB-RESULT) WILL ALSO CONTAIN ZERO.
*
*   TCB-RESULT WILL BE SET TO NON-ZERO IF ANY OF THE INDIVIDUAL
*   DATA GROUPS IS NON-ZERO.  IT IS THE RESPONSIBILITY OF THE
*   APPLICATION PROGRAM TO DETERMINE THE CAUSE AND SEVERITY OF
*   THE RESULT CONDITION.  AN END-OF-DATA ON THE BASE DATA
*   GROUP (EOF) WILL CAUSE A NON-ZERO VALUE TO BE RETURNED TO
*   THE TCB-RESULT FIELD.
*-----
000100*---* START OF P49023D *---* PEM RESULT CODES *-----*
000200*
000300 01  PEM-DATA-GROUP-RESULT-CODES.
000400      05 DGR-CODE-VALUES.
000500      10 DGR-OK.
000600                               15 FILLER PIC S9(4) COMP VALUE +0000
000700      10 DGR-END-DATA.          15 FILLER PIC S9(4) COMP VALUE +0001
000800      10 DGR-I-O-ERR.          15 FILLER PIC S9(4) COMP VALUE +0002
000900      10 DGR-DUP-KEY.          15 FILLER PIC S9(4) COMP VALUE +0003
001000      10 DGR-NO-FIND.          15 FILLER PIC S9(4) COMP VALUE +0004
001100      10 DGR-SKEY-NO-FIND.     15 FILLER PIC S9(4) COMP VALUE +0005
001200      10 DGR-KRID-DISABLED.    15 FILLER PIC S9(4) COMP VALUE +0006
001300      10 DGR-DB-NOT-AVAIL.     15 FILLER PIC S9(4) COMP VALUE +0007
001400      10 DGR-SQLCA-ERROR.      15 FILLER PIC S9(4) COMP VALUE +0008
001500      10 DGR-LEN-ERR.          15 FILLER PIC S9(4) COMP VALUE +0009
001600      10 DGR-KRID-DISABLED.    15 FILLER PIC S9(4) COMP VALUE +0010
001700      10 DGR-DB-NOT-AVAIL.     15 FILLER PIC S9(4) COMP VALUE +0011
001800      10 DGR-SQLCA-ERROR.      15 FILLER PIC S9(4) COMP VALUE +0012
001900      10 DGR-LEN-ERR.          15 FILLER PIC S9(4) COMP VALUE +0013
002000      10 DGR-LEN-ERR.          15 FILLER PIC S9(4) COMP VALUE +0014
002100      10 DGR-LEN-ERR.          15 FILLER PIC S9(4) COMP VALUE +0015
002200      10 DGR-LEN-ERR.          15 FILLER PIC S9(4) COMP VALUE +0016
002300      10 DGR-LEN-ERR.          15 FILLER PIC S9(4) COMP VALUE +0017
002400      10 DGR-LEN-ERR.          15 FILLER PIC S9(4) COMP VALUE +0018
002500*
002600*----* END OF P49023D *-----*
      EJECT
*-----
*   THE FOLLOWING COPYBOOK I57104D CONTAINS THE ACTION CODES
*   USED IN ACCESSING CONVERTED PCD.
*
*   THE REQUESTED ACTION MUST BE PLACED INTO PCD-ACTION FIELD

```



Umbrella Programming

Problem Specifications—Batch Program

```
* IN THE PCD CONTROL BLOCK PRIOR TO ISSUING THE LINK ACTIVITY
* TO PCD PROCESSING, ACTIVITY 1013 (READ ONLY) OR ACTIVITY
* 1014 (READ/UPDATE).
* COPYBOOK I57101D CONTAINS DG 1452 DEFINITION AND HAS BEEN
* INCLUDED IN THE LINKAGE SECTION OF THIS PROGRAM.
*-----
000100**** START OF I57104D ***** PCD ACTION LIST *****
000200*
000300 01 PCD-ACTION-LIST.
000400 05 PCD-ACTION-CODES.
000500 10 PCD-NO-OPERATION.
000600 15 FILLER PIC S9(4) COMP VALUE +0000
000700 10 PCD-CREATE.
000800 15 FILLER PIC S9(4) COMP VALUE +0001
000900 10 PCD-ADD.
001000 15 FILLER PIC S9(4) COMP VALUE +0002
001100 10 PCD-READ.
001200 15 FILLER PIC S9(4) COMP VALUE +0003
001300 10 PCD-READ-NEXT.
001400 15 FILLER PIC S9(4) COMP VALUE +0004
001500 10 PCD-UPDATE.
001600 15 FILLER PIC S9(4) COMP VALUE +0005
001700 10 PCD-COPY.
001800 15 FILLER PIC S9(4) COMP VALUE +0006
001900 10 PCD-DELETE.
002000 15 FILLER PIC S9(4) COMP VALUE +0007
002100 10 PCD-CLEAR.
002200 15 FILLER PIC S9(4) COMP VALUE +0008
002300 10 PCD-REPLACE.
002400 15 FILLER PIC S9(4) COMP VALUE +0009
002500 10 PCD-ADD-IP.
002600 15 FILLER PIC S9(4) COMP VALUE +0010
002700 10 PCD-DELETE-IP.
002800 15 FILLER PIC S9(4) COMP VALUE +0011
002900 10 PCD-READ-NEXT-SET.
003000 15 FILLER PIC S9(4) COMP VALUE +0012
003100 10 PCD-READ-NEXT-EFFECT-SET.
003200 15 FILLER PIC S9(4) COMP VALUE +0013
003300 10 PCD-READ-NEXT-SET-SEQ.
003400 15 FILLER PIC S9(4) COMP VALUE +0014
003500*
003600***** END OF I57104D *****
EJECT
*-----
* THE FOLLOWING COPYBOOK I57105D CONTAINS RESULT VALUES
* RETURNED IN FIELD PCD-RESULT DG 1452 THE PCD CONTROL BLOCK
* AFTER A PCD ACTION HAS BEEN REQUESTED.
* COPYBOOK I57101D CONTAINS DG 1452 DEFINITION AND HAS BEEN
* INCLUDED IN THE LINKAGE SECTION OF THIS PROGRAM.
*-----
000100*****
000200* I57105D - PCD RESULT LIST
000300*****
000400 01 PCD-RESULT-LIST.
000500 05 PCD-RESULT-CODES.
000600* PCD-NO-ERRORS
000700 10 FILLER PIC S9(8) COMP VALUE +00000.
000800* PCD-INVALID-ACTION
000900 10 FILLER PIC S9(8) COMP VALUE +57001.
001000* PCD-ACTN-ACTV-ERROR
001100 10 FILLER PIC S9(8) COMP VALUE +57002.
001200* PCD-NO-GROUP-FOUND
001300 10 FILLER PIC S9(8) COMP VALUE +57003.
001400* PCD-NO-SET-FOUND
001500 10 FILLER PIC S9(8) COMP VALUE +57004.
```



Umbrella Programming

Problem Specifications—Batch Program

001600*	PCD-NO-ELEMENT-FOUND	
001700	10 FILLER	PIC S9(8) COMP VALUE +57005.
001800*	PCD-DUPLICATE-GROUP	
001900	10 FILLER	PIC S9(8) COMP VALUE +57006.
002000*	PCD-DUPLICATE-SET	
002100	10 FILLER	PIC S9(8) COMP VALUE +57007.
002200*	PCD-DUPLICATE-ELEMENT	
002300	10 FILLER	PIC S9(8) COMP VALUE +57008.
002400*	PCD-END-OF-FILE	
002500	10 FILLER	PIC S9(8) COMP VALUE +57009.
002600*	PCD-END-OF-GROUP	
002700	10 FILLER	PIC S9(8) COMP VALUE +57010.
002800*	PCD-END-OF-SET	
002900	10 FILLER	PIC S9(8) COMP VALUE +57011.
003000*	PCD-INPUT-EXIT-ERROR	
003100	10 FILLER	PIC S9(8) COMP VALUE +57012.
003200*	PCD-OUTPUT-EXIT-ERROR	
003300	10 FILLER	PIC S9(8) COMP VALUE +57013.
003400*	PCD-MAINT-EXIT-ERROR	
003500	10 FILLER	PIC S9(8) COMP VALUE +57014.
003600*	PCD-INVALID-EFF-DATE	
003700	10 FILLER	PIC S9(8) COMP VALUE +57015.
003800*	PCD-INVALID-EXP-DATE	
003900	10 FILLER	PIC S9(8) COMP VALUE +57016.
004000*	PCD-INVALID-OUT-EFF-DATE	
004100	10 FILLER	PIC S9(8) COMP VALUE +57017.
004200*	PCD-INVALID-OUT-EXP-DATE	
004300	10 FILLER	PIC S9(8) COMP VALUE +57018.
004400*	PCD-INVALID-DATA-GROUP	
004500	10 FILLER	PIC S9(8) COMP VALUE +57019.
004600*	PCD-INVALID-DISP-ACTV	
004700	10 FILLER	PIC S9(8) COMP VALUE +57020.
004800*	PCD-INVALID-DBLK-ACTV	
004900	10 FILLER	PIC S9(8) COMP VALUE +57021.
005000*	PCD-INVALID-DB-ACTV	
005100	10 FILLER	PIC S9(8) COMP VALUE +57022.
005200*	PCD-MISSING-USER-DATA	
005300	10 FILLER	PIC S9(8) COMP VALUE +57023.
005400*	PCD-MISSING-CTL-DATA	
005500	10 FILLER	PIC S9(8) COMP VALUE +57024.
005600*	PCD-INPUT-SET-NOT-FOUND	
005700	10 FILLER	PIC S9(8) COMP VALUE +57025.
005800*	PCD-OUTPUT-SET-NOT-FOUND	
005900	10 FILLER	PIC S9(8) COMP VALUE +57026.
006000*	PCD-INPUT-GROUP-NOT-FOUND	
006100	10 FILLER	PIC S9(8) COMP VALUE +57027.
006200*	PCD-OUTPUT-GROUP-NOT-FOUND	
006300	10 FILLER	PIC S9(8) COMP VALUE +57028.
006400*	PCD-UNAUTHORIZED-ACTION	
006500	10 FILLER	PIC S9(8) COMP VALUE +57029.
006600*	PCD-CC-NO-IS-ZERO-ON-UPDT	
006700	10 FILLER	PIC S9(8) COMP VALUE +57030.
006800*	PCD-NO-ALTERNATE-WO-PRIMARY	
006900	10 FILLER	PIC S9(8) COMP VALUE +57031.
007000*	PCD-UNDETERMINED-ERROR	
007100	10 FILLER	PIC S9(8) COMP VALUE +57999.
007200	05 FILLER	REDEFINES PCD-RESULT-CODES.
007300	10 FILLER.	
007400	15 FILLER	PIC XX.
007500	15 PCD-NO-ERRORS	PIC XX.
007600	10 FILLER.	
007700	15 FILLER	PIC XX.
007800	15 PCD-INVALID-ACTION	PIC XX.
007900	10 FILLER.	
008000	15 FILLER	PIC XX.



Umbrella Programming

Problem Specifications—Batch Program

008100	15 PCD-ACTN-ACTV-ERROR	PIC XX.
008200	10 FILLER.	
008300	15 FILLER	PIC XX.
008400	15 PCD-NO-GROUP-FOUND	PIC XX.
008500	10 FILLER.	
008600	15 FILLER	PIC XX.
008700	15 PCD-NO-SET-FOUND	PIC XX.
008800	10 FILLER.	
008900	15 FILLER	PIC XX.
009000	15 PCD-NO-ELEMENT-FOUND	PIC XX.
009100	10 FILLER.	
009200	15 FILLER	PIC XX.
009300	15 PCD-DUPLICATE-GROUP	PIC XX.
009400	10 FILLER.	
009500	15 FILLER	PIC XX.
009600	15 PCD-DUPLICATE-SET	PIC XX.
009700	10 FILLER.	
009800	15 FILLER	PIC XX.
009900	15 PCD-DUPLICATE-ELEMENT	PIC XX.
010000	10 FILLER.	
010100	15 FILLER	PIC XX.
010200	15 PCD-END-OF-FILE	PIC XX.
010300	10 FILLER.	
010400	15 FILLER	PIC XX.
010500	15 PCD-END-OF-GROUP	PIC XX.
010600	10 FILLER.	
010700	15 FILLER	PIC XX.
010800	15 PCD-END-OF-SET	PIC XX.
010900	10 FILLER.	
011000	15 FILLER	PIC XX.
011100	15 PCD-INPUT-EXIT-ERROR	PIC XX.
011200	10 FILLER.	
011300	15 FILLER	PIC XX.
011400	15 PCD-OUTPUT-EXIT-ERROR	PIC XX.
011500	10 FILLER.	
011600	15 FILLER	PIC XX.
011700	15 PCD-MAINT-EXIT-ERROR	PIC XX.
011800	10 FILLER.	
011900	15 FILLER	PIC XX.
012000	15 PCD-INVALID-EFF-DATE	PIC XX.
012100	10 FILLER.	
012200	15 FILLER	PIC XX.
012300	15 PCD-INVALID-EXP-DATE	PIC XX.
012400	10 FILLER.	
012500	15 FILLER	PIC XX.
012600	15 PCD-INVALID-OUT-EFF-DATE	PIC XX.
012700	10 FILLER.	
012800	15 FILLER	PIC XX.
012900	15 PCD-INVALID-OUT-EXP-DATE	PIC XX.
013000	10 FILLER.	
013100	15 FILLER	PIC XX.
013200	15 PCD-INVALID-DATA-GROUP	PIC XX.
013300	10 FILLER.	
013400	15 FILLER	PIC XX.
013500	15 PCD-INVALID-DISP-ACTV	PIC XX.
013600	10 FILLER.	
013700	15 FILLER	PIC XX.
013800	15 PCD-INVALID-DBLK-ACTV	PIC XX.
013900	10 FILLER.	
014000	15 FILLER	PIC XX.
014100	15 PCD-INVALID-DB-ACT	PIC XX.
014200	10 FILLER.	
014300	15 FILLER	PIC XX.
014400	15 PCD-MISSING-USER-DATA	PIC XX.
014500	10 FILLER.	



Umbrella Programming

Problem Specifications—Batch Program

```

014600      15 FILLER          PIC XX.
014700      15 PCD-MISSING-CTL-DATA  PIC XX.
014800      10 FILLER.
014900      15 FILLER          PIC XX.
015000      15 PCD-INPUT-SET-NOT-FOUND  PIC XX.
015100      10 FILLER.
015200      15 FILLER          PIC XX.
015300      15 PCD-OUTPUT-SET-NOT-FOUND  PIC XX.
015400      10 FILLER.
015500      15 FILLER          PIC XX.
015600      15 PCD-INPUT-GROUP-NOT-FOUND  PIC XX.
015700      10 FILLER.
015800      15 FILLER          PIC XX.
015900      15 PCD-OUTPUT-GROUP-NOT-FOUND  PIC XX.
016000      10 FILLER.
016100      15 FILLER          PIC XX.
016200      15 PCD-UNAUTHORIZED-ACTION  PIC XX.
016300      10 FILLER.
016400      15 FILLER          PIC XX.
016500      15 PCD-CC-NO-IS-ZERO-ON-UPDT  PIC XX.
016600      10 FILLER.
016700      15 FILLER          PIC XX.
016800      15 PCD-NO-ALT-WO-PRIMARY  PIC XX.
016900      10 FILLER.
017000      15 FILLER          PIC XX.
017100      15 PCD-UNDETERMINED-ERROR  PIC XX.
017200*     15 FILLER          PIC XX.
017300*****EJECT*****
*-----*
*   THE FOLLOWING COPYBOOK T58007D CONTAINS ALL THE POSSIBLE
*   ACTIONS THAT MIGHT BE REQUESTED OF THE DATE SERVICES SYSTEM.
*
*   THE REQUESTED ACTION CODE MUST BE LOADED INTO THE ACTION
*   FIELD DCB-ACTION IN THE DATE CONTROL BLOCK DATA GROUP 2000
*   PRIOR TO THE LINK TO DATE SERVICES ACTIVITY 1900.
*   THE DATE CONTROL BLOCK IS DEFINED BY COPYBOOK T58001D
*   INCLUDED IN THE LINKAGE SECTION OF THIS PROGRAM.
*
*-----*
000100*-- START OF T58007D *---* DATE ROUTINE ACTIONS *-----*
000200*
000300*
000400*
000500*****000600*
000700 01 DCB-ACTIONS.
000800      05 DCB-PROG-LINK.
000900      10 DCB-LINK-DSV-LONG.
001000      15 FILLER          PIC S9(9) COMP VALUE +1900.
001100      10 FILLER          REDEFINES DCB-LINK-DSV-LONG.
001200      15 FILLER          PIC XX.
001300      15 DCB-LINK-DSV    PIC XX.
001400*
001500      05 DCB-PROG-ACTIONS.
001600* GREGORIAN TO JULIAN
001700      10 DCB-AC-GREG-TO-JUL.
001800      15 FILLER          PIC S9(4) COMP VALUE +0001.
001900* GREGORIAN TO CALCULATION
002000      10 DCB-AC-GREG-TO-CALC.
002100      15 FILLER          PIC S9(4) COMP VALUE +0002.
002200* GREGORIAN TO NUMERIC DISPLAY
002300      10 DCB-AC-GREG-TO-NUM-DSP.
002400      15 FILLER          PIC S9(4) COMP VALUE +0003.
002500* GREGORIAN TO ALPHANUMERIC DISPLAY
002600      10 DCB-AC-GREG-TO-ALPHA-DSP.

```



Umbrella Programming

Problem Specifications—Batch Program

```
002700           15 FILLER          PIC S9(4) COMP VALUE +0004.  
002800* JULIAN TO GREGORIAN  
002900           10 DCB-AC-JUL-TO-GREG.  
003000           15 FILLER          PIC S9(4) COMP VALUE +0010.  
003100* JULIAN TO CALCULATION  
003200           10 DCB-AC-JUL-TO-CALC.  
003300           15 FILLER          PIC S9(4) COMP VALUE +0011.  
003400* JULIAN TO NUMERIC DISPLAY  
003500           10 DCB-AC-JUL-TO-NUM-DSP.  
003600           15 FILLER          PIC S9(4) COMP VALUE +0012.  
003700* JULIAN TO ALPHANUMERIC DISPLAY  
003800           10 DCB-AC-JUL-TO-ALPHA-DSP.  
003900           15 FILLER          PIC S9(4) COMP VALUE +0013.  
004000* CALCULATION TO GREGORIAN  
004100           10 DCB-AC-CALC-TO-GREG.  
004200           15 FILLER          PIC S9(4) COMP VALUE +0020.  
004300* CALCULATION TO JULIAN  
004400           10 DCB-AC-CALC-TO-JUL.  
004500           15 FILLER          PIC S9(4) COMP VALUE +0021.  
004600* CALCULATION TO NUMERIC DISPLAY  
004700           10 DCB-AC-CALC-TO-NUM-DSP.  
004800           15 FILLER          PIC S9(4) COMP VALUE +0022.  
004900* CALCULATION TO ALPHANUMERIC DISPLAY  
005000           10 DCB-AC-CALC-TO-ALPHA-DSP.  
005100           15 FILLER          PIC S9(4) COMP VALUE +0023.  
005200* BUMP TO NEXT DATE GREGORIAN  
005300           10 DCB-AC-BND-GREG.  
005400           15 FILLER          PIC S9(4) COMP VALUE +0030.  
005500* BUMP TO NEXT DATE JULIAN  
005600           10 DCB-AC-BND-JUL.  
005700           15 FILLER          PIC S9(4) COMP VALUE +0031.  
005800* BUMP TO NEXT DATE CALCULATION  
005900           10 DCB-AC-BND-CALC.  
006000           15 FILLER          PIC S9(4) COMP VALUE +0032.  
006100* BUMP TO NEXT DATE GREGORIAN 360  
006200           10 DCB-AC-BND-G360.  
006300           15 FILLER          PIC S9(4) COMP VALUE +0033.  
006400* BUMP TO NEXT BUSINESS DATE GREGORIAN  
006500           10 DCB-AC-BNBD-GREG.  
006600           15 FILLER          PIC S9(4) COMP VALUE +0040.  
006700* BUMP TO NEXT BUSINESS DATE JULIAN  
006800           10 DCB-AC-BNBD-JUL.  
006900           15 FILLER          PIC S9(4) COMP VALUE +0041.  
007000* BUMP TO NEXT BUSINESS DATE CALCULATION  
007100           10 DCB-AC-BNBD-CALC.  
007200           15 FILLER          PIC S9(4) COMP VALUE +0042.  
007300* GREGORIAN DAY DIFFERENCE  
007400           10 DCB-AC-DIFF-GREG.  
007500           15 FILLER          PIC S9(4) COMP VALUE +0050.  
007600* JULIAN DAY DIFFERENCE  
007700           10 DCB-AC-DIFF-JUL.  
007800           15 FILLER          PIC S9(4) COMP VALUE +0051.  
007900* CALCULATION DAY DIFFERENCE  
008000           10 DCB-AC-DIFF-CALC.  
008100           15 FILLER          PIC S9(4) COMP VALUE +0052.  
008200* GREGORIAN 360 DAY DIFFERENCE  
008300           10 DCB-AC-DIFF-G360.  
008400           15 FILLER          PIC S9(4) COMP VALUE +0053.  
008500* BUSINESS DAY DIFFERENCE GREGORIAN  
008600           10 DCB-AC-BUS-DIFF-GREG.  
008700           15 FILLER          PIC S9(4) COMP VALUE +0060.  
008800* BUSINESS DAY DIFFERENCE JULIAN  
008900           10 DCB-AC-BUS-DIFF-JUL.  
009000           15 FILLER          PIC S9(4) COMP VALUE +0061.  
009100* BUSINESS DAY DIFFERENCE CALCULATION
```



Umbrella Programming

Problem Specifications—Batch Program

```

009200      10 DCB-AC-BUS-DIFF-CALC.
009300          15 FILLER      PIC S9(4) COMP VALUE +0062.
009400* GREGORIAN MONTHS DIFFERENCE
009500          10 DCB-AC-MO-DIFF-GREG.
009600          15 FILLER      PIC S9(4) COMP VALUE +0070.
009700* HOLIDAY TABLE REQUEST
009800          10 DCB-AC-HOLIDAYS.
009900          15 FILLER      PIC S9(4) COMP VALUE +0080.
010000* HOLIDAY TABLE RENEW REQUEST
010100          10 DCB-AC-RENEW-HOLIDAYS.
010200          15 FILLER      PIC S9(4) COMP VALUE +0081.
010300* FORMATED DATE (CYYMMDD) TO GREGORIAN
010400          10 DCB-AC-FMT00-GREG.
010500          15 FILLER      PIC S9(4) COMP VALUE +0100.
010600* FORMATED DATE (YY-MM-DD) TO GREGORIAN
010700          10 DCB-AC-FMT01-GREG.
010800          15 FILLER      PIC S9(4) COMP VALUE +0101.
010900* FORMATED DATE (MMDDYY) TO GREGORIAN
011000          10 DCB-AC-FMT02-GREG.
011100          15 FILLER      PIC S9(4) COMP VALUE +0102.
011200* FORMATED DATE (MM-DD-YY) TO GREGORIAN
011300          10 DCB-AC-FMT03-GREG.
011400          15 FILLER      PIC S9(4) COMP VALUE +0103.
011500* FORMATED DATE GREGORIAN TO (MMDDYYC)
011600          10 DCB-AC-FMT04-GREG.
011700          15 FILLER      PIC S9(4) COMP VALUE +0104.
011800* FORMATED DATE GREGORIAN TO (MMDDYY)
011900          10 DCB-AC-FMT05-GREG.
012000          15 FILLER      PIC S9(4) COMP VALUE +0105.
012100* DATE STATUS CHECK ALL
012200          10 DCB-AC-DSC-ALL.
012300          15 FILLER      PIC S9(4) COMP VALUE +0130.
012400* DATE STATUS CHECK PHYSICAL FIRST OF MONTH
012500          10 DCB-AC-DSC-PHY-FOM.
012600          15 FILLER      PIC S9(4) COMP VALUE +0131.
012700* DATE STATUS CHECK PHYSICAL FIRST OF QUARTER
012800          10 DCB-AC-DSC-PHY-FOQ.
012900          15 FILLER      PIC S9(4) COMP VALUE +0132.
013000* DATE STATUS CHECK PHYSICAL FIRST OF YEAR
013100          10 DCB-AC-DSC-PHY-FOY.
013200          15 FILLER      PIC S9(4) COMP VALUE +0133.
013300* DATE STATUS CHECK PHYSICAL END OF MONTH
013400          10 DCB-AC-DSC-PHY-EOM.
013500          15 FILLER      PIC S9(4) COMP VALUE +0134.
013600* DATE STATUS CHECK PHYSICAL END OF QUARTER
013700          10 DCB-AC-DSC-PHY-EOQ.
013800          15 FILLER      PIC S9(4) COMP VALUE +0135.
013900* DATE STATUS CHECK PHYSICAL END OF YEAR
014000          10 DCB-AC-DSC-PHY-EOY.
014100          15 FILLER      PIC S9(4) COMP VALUE +0136.
014200* DATE STATUS CHECK BUSINESS FIRST OF MONTH
014300          10 DCB-AC-DSC-BUS-FOM.
014400          15 FILLER      PIC S9(4) COMP VALUE +0137.
014500* DATE STATUS CHECK BUSINESS FIRST OF QUARTER
014600          10 DCB-AC-DSC-BUS-FOQ.
014700          15 FILLER      PIC S9(4) COMP VALUE +0138.
014800* DATE STATUS CHECK BUSINESS FIRST OF YEAR
014900          10 DCB-AC-DSC-BUS-FOY.
015000          15 FILLER      PIC S9(4) COMP VALUE +0139.
015100* DATE STATUS CHECK BUSINESS END OF MONTH
015200          10 DCB-AC-DSC-BUS-EOM.
015300          15 FILLER      PIC S9(4) COMP VALUE +0140.
015400* DATE STATUS CHECK BUSINESS END OF QUARTER
015500          10 DCB-AC-DSC-BUS-EOQ.
015600          15 FILLER      PIC S9(4) COMP VALUE +0141.

```



Umbrella Programming

Problem Specifications—Batch Program

```
015700* DATE STATUS CHECK BUSINESS END OF YEAR
015800          10 DCB-AC-DSC-BUS-EOY.
015900          15 FILLER      PIC S9(4) COMP VALUE +0142.
016000* DATE STATUS CHECK BUSINESS DAY OR NON BUSINESS DAY
016100          10 DCB-AC-DSC-BUS-DAY.
016200          15 FILLER      PIC S9(4) COMP VALUE +0143.
016300* DATE STATUS CHECK DAY OF WEEK
016400          10 DCB-AC-DSC-WEEK-DAY.
016500          15 FILLER      PIC S9(4) COMP VALUE +0144.
016600* DATE STATUS CHECK LEAP YEAR
016700          10 DCB-AC-DSC-LEAP-YEAR.
016800          15 FILLER      PIC S9(4) COMP VALUE +0145.
016900* DATE STATUS CHECK PHYSICAL FIRST OF WEEK
017000          10 DCB-AC-DSC-PHY-FOW.
017100          15 FILLER      PIC S9(4) COMP VALUE +0146.
017200* DATE STATUS CHECK PHYSICAL END OF WEEK
017300          10 DCB-AC-DSC-PHY-EOW.
017400          15 FILLER      PIC S9(4) COMP VALUE +0147.
017500* DATE STATUS CHECK BUSINESS FIRST OF WEEK
017600          10 DCB-AC-DSC-BUS-FOW.
017700          15 FILLER      PIC S9(4) COMP VALUE +0148.
017800* DATE STATUS CHECK BUSINESS END OF WEEK
017900          10 DCB-AC-DSC-BUS-EOW.
018000          15 FILLER      PIC S9(4) COMP VALUE +0149.
018100* CALCULATE CURRENT BUSINESS DAY
018200          10 DCB-AC-CALC-CURR-BUS.
018300          15 FILLER      PIC S9(4) COMP VALUE +0160.
018400* CALCULATE NEXT WEEK DAY
018500          10 DCB-AC-CALC-NXT-WKDAY.
018600          15 FILLER      PIC S9(4) COMP VALUE +0161.
018700* CONVERT GREGORIAN TO SQL FORMATTED DATE
018800          10 DCB-HOGAN-TO-SQL-DATE.
018900          15 FILLER      PIC S9(4) COMP VALUE +0200.
019000* CONVERT SQL FORMATTED DATE TO GREGORIAN
019100          10 DCB-SQL-TO-HOGAN-DATE.
019200          15 FILLER      PIC S9(4) COMP VALUE +0201.
019300* CONVERT HOGAN TO SQL TIME
019400          10 DCB-HOGAN-TO-SQL-TIME.
019500          15 FILLER      PIC S9(4) COMP VALUE +0202.
019600* CONVERT SQL TO HOGAN TIME
019700          10 DCB-SQL-TO-HOGAN-TIME.
019800          15 FILLER      PIC S9(4) COMP VALUE +0203.
019900*
020000*
020100*----* END OF T58007D *-----*
          EJECT
*-----
*   THE FOLLOWING COPYBOOK T58008D IS USED TO DEFINE THE POSSIBLE
*   RESULT VALUES RETURNED FROM A DATE SERVICES REQUEST IN THE
*   DCB-RESULT FIELD OF THE DATE CONTROL BLOCK DATA GROUP 2000.
*-----
000100*---* START OF T58008D *---* DATE ROUTINE RESULTS *-----
000200*
000300*
000400*
000500*****000600*
000600*
000700 01 DCB-RESULTS.
000800 05 DCB-RESULT-VALUES.
000900* SUCCESSFUL COMPLETION
001000          10 FILLER      PIC S9(8) COMP VALUE +00000.
001100* INVALID ACTION
001200          10 FILLER      PIC S9(8) COMP VALUE +58001.
001300* INVALID GREGORIAN MONTH
001400          10 FILLER      PIC S9(8) COMP VALUE +58002.
```



Umbrella Programming

Problem Specifications—Batch Program

```

001500* INVALID GREGORIAN DAY
001600      10 FILLER          PIC S9(8) COMP  VALUE +58003.
001700* INVALID JULIAN DAY
001800      10 FILLER          PIC S9(8) COMP  VALUE +58004.
001900* INVALID CALCULATION DATE
002000      10 FILLER          PIC S9(8) COMP  VALUE +58005.
002100* INVALID DIFFERENCE FACTOR
002200      10 FILLER          PIC S9(8) COMP  VALUE +58006.
002300* INVALID CYCLE PERIOD
002400      10 FILLER          PIC S9(8) COMP  VALUE +58050.
002500* INVALID FREQ CODE
002600      10 FILLER          PIC S9(8) COMP  VALUE +58051.
002700* INVALID FREQ WEEK
002800      10 FILLER          PIC S9(8) COMP  VALUE +58052.
002900* INVALID FREQ DATE
003000      10 FILLER          PIC S9(8) COMP  VALUE +58053.
003100* INVALID FREQ BUMP
003200      10 FILLER          PIC S9(8) COMP  VALUE +58054.
003300* INVALID FREQ CYCLE
003400      10 FILLER          PIC S9(8) COMP  VALUE +58055.
003500* INVALID CYCLE EXCEPT
003600      10 FILLER          PIC S9(8) COMP  VALUE +58056.
003700* INVALID FREQ AMT
003800      10 FILLER          PIC S9(8) COMP  VALUE +58057.
003900* INVALID GREGORIAN MONTH IN DATE OUT
004000      10 FILLER          PIC S9(8) COMP  VALUE +58102.
004100* INVALID GREGORIAN DAY IN DATE OUT
004200      10 FILLER          PIC S9(8) COMP  VALUE +58103.
004300* INVALID JULIAN DAY IN DATE OUT
004400      10 FILLER          PIC S9(8) COMP  VALUE +58104.
004500* INVALID CALCULATION DATE IN DATE OUT
004600      10 FILLER          PIC S9(8) COMP  VALUE +58105.
004700* INVALID ACTION FOR DATE
004800      10 FILLER          PIC S9(8) COMP  VALUE +58106.
004900* NOT EFFECTIVE
005000      10 FILLER          PIC S9(8) COMP  VALUE +58150.
005100* MORE THAN 15 CYCLES
005200      10 FILLER          PIC S9(8) COMP  VALUE +58151.
005300* INVALID FROM-THRU RANGE
005400      10 FILLER          PIC S9(8) COMP  VALUE +58152.
005500* INVALID FIRST-LAST RANGE
005600      10 FILLER          PIC S9(8) COMP  VALUE +58153.
005700* WARNING CALCULATED DATES HAVE BEEN ADJUSTED.
005800      10 FILLER          PIC S9(8) COMP  VALUE +58154.
005900* SQL DATE IS INVALID
006000      10 FILLER          PIC S9(8) COMP  VALUE +58200.
006100* SQL FORMAT ON SITE CONTROL RECORD NOT SUPPORTED
006200      10 FILLER          PIC S9(8) COMP  VALUE +58201.
006300*
006400*-----*
006500*
006600*
006700      05 DCB-RESULT-NAMES    REDEFINES DCB-RESULT-VALUES.
006800      10 FILLER.
006900      15 FILLER          PIC XX.
007000      15 DCB-RS-OK        PIC XX.
007100      10 FILLER.
007200      15 FILLER          PIC XX.
007300      15 DCB-RS-INVALID-ACTION PIC XX.
007400      10 FILLER.
007500      15 FILLER          PIC XX.
007600      15 DCB-RS-INVALID-GREG-MONTH PIC XX.
007700      10 FILLER.
007800      15 FILLER          PIC XX.
007900      15 DCB-RS-INVALID-GREG-DAY  PIC XX.

```



Umbrella Programming

Problem Specifications—Batch Program

```
008000      10 FILLER.  
008100          15 FILLER          PIC XX.  
008200          15 DCB-RS-INVALID-JUL-DAY    PIC XX.  
008300      10 FILLER.  
008400          15 FILLER          PIC XX.  
008500          15 DCB-RS-INVALID-CALC-DATE    PIC XX.  
008600      10 FILLER.  
008700          15 FILLER          PIC XX.  
008800          15 DCB-RS-INVALID-DIFF-FACTOR    PIC XX.  
008900      10 FILLER.  
009000          15 FILLER          PIC XX.  
009100          15 DCB-RS-INVALID-CYCLE-PERIOD    PIC XX.  
009200      10 FILLER.  
009300          15 FILLER          PIC XX.  
009400          15 DCB-RS-INVALID-FREQ-CODE    PIC XX.  
009500      10 FILLER.  
009600          15 FILLER          PIC XX.  
009700          15 DCB-RS-INVALID-FREQ-WEEK    PIC XX.  
009800      10 FILLER.  
009900          15 FILLER          PIC XX.  
010000          15 DCB-RS-INVALID-FREQ-DATE    PIC XX.  
010100      10 FILLER.  
010200          15 FILLER          PIC XX.  
010300          15 DCB-RS-INVALID-FREQ-BUMP    PIC XX.  
010400      10 FILLER.  
010500          15 FILLER          PIC XX.  
010600          15 DCB-RS-INVALID-FREQ-CYCLE    PIC XX.  
010700      10 FILLER.  
010800          15 FILLER          PIC XX.  
010900          15 DCB-RS-INVALID-CYCLE-EXCEPT    PIC XX.  
011000      10 FILLER.  
011100          15 FILLER          PIC XX.  
011200          15 DCB-RS-INVALID-FREQ-AMT    PIC XX.  
011300      10 FILLER.  
011400          15 FILLER          PIC XX.  
011500          15 DCB-RS-INVALID-GREG-MO-DO    PIC XX.  
011600      10 FILLER.  
011700          15 FILLER          PIC XX.  
011800          15 DCB-RS-INVALID-GREG-DAY-DO    PIC XX.  
011900      10 FILLER.  
012000          15 FILLER          PIC XX.  
012100          15 DCB-RS-INVALID-JUL-DAY-DO    PIC XX.  
012200      10 FILLER.  
012300          15 FILLER          PIC XX.  
012400          15 DCB-RS-INVALID-CALC-DATE-DO    PIC XX.  
012500      10 FILLER.  
012600          15 FILLER          PIC XX.  
012700          15 DCB-RS-INVALID-DATE-ACT    PIC XX.  
012800      10 FILLER.  
012900          15 FILLER          PIC XX.  
013000          15 DCB-RS-NOT-EFFECTIVE    PIC XX.  
013100      10 FILLER.  
013200          15 FILLER          PIC XX.  
013300          15 DCB-RS-MORE-CYCLES    PIC XX.  
013400      10 FILLER.  
013500          15 FILLER          PIC XX.  
013600          15 DCB-RS-INVALID-FT-RANGE    PIC XX.  
013700      10 FILLER.  
013800          15 FILLER          PIC XX.  
013900          15 DCB-RS-INVALID-FL-RANGE    PIC XX.  
014000      10 FILLER.  
014100          15 FILLER          PIC XX.  
014200          15 DCB-RS-DATE-ADJUST    PIC XX.  
014300      10 FILLER.  
014400          15 FILLER          PIC XX.
```



Umbrella Programming

Problem Specifications—Batch Program

```

014500      15 DCB-RS-SQL-DATE-INVALID      PIC XX.
014600      10 FILLER.
014700      15 FILLER                  PIC XX.
014800      15 DCB-RS-SQL-FORMAT-NOT-SUP   PIC XX.
014900*
015000*----* END OF T58008D *-----*
      EJECT
      LINKAGE SECTION.
*****
*          L I N K A G E   S E C T I O N
*
*****-----*
* THE FOLLOWING COPYBOOK P49000D DEFINES DATA GROUP 1
* TRANSACTION CONTROL BLOCK (TCB)
*
* THE APPLICATION CONTROL BLOCK IS CONTAINED WITHIN THE TCB.
* THE ACB IS USED FOR CDMF/PCD PROCESSING.
* COPYBOOK U48004D CONTAINS THE CDMF ACTION CODES
* COPYBOOK U48024D CONTAINS THE CDMF RESULT CODES
* ACTIVITY ID 48000
*
* THE TCB MUST BE THE FIRST DATA GROUP IN THE LINKAGE SECTION.
* THE TCB DOES NOT NEED TO BE LISTED IN THE UMB PROGRAM DEF.
* ALL OTHER DATA GROUPS MUST BE LISTED IN THE UMB PROGRAM DEF.
* IN THE SAME ORDER AS THEY ARE LISTED IN THE LINKAGE SECTION
* AND IN THE USING STATEMENT OF THIS APPLICATION PROGRAM.
*-----*
000100**** START OF P49000D ***** TCB *****
000200* THIS COPYBOOK NOW DEFINES FOUR DISTINCT DATA GROUPS:
000300* 1) D.G. 00001, THE USER TRANSACTION CONTROL BLOCK AND THE CDMF
000400*     APPLICATION CONTROL BLOCK;
000500* 2) D.G. 00012, THE SYSIN INPUT DATA GROUP;
000600* 3) D.G. 00013, THE SYSPRINT OUPUT DATA GROUP; AND
000700* 4) D.G. 00010, THE SECURITY CONTROL BLOCK, THE TCB USER AREA,
000800*     THE USER TCB EXTENSION AREA, AND THE INTERNAL TCB EXTENSION
000900*     AREA (ALC PROGRAMS ONLY).
001000*****-----*
001100      SKIP1
001200*****-----*
001300* DATA GROUP 00001 (USER TRANSACTION CONTROL BLOCK)
001400*****-----*
001500*
001600 01 TRANSACTION-CONTROL-BLOCK.
001700      05 FILLER                  PIC XXXX.
001800      05 TCB-TRANS-NO            PIC XXXX.
001900      05 TCB-CO-ID              PIC XX.
002000      05 TCB-APPL-ID            PIC XX.
002100      05 TCB-FUNC-ID            PIC XX.
002200      05 TCB-SOURCE-TYPE.
002300          10 TCB-SOURCE-TYPE-N    PIC S9(4) COMP.
002400          88 TCB-ONLINE           VALUE +3 +5 +6.
002500          88 TCB-BTCH             VALUE +4.
002600          88 TCB-APPL-SOURCE       VALUE +2.
002700          88 TCB-AUTHORIZATIONS    VALUE +5 +6.
002800          88 TCB-MANNED-TELLER     VALUE +5.
002900          88 TCB-UNMANNED-TELLER   VALUE +6.
003000      05 TCB-ACTIVITY.
003100          10 TCB-ACTIVITY-N      PIC S9(04) COMP.
003200      05 TCB-RESULT              PIC XX.
003300      05 TCB-USER-DATA.
003400          10 FILLER                PIC X(6).
003500          10 TCB-DATA-GROUP       PIC XX.
003600          10 TCB-PARM-POS         PIC XX.

```



Umbrella Programming

Problem Specifications—Batch Program

```

003700      15 TCB-PARM-POS-N      PIC S9(4) COMP.
003800      05 FILLER           REDEFINES TCB-USER-DATA.
003900      10 TCB-USER-INFO      PIC XXXX.
004000      10 TCB-USER-COND      PIC XX.
004100      10 FILLER           PIC X(4).
004200      05 FILLER           REDEFINES TCB-USER-DATA.
004300      10 TCB-USER-CC       PIC XX.
004400      10 TCB-USER-RESULT     PIC XX.
004500      10 TCB-USER-ENVMT     PIC XX.
004600      10 FILLER           PIC X(4).
004700      05 FILLER           REDEFINES TCB-USER-DATA.
004800      10 TCB-EOJ-CALL      PIC XXXX.
004900      10 FILLER           REDEFINES TCB-EOJ-CALL.
005000      15 TCB-SOT-CALL      PIC XXXX.
005100      10 FILLER           PIC X(6).
005200      05 TCB-TIME          PIC S9(7) COMP-3.
005300      05 TCB-SYS-DATE      PIC S9(7) COMP-3.
005400      05 TCB-SOURCE         PIC X(8).
005500      05 TCB-OPERATOR        .
005600      10 TCB-UMBRELLA-OPERATOR PIC X(8).
005700      10 TCB-OPERATOR-FILLER PIC X(12).
005800      10 TCB-DXRF           REDEFINES TCB-OPERATOR-FILLER.
005900      15 TCB-DXRF-ID        PIC X(4).
006000      15 FILLER           PIC X(8).
006100      10 TCB-DYN-KEY-RANGE   REDEFINES TCB-OPERATOR-FILLER.
006200      15 TCB-DKR-ID         PIC X(4).
006300      15 FILLER           PIC X(8).
006400      10 TCB-ENQ            REDEFINES TCB-OPERATOR-FILLER.
006500      15 TCB-ENQ-ID         PIC X(8).
006600      15 FILLER           PIC X(4).
006700      05 TCB-DESTINATION     .
006800      05 TCB-TERM-DATA      .
006900      10 TCB-PFKEY          PIC X.
007000      88 TCB-ENTER          VALUE QUOTE.
007100      88 TCB-PF01          VALUE '1'.
007200      88 TCB-PF02          VALUE '2'.
007300      88 TCB-PF03          VALUE '3'.
007400      88 TCB-PF04          VALUE '4'.
007500      88 TCB-PF05          VALUE '5'.
007600      88 TCB-PF06          VALUE '6'.
007700      88 TCB-PF07          VALUE '7'.
007800      88 TCB-PF08          VALUE '8'.
007900      88 TCB-PF09          VALUE '9'.
008000      88 TCB-PF10          VALUE ':'.
008100      88 TCB-PF11          VALUE '#'.
008200      88 TCB-PF12          VALUE '@'.
008300      88 TCB-PF13          VALUE 'A'.
008400      88 TCB-PF14          VALUE 'B'.
008500      88 TCB-PF15          VALUE 'C'.
008600      88 TCB-PF16          VALUE 'D'.
008700      88 TCB-PF17          VALUE 'E'.
008800      88 TCB-PF18          VALUE 'F'.
008900      88 TCB-PF19          VALUE 'G'.
009000      88 TCB-PF20          VALUE 'H'.
009100      88 TCB-PF21          VALUE 'I'.
009200      88 TCB-PF22          VALUE '¢'.
009300      88 TCB-PF23          VALUE '.'.
009400      88 TCB-PF24          VALUE '<'.
009500      88 TCB-PFKEY-NOT-PRESENT VALUE LOW-VALUE.
009600      10 FILLER           PIC XX.
009700      05 TCB-GENP-LOG       PIC X(1).
009800      88 TCB-GENP-NO-LOGGING VALUE 'N'.
009900      88 TCB-GENP-LOGGING    VALUE 'Y' ''.
010000      10 FILLER           LOW-VALUE.
010100      05 TCB-EFFECTIVE-DATE PIC S9(7) COMP-3.

```



Umbrella Programming

Problem Specifications—Batch Program

```

010200      05 TCB-DEVICE-TYPE-2.
010300      10 TCB-DEVICE-TYPE          PIC X.
010400      88 TCB-3270-2             VALUE 'A'.
010500      88 TCB-BATCH              VALUE 'B'.
010600      88 TCB-3270-1              VALUE 'C'.
010700      88 TCB-3604-DS1            VALUE 'D'.
010800      88 TCB-3604-DS3            VALUE 'E'.
010900      88 TCB-3604-DS4            VALUE 'F'.
011000      88 TCB-3600-JP             VALUE 'G'.
011100      88 TCB-3600-PB             VALUE 'H'.
011200      88 TCB-3600-LP             VALUE 'I'.
011300      88 TCB-3270-MOD1-PRINTER  VALUE 'J'.
011400      88 TCB-3270-MOD2-PRINTER  VALUE 'K'.
011500      88 TCB-TWX                VALUE 'L'.
011600      88 TCB-2470-MOD2            VALUE 'M'.
011700      88 TCB-2740-MOD1            VALUE 'N'.
011800      88 TCB-ALIEN-X             VALUE 'X'.
011900      88 TCB-ALIEN-Y             VALUE 'Y'.
012000      88 TCB-ALIEN-Z             VALUE 'Z'.
012100*
012200***** RESERVED FOR TCB-DEVICE EXPANSION TO PIC X(2)
012300      10 FILLER                PIC X.
012400      05 TCB-LONG-ACTIVITY-N    PIC S9(09) COMP.
012500      05 TCB-LONG-ACTIVITY      REDEFINES TCB-LONG-ACTIVITY-N.
012600      10 TCB-LONG-ACT-HI        PIC XX.
012700      10 TCB-LONG-ACT-LO        PIC XX.
012800      05 TCB-LONG-DGID-N       PIC S9(09) COMP.
012900      05 TCB-LONG-DGID          REDEFINES TCB-LONG-DGID-N.
013000      10 TCB-LONG-DG-HI         PIC XX.
013100      10 TCB-LONG-DG-LO         PIC XX.
013200      05 TCB-USER-CC-APP       PIC XX.
013300      05 TCB-RESULT-2          PIC XX.
013400      EJECT
013500*****
013600* CDMF APPLICATION CONTROL BLOCK
013700*****
013800      05 CDMF-CONTROL-BLOCK.
013900      10 CDMF-ACTION           PIC XX.
014000      10 CDMF-RESULT           PIC XX.
014100      10 CDMF-KEY-FIELDS.
014200      15 CDMF-FORMAT          PIC XXXX.
014300      15 CDMF-COID             PIC XX.
014400      15 CDMF-EFF-DATE         PIC S9(7)  COMP-3.
014500      10 CDMF-EXP-DATE         PIC S9(7)  COMP-3.
014600      10 CDMF-COID-FOUND       PIC XX.
014700      88 CDMF-DEFAULT-COID-FOUND  VALUE HIGH-VALUES.
014800      10 CDMF-EFF-DATE-FOUND    PIC S9(7)  COMP-3.
014900      10 CDMF-HIGH-USE-FLAG     PIC X.
015000      88 CDMF-HIGH-USE-ITEM    VALUE 'Y'.
015100      88 CDMF-NON-PURGEABLE    VALUE 'P'.
015200*
015300***** ITEM OWNERSHIP IS ALWAYS RETURNED IN CDMF-OWNER-APPLICATION
015400***** OWNERSHIP MAY BE RETRIEVED AND UPDATED FROM DATA GROUP 480
015500***** IF THIS FLAG IS SET TO A 'Y'. OWNERSHIP MAY BE UPDATED FR
015600***** CDMF-OWNER-APPLICATION IF THIS FLAG IS SET TO A 'C'.
015700      10 CDMF-OWNER-APP-FLAG    PIC X.
015800      88 CDMF-OWNER-APP-REQUEST  VALUE 'Y'.
015900      88 CDMF-OWNER-APP-IN-CTL-BLK  VALUE 'C'.
016000      10 CDMF-ITEM-LOCATION      PIC X.
016100      88 CDMF-ITEM-FOUND-IN-TABLE  VALUE 'Y'.
016200      10 FILLER                PIC X.
016300      10 CDMF-CC-NO             PIC X(4).
016400      10 CDMF-LAST-CHANGE-DATA.
016500      15 CDMF-LAST-CHANGE-DATE   PIC S9(7)  COMP-3.
016600      15 CDMF-LAST-CHANGE-TIME   PIC S9(7)  COMP-3.

```



Umbrella Programming

Problem Specifications—Batch Program

```
016700      15 CDMF-LAST-CHANGE-CC-NO PIC X(4).
016800      15 CDMF-LAST-CHANGE-SOURCE PIC X(8).
016900      15 CDMF-LAST-CHANGE-OPER  PIC X(8).
017000      10 CDMF-SECONDARY-KEY-ID  PIC X(4).
017100      10 CDMF-SUBSTITUTE-DGID  PIC X(4).
017200*
017300***** THIS FIELD IS FOR INTERNAL UMBRELLA USE ONLY.
017400      10 CDMF-RELEASE-CTL-DG-LEN  PIC XX.
017500*
017600***** THIS FIELD IS FOR INTERNAL UMBRELLA USE ONLY.
017700      10 CDMF-RELEASE-CTL-FLAGS  PIC X.
017800*
017900***** THE ITEM APPLICATION OWNERSHIP IS ALWAYS RETURNED IN THIS
018000***** FIELD.  THIS FIELD MAY ONLY BE USED IN AN UPDATE WHEN
018100***** CDMF-OWNER-APP-FLAG IS SET TO A 'C'.
018200      10 CDMF-OWNER-APPLICATION  PIC X(3).
018300      10 FILLER                PIC X(2).
018400***** END OF DATA GROUP 00001 ****
018500      EJECT
018600*****
018700* DATA GROUP 12 (SYSIN INPUT DATA GROUP)
018800*-----
018900* NOTE: PROGRAMS THAT REFERENCE DATA GROUP 12 NEED NOT CODE THE
019000* DATA GROUP ON THE PROGRAM DEFINITION.  INSTEAD, YOU MAY REFER
019100* TO THIS AREA DIRECTLY & ISSUE ACTIVITIES THAT REFERENCE DG 12.
019200*****
019300*
019400***** DATA GROUP CHAIN (DO NOT DESTROY)
019500      05 FILLER                PIC X(8).
019600      05 DATA-GROUP-12.
019700      10 DG12-ACTION          PIC XX.
019800      10 DG12-RESULT           PIC XX.
019900      10 DG12-CARD-IMAGE      PIC X(80).
020000***** END OF DATA GROUP 00012 ****
020100      SKIP1
020200*****
020300* DATA GROUP 13 (SYSPRINT OUTPUT DATA GROUP)
020400*-----
020500* NOTE: PROGRAMS THAT REFERENCE DATA GROUP 13 NEED NOT CODE THE
020600* DATA GROUP ON THE PROGRAM DEFINITION.  INSTEAD, YOU MAY REFER
020700* TO THIS AREA DIRECTLY & ISSUE ACTIVITIES THAT REFERENCE DG 13.
020800*****
020900*
021000***** DATA GROUP CHAIN (DO NOT DESTROY)
021100      05 FILLER                PIC X(8).
021200      05 DATA-GROUP-13.
021300      10 DG13-ACTION          PIC XX.
021400      10 DG13-RESULT           PIC XX.
021500      10 DG13-CONTROL-CHAR    PIC X.
021600      10 DG13-PRINT-DATA     PIC X(132).
021700      10 FILLER                PIC X(7).
021800***** END OF DATA GROUP 00013 ****
021900      SKIP1
022000*****
022100* DATA GROUP 10 (TCB EXTENSION AREAS)
022200*-----
022300* THIS DATA GROUP CONTAINS THE SECURITY CONTROL BLOCK, THE TCB
022400* USER AREA (FOR CLIENT USE), THE USER TCB EXTENSION AREA, AND
022500* THE INTERNAL TCB EXTENSION AREA.
022600* ***** NOTE: DO NOT CODE THIS DATA GROUP ON YOUR PROGRAM
022700* ***** DEFINITION.  INSTEAD, YOU SHOULD REFER TO THIS
022800* ***** AREA DIRECTLY SINCE IT IS PART OF THE USER
022900* ***** TRANSACTION CONTROL BLOCK.
023000*****
023100*
```



Umbrella Programming

Problem Specifications—Batch Program

```

023200***** DATA GROUP CHAIN (DO NOT DESTROY)
023300      05 FILLER                               PIC X(8).
023400      05 DATA-GROUP-10.
023500      EJECT
023600*****
023700* SECURITY CONTROL BLOCK
023800*****
023900      10 SECURITY-CONTROL-BLOCK.
024000      15 SCB-ACTION          PIC S9(4)  COMP.
024100      15 SCB-ACTION-X REDEFINES SCB-ACTION
024200                           PIC XX.
024300      15 SCB-RESULT          PIC S9(4)  COMP.
024400      88 SCB-AUTHORIZATION-VALID VALUE +0.
024500      88 SCB-AUTHORIZATION-FAILED VALUE +4.
024600      88 SCB-AUTHORIZATION-ERROR VALUE +8.
024700      88 SCB-EXT-SECURITY-INACTIVE VALUE +12.
024800      15 SCB-RESULT-X REDEFINES SCB-RESULT
024900                           PIC XX.
025000      15 SCB-VIOLATION-ACTION PIC S9(4)  COMP.
025100      88 SCB-ABEND-TASK    VALUE +0.
025200      88 SCB-RETURN        VALUE +1.
025300      15 SCB-LOGGING-FLAG  PIC X.
025400      88 SCB-LOG-EXCPTNS   VALUE 'Y'.
025500      88 SCB-BYPASS-LOG   VALUE 'N'.
025600      15 SCB-PROCESSING-TYPE PIC X.
025700      15 SCB-PEM-FLAG1     PIC X.
025800      15 SCB-FUTURE-FLAGS  PIC X(3).
025900      15 SCB-FORMAT-NUMBER PIC XXXX.
026000      15 SCB-FORMAT-NAME  PIC X(10).
026100*
026200***** THIS FIELD IS FOR ALC PROGRAMS ONLY
026300      15 SCB-ADDR-FMT-TARGET-DG  PIC XXXX.
026400*
026500      15 SCB-TARGET-DG-ID    PIC XXXX.
026600      15 SCB-ITEM-OWNER    PIC X(3).
026700      15 SCB-PREV-OWNER    PIC X(3).
026800      15 SCB-MESSAGE-NO.
026900      20 SCB-MESSAGE-NO-N  PIC S9(4)  COMP.
027000      15 SCB-EXCEPTION-MESSAGE.
027100      20 SCB-RULE-NAME    PIC X(40).
027200      20 FILLER          PIC X(4).
027300      15 SCB-USER-DATA    PIC X(25).
027400      15 SCB-RESERVED     PIC X(8).
027500      15 FILLER          PIC X(1).
027600      EJECT
027700*****
027800* TCB USER AREA
027900*-----
028000* THIS AREA IS RESERVED FOR CLIENTS AND WILL NEVER BE USED BY
028100* HOGAN SYSTEMS.
028200*****
028300      10 TCB-USER-AREA.
028400      15 FILLER          PIC X(104).
028500      SKIP1
028600*****
028700* USER TCB EXTENSION AREA
028800*-----
028900* THIS AREA IS RESERVED FOR NEW TCB FIELDS TO BE ADDED AND
029000* UPDATED BY HOGAN SYSTEMS.
029100*****
029200      10 TCB-EXTENSION-AREA.
029300*          CURSOR POSITION AFTER A DEBLOCK; ROW AND COLUMN
029400      15 TCB-ROW           PIC S9(4)  COMP.
029500      15 TCB-COLUMN         PIC S9(4)  COMP.
029600      15 TCB-BATCH-DISP-OPTION PIC X.

```



Umbrella Programming

Problem Specifications—Batch Program

```

029700          88 TCB-BATCH-DISP-DUMP      VALUE LOW-VALUES.
029800*          88 BATCH DISPLAY TO SYSPRINT IN DUMP FORMAT.
029900          88 TCB-BATCH-DISP-FORMAT  VALUE 'F'.
030000*          88 BATCH DISPLAY TO SYSPRINT IN SCREEN FORMAT
030100          88 TCB-BATCH-DISP-RETURN  VALUE 'R'.
030200*          88 BATCH DISPLAY DATA IN DG 47. NOT PRINTED.
030300          15 FILLER             PIC X.
030400          15 TCB-DYN-TXN-ID    PIC X(008).
030500          15 TCB-OPTIONS-1   PIC X.
030600          15 TCB-OPTIONS-2   PIC X.
030700          15 FILLER             PIC X(040).
030800          15 TCB-SQL-ACTION.
030900          20 TCB-SQL-ACTION-N  PIC S9(4) COMP.
031000          15 TCB-SQL-RESULT.
031100          20 TCB-SQL-RESULT-N  PIC S9(4) COMP.
031200          15 TCB-SQL-DYNPLAN   PIC X(008).
031300          15 TCB-SQL-CURPLAN   PIC X(008).
031400          15 TCB-SQL-DYN-SUBSID  PIC X(004).
031500          15 TCB-SQL-SUBSID   PIC X(004).
031600          15 TCB-CKPT-COUNT   PIC S9(9) COMP.
031700          15 FILLER             PIC X(005).
031800          15 TCB-APPC-SERVICE-AREAS.
031900          20 TCB-APPC-DATA-GROUP  PIC X(4).
032000          20 TCB-APPC-SYSTEM-KEY  PIC X(4).
032100          20 TCB-APPC-APPL-KEY   PIC X(8).
032200          20 TCB-APPC-XMIT-IMMED  PIC X(1).
032300          88 TRANSMIT           VALUE 'Y'.
032400          88 TRAN-PREPARE-RECEIVE  VALUE 'R'.
032500          20 TCB-APPC-XMIT-ERROR  PIC X(1).
032600*          **88 ISSUE-ERROR     VALUE +1.
032700*          **88 ISSUE-ERROR-W-DATA  VALUE +17.
032800*          **88 ISSUE-ABEND     VALUE +2.
032900          20 TCB-APPC-MORE-DATA  PIC X(1).
033000          20 TCB-APPC-BYPSS-ERR  PIC X(1).
033100          20 TCB-APPC-RETURN-CDE  PIC X(6).
033200          15 FILLER             PIC X.
033300          15 TCB-VSAM-RELATIVE-NUMB  PIC X(4).
033400          15 TCB-MONETARY-KEY   PIC X(3).
033401          15 TCB-PRES-CURRENCY-CD  PIC X(3).
033500          15 TCB-LANGUAGE-KEY  PIC X(3).
033600          15 TCB-PACK-COLLECT-NAME  PIC X(18).
033700          15 TCB-DYN-COLLECT-NAME.
033800          20 TCB-DYN-PLAN-PREF   PIC X(2).
033900          20 TCB-DYN-COMP-GRP   PIC X(8).
034000          20 TCB-DYN-PROC-GRP   PIC X(4).
034100          20 TCB-DYN-KEY-RANGE  PIC X(4).
034200          15 FILLER             PIC X(8).
034300          15 TCB-DB2-KRID      PIC X(4).
034400          15 TCB-LANG-ENABLED   PIC X(1).
034500          15 TCB-DEFAULT-LANG  PIC X(3).
034600          15 TCB-LANG-ENCODE-FLAG  PIC X(1).
034700          15 TCB-PROC-GROUP    PIC X(4).
034800          15 TCB-PROC-GROUP-BRANCH  PIC 9(5) COMP-3.
034900          15 TCB-PROCESSING-ID   REDEFINES
035000          TCB-PROC-GROUP-BRANCH  PIC 9(5) COMP-3.
035100          15 TCB-UDFL-LANG     PIC X(3).
035200          15 TCB-SPS-IMplode-LANG  PIC X(3).
035300          15 TCB-OPERATOR-REGION  PIC 9(5) COMP-3.
035400          15 TCB-OPERATOR-BRANCH  PIC 9(5) COMP-3.
035500          15 TCB-DYN-LANG      PIC X(3).
035600          15 TCB-DMAP-ID       PIC X(7).
035700          15 TCB-DB2-PROC-GRP-ID  PIC X(4).
035800          15 TCB-DB2-PROC-GRP-BRANCH  PIC 9(5) COMP-3.
035900          15 TCB-DB2-PROCESSING-ID  REDEFINES
036000          TCB-DB2-PROC-GRP-BRANCH  PIC 9(5) COMP-3.

```



Umbrella Programming

Problem Specifications—Batch Program

```

036100      15 TCB-USER-WORK-AREA.
036200          20 TCB-DYNAMIC-DG-ADDR POINTER.
036201          20 TCB-DYNAMIC-DG-ADDRESS REDEFINES
036202              TCB-DYNAMIC-DG-ADDR POINTER.
036300          20 FILLER             PIC X(8).
036400      15 TCB-HDP-RESERVED     PIC X(10).
036500      15 TCB-DFLT-LANG-ENCODE-BYTE
036600                      PIC X(01).
036700      15 TCB-DB2-TEST-POOL-ID    PIC X(02).
036800      15 TCB-PRES-CURR-RND-IN    PIC X(01).
036801* DO NOT ROUND PRESENTATION CURRENCY - ADDS AN EXTRA DECIMAL DIG
036802          88 TCB-DO-NOT-ROUND-PRES-CURR           VALUE 'N
036803* ROUND PRESENTATION CURRENCY
036804          88 TCB-ROUND-PRES-CURR           VALUE 'Y
036900      15 TCB-RAND-REC-ADDR-NR    PIC S9(9) COMP.
037000      15 TCB-JOB-ID          PIC X(8).
037100      15 TCB-OPERATOR-PROCESSING-ID PIC 9(5) COMP-3.
037200      15 TCB-DFLT-CENTURYWINDOW-CUTOFF PIC S999 COMP-3.
037300      15 TCB-CICS-STARTCODE    PIC XX.
037301          88 TCB-CICS-START-DPL           VALUE 'D '
037302          88 TCB-CICS-START-DPL-SYNC        VALUE 'DS'
037303          88 TCB-CICS-START-TD-TRIGGER      VALUE 'QD'
037304          88 TCB-CICS-START-CMD           VALUE 'S '
037305          88 TCB-CICS-START-CMD-DATA        VALUE 'SD'
037306          88 TCB-CICS-START-FEPI          VALUE 'SZ'
037307          88 TCB-CICS-START-TERMINAL-INPUT    VALUE 'TD'
037308          88 TCB-CICS-START-USER-ATTACH      VALUE 'U '
037309      15 FILLER             PIC X(661).
037400      15 FILLER             PIC X(200).

037500      SKIP1
037600***** END OF DATA GROUP 00010 *****
037700      SKIP1
037800*
037900***** END OF P49000D *****
          EJECT
*-----
*   THE FOLLOWING COPYBOOK Z47100D DEFINES DATA GROUP 47100
*   BASE DATA GROUP EMP DATA BASE
*   EMPLOYEE INFORMATION
*   EMP-KEY-GROUP
*-----
000010***** START OF Z47100D ***** EMP KEY *****
000020*
000030*   COPYBOOK Z47100D DEFINES DATA GROUP 47100, WHICH IS A KEY
000040*   DATA GROUP IN AN EMPLOYEE INFORMATION RECORD ON THE "EMP"
000050*   DATA BASE FOR EDUCATION CLASS USE.
000060*
000070***** *****
000080*
000090*   DATA GROUP NUMBER    47100
000100*
000110***** *****
000120 01   EMP-KEY-GROUP.
000130      10  EMP-ACTION          PIC  XX.
000140      10  EMP-RESULT          PIC  XX.
000150      05  EMP-KEY-GROUP-MOVE.
000160      10  EMP-CO-ID           PIC  XX.
000170      10  EMP-KEY-ID          PIC  9(11) COMP-3.
000180      10  EMP-FILLER          PIC  X(38).
000190***** END OF Z47100D *****
          EJECT
*-----
*   THE FOLLOWING COPYBOOK Z47110D DEFINES DATA GROUP 47110
*   POSITIONAL DATA GROUP EMP DATA BASE
*   EMPLOYEE GENERAL INFORMATION

```



Umbrella Programming

Problem Specifications—Batch Program

```
*     EMP-INFO-GROUP
*-----
000010***** START OF Z47110D ***** EMP INFORMATION *****
000020*
000030*   COPYBOOK Z47110D DEFINES DATA GROUP 47110, WHICH IS A
000040*   POSITIONAL DATA GROUP USED FOR "EMP" DATA BASE RECORDS
000050*   FOR EDUCATION CLASSES.
000060*
000070*****
000080*
000090*       DATA GROUP NUMBER    47110
000100*
000110*****
000120 01  EMP-INFO-GROUP.
000130      10  EMP-INFO-ACTION          PIC XX.
000140      10  EMP-INFO-RESULT         PIC XX.
000150      05  EMP-INFO-GROUP-MOVE.
000160      10  EMP-L-NAME            PIC X(15).
000170      10  EMP-F-NAME            PIC X(15).
000180      10  EMP-ADDRESS           PIC X(30).
000190      10  EMP-CITY              PIC X(20).
000200      10  EMP-STATE             PIC X(2).
000210      10  EMP-ZIP               PIC 9(9)  COMP-3.
000220      10  EMP-RESERVED          PIC X(5).
000230      10  EMP-EOC-CODE         PIC XX.
000240      10  EMP-SEX               PIC X.
000250      10  EMP-BIRTHDATE         PIC 9(7)  COMP-3.
000260      10  EMP-EXEMP             PIC 999   COMP-3.
000270      10  EMP-INSUR-EXEMP       PIC 999   COMP-3.
000280      10  EMP-PHONE             PIC 9(11) COMP-3.
000290      10  EMP-INFO-FILLER        PIC X(87).
000300***** END OF Z47110D *****
EJECT
*-----
*   THE FOLLOWING COPYBOOK Z47120 DEFINES DATA GROUP 47120
*   POSITIONAL DATA GROUP EMP DATA BASE
*   EMPLOYEE JOB STATUS INFORMATION
*   EMP-JOB-STATUS
*-----
000010***** START OF Z47120D ***** EMP JOB STATUS *****
000020*
000030*   COPYBOOK Z47120 DEFINES DATA GROUP 47120, WHICH IS A
000040*   POSITIONAL DATA GROUP USED FOR "EMP" DATA BASE RECORDS
000050*   FOR EDUCATION CLASSES.
000060*
000070*****
000080*
000090*       DATA GROUP NUMBER    47120
000100*
000110*****
000120 01  EMP-JOB-STATUS.
000130      10  JOB-STAT-ACTION        PIC XX.
000140      10  JOB-STAT-RESULT        PIC XX.
000150      05  EMP-JOB-STATUS-MOVE.
000160      10  JOB-STAT-DATE-HIRED    PIC 9(7)  COMP-3.
000170      10  JOB-STAT-DATE-TERMD    PIC 9(7)  COMP-3.
000180      10  JOB-STAT-MGR           PIC X(30).
000190      10  JOB-STAT-DEPT          PIC XXX.
000200      10  JOB-STAT-COST-CENTER    PIC XXX.
000210      10  JOB-STAT-CLASS-POS.
000211      15  JOB-STAT-CLASS          PIC XX.
000212      15  JOB-STAT-POSITION        PIC XX.
000230      10  JOB-STAT-FILLER         PIC X(98).
000240***** END OF Z47120D *****
EJECT
```



Umbrella Programming

Problem Specifications—Batch Program

```

*-----
* THE FOLLOWING COPYBOOK Z47130 DEFINES DATA GROUP 47130
* POSITIONAL DATA GROUP EMP DATA BASE
* EMPLOYEE CURRENT PAY INFORMATION
* EMP-CURRENT-PAY
*-----

000010***** START OF Z47130D ***** EMP CURRENT PAY *****
000020*
000030*      COPYBOOK Z47130D DEFINES DATA GROUP 47130, WHICH IS A
000040*      POSITIONAL DATA GROUP USED FOR "EMP" DATA BASE RECORDS
000050*      FOR EDUCATION CLASSES.
000060*
000070*****
000080*
000090*      DATA GROUP NUMBER    47130
000100*
000110*****
000120 01  EMP-CURRENT-PAY.
000130      10  EMP-C-ACTION          PIC XX.
000140      10  EMP-C-RESULT          PIC XX.
000150      05  EMP-CURRENT-PAY-MOVE.
000160      10  EMP-C-BEG-DATE        PIC 9(7)     COMP-3.
000170      10  EMP-C-END-DATE        PIC 9(7)     COMP-3.
000180      10  EMP-C-EARN-TOT       PIC S9(7)V99  COMP-3.
000190      10  EMP-C-FED-INC        PIC S9(7)V99  COMP-3.
000200      10  EMP-C-FICA           PIC S9(7)V99  COMP-3.
000210      10  EMP-C-OTHER-TAX       PIC S9(7)V99  COMP-3.
000220      10  EMP-C-DED-ADJ-TOT     PIC S9(7)V99  COMP-3.
000230      10  EMP-C-NET-PAY        PIC S9(7)V99  COMP-3.
000240      10  EMP-C-DED-ADJ OCCURS 10 TIMES.
000250          15  EMP-C-DED-ADJ-CODE   PIC XX.
000260          15  EMP-C-DED-ADJ-AMT   PIC S9(7)V99  COMP-3.
000270      10  EMP-C-EARNINGS OCCURS 5 TIMES.
000280          15  EMP-C-TYPE-CODE    PIC XX.
000290          15  EMP-C-UNITS         PIC S9(5)    COMP-3.
000300          15  EMP-C-AMTS          PIC S9(7)V99  COMP-3.
000310      10  EMP-C-FILLER         PIC X(138).
000320***** END OF Z47130D *****
EJECT
*-----
* THE FOLLOWING COPYBOOK Z47140 DEFINES DATA GROUP 47140
* POSITIONAL DATA GROUP EMP DATA BASE
* EMPLOYEE YEAR TO DATE PAY INFORMATION
* EMP-YEAR-TO-DATE-PAY
*-----

000010***** START OF Z47140D ***** EMP YEAR-TO-DATE PAY *****
000020*
000030*      COPYBOOK Z47140D DEFINES DATA GROUP 47140, WHICH IS A
000040*      POSITIONAL DATA GROUP USED FOR "EMP" DATA BASE RECORDS
000050*      FOR EDUCATION CLASSES.
000060*
000070*****
000080*
000090*      DATA GROUP NUMBER    47140
000100*
000110*****
000120 01  EMP-YEAR-TO-DATE-PAY.
000130      10  EMP-Y-ACTION          PIC XX.
000140      10  EMP-Y-RESULT          PIC XX.
000150      05  EMP-YEAR-TO-DATE-PAY-MOVE.
000160      10  EMP-Y-BEG-DATE        PIC 9(7)     COMP-3.
000170      10  EMP-Y-END-DATE        PIC 9(7)     COMP-3.
000180      10  EMP-Y-EARN-TOT       PIC S9(7)V99  COMP-3.
000190      10  EMP-Y-FED-INC        PIC S9(7)V99  COMP-3.
000200      10  EMP-Y-FICA           PIC S9(7)V99  COMP-3.

```



Umbrella Programming

Problem Specifications—Batch Program

```
000210      10 EMP-Y-OTHER-TAX          PIC S9(7)V99 COMP-3.
000220      10 EMP-Y-DED-ADJ-TOT        PIC S9(7)V99 COMP-3.
000230      10 EMP-Y-NET-PAY          PIC S9(7)V99 COMP-3.
000240      10 EMP-Y-DED-ADJ OCCURS 10 TIMES.
000250          15 EMP-Y-DED-ADJ-CODE    PIC XX.
000260          15 EMP-Y-DED-ADJ-AMT    PIC S9(7)V99 COMP-3.
000270      10 EMP-Y-EARNINGS OCCURS 5 TIMES.
000280          15 EMP-Y-TYPE-CODE     PIC XX.
000290          15 EMP-Y-UNITS         PIC 9(5)   COMP-3.
000300          15 EMP-Y-AMTS          PIC S9(7)V99 COMP-3.
000310      10 EMP-Y-FILLER          PIC X(138).
000320***** END OF Z47140D ****
EJECT
*-----
* THE FOLLOWING COPYBOOK T58001D DEFINES DATA GROUP 2000
* DATE CONTROL BLOCK.
*
* THE DATE CONTROL BLOCK IS USED FOR DATE SERVICES
* COMMUNICATION. ACTIVITY ID 1900
*
* COPYBOOK T58007D CONTAINS DCB ACTION FIELDS
* COPYBOOK T58008D CONTAINS DCB RESULT FIELDS
*-----
000100*** START OF T58001D *** DATE CONTROL BLOCK ***
000200*
000300* DATA GROUP - 2000
000400* LINK ACTIVITY - 1900
000500*
000600*****
000700*
000800 01 DATE-CONTROL-BLOCK.
000900 05 DCB-ACTION          PIC XX.
001000 05 DCB-RESULT          PIC XX.
001100 05 DCB-DATE-IN         PIC S9(7) COMP-3.
001200 05 DCB-DATE-OUT        PIC S9(7) COMP-3.
001300 05 DCB-DIFF-FACTOR     PIC X(1).
001400          88 DCB-DIFF-DAYS       VALUE 'D'.
001500          88 DCB-DIFF-MONTHS     VALUE 'M'.
001600          88 DCB-DIFF-YEARS     VALUE 'Y'.
001700 05 DCB-DIFF-AMT         PIC S9(7) COMP-3.
001800 05 DCB-EDIT-CHAR        PIC X(1).
001900 05 DCB-RAW-DATE-AREA    PIC X(25).
002000*
002100 05 DCB-DISPLAY         REDEFINES DCB-RAW-DATE-AREA.
002200 10 DCB-NUM-WC.
002300          15 DCB-NUM-C          PIC X.
002400          15 DCB-NUM-WOC         .
002500          20 FILLER          PIC X(8).
002600          15 FILLER          PIC X(16).
002700*
002800 05 DCB-HOLIDAYS OCCURS 15 TIMES.
002900 10 DCB-HOLIDAY          PIC S9(7) COMP-3.
003000 05 DCB-DAY-OF-WEEK      PIC 9(1).
003100          88 DCB-DOW-SUN        VALUE 1.
003200          88 DCB-DOW-MON        VALUE 2.
003300          88 DCB-DOW-TUE        VALUE 3.
003400          88 DCB-DOW-WED        VALUE 4.
003500          88 DCB-DOW-THR        VALUE 5.
003600          88 DCB-DOW-FRI        VALUE 6.
003700          88 DCB-DOW-SAT        VALUE 7.
003800 05 DCB-DATE-STATUS.
003900 10 DCB-FOM              PIC X(1).
004000          88 DCB-FOM-BUS        VALUE 'B' 'T'.
004100          88 DCB-FOM-CLNDR      VALUE 'P' 'T'.
004200 10 DCB-FOQ              PIC X(1).
```



Umbrella Programming

Problem Specifications—Batch Program

```

004300          88 DCB-FOQ-BUS           VALUE 'B' 'T'.
004400          88 DCB-FOQ-CLNDR        VALUE 'P' 'T'.
004500      10 DCB-FOY             PIC X(1).
004600          88 DCB-FOY-BUS         VALUE 'B' 'T'.
004700          88 DCB-FOY-CLNDR       VALUE 'P' 'T'.
004800      10 DCB-EOM             PIC X(1).
004900          88 DCB-EOM-BUS         VALUE 'B' 'T'.
005000          88 DCB-EOM-CLNDR       VALUE 'P' 'T'.
005100      10 DCB-EOQ             PIC X(1).
005200          88 DCB-EOQ-BUS         VALUE 'B' 'T'.
005300          88 DCB-EOQ-CLNDR       VALUE 'P' 'T'.
005400      10 DCB-EOY             PIC X(1).
005500          88 DCB-EOY-BUS         VALUE 'B' 'T'.
005600          88 DCB-EOY-CLNDR       VALUE 'P' 'T'.
005700      10 DCB-BUS-DAY          PIC X(1).
005800          88 DCB-BDAY            VALUE 'B'.
005900      10 DCB-WEEK-DAY          PIC X(1).
006000          88 DCB-WDAY            VALUE 'W'.
006100      10 DCB-LEAP-YEAR         PIC X(1).
006200          88 DCB-LYEAR            VALUE 'L'.
006300      10 DCB-FOW              PIC X.
006400          88 DCB-FOW-BUS          VALUE 'B' 'T'.
006500          88 DCB-FOW-CLNDR        VALUE 'P' 'T'.
006600      10 DCB-EOW              PIC X.
006700          88 DCB-EOW-BUS          VALUE 'B' 'T'.
006800          88 DCB-EOW-CLNDR        VALUE 'P' 'T'.
006900      05 DCB-CURR-BUSINESS-DAY   PIC 99.
007000      05 DCB-FISCAL-STATUS.
007100      10 DCB-FISCAL-FOQ          PIC X(1).
007200          88 DCB-FISCAL-FOQ-BUS    VALUE 'B' 'T'.
007300          88 DCB-FISCAL-FOQ-CLNDR  VALUE 'P' 'T'.
007400      10 DCB-FISCAL-FOY          PIC X(1).
007500          88 DCB-FISCAL-FOY-BUS    VALUE 'B' 'T'.
007600          88 DCB-FISCAL-FOY-CLNDR  VALUE 'P' 'T'.
007700      10 DCB-FISCAL-EOQ          PIC X(1).
007800          88 DCB-FISCAL-EOQ-BUS    VALUE 'B' 'T'.
007900          88 DCB-FISCAL-EOQ-CLNDR  VALUE 'P' 'T'.
008000      10 DCB-FISCAL-EOY          PIC X(1).
008100          88 DCB-FISCAL-EOY-BUS    VALUE 'B' 'T'.
008200          88 DCB-FISCAL-EOY-CLNDR  VALUE 'P' 'T'.
008300*
008400      05 DCB-SQL-FIELDS.
008500      10 DCB-SQL-DATE-IN        PIC X(10).
008600      10 DCB-SQL-DATE-OUT       PIC X(10).
008700      10 DCB-SQL-TIME          PIC X(08).
008800      10 DCB-PACK-TIME         PIC S9(07)  COMP-3.
008900      10 DCB-SQL-OVRD-DATE      PIC X(03).
009000          88 DCB-SQL-EUR-DATE     VALUE 'EUR'.
009100          88 DCB-SQL-ISO-DATE     VALUE 'ISO'.
009200          88 DCB-SQL-JIS-DATE     VALUE 'JIS'.
009300          88 DCB-SQL-USA-DATE     VALUE 'USA'.
009400      10 DCB-SQL-OVRD-TIME      PIC X(03).
009500          88 DCB-SQL-EUR-TIME     VALUE 'EUR'.
009600          88 DCB-SQL-ISO-TIME     VALUE 'ISO'.
009700          88 DCB-SQL-JIS-TIME     VALUE 'JIS'.
009800          88 DCB-SQL-USA-TIME     VALUE 'USA'.
009900*
010000      05 DCB-BRANCH-ID-X.
010100      10 DCB-BRANCH-ID          PIC 9(05)  COMP-3.
010200*
010300 01 FILLER REDEFINES DATE-CONTROL-BLOCK.
010400      05 DCB-DATA             PIC X(17).
010500*
010600*----* END OF T58001D *-----*
EJECT

```



Umbrella Programming

Problem Specifications—Batch Program

```
*-----  
* THE FOLLOWING COPYBOOK I57101D DEFINES DATA GROUP 1452  
* PCD CONTROL BLOCK  
*  
* THE PCD CONTROL BLOCK IS USED FOR PCD PROCESSING  
* ACTIVITIES 1013 AND 1014  
*  
* COPYBOOK I57104D CONTAINS PCD ACTION FIELDS  
* COPYBOOK I57105D CONTAINS PCD RESULT FIELDS  
*-----  
000100*****  
000200*           I57101D - PCD CONTROL BLOCK  
000300*****  
000400*  
000500*           PEM - DG #1452, ALLOCATE ACTIVITY #1530  
000600*****  
000700 01 PCD-CONTROL-BLOCK.  
000800    05 PCD-ACTION          PIC XX.  
000900    05 PCD-RESULT          PIC XX.  
001000    05 PCD-INPUT-SECTION.  
001100    10 PCD-COID            PIC XX.  
001200    10 PCD-ID              PIC XX.  
001300    10 PCD-USER-KEY        PIC X(11).  
001400    10 PCD-EFFECT-DATE    PIC S9(7) COMP-3.  
001500    10 PCD-EXPIRE-DATE    PIC S9(7) COMP-3.  
001600    10 FILLER              PIC X.  
001700    05 PCD-OUTPUT-SECTION.  
001800    10 PCD-OUT-COID        PIC XX.  
001900    10 PCD-OUT-ID          PIC XX.  
002000    10 PCD-OUT-USER-KEY    PIC X(11).  
002100    10 PCD-OUT-EFFECT-DATE PIC S9(7) COMP-3.  
002200    10 PCD-OUT-EXPIRE-DATE PIC S9(7) COMP-3.  
002300    10 FILLER              PIC X.  
002400    05 PCD-EFFECT-PERIOD.  
002500    10 PCD-FIRST-EFFECT    PIC S9(7) COMP-3.  
002600    10 PCD-LAST-EFFECT    PIC S9(7) COMP-3.  
002700    05 PCD-SWTCH          PIC XX.  
002800    05 PCD-AMSG            PIC X(8).  
002900    05 PCD-RMSG            PIC X(17).  
003000    05 PCD-CCNO            PIC S9(8)  COMP.  
003100    05 FILLER              PIC X.  
003200    05 PCD-COID-USED      PIC XX.  
003300    05 PCD-USER-FIELD-1   PIC X(40).  
003400    05 PCD-OWNER           PIC X(03).  
003500    05 FILLER              PIC X(01).  
003600*  
003700*****  
EJECT  
*-----  
* THE FOLLOWING COPYBOOK Z47190D DEFINES DATA GROUP 47190  
* TARGET DATA GROUP FOR PCD 47190  
* THIS IS A CONVERTED PCD  
* CONTAINS POSITION CODE DESCRIPTIONS WITHIN JOB CLASS.  
*-----  
*****  
* START OF Z47190D - POSITION CODE DESCRIPTIONS  
* DATA GROUP NUMBER 47190  
* USER KEY = JOB CLASS (TQE00000 FROM DATA GROUP 47120)  
* RECURSIVE KEY = POSITION CODE  
*****  
01 EMP-POSITION-CODE-DESCRIPTION.  
    05 EMP-POS-ACT-TQE00000  PIC XX.  
    05 EMP-POS-RES-TQE00000  PIC XX.  
    05 EMP-POS-KEY-TQE00000  PIC XX.  
    05 EMP-POSITION-CODE     REDEFINES EMP-POS-KEY-TQE00000
```



Umbrella Programming

Problem Specifications—Batch Program

```

          PIC XX.
      05 EMP-POS-DESCRIPTION      PIC X(40).
*
*   END OF Z47190D
    EJECT
*-----
*   THE FOLLOWING COPYBOOK Z47192D DEFINES DATA GROUP 47192
*   TARGET DATA GROUP FOR PCD 47192
*   THIS IS A CDMF/PCD
*   CONTAINS POSITION CODE DESCRIPTIONS WITHIN JOB CLASS.
*-----
***** Z 4 7 1 9 2 D *****
*
*   THIS DG DEFINES THE POSITION DESCRIPTION RECORD WHICH IS
*   THE TARGET DATA GROUP FOR CDMF FORMAT 47192 USED IN THE
*   APPLICATIONS PROGRAMMING CLASS.  THE KEY IS COMPRISED OF
*   THE JOB CLASS AND THE POSITION CODE (BOTH FOUND IN
*   DG 47120 OF THE EMP DATA BASE).  THE DESCRIPTION IS USED
*   IN THE REPORT GENERATED BY PROGRAM 478XX.
*   *** DGID = 47192 ***
***** 01 POSITION-DESCRIPTION-RECORD.
  05 POSITION-CODE-ACTION      PIC XX.
  05 POSITION-CODE-RESULT      PIC XX.
  05 POSITION-JOB-CLASS        PIC XX.
  05 POSITION-CODE             PIC XX.
  05 POSITION-DESCRIPTION      PIC X(40).
*
*   END OF Z47192D
    EJECT
***** *          P R O C E D U R E      D I V I S I O N *
*-----*
*   FOR EASE IN DE-BUGGING, PLEASE MODIFY CODE ONLY IN CLASS
*   CODING SECTION AND MARKED AREAS.
***** PROCEDURE DIVISION
      USING     TRANSACTION-CONTROL-BLOCK
                EMP-KEY-GROUP
                EMP-INFO-GROUP
                EMP-JOB-STATUS
                EMP-CURRENT-PAY
                EMP-YEAR-TO-DATE-PAY
                DATE-CONTROL-BLOCK
                PCD-CONTROL-BLOCK
                EMP-POSITION-CODE-DESCRIPTION
                POSITION-DESCRIPTION-RECORD.

      EJECT
*-----*
*   HOUSEKEEPING SECTION.
*-----*
      PERFORM  XX000-HOUSE-KEEPING.

      PERFORM  DA000-USE-DATE-SVCS  THRU  DA999-EXIT.

      *-*  IF TCB-BTCH
      *-*      AND TCB-FUNC-ID = FUNC-ID-99?? (\\\+\+20)
      *-*      PERFORM UP000-UPDATE-EMP-DATA-BASE  THRU UP999-EXIT.

      SKIP3

```



Umbrella Programming

Problem Specifications—Batch Program

*-----
BA000-MAIN-LINE SECTION.
*-----

BA100-READ-LOOP.

PERFORM RD000-READ-EMP-DB-SEQUENTIAL THRU RD099-EXIT.

PERFORM PC000-READ-CDMF-FORMAT THRU PC099-EXIT.

PERFORM PC100-READ-PCD-FORMAT THRU PC199-EXIT.

PERFORM CC000-LINK-TO-CCP THRU CC999-EXIT.

IF TCB-BTCH
 PERFORM PR000-PRINT-REPORT THRU PR999-EXIT
 GO TO BA100-READ-LOOP

ELSE
 PERFORM ZZ000-END-OF-PROCESSING.

SKIP3

*-----
CA000-CALL-PEM SECTION.
*-----

CALL 'PEM' USING TRANSACTION-CONTROL-BLOCK.

CA999-EXIT.
 EXIT.
 EJECT

*-----
CLASS-CODING SECTION.

*-----
DA000-USE-DATE-SVCS.
*-----

*-----
DA999-EXIT.
 EXIT.
 EJECT

*-----
PC000-READ-CDMF-FORMAT.
*-----

PC099-EXIT.
 EXIT.
 SKIP3

*-----
PC100-READ-PCD-FORMAT.
*-----



Umbrella Programming

Problem Specifications—Batch Program



Umbrella Programming

Problem Specifications—Batch Program

```
MOVE CA-LONG-PEM-TRANS-DUMP-END TO TCB-LONG-ACTIVITY.  
PERFORM CA000-CALL-PEM.  
  
YY999-EXIT.  
  EXIT.  
  SKIP3  
*-----  
ZZ000-END-OF-PROCESSING SECTION.  
*-----  
  
MOVE CA-LONG-PEM-END-PROG TO TCB-LONG-ACTIVITY.  
PERFORM CA000-CALL-PEM.  
  
STOP RUN.  
// ****
```

Notes:



Batch Problem Execute JCL

```
MODULE NAME ZUPCXXEX

//ZUP{J}EX JOB (HOGN,{B},BEF),'PGM EXEC Z9994\\\\\\',MSGCLASS=9,
//                      TIME=(00,04),REGION=4M,NOTIFY=&SYSUID
//P$$LIB  JCLLIB ORDER=( {TL}.PROCLIB)
//*
//JS010      EXEC HOGNBPEM,
//                  TEMPLIB=' {L}.TESTLIB',
//                  VSAM=' {V}'
//*-----*
//*   EXEC YOUR PROGRAM IN BATCH                         *
//*-----*
//EMP        DD DSN=&VSAM..EMP,DISP=SHR,DCB=BUFNO=10
//SYSIN     DD *
                           <===== PLACE YOUR TRANSACTIONS INFO HERE
//*****
```

Notes:



Summary

CSC

Summary



- PEM is in control with other programs acting as subroutines
- PEM addresses data through data groups
- Data groups in COBOL programs must match
 - Program definition on the Process Dictionary
 - LINKAGE SECTION of the program
 - 01 levels listed on the PROCEDURE DIVISION “USING” statement

© 2000 CSC. All rights reserved

CSC

Summary

- Transaction Control Block is key communication data group among PEM and all programs
- Never list TCB on program definition on the Process Dictionary
- Activities 1 - 100 are authorized for all programs
- Subsystems have other communication data groups, *control blocks*, to pass information

© 2000 CSC. All rights reserved





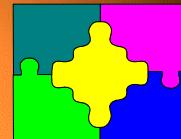
Summary

- Only static values are coded in working storage
- COBOL considers some fields character while PEM treats them as binary
- Application programs must check for results during processing flow

© 2000 CSC. All rights reserved



Summary



- The precompiler will compare linkage section and using statements of a COBOL program with data groups referenced on the Process Dictionary program definition. It will generate error messages that protect the programmer.
- Batch execution occurs by adding a transaction definition to Process Dictionary with source type 4 and placing the Company, Application ID, and Function of the transaction into SYSIN.

© 2000 CSC. All rights reserved



Umbrella Programming

Summary

- PEM is in control of programming flow when coding under any part of the Hogan System. All other programs run as subroutines of PEM.
- PEM establishes addressability to data through data groups.
- The order of data groups in COBOL programs must match in three key places:
 - Program definition on the Process Dictionary
 - LINKAGE SECTION of the program
 - 01 levels listed on the PROCEDURE DIVISION "USING" statement.
- The Transaction Control Block is the key communication data group among PEM and all application programs.
- Never list the TCB on a program definition on the Process Dictionary.
- Activities 1 through 100 are authorized for all programs to use.
- Subsystems have other important communication data groups, control blocks, to pass information among Hogan programs.
- Only static values are coded in working storage.
- Many key fields within the Hogan System are halfword and fullword binary values. Through redefining, COBOL considers the fields character while PEM treats them as binary.
- Application programs must check for results during processing flow. The type of activity issued determines the appropriate check.
- The precompiler will compare the linkage section and using statements of a COBOL program with the data groups referenced on the Process Dictionary program definition. It will generate error messages that will protect the programmer from many 0C4, 0C7, and other abends.
- Batch execution occurs by adding a transaction definition to the Process Dictionary with source type 4 and placing the Company, Application ID, and Function of the transaction into SYSIN.



Date Services System—DTS

13

Purpose



Date Services System - DTS

- ✿ Introduce a subsystem of the Umbrella
- ✿ Code use of Date Services System in a COBOL program
- ✿ Illustrate use of communication data group by subsystem

© 2001 CSC. All rights reserved.

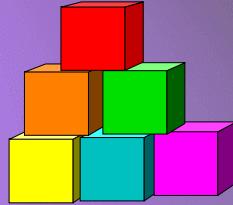
Notes:



Topics

CSC

Topics



- ★ Date formats
- ★ Date calculations
- ★ Date Services online screen
- ★ Data group 2000
- ★ Application programming code to invoke Date Services

© 2000 CSC. All rights reserved.

Objectives

CSC

Objectives



- List formats for storing dates under Hogan
- Give examples of the types of date calculations performed by the subsystem
- Use the online screen to do date manipulations
- Explain purpose of data group 2000 within Date Services
- Identify date services fields by comparing their COBOL names with location on date services map
- Invoke Date Services from an application program to do date manipulations.

© 2000 CSC. All rights reserved.



Date Services System (DTS)

The purpose of Date Services System is to provide a common routine for all Hogan Systems to request date manipulations.

Date Formats Supported

The requests will be performed in a uniform manner so that date calculations will be standard system wide. Five date formats are supported under the date services system.

The formats use the alpha codes:



Where:

C = CENTURY (0 = 20TH CENTURY)
Y = YEAR
M = MONTH
D = DAY
E = EDIT

© 2000 CSC. All rights reserved.

Notes:



Umbrella Programming

Date Services System (DTS)



Date Formats Supported

Format	Description
GREGORIAN (CYYMMDD)	This format is the system standard date format, usually packed decimal, century, year, month, and day.
JULIAN (0CYYDDD)	This date format consists of the century, year followed by the number of days since the first of the year. Usually packed decimal.
CALCULATION (0DDDDDD)	This date format is the number of days since January 1, 1801. Usually packed decimal.
NUMERIC (CYYeMMeDD)	This is the edited display format of the Gregorian date. (e = edit character)
ALPHANUMERIC (MONTH DD, YEAR)	This is the written form of the Gregorian date for display.

© 2000 CSC. All rights reserved.

Notes:



Umbrella Programming

Date Services System (DTS)

The following example will illustrate the different formats for a single date.

FORMAT	FORM
GREG	0761220
JUL	0076355
CALC	0064272
NUM	076/12/20
ALPHA	DECEMBER 20, 1976

NOTE

- 0 is used for the century code for dates in the 20th century (1900).
- In the century, the value 1 indicates dates in the 21st century (2000).
- Use 9 for the century code for dates in the 19th century (1800).
- Holiday calculations or business day calculations cannot be executed on 19th century dates.

Notes:

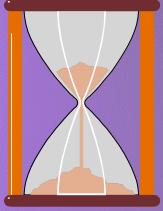


Date Calculations Supported

CSC

Date Calculations

- DTS converts and computes dates in any of the five supported formats
- DTS projects dates forwards and backwards in time
- For accurate projection, company holidays must be defined



© 2000 CSC. All rights reserved.

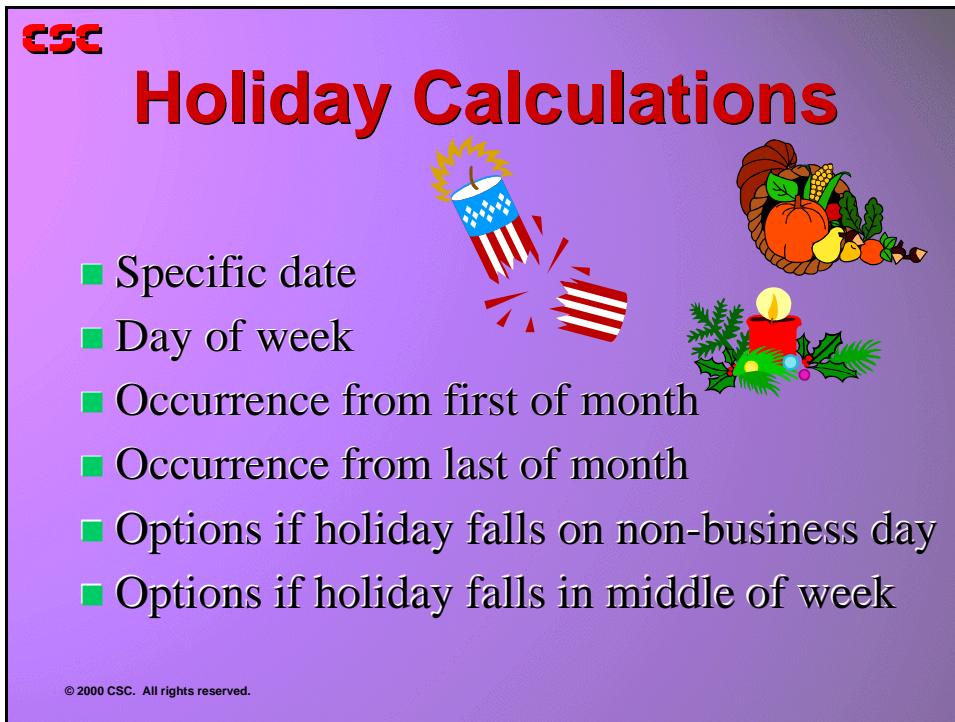
The Date Services routine can convert a Gregorian, Julian, or a Calculation Date into any of the other date formats. The routine can compute the number of calendar days, or the number of business days between two Gregorian, Julian, or Calculation dates. It can compute a new date from a Gregorian, Julian, or Calculation date. Either calendar or business days can be computed in the past or into the future.

It can convert a limited number of different display dates to a Gregorian date and determine if a Gregorian date is a calendar or business day; first of the month, quarter, or year; end of the month, quarter, or year; day of week; or if the year is a leap year.

Notes:



Holiday Calculations



The slide has a purple background. In the top left corner is the red CSC logo. The title "Holiday Calculations" is centered in large red letters. To the right of the title are three decorative icons: a lit firecracker, a cornucopia overflowing with fruit and vegetables, and a red candle surrounded by holly leaves. On the left side, there is a bulleted list of six items, each preceded by a green square icon. At the bottom left is a small copyright notice: "© 2000 CSC. All rights reserved."

- Specific date
- Day of week
- Occurrence from first of month
- Occurrence from last of month
- Options if holiday falls on non-business day
- Options if holiday falls in middle of week

The Date Services routine will automatically calculate a holiday table for an entire year and store that table on PCD when a holiday of the year is first referenced. The user can add to the holiday table as necessary to augment the holiday calculation. Note, the system cannot handle dates in the years prior to 1900 for holiday calculations.

HOLIDAYS CAN BE DEFINED AS:

- SPECIFIC DATE
- DAY OF WEEK
- OCCURRENCE FROM FIRST OF MONTH
- OCCURRENCE FROM LAST OF MONTH
- OPTIONS IF HOLIDAY FALLS ON NON-BUSINESS DAY
- OPTIONS IF HOLIDAY FALLS IN MIDDLE OF WEEK

On the Date Services Holiday Table Screen, there is a command to rebuild the holiday table for the year specified in the effective date field.



Application Communication With Date Services

Date Service Actions

Select option 5 on the "Umbrella System Master Menu" to access the Date Services System online.

To use the Date Service Request Test, an action is required. Following is the list of the action numerical codes and their functions.

The fields in parentheses are used as input. The resultant output will be placed in the arrowed field.

Note: Actions asterisk (*), the SQL-DATE-IN can be used instead of the DATE-IN field.

Note: All file names begin with DCB - i.e. DCB - Date Out.

ACTION	FUNCTION	DATE-OUT
.	GREGORIAN TO 'DATE' (DCBDATE-IN)	→ DATE-OUT
1 *	GREGORIAN TO JULIAN	
2 *	GREGORIAN TO CALCULATION	
3 *	GREGORIAN TO NUMERIC DISPLAY	
4 *	GREGORIAN TO ALPHANUMERIC DISPLAY	
.	JULIAN TO 'DATE' (DATE-IN)	→ DATE-OUT
10	JULIAN TO GREGORIAN	
11	JULIAN TO CALCULATION	
12	JULIAN TO NUMERIC DISPLAY	
13	JULIAN TO ALPHANUMERIC DISPLAY	
.	CALCULATION TO 'DATE' (DATE-IN)	→ DATE-OUT
20	CALCULATION TO GREGORIAN	
21	CALCULATION TO JULIAN	
22	CALCULATION TO NUMERIC DISPLAY	
23	CALCULATION TO ALPHANUMERIC DISPLAY	
.	BUMP TO NEXT CALENDAR DATE (DATE-IN, DIFF-FACTOR, DIFF-AMT)	→ DATE-OUT
30 *	BUMP TO NEXT DATE GREGORIAN	
31	BUMP TO NEXT DATE JULIAN	
32	BUMP TO NEXT DATE CALCULATION	
33 *	BUMP TO NEXT DATE GREGORIAN - 360 DAY YEAR	
.	BUMP TO NEXT BUSINESS DATE (DATE-IN, DIFF-FACTOR, DIFF-AMT)	→ DATE-OUT



Umbrella Programming

Application Communication With Date Services

```

40 *      BUMP TO NEXT BUSINESS DATE GREGORIAN
41      BUMP TO NEXT BUSINESS DATE JULIAN
42      BUMP TO NEXT BUSINESS DATE CALCULATION

ACTION      FUNCTION
-----      -----
. DAY DIFFERENCE (DATE-IN, DATE-OUT) → DIFF-AMT

50      DAY DIFFERENCE GREGORIAN
51      DAY DIFFERENCE JULIAN
52      DAY DIFFERENCE CALCULATION
53      DAY DIFFERENCE GREGORIAN - 360 DAY YEAR

. BUSINESS DAY DIFFERENCE (DATE-IN, DATE-OUT) → DIFF-AMT

60      BUSINESS DAY DIFFERENCE GREGORIAN
61      BUSINESS DAY DIFFERENCE JULIAN
62      BUSINESS DAY DIFFERENCE CALCULATION

. MONTH DIFFERENCE (DATE-IN, DATE-OUT) → DIFF-AMT

70      MONTH DIFFERENCE GREGORIAN → DIFF-AMT

. HOLIDAY DISPLAY (DATE-IN) → HOLIDAYS

80 *      DISPLAY HOLIDAYS

. FORMATTED DATE TO GREGORIAN (RAW-DATE-AREA) → DATE-OUT

100      FORMATTED DATE (CYYMMDD) TO GREGORIAN
101      FORMATTED DATE (CYY-MM-DD) TO GREGORIAN
102      FORMATTED DATE (MMDDYY) TO GREGORIAN
103      FORMATTED DATE (MM-DD-YY) TO GREGORIAN
104 *      GREGORIAN TO FORMATTED DATE (MMDDYYC)
105 *      GREGORIAN TO FORMATTED DATE (MMDDYY)

```



Umbrella Programming

Application Communication With Date Services

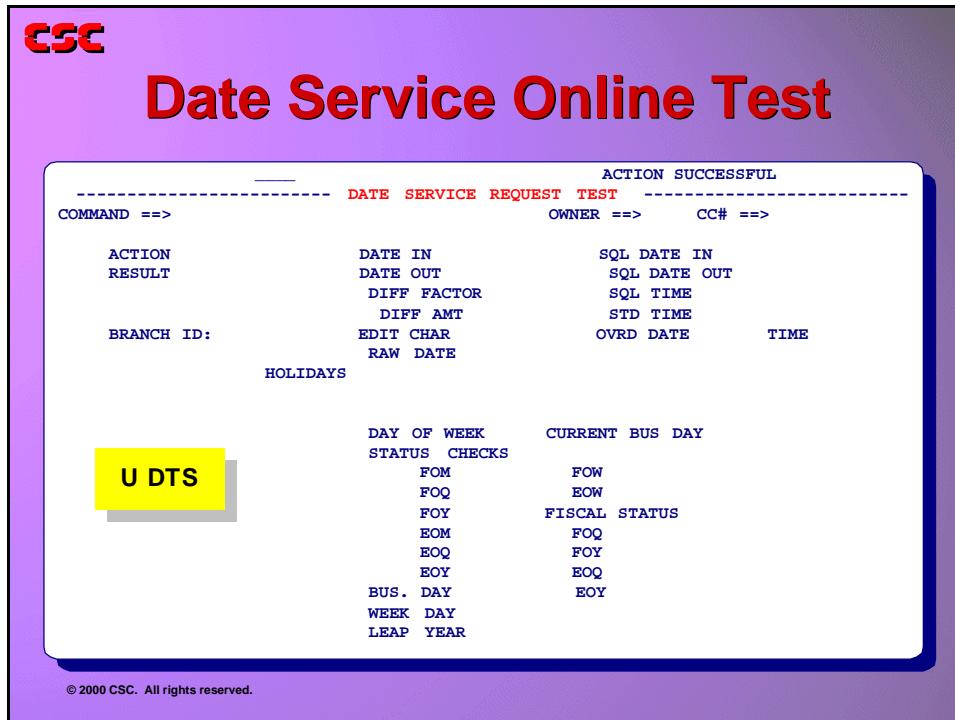
ACTION	FUNCTION	

	.	DATE STATUS CHECK (DATE-IN) → VARIOUS
130 *	DATE STATUS CHECK ALL	
131 *	DATE STATUS CHECK PHYSICAL FIRST OF MONTH	
132 *	DATE STATUS CHECK PHYSICAL FIRST OF QUARTER	
133 *	DATE STATUS CHECK PHYSICAL FIRST OF YEAR	
134 *	DATE STATUS CHECK PHYSICAL END OF MONTH	
135 *	DATE STATUS CHECK PHYSICAL END OF QUARTER	
136 *	DATE STATUS CHECK PHYSICAL END OF YEAR	
137 *	DATE STATUS CHECK BUSINESS FIRST OF MONTH	
138 *	DATE STATUS CHECK BUSINESS FIRST OF QUARTER	
139 *	DATE STATUS CHECK BUSINESS FIRST OF YEAR	
140 *	DATE STATUS CHECK BUSINESS END OF MONTH	
141 *	DATE STATUS CHECK BUSINESS END OF QUARTER	
142 *	DATE STATUS CHECK BUSINESS END OF YEAR	
143 *	DATE STATUS CHECK BUSINESS DAY OR NON BUSINESS DAY	
144 *	DATE STATUS CHECK DAY OF WEEK	
145 *	DATE STATUS CHECK FOR LEAP YEAR	
	.	MISCELLANEOUS DATE SERVICES (DATE-IN)
160 *	CALCULATE CURRENT BUSINESS DAY WITHIN MONTH	
161 *	CALCULATE NEXT WEEK DAY	
	.	SQL FUNCTIONS
200	CONVERT GREGORIAN TO SQL	
201	CONVERT SQL TO GREGORIAN	
202	CONVERT PACKED TO SQL TIME	
203	CONVERT SQL TO PACKED TIME	



Date Services Online Test Screen

U DTS from a cleared screen will display the Date Service Request Test Screen.



Press PF1 to display HELP information.

Notes:



Umbrella Programming

Application Communication With Date Services

Field Description/Values

ACTION

See list of actions on previous pages to enter the numeric code equivalent to function required.

RESULT

Error code will be returned in this field if necessary.

DATE IN

For actions 1-80 and 130-161 format is YYMMDD.

DATE OUT

For actions 50-70 (only) format is YYMMDD.

DIFF FACTOR

For actions 30-42 (only) format is **D** for day, **M** for month, **Y** for year.

DIFF AMT

For actions 30-42 (only).

BRANCH ID

This field is only used when Holiday Tables are generated for a branch, in which case the branch Holiday Table takes precedence over the company Holiday Table.

EDIT CHAR

For example, /, -, or blank.

RAW DATE

For actions 100-105 (only).

HOLIDAYS

System will return holidays for year entered in DATE IN.

DAY OF WEEK

System will return day of week for action 144. **1** for Sunday, **2** for Monday, and so on.

STATUS CHECKS

System will return.

SQL DATE IN

Date in the standard SQL format defined on the site control record.

The SQL formatted input date will only be used if the normal input date field (DCB-DATE-IN) is packed zero and the SQL date field is not blank.



SQL DATE OUT

This field will display the DATE OUT field contents in the standard SQL format defined on the site control record.

SQL TIME

Time in the standard SQL format defined on the site control record. This field is only used for the time conversion command. If Hogan time is being converted, the result will be displayed in this field.

STD TIME

Hogan formatted time assuming the sequence of years, months, and days as defined by the default edit pattern on the site control record. This field is only used for the time conversion command. If SQL time is being converted, the result will be displayed in this field.

OVRD DATE

This field overrides the SQL date format defined on the site control record.

TIME

This field overrides the SQL time format defined on the site control record.

Notes:



Umbrella Programming

Application Communication With Date Services

Date Service Research Exercise



1. How many days have you been living?
2. On which day of the week were you born?
3. Will 2004 be a leap year? Yes No
4. Which day of the week was new year's day, the year you were born?
5. What is 10 calendar days from today?
6. What is the alphanumeric date of the day you were born?



Date Control Block

Application programs communicate with Date Services through data group 2000. This data group is known as the Date Control Block and is defined by copybook T58001D (COBOL) and T58001E (ALC).

All communication with the Date Services System is achieved by loading the appropriate data into the Date Control Block and passing control to Date Services by issuing activity 1900.

On return from Date Services, the Date Control Block result field should be examined to determine if and how the date request was honored.

Notes:



Umbrella Programming

Application Communication With Date Services

```
DATA GROUP          COPYBOOK NAME T58001D

*--* START OF T58001D *----* DATE CONTROL BLOCK *-----*
*
*   DATA GROUP - 2000
*   LINK ACTIVITY - 1900
*
*****
01  DATE-CONTROL-BLOCK.

05  DCB-ACTION      PIC XX.
05  DCB-RESULT       PIC XX.
05  DCB-DATE-IN     PIC S9(7) COMP-3.
05  DCB-DATE-OUT    PIC S9(7) COMP-3.
05  DCB-DIFF-FACTOR PIC X(1).
   88  DCB-DIFF-DAYS           VALUE 'D'.
   88  DCB-DIFF-MONTHS        VALUE 'M'.
   88  DCB-DIFF-YEARS         VALUE 'Y'.
05  DCB-DIFF-AMT      PIC S9(7) COMP-3.
05  DCB-EDIT-CHAR    PIC X(1).
05  DCB-RAW-DATE-AREA PIC X(25).
*
05  DCB-DISPLAY      REDEFINES DCB-RAW-DATE-AREA.
  10  DCB-NUM-WC.
    15  DCB-NUM-C      PIC X.
    15  DCB-NUM-WOC.
      20  FILLER      PIC X(8).
    15  FILLER      PIC X(16).
*
05  DCB-HOLIDAYS OCCURS 15 TIMES.
  10  DCB-HOLIDAY    PIC S9(7) COMP-3.
05  DCB-DAY-OF-WEEK  PIC 9(1).
   88  DCB-DOW-SUN    VALUE 1.
   88  DCB-DOW-MON    VALUE 2.
   88  DCB-DOW-TUE    VALUE 3.
   88  DCB-DOW-WED    VALUE 4.
   88  DCB-DOW-THR    VALUE 5.
   88  DCB-DOW-FRI    VALUE 6.
   88  DCB-DOW-SAT    VALUE 7.
05  DCB-DATE-STATUS.
  10  DCB-FOM        PIC X(1).
    88  DCB-FOM-BUS   VALUE 'B' 'T'.
    88  DCB-FOM-CLNDR VALUE 'P' 'T'.
  10  DCB-FOQ        PIC X(1).
    88  DCB-FOQ-BUS   VALUE 'B' 'T'.
    88  DCB-FOQ-CLNDR VALUE 'P' 'T'.
  10  DCB-FOY        PIC X(1).
    88  DCB-FOY-BUS   VALUE 'B' 'T'.
    88  DCB-FOY-CLNDR VALUE 'P' 'T'.
  10  DCB-EOM        PIC X(1).
    88  DCB-EOM-BUS   VALUE 'B' 'T'.
    88  DCB-EOM-CLNDR VALUE 'P' 'T'.
  10  DCB-EOQ        PIC X(1).
    88  DCB-EOQ-BUS   VALUE 'B' 'T'.
    88  DCB-EOQ-CLNDR VALUE 'P' 'T'.
  10  DCB-EOY        PIC X(1).
    88  DCB-EOY-BUS   VALUE 'B' 'T'.
    88  DCB-EOY-CLNDR VALUE 'P' 'T'.
  10  DCB-BUS-DAY    PIC X(1).
    88  DCB-BDAY      VALUE 'B'.
  10  DCB-WEEK-DAY   PIC X(1).
    88  DCB-WDAY      VALUE 'W'.
  10  DCB-LEAP-YEAR  PIC X(1).
```



Umbrella Programming

Application Communication With Date Services

```

          88 DCB-LYEAR                      VALUE 'L'.
10  DCB-FOW      PIC X.                 VALUE 'B' 'T'.
          88 DCB-FOW-BUS                  VALUE 'P' 'T'.
          88 DCB-FOW-CLNDR
10  DCB-EOW      PIC X.                 VALUE 'B' 'T'.
          88 DCB-EOW-BUS                  VALUE 'P' 'T'.
          88 DCB-EOW-CLNDR
05  DCB-CURR-BUSINESS-DAY   PIC 99.
05  DCB-FISCAL-STATUS.
10  DCB-FISCAL-FOQ  PIC X(1).        VALUE 'B' 'T'.
          88 DCB-FISCAL-FOQ-BUS          VALUE 'P' 'T'.
          88 DCB-FISCAL-FOQ-CLNDR
10  DCB-FISCAL-FOY  PIC X(1).        VALUE 'B' 'T'.
          88 DCB-FISCAL-FOY-BUS          VALUE 'P' 'T'.
          88 DCB-FISCAL-FOY-CLNDR
10  DCB-FISCAL-EOQ  PIC X(1).        VALUE 'B' 'T'.
          88 DCB-FISCAL-EOQ-BUS          VALUE 'P' 'T'.
          88 DCB-FISCAL-EOQ-CLNDR
10  DCB-FISCAL-EOY  PIC X(1).        VALUE 'B' 'T'.
          88 DCB-FISCAL-EOY-BUS          VALUE 'P' 'T'.
          88 DCB-FISCAL-EOY-CLNDR
*                                         *
05  DCB-SQL-FIELDS.
10  DCB-SQL-DATE-IN     PIC X(10).
10  DCB-SQL-DATE-OUT    PIC X(10).
10  DCB-SQL-TIME        PIC X(08).
10  DCB-PACK-TIME       PIC S9(07)   COMP-3.
10  DCB-SQL-OVRD-DATE   PIC X(03).
          88 DCB-SQL-EUR-DATE          VALUE 'EUR'.
          88 DCB-SQL-ISO-DATE          VALUE 'ISO'.
          88 DCB-SQL-JIS-DATE          VALUE 'JIS'.
          88 DCB-SQL-USA-DATE          VALUE 'USA'.
10  DCB-SQL-OVRD-TIME   PIC X(03).
          88 DCB-SQL-EUR-TIME          VALUE 'EUR'.
          88 DCB-SQL-ISO-TIME          VALUE 'ISO'.
          88 DCB-SQL-JIS-TIME          VALUE 'JIS'.
          88 DCB-SQL-USA-TIME          VALUE 'USA'.
*                                         *
05  DCB-BRANCH-ID-X.
10  DCB-BRANCH-ID       PIC 9(05)   COMP-3.
*                                         *
01  FILLER REDEFINES DATE-CONTROL-BLOCK.
05  DCB-DATA            PIC X(17).
*                                         *
*----* END OF T58001D *-----*

```

Notes:



Umbrella Programming

Application Communication With Date Services

Date Services Actions Copybook T58007D

```
DATA GROUP          COPYBOOK NAME T58007D

*--* START OF T58007D *----* DATE ROUTINE ACTIONS *-----*
*
*
*
*****
*
01  DCB-ACTIONS.

05  DCB-PROG-LINK.
    10  DCB-LINK-DSV-LONG.
        15  FILLER      PIC S9(9) COMP VALUE +1900.
    10  FILLER      REDEFINES DCB-LINK-DSV-LONG.
        15  FILLER      PIC XX.
        15  DCB-LINK-DSV  PIC XX.

*
    05  DCB-PROG-ACTIONS.
*
* GREGORIAN TO JULIAN
    10  DCB-AC-GREG-TO-JUL.
        15  FILLER      PIC S9(4) COMP VALUE +0001.
*
* GREGORIAN TO CALCULATION
    10  DCB-AC-GREG-TO-CALC.
        15  FILLER      PIC S9(4) COMP VALUE +0002.
*
* GREGORIAN TO NUMERIC DISPLAY
    10  DCB-AC-GREG-TO-NUM-DSP.
        15  FILLER      PIC S9(4) COMP VALUE +0003.
*
* GREGORIAN TO ALPHANUMERIC DISPLAY
    10  DCB-AC-GREG-TO-ALPHA-DSP.
        15  FILLER      PIC S9(4) COMP VALUE +0004.
*
* JULIAN TO GREGORIAN
    10  DCB-AC-JUL-TO-GREG.
        15  FILLER      PIC S9(4) COMP VALUE +0010.
*
* JULIAN TO CALCULATION
    10  DCB-AC-JUL-TO-CALC.
        15  FILLER      PIC S9(4) COMP VALUE +0011.
*
* JULIAN TO NUMERIC DISPLAY
    10  DCB-AC-JUL-TO-NUM-DSP.
        15  FILLER      PIC S9(4) COMP VALUE +0012.
*
* JULIAN TO ALPHANUMERIC DISPLAY
    10  DCB-AC-JUL-TO-ALPHA-DSP.
        15  FILLER      PIC S9(4) COMP VALUE +0013.
*
* CALCULATION TO GREGORIAN
    10  DCB-AC-CALC-TO-GREG.
        15  FILLER      PIC S9(4) COMP VALUE +0020.
*
* CALCULATION TO JULIAN
    10  DCB-AC-CALC-TO-JUL.
        15  FILLER      PIC S9(4) COMP VALUE +0021.
*
* CALCULATION TO NUMERIC DISPLAY
    10  DCB-AC-CALC-TO-NUM-DSP.
        15  FILLER      PIC S9(4) COMP VALUE +0022.
*
* CALCULATION TO ALPHANUMERIC DISPLAY
    10  DCB-AC-CALC-TO-ALPHA-DSP.
        15  FILLER      PIC S9(4) COMP VALUE +0023.
*
* BUMP TO NEXT DATE GREGORIAN
    10  DCB-AC-BND-GREG.
        15  FILLER      PIC S9(4) COMP VALUE +0030.
*
* BUMP TO NEXT DATE JULIAN
    10  DCB-AC-BND-JUL.
        15  FILLER      PIC S9(4) COMP VALUE +0031.
*
* BUMP TO NEXT DATE CALCULATION
    10  DCB-AC-BND-CALC.
```



Umbrella Programming

Application Communication With Date Services

```

        15 FILLER          PIC S9(4) COMP VALUE +0032.
*   BUMP TO NEXT DATE GREGORIAN 360
        10 DCB-AC-BND-G360.
        15 FILLER          PIC S9(4) COMP VALUE +0033.
*   BUMP TO NEXT BUSINESS DATE GREGORIAN
        10 DCB-AC-BNBD-GREG.
        15 FILLER          PIC S9(4) COMP VALUE +0040.
*   BUMP TO NEXT BUSINESS DATE JULIAN
        10 DCB-AC-BNBD-JUL.
        15 FILLER          PIC S9(4) COMP VALUE +0041.
*   BUMP TO NEXT BUSINESS DATE CALCULATION
        10 DCB-AC-BNBD-CALC.
        15 FILLER          PIC S9(4) COMP VALUE +0042.
*   GREGORIAN DAY DIFFERENCE
        10 DCB-AC-DIFF-GREG.
        15 FILLER          PIC S9(4) COMP VALUE +0050.
*   JULIAN DAY DIFFERENCE
        10 DCB-AC-DIFF-JUL.
        15 FILLER          PIC S9(4) COMP VALUE +0051.
*   CALCULATION DAY DIFFERENCE
        10 DCB-AC-DIFF-CALC.
        15 FILLER          PIC S9(4) COMP VALUE +0052.
*   GREGORIAN 360 DAY DIFFERENCE
        10 DCB-AC-DIFF-G360.
        15 FILLER          PIC S9(4) COMP VALUE +0053.
*   BUSINESS DAY DIFFERENCE GREGORIAN
        10 DCB-AC-BUS-DIFF-GREG.
        15 FILLER          PIC S9(4) COMP VALUE +0060.
*   BUSINESS DAY DIFFERENCE JULIAN
        10 DCB-AC-BUS-DIFF-JUL.
        15 FILLER          PIC S9(4) COMP VALUE +0061.
*   BUSINESS DAY DIFFERENCE CALCULATION
        10 DCB-AC-BUS-DIFF-CALC.
        15 FILLER          PIC S9(4) COMP VALUE +0062.
*   GREGORIAN MONTHS DIFFERENCE
        10 DCB-AC-MO-DIFF-GREG.
        15 FILLER          PIC S9(4) COMP VALUE +0070.
*   HOLIDAY TABLE REQUEST
        10 DCB-AC-HOLIDAYS.
        15 FILLER          PIC S9(4) COMP VALUE +0080.
*   HOLIDAY TABLE RENEW REQUEST
        10 DCB-AC-RENEW-HOLIDAYS.
        15 FILLER          PIC S9(4) COMP VALUE +0081.
*   FORMATED DATE (CYYMMDD) TO GREGORIAN
        10 DCB-AC-FMT00-GREG.
        15 FILLER          PIC S9(4) COMP VALUE +0100.
*   FORMATED DATE (YY-MM-DD) TO GREGORIAN
        10 DCB-AC-FMT01-GREG.
        15 FILLER          PIC S9(4) COMP VALUE +0101.
*   FORMATED DATE (MMDDYY) TO GREGORIAN
        10 DCB-AC-FMT02-GREG.
        15 FILLER          PIC S9(4) COMP VALUE +0102.
*   FORMATED DATE (MM-DD-YY) TO GREGORIAN
        10 DCB-AC-FMT03-GREG.
        15 FILLER          PIC S9(4) COMP VALUE +0103.
*   FORMATED DATE GREGORIAN TO (MMDDYYC)
        10 DCB-AC-FMT04-GREG.
        15 FILLER          PIC S9(4) COMP VALUE +0104.
*   FORMATED DATE GREGORIAN TO (MMDDYY)
        10 DCB-AC-FMT05-GREG.
        15 FILLER          PIC S9(4) COMP VALUE +0105.
*   DATE STATUS CHECK ALL
        10 DCB-AC-DSC-ALL.
        15 FILLER          PIC S9(4) COMP VALUE +0130.
*   DATE STATUS CHECK PHYSICAL FIRST OF MONTH

```



Umbrella Programming

Application Communication With Date Services

```
10 DCB-AC-DSC-PHY-FOM.  
      15 FILLER          PIC S9(4) COMP VALUE +0131.  
*   DATE STATUS CHECK PHYSICAL FIRST OF QUARTER  
      10 DCB-AC-DSC-PHY-FOQ.  
      15 FILLER          PIC S9(4) COMP VALUE +0132.  
*   DATE STATUS CHECK PHYSICAL FIRST OF YEAR  
      10 DCB-AC-DSC-PHY-FOY.  
      15 FILLER          PIC S9(4) COMP VALUE +0133.  
*   DATE STATUS CHECK PHYSICAL END OF MONTH  
      10 DCB-AC-DSC-PHY-EOM.  
      15 FILLER          PIC S9(4) COMP VALUE +0134.  
*   DATE STATUS CHECK PHYSICAL END OF QUARTER  
      10 DCB-AC-DSC-PHY-EOQ.  
      15 FILLER          PIC S9(4) COMP VALUE +0135.  
*   DATE STATUS CHECK PHYSICAL END OF YEAR  
      10 DCB-AC-DSC-PHY-EOY.  
      15 FILLER          PIC S9(4) COMP VALUE +0136.  
*   DATE STATUS CHECK BUSINESS FIRST OF MONTH  
      10 DCB-AC-DSC-BUS-FOM.  
      15 FILLER          PIC S9(4) COMP VALUE +0137.  
*   DATE STATUS CHECK BUSINESS FIRST OF QUARTER  
      10 DCB-AC-DSC-BUS-FOQ.  
      15 FILLER          PIC S9(4) COMP VALUE +0138.  
*   DATE STATUS CHECK BUSINESS FIRST OF YEAR  
      10 DCB-AC-DSC-BUS-FOY.  
      15 FILLER          PIC S9(4) COMP VALUE +0139.  
*   DATE STATUS CHECK BUSINESS END OF MONTH  
      10 DCB-AC-DSC-BUS-EOM.  
      15 FILLER          PIC S9(4) COMP VALUE +0140.  
*   DATE STATUS CHECK BUSINESS END OF QUARTER  
      10 DCB-AC-DSC-BUS-EOQ.  
      15 FILLER          PIC S9(4) COMP VALUE +0141.  
*   DATE STATUS CHECK BUSINESS END OF YEAR  
      10 DCB-AC-DSC-BUS-EOY.  
      15 FILLER          PIC S9(4) COMP VALUE +0142.  
*   DATE STATUS CHECK BUSINESS DAY OR NON BUSINESS DAY  
      10 DCB-AC-DSC-BUS-DAY.  
      15 FILLER          PIC S9(4) COMP VALUE +0143.  
*   DATE STATUS CHECK DAY OF WEEK  
      10 DCB-AC-DSC-WEEK-DAY.  
      15 FILLER          PIC S9(4) COMP VALUE +0144.  
*   DATE STATUS CHECK LEAP YEAR  
      10 DCB-AC-DSC-LEAP-YEAR.  
      15 FILLER          PIC S9(4) COMP VALUE +0145.  
*   DATE STATUS CHECK PHYSICAL FIRST OF WEEK  
      10 DCB-AC-DSC-PHY-FOW.  
      15 FILLER          PIC S9(4) COMP VALUE +0146.  
*   DATE STATUS CHECK PHYSICAL END OF WEEK  
      10 DCB-AC-DSC-PHY-EOW.  
      15 FILLER          PIC S9(4) COMP VALUE +0147.  
*   DATE STATUS CHECK BUSINESS FIRST OF WEEK  
      10 DCB-AC-DSC-BUS-FOW.  
      15 FILLER          PIC S9(4) COMP VALUE +0148.  
*   DATE STATUS CHECK BUSINESS END OF WEEK  
      10 DCB-AC-DSC-BUS-EOW.  
      15 FILLER          PIC S9(4) COMP VALUE +0149.  
*   CALCULATE CURRENT BUSINESS DAY  
      10 DCB-AC-CALC-CURR-BUS.  
      15 FILLER          PIC S9(4) COMP VALUE +0160.  
*   CALCULATE NEXT WEEK DAY  
      10 DCB-AC-CALC-NXT-WKDAY.  
      15 FILLER          PIC S9(4) COMP VALUE +0161.  
*   CONVERT GREGORIAN TO SQL FORMATTED DATE  
      10 DCB-HOGAN-TO-SQL-DATE.  
      15 FILLER          PIC S9(4) COMP VALUE +0200.
```



Umbrella Programming

Application Communication With Date Services

```
*      CONVERT SQL FORMATTED DATE TO GREGORIAN
      10  DCB-SQL-TO-HOGAN-DATE.
          15  FILLER           PIC S9(4) COMP VALUE +0201.
*      CONVERT HOGAN TO SQL TIME
      10  DCB-HOGAN-TO-SQL-TIME.
          15  FILLER           PIC S9(4) COMP VALUE +0202.
*      CONVERT SQL TO HOGAN TIME
      10  DCB-SQL-TO-HOGAN-TIME.
          15  FILLER           PIC S9(4) COMP VALUE +0203.
*
*
*-----* END OF T58007D *-----*
```

Notes:



Umbrella Programming

Application Communication With Date Services

Date Services Results Copybook T58008D

```
DATA GROUP          COPYBOOK NAME T58008D

*--* START OF T58008D *----* DATE ROUTINE RESULTS *-----*
*
*
*
*****
*
01  DCB-RESULTS.
    05  DCB-RESULT-VALUES.
*   SUCCESSFUL COMPLETION
    10  FILLER      PIC S9(8) COMP  VALUE +00000.
*   INVALID ACTION
    10  FILLER      PIC S9(8) COMP  VALUE +58001.
*   INVALID GREGORIAN MONTH
    10  FILLER      PIC S9(8) COMP  VALUE +58002.
*   INVALID GREGORIAN DAY
    10  FILLER      PIC S9(8) COMP  VALUE +58003.
*   INVALID JULIAN DAY
    10  FILLER      PIC S9(8) COMP  VALUE +58004.
*   INVALID CALCULATION DATE
    10  FILLER      PIC S9(8) COMP  VALUE +58005.
*   INVALID DIFFERENCE FACTOR
    10  FILLER      PIC S9(8) COMP  VALUE +58006.
*   INVALID CYCLE PERIOD
    10  FILLER      PIC S9(8) COMP  VALUE +58050.
*   INVALID FREQ CODE
    10  FILLER      PIC S9(8) COMP  VALUE +58051.
*   INVALID FREQ WEEK
    10  FILLER      PIC S9(8) COMP  VALUE +58052.
*   INVALID FREQ DATE
    10  FILLER      PIC S9(8) COMP  VALUE +58053.
*   INVALID FREQ BUMP
    10  FILLER      PIC S9(8) COMP  VALUE +58054.
*   INVALID FREQ CYCLE
    10  FILLER      PIC S9(8) COMP  VALUE +58055.
*   INVALID CYCLE EXCEPT
    10  FILLER      PIC S9(8) COMP  VALUE +58056.
*   INVALID FREQ AMT
    10  FILLER      PIC S9(8) COMP  VALUE +58057.
*   INVALID GREGORIAN MONTH IN DATE OUT
    10  FILLER      PIC S9(8) COMP  VALUE +58102.
*   INVALID GREGORIAN DAY IN DATE OUT
    10  FILLER      PIC S9(8) COMP  VALUE +58103.
*   INVALID JULIAN DAY IN DATE OUT
    10  FILLER      PIC S9(8) COMP  VALUE +58104.
*   INVALID CALCULATION DATE IN DATE OUT
    10  FILLER      PIC S9(8) COMP  VALUE +58105.
*   INVALID ACTION FOR DATE
    10  FILLER      PIC S9(8) COMP  VALUE +58106.
*   NOT EFFECTIVE
    10  FILLER      PIC S9(8) COMP  VALUE +58150.
*   MORE THAN 15 CYCLES
    10  FILLER      PIC S9(8) COMP  VALUE +58151.
*   INVALID FROM-THRU RANGE
    10  FILLER      PIC S9(8) COMP  VALUE +58152.
*   INVALID FIRST-LAST RANGE
    10  FILLER      PIC S9(8) COMP  VALUE +58153.
*   WARNING CALCULATED DATES HAVE BEEN ADJUSTED.
    10  FILLER      PIC S9(8) COMP  VALUE +58154.
*   SQL DATE IS INVALID
    10  FILLER      PIC S9(8) COMP  VALUE +58200.
```



Umbrella Programming

Application Communication With Date Services

```
*      SQL FORMAT ON SITE CONTROL RECORD NOT SUPPORTED
      10  FILLER          PIC S9(8) COMP      VALUE +58201.
*
*-----*
*
*      05 DCB-RESULT-NAMES      REDEFINES DCB-RESULT-VALUES.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-OK        PIC XX.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-INVALID-ACTION PIC XX.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-INVALID-GREG-MONTH PIC XX.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-INVALID-GREG-DAY   PIC XX.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-INVALID-JUL-DAY   PIC XX.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-INVALID-CALC-DATE PIC XX.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-INVALID-DIFF-FACTOR PIC XX.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-INVALID-CYCLE-PERIOD PIC XX.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-INVALID-FREQ-CODE  PIC XX.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-INVALID-FREQ-WEEK  PIC XX.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-INVALID-FREQ-DATE  PIC XX.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-INVALID-FREQ-BUMP  PIC XX.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-INVALID-FREQ-CYCLE  PIC XX.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-INVALID-CYCLE-EXCEPT PIC XX.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-INVALID-FREQ-AMT   PIC XX.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-INVALID-GREG-MO-DO  PIC XX.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-INVALID-GREG-DAY-DO PIC XX.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-INVALID-JUL-DAY-DO  PIC XX.
  10  FILLER.
      15 FILLER          PIC XX.
      15 DCB-RS-INVALID-CALC-DATE-DO PIC XX.
```



Umbrella Programming

Application Communication With Date Services

```
      15  FILLER          PIC XX.  
      15  DCB-RS-INVALID-DATE-ACT  PIC XX.  
10  FILLER.  
      15  FILLER          PIC XX.  
      15  DCB-RS-NOT-EFFECTIVE  PIC XX.  
10  FILLER.  
      15  FILLER          PIC XX.  
      15  DCB-RS-MORE-CYCLES  PIC XX.  
10  FILLER.  
      15  FILLER          PIC XX.  
      15  DCB-RS-INVALID-FT-RANGE  PIC XX.  
10  FILLER.  
      15  FILLER          PIC XX.  
      15  DCB-RS-INVALID-FL-RANGE  PIC XX.  
10  FILLER.  
      15  FILLER          PIC XX.  
      15  DCB-RS-DATE-ADJUST    PIC XX.  
10  FILLER.  
      15  FILLER          PIC XX.  
      15  DCB-RS-SQL-DATE-INVALID  PIC XX.  
10  FILLER.  
      15  FILLER          PIC XX.  
      15  DCB-RS-SQL-FORMAT-NOT-SUP  PIC XX.  
*                                         *  
*----* END OF T58008D *-----*
```

Notes:



Umbrella Programming

Application Communication With Date Services

Sample Program Using Date Services

```
IDENTIFICATION DIVISION.
```

```
WORKING-STORAGE SECTION.
```

```
DCB-ACTIONS    COPY    T58007D.  
DCB-RESULTS    COPY    T58008D.
```

```
LINKAGE SECTION.
```

```
PEM-TCB        COPY    P49000D.  
DATE-CONTROL-BLOCK  COPY    T58001D.
```

```
PROCEDURE DIVISION  
  USING TRANSACTION-CONTROL-BLOCK  
        DATE-CONTROL-BLOCK.
```

```
AA000-MAIN-SECTION.
```

```
MOVE  TCB-SYS-DATE           TO  DCB-DATE-IN.  
MOVE  DCB-AC-GREG-TO-ALPHA-DSP  TO  DCB-ACTION.  
MOVE  DCB-LINK-DSV-LONG       TO  TCB-LONG-ACTIVITY.  
CALL 'PEM' USING TRANSACTION-CONTROL-BLOCK.  
IF DCB-RESULT NOT EQUAL TO DCB-RS-OK PERFORM ERROR-RTN.
```

```
*** THE ALPHANUMERIC DISPLAY OF TODAY'S DATE IS  
*** RETURNED IN THE DATE-CONTROL-BLOCK FIELD OF  
*** DCB-RAW-DATE-AREA.
```



Umbrella Programming

Problem Specifications—DTS



Problem Specifications—DTS

The purpose of this problem is to develop the logic to convert the system IPL date from its packed format to an alphanumeric print format that is 10 business days greater than the IPL date.

The IPL date is contained in the TCB and may be referenced by use of the COBOL data-name TCB-SYS-DATE.

Note: The problem may require more than one Date Service action.

Leave the reformatted date in the appropriate output field in the date control block after conversion. SPS looks for the returned date in the DCB-RAW-DATE-AREA field.

Activity 1900 is used to link to Date Services. A COBOL definition for this activity has been included in your program.

The DTS communication data group 2000, known as the Date Control Block, is defined by COBOL copybook T58001D. T58001D has been included in the linkage section of your program.

The DTS action and result code static values, defined in COBOL copy books T58007D and T58008D respectively, have been included in working storage of your program.

Add the required code to your program.

Submit for compile/link.

To test your program, submit the execute job.

Notes:



Summary



Summary



- Date Services System performs a wide variety of date manipulations
- Five date formats are maintained. Gregorian (CYYMMDD) is system standard
- Date Services allows online manipulation by entering **U 5.1** or **U DTS** from cleared screen
- Data group 2000 is the DTS communication data group
- Link activity 1900 is delivered Process Dictionary activity to invoke Data Services System
- ~~Each functional date manipulation requires a separate transaction code.~~

Notes:



Umbrella Programming

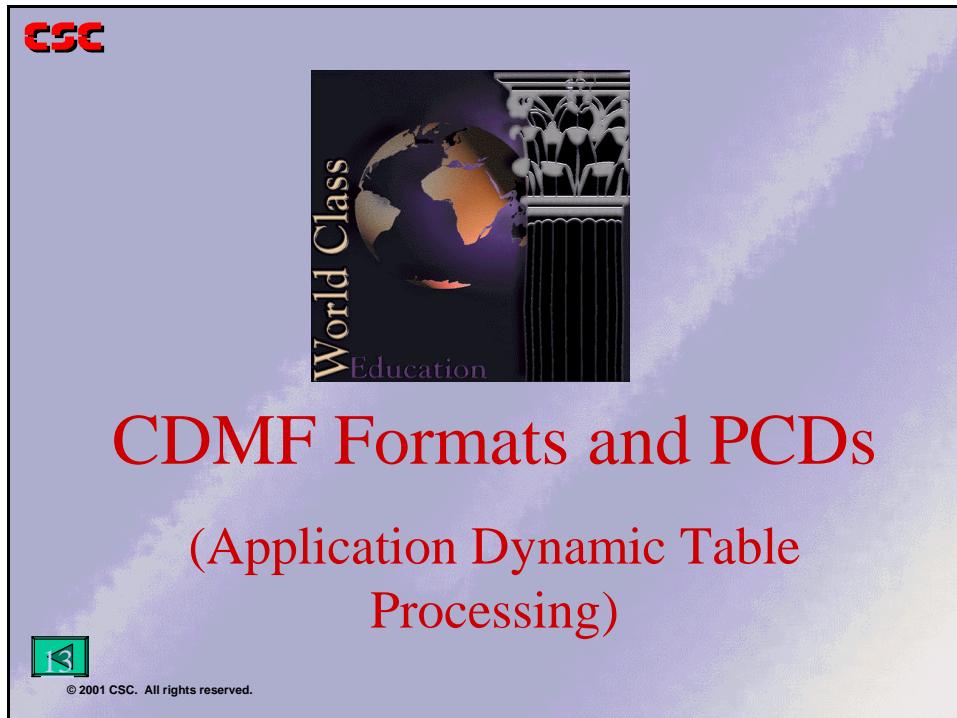
Summary



CDMF Formats and PCD Formats

14

Purpose



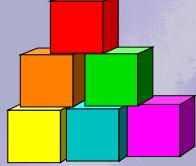
Notes:



Topics



Topics



- ▶ Process Control Data
- ▶ CDMF data base
(format, item, target
data group)
- ▶ CDMF key
- ▶ Effective dating
- ▶ Format definition
- ▶ CDMF actions and
PF keys
- ▶ Defaulting concepts
- ▶ Accessing CDMF
- ▶ Items converted
from the PCD
system
- ▶ Accessing PCD

© 2000 CSC. All rights reserved.

Notes:



Objectives

CSC

Objectives



To become familiar with:

- PCDs within Hogan's Systems
- Major components of CDMF data base
- Key fields for CDMF formats - new and converted PCD
- Effective-dated series
- Format definition and three key fields on definition
- FMT and NXTE performance and PF key assignment
- Item and format level defaulting
- CDMF format access coding
- PCD and CDMF format differences
- Converted PCD access coding

© 2000 CSC. All rights reserved.

1. To list two examples of the use of Process Control Data within HOGAN Systems
2. To describe the three major components of the CDMF data base
3. To list the key fields for CDMF formats—new and converted PCD
4. To give an example of an effective-dated series
5. To explain the purpose of a format definition and identify three key fields on the definition
6. To explain what the actions FMT and NXTE perform and give their corresponding PF key assignments
7. To distinguish between item and format level defaulting
8. To code the access of a CDMF format
9. To identify three differences between a converted PCD and CDMF format
10. To code the access of a converted PCD.



Process Control Data

Process Control Data Introduction



Process Control Data

■ System Variables

- Interest Rates
- Location Codes
- Fees

■ Data Stored on CDMF

- Formats
- PCD

© 2000 CSC. All rights reserved.

Traditionally, system variables like interest rates, location codes and a host of others are stored in copybooks. They are included at compile time or read in as internal tables at execution time. Many application systems use indexed files to store the data. Both of these methods require special handling or additional programming when changes are made. If indexed files are used, special programs need to be written to add, change, and delete data on the file along with the utility processes to load, backup, and restore the data.

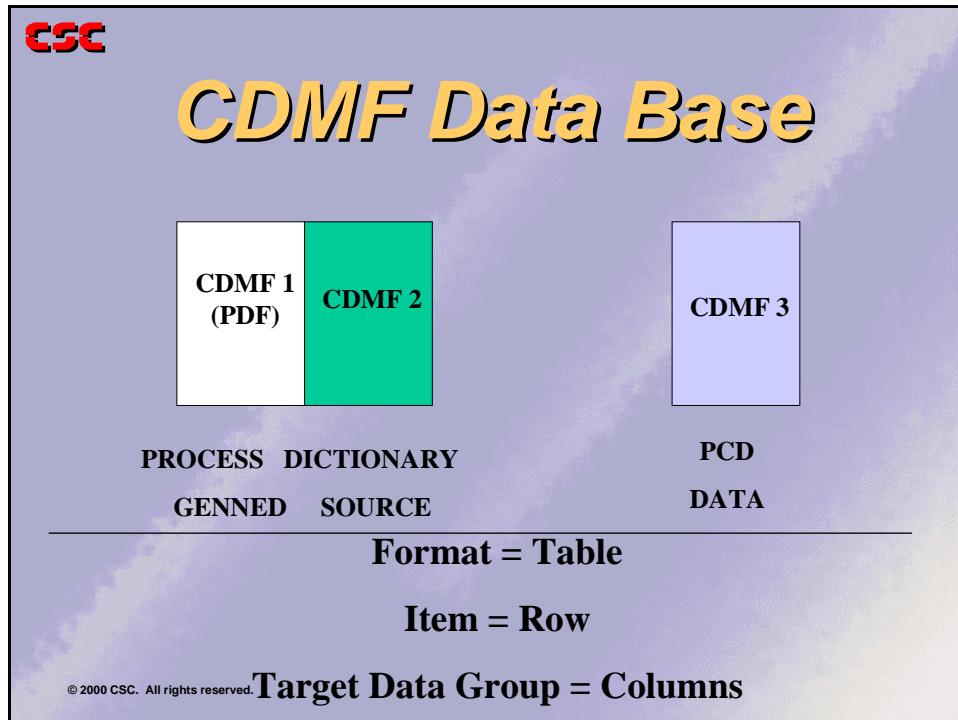
Keeping PCDs on the CDMF data base allows application processing criteria to be stored in a location external to the application programs. Changes to the data do not force a recompile of the application program(s) and PCDs can be shared by many applications. All of the processes needed to add, change, delete, and access PCDs are provided by the Umbrella System.

Process Control Data, or data that controls processing, is stored on the CDMF data base. Prior to the CDMF release of the Umbrella System, the data was stored on separate data bases used only for PCD processing. Part of the original CDMF release installation was to convert the old PCDs to the new CDMF technology. These converted PCDs are also referred to as simulated PCDs. New PCDs created today do not need to be converted. Both kinds of PCDs are stored on the CDMF data base and each has its own access method.



CDMF Concepts and Terminology

CDMF Components



Format

FORMAT = TABLE = FILE = PCDID = GROUP

In CDMF terminology groups of like records are called formats. The characteristics of each format are defined in a format definition.

Item

ITEM = ROW = ENTRY = RECORD = SET & ELEMENT

Each record in the format is called an item. All items in a format have the same data record layout. An items structure will be defined by a Data Group Definition, called Target Data Group.

FIELD = COLUMN = ELEMENTARY DATA

Each item is a collection of fields. Each field has its own unique size and structure, and shares the identical location across all the items.



Umbrella Programming

CDMF Concepts and Terminology

Target Data Group

The items in each format are associated with a specific data group known as the target data group.

When an item is needed from a table or format, keys to the table and item are set up so that the retrieval can be made. The data is returned to the application program in the target data group. When items need to be updated, the data is assumed to be in the target data group.

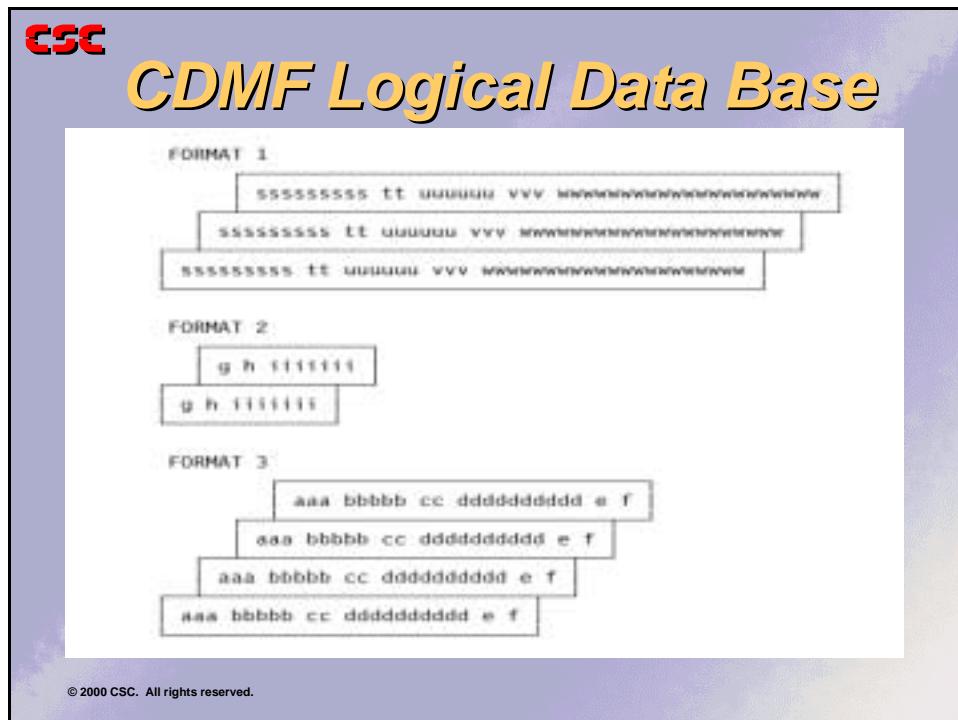


Figure 14-1. CDMF Logical Data Base

Target data group defines the data fields contained in the item.

Each item has the same data structure, but contains different data values.

Many different formats are stored on the CDMF data base.



CDMF/PCD Key

The logical key of each CDMF item is composed of four parts:

FORMAT ID	A fullword binary number assigned to uniquely identify the format.
COMPANY ID	A halfword binary number used to relate a format to a particular company. The default company is identified by DFLT.
ITEM KEY	The item key is user-defined. The length of the key is defined on the format definition and may be up to 244 positions in length. The Change Control System has a restriction on the item key length of 64 bytes. Because Hogan Systems uses change control to deliver its releases, delivered PCDs have item keys of 64 bytes or less.
EFFECTIVE DATE	The item key is located after the action and result fields in the target data group.

Notes:



Effective Dating Concepts

CSC

Effective Dating Concepts

- Effective date part of key
- Effective series
- Expiration date assigned

© 2000 CSC. All rights reserved.

Each item on the CDMF data base has an effective date associated with it. The effective date is used to allow CDMF items that otherwise have identical keys to exist.

Items that have the same key except for different effective dates are said to be an effective series. The date range for one item is called an effective period.

The effective period of an item is bounded by an effective date and an expiration date. The effective date is entered at the time a CDMF item is created. The expiration date is assigned automatically by CDMF based on the dates of other items in the effective series. It is adjusted automatically as new items are created. Thus, effective ranges will not overlap or have a gap.

Whenever items with identical keys, except for different effective periods, are requested using the NXT action, the item with the highest effective period will be returned first. This is due to the order in which the items are stored on the data base. Effective dates are in 1's compliment form, causing the highest key to be stored first in the series.

If no effective date is entered, the default is the current processing date. For retrieval, normally the effective date is allowed to default. When creating an item the effective date should typically be input.



Defining a Format

Steps to Define a Format

A format is a collection of items, each of which has the same layout but contains unique data. When defining a new format to the CDMF data base, certain Process Dictionary components must be established. They are:

- CDMF Format Definition
- Data Group Definition—target data group
- Map Definition—for online maintenance.

The following discussion is a general overview of the screens and fields used in defining a format. The screens and Process Dictionary entries for format 47192 will be used as examples during the discussion.

Format Definition Screen

The purpose of the format definition is to identify the format processing components to PEM such as:

- Format ID
- CDMF Data Base or DB2 Table storing the items
- Target data group and key length
- Map ID for the display if the items are to be maintained online.

Format ID is a fullword field. This field is required and must be unique within Company ID.

The format mnemonic is an alphanumeric field that is required. Formats can be retrieved by mnemonic sequence by using the command BY MNEM and then using either NXT or INQ command.

Company number may be 1 through 65535, or the value of DFLT, which is internally represented as 65535. The use of company specific formats allows unique items for each company processed.

Format description field is for documentation.



Umbrella Programming

Defining a Format

The CDMF Database ID or Relational DBID indicates where the items in the format will reside. CDMF3 is the Data Base ID used to contain application related formats (PCDs). CDMF1 and CDMF2 are reserved for Process Dictionary entries.

SQL Activity is used to specify the Data Base Service Module (DBSM) to be used to access the Relational DBID.

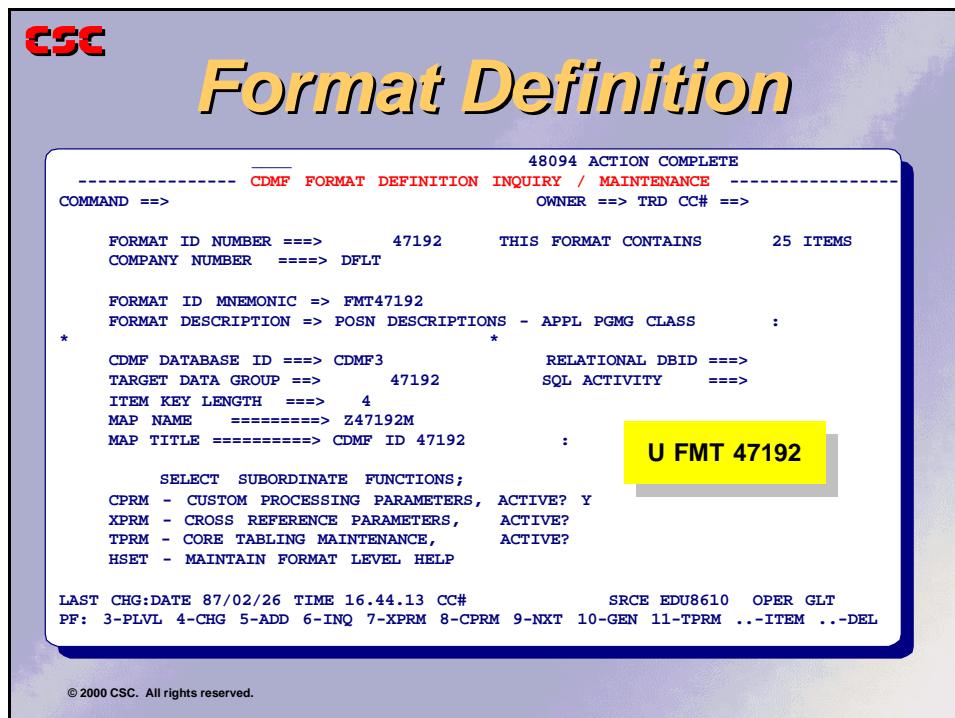
Target data group is a required field. The target data group provides addressability to the data in each format item and describes the layout of the data. This field will support fullword data group IDs.

Item key length is a required field. It should not include the four bytes of action and result within the data group.

Map name should contain a Map ID if the format items are to be maintained online.

The map title will be displayed with the map screen during online maintenance.

The format definition base screen for format 47192 can be obtained with the command **U FMT 47192**.



Placing an S in the 1-byte field to the left of the TARGET DATA GROUP field and pressing the PF6 key will cause the Data Group Definition to be displayed for the Target Data Group ID.

CSC

Select Target Data Group

```
----- 48094 ACTION COMPLETE
----- CDMF FORMAT DEFINITION INQUIRY / MAINTENANCE -----
COMMAND ==> OWNER ==> TRD CC# ==>

FORMAT ID NUMBER ===> 47192      THIS FORMAT CONTAINS      25 ITEMS
COMPANY NUMBER ===> DFLT

FORMAT ID MNEMONIC => FMT47192
FORMAT DESCRIPTION => POSN DESCRIPTIONS - APPL PGMG CLASS      :
* 
CDMF DATABASE ID ===> CDMF3          RELATIONAL DBID ===>
S TARGET DATA GROUP ==> 47192          SQL ACTIVITY      ===>
ITEM KEY LENGTH ===> 4
MAP NAME ======> Z47192M
MAP TITLE ======> CDMF ID 47192      :

SELECT SUBORDINATE FUNCTIONS;
CPRM - CUSTOM PROCESSING PARAMETERS, ACTIVE? Y
XPRM - CROSS REFERENCE PARAMETERS, ACTIVE?
TPRM - CORE TABLING MAINTENANCE, ACTIVE?
HSET - MAINTAIN FORMAT LEVEL HELP

LAST CHG:DATE 87/02/26 TIME 16.44.13 CC#           SRCE EDU8610 OPER GLT
PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-XPRM 8-CPRM 9-NXT 10-GEN 11-TPRM ...-ITEM ..-DEL

© 2000 CSC. All rights reserved.
```

Notes:



Umbrella Programming

Defining a Format

The data group definition screen is returned.

The screenshot shows a terminal window titled "Data Group Definition 47192". The title bar has the CSC logo. The window contains several lines of text defining a data group. At the top, there is a header with fields like "ACTION COMPLETE", "PROCESS", "DICTIONARY", "DATA", "GROUP", "DEFINITION", and "SETUP". Below this, a command "COMMAND ==> INQ" is shown, along with an owner "OWNER ==> TRD CC# ==>". The main body of the screen displays parameters for the data group, including its ID (47192), dictionary ID (Z47192D), and effective date (78/01/01). It also specifies the data group's description (PCD POSITION DESC'S), length (48), initialize status (FLD), and initialization character (X' 00'). The data group type is listed as "TEMPORARY" with "CSA ELIGIBLE ==>". A copy book name (COBOL) and copy number (Z47192D) are also mentioned. A section for new data group keys follows, with fields for "DATA GROUP ID==>" and "ALIAS==>". At the bottom, there is a footer with copyright information: "© 2000 CSC. All rights reserved." and a series of command codes: LAST CHG:DATE 92/07/20 TIME 16.26.59 CC# 1001 SRCE WOOF OPER GLT, LAST GEN:DATE 92/08/06 TIME 6.27.01 CC# 1001 SRCE WOOF OPER GLT, PF: 2-XREF 4-CHG 5-ADD 6-INQ 7-BROWSE 8-EDIT 9-NXT 10-GEN 11-CPY ..-DEL.

Press the PF7 key to browse the fields of the data group or press the PF3 key to return to the Format Definition Screen.

Notes:



CSC

Data Group Field Definition

```

ACTION COMPLETE
----- DATA GROUP DEFINITION INQUIRY / MAINTENANCE -----
COMMAND INPUT ==> HP                                     CC# ==> 0
DATA GROUP ID ==> 0000047192 EFP DATE==> 78/01/01 COLS: 9 80 MODE ==> BRWS
***** -1-----+---2-----+---3-----+---4-----+---5-----+---6-----+---7-----+---8
000001 ****
000002          Z 4 7 1 9 2 D
000003
000004 THIS DG DEFINES THE POSITION DESCRIPTION RECORD WHICH IS *
000005 THE TARGET DATA GROUP FOR CDMF FORMAT 47192 USED IN THE *
000006 APPLICATIONS PROGRAMMING CLASS. THE KEY IS COMPRISED OF *
000007 THE JOB CLASS AND THE POSITION CODE (BOTH FOUND IN *
000008 DG 47120 OF THE EMP DATA BASE). THE DESCRIPTION IS USED *
000009 IN THE REPORT GENERATED BY PROGRAM 478XX.
000010 *** DGID = 47192 ***
000011 ****
000012 1 POSITION-DESCRIPTION-RECORD.
000014 05 POSITION-CODE-ACTION      PIC XX.
000015 05 POSITION-CODE-RESULT     PIC XX.
000016 05 POSITION-JOB-CLASS       PIC XX.           TQE47921
000018 05 POSITION-CODE           PIC XX.           TQE47922
000020 05 POSITION-DESCRIPTION    PIC X(40).        TQE47923
***** -1-----+---2-----+---3-----+---4-----+---5-----+---6-----+---7-----+---8
PF: 4-SAVE 5-HP 6-BROWSE 7-SB 8-SF 9-EDIT 10-SL 11-SR ...EID ...LTYP ...CANCEL

```

PF7 or
BROWSE
Command

© 2000 CSC. All rights reserved.

Press the PF3 key to return to the Format Definition Screen.

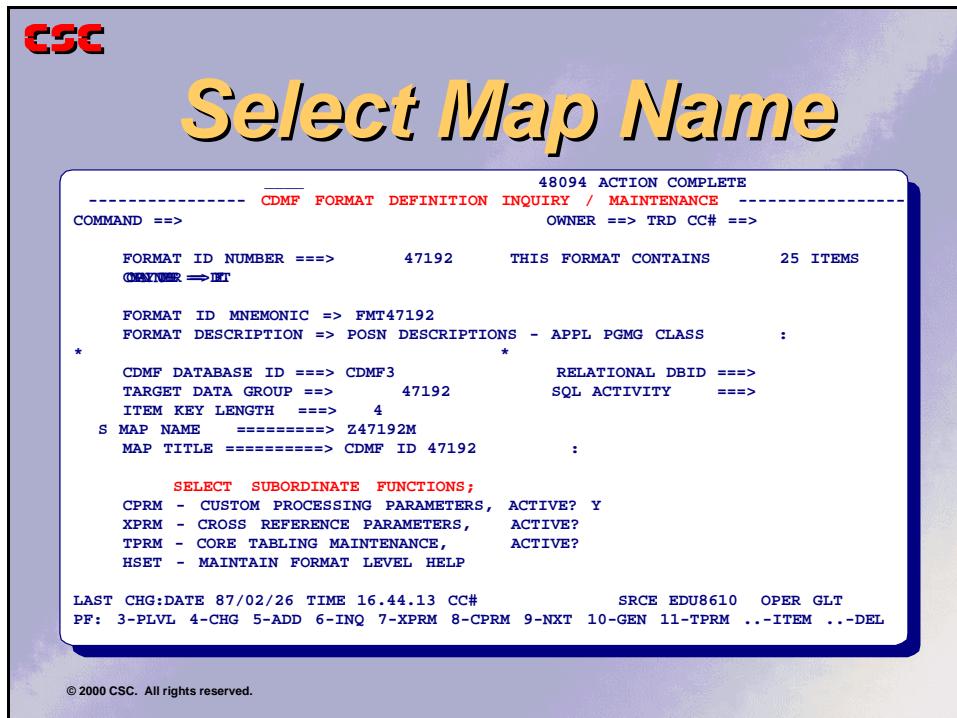
Notes:



Umbrella Programming

Defining a Format

Placing an S in the 1-byte field to the left of the MAP NAME field and pressing the PF6 key will cause the Map Definition to be displayed for the Map ID.



Notes:



The map definition for a format is the same as any other map definition. If the format is going to be maintained by batch programs only, a map definition is not needed.

CSC

Map Definition Z47192M

ACTION COMPLETE									
----- MAP DEFINITION BASE SCREEN -----									
COMMAND ==>	NXT	OWNER ==>	TRD CC# ==>						
LANGUAGE =>	ENU								
MAP NAME =>	Z47192M	DEVICE =>	A	CO GROUP =>	ALL	EFF DATE =>	78/01/01		
DESCRIPTION =====> JOB CLASS/DESCRIPTION FORMAT :									
LINKNAME =====> Z47192MM									
TYPE (I,O,U,S) =====>	U	TRANCODE DYNAMIC? =====>	N						
TRANCODE =====>	U140	TRANCODE PROTECTED? =====>	N						
EXTENDED ATTRIBUTES? ==>	N	NON-PEM MAP? =====>	N						
MAP TYPE (DOC SCR) =====>	SCR	MAXIMUM ROWS =>	24	MAXIMUM CLMS =>	80				
EXTRA FPS DATA GROUPS									
*	DGID	*	DGID	*	DGID	*	DGID	*	DGID
NEW MAP KEY FOR COPY:									
MAP NAME =>	DEVICE =>	CO GROUP =>	EFF DATE =>						
LANGUAGE =>									
LAST CHG:DATE 92/07/20 TIME 12.44.07 CC#	1002	SRCE F10F	OPER GLT						
LAST GEN:DATE 92/08/06 TIME 11.40.36 CC#	SRCE GXB6	OPER							
PF: 4-CHG 5-ADD 6-INQ 7-BROWSE 8-EDIT 9-NEXT 10-GEN 11-DSP ...DOC	...CPY	...-GRP							

© 2000 CSC. All rights reserved.

Note: The transaction code is U140 on the "Map Definition Base Screen". The trancode for all item maintenance screens, regardless of which format is being processed, should be U140.

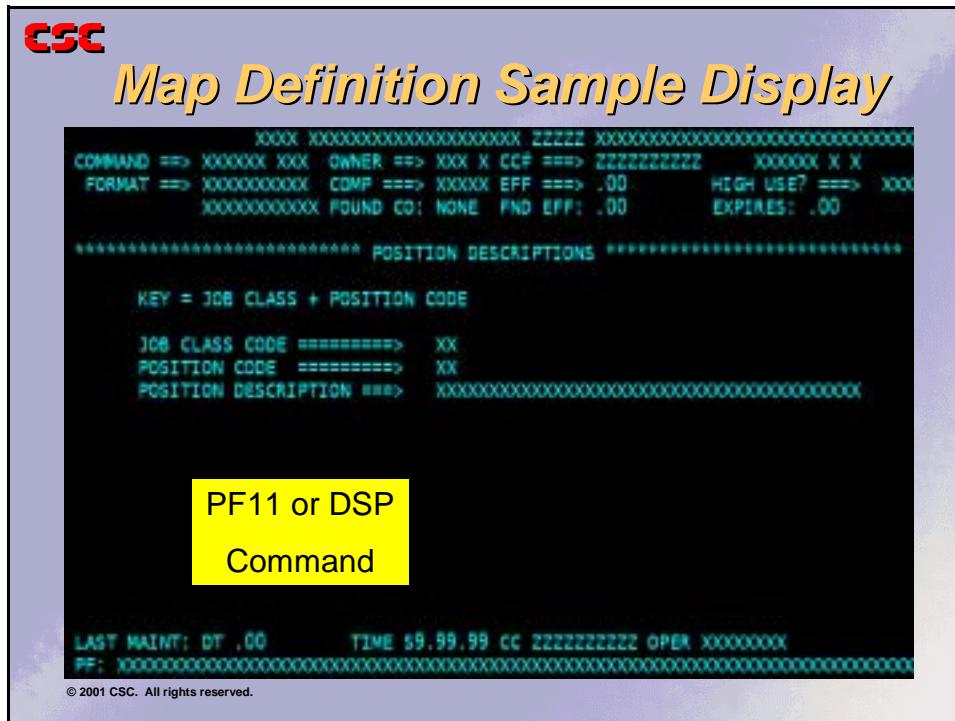
Notes:



Umbrella Programming

Defining a Format

Using the DSP command, the map is displayed below. The first four lines are the standard item maintenance FPS header, which is included map U48140E.



CSC
Map Definition Sample Display

```
XXXX XXXXXXXXXXXXXXXXXXXX ZZZZ XXXXXXXXXXXXXXXXXXXXXXXXX  
COMMAND ==> XXXXXX XXX OWNER ==> XXX X CCF ==> ZZZZZZZZ XXXXXXX X X  
FORMAT ==> XXXXXXXXXX COMP ==> XXXXX EFF ==> .00 HIGH USE? ==> XXX  
XXXXXXXXXXXX FOUND CO: NONE FND EFF: .00 EXPIRES: .00  
***** POSITION DESCRIPTIONS *****  
KEY = JOB CLASS + POSITION CODE  
JOB CLASS CODE ======> XX  
POSITION CODE ======> XX  
POSITION DESCRIPTION ===> XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  
LAST MAINT: DT ,00 TIME 59.99.99 CC ZZZZZZZZ OPEN XXXXXXXX  
PF: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  
© 2001 CSC. All rights reserved.
```

PF11 or DSP
Command

The last two lines are actually two include maps. The include map U48140F contains the last maintenance information fields. The include map U48100H, the last line on the map, is the PF key usage documentation line. All other lines are programmer designed and are used to display the item data from the target data group.

Notes:



Note: The positioning of the three include maps. Remember, an include map takes the row from the base map, and the column from the included map itself.

Each of the modifiable fields shown on the map definition are associated with one field in the target data group.

The screenshot shows a window titled "Map Definition Design" with the CSC logo at the top left. The main area displays a map definition script with several lines of code. A yellow callout box highlights the text "PF7 or BROWSE Command".

```

ACTION COMPLETE
ORIGIN=( 1      , 1      ) ---- MAP DEFINITION SCREEN ----- MAX-ORIGIN=( 24      , 80      )
COMMAND   ROW CLM LEN (THRU ROW) TO/AT ROW CLM   LANGUAGE==> ENU   MODE==> BRWS
-----+---1-----+---2-----+---3-----+---4-----+---5-----+---6-----+---7-----+
IU48140E                                     001
002
003
004
005
006 ***** POSITION DESCRIPTIONS *****
007
008     "KEY = JOB CLASS + POSITION CODE"
009
010    "JOB CLASS CODE ======>"  XX
011    "POSITION CODE ======>"  XX
012    "POSITION DESCRIPTION ==>"  XXXXXXXXXXXXXXXXX.....XXXXXXX
013
014
015
016
-----+---1-----+---2-----+---3-----+---4-----+---5-----+---6-----+---7-----+
PF: 3-PLVL 6-FDM2 7-SB 8-SF 9-FDM 10-SL 11-SR

```

© 2000 CSC. All rights reserved.

Notes:



Umbrella Programming

Defining a Format

Under the headings SELECT SUBORDINATE FUNCTIONS are the four labels: CUSTOM PROCESSING PARAMETERS, CROSS REFERENCE PARAMETERS, CORE TABLING MAINTENANCE, and MAINTAIN FORMAT LEVEL HELP. A 'Y' next to ACTIVE? indicates there are parameters for this function.

Whenever a format is created, custom processing parameters are automatically created for the format. To view or modify the default parameters, 1) Press the PF8 key, or 2) Enter CPRM in the COMMAND field, or 3) place an S to the left of the function and press the PF6 key.

The screenshot shows a terminal window with the CSC logo at the top. The title bar reads "Format Processing Features". The main area displays a successful format definition inquiry. The output includes:

```
ACTION SUCCESSFUL
----- CDMF FORMAT DEFINITION INQUIRY / MAINTENANCE -----
COMMAND ==> INQ OWNER ==> TRD CC# ==>
FORMAT ID NUMBER ==> 47192 THIS FORMAT CONTAINS 25 ITEMS
COMPANY NUMBER ==> DFLT

FORMAT ID MNEMONIC => FMT47192
FORMAT DESCRIPTION => POSN DESCRIPTIONS - APPL PGMG CLASS :
*
* CDMF DATABASE ID ==> CDMF3 RELATIONAL DBID ====
TARGET DATA GROUP ==> 47192 SQL ACTIVITY ====
ITEM KEY LENGTH ==> 4
MAP NAME =====> Z47192M
MAP TITLE =====> CDMF ID 47192 :

SELECT SUBORDINATE FUNCTIONS;
S CPRM - CUSTOM PROCESSING PARAMETERS, ACTIVE? Y
XPRM - CROSS REFERENCE PARAMETERS, ACTIVE?
TPRM - CORE TABLING MAINTENANCE, ACTIVE?
HSET - MAINTAIN FORMAT LEVEL HELP

LAST CHG:DATE 87/02/26 TIME 16.44.13 CC# SRCE EDU8610 OPER GLT
PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-XPRM 8-CPRM 9-NXT 10-GEN 11-TPRM ..-ITEM ..-DEL
```

At the bottom of the window, it says "© 2000 CSC. All rights reserved."

Notes:



Format Custom Processing Parameters

The custom parameters are divided into functional groups. These groups are:

- Online Maintenance Options
- Processing Options
- RCS Processing Data.

Parameters under Online Maintenance are invoked whenever items are accessed using standard online item maintenance. The parameters under Processing Options are invoked whenever accessed by a program or standard item maintenance. The RCS Processing data is used by the Release Control System.

```

48000 ACTION COMPLETE
--- CDMF CUSTOM PROCESSING PARAMETERS FOR FORMAT      47192 , COMPANY DFLT ---
COMMAND ==> CPRM                                         CC# ==>

ONLINE MAINTENANCE OPTIONS:
POST-DEBLOCK ACT =>
USER ACTION 1 ===>
USER ACTION 2 ===>
EFS TABLE NAME ===> X4719200
OWNER APP REQD? ==> NO

PROCESSING OPTIONS:
COMPANY DEFAULTING LEVEL =====> FORMAT
STORE CDMF CONTROL BLOCK ? ===> NO
ITEMS FIXED OR VARIABLE LTH? => FIXED
KEY COMPRESSION ROUTINE =====>
LOG FOR AUDIT REPORTING? =====> YES
DATABASE TECHNOLOGY (H/R) =====> H

RCS PROCESSING DATA:
PROCESS DICTIONARY? (Y/N) =====> N
GROUP FORMAT TYPE (P/S) =====>
COMMON KEY LENGTH =====>

PF: 3-PLVL 4-CHG 6-INQ

```

© 2000 CSC. All rights reserved.

Notes:

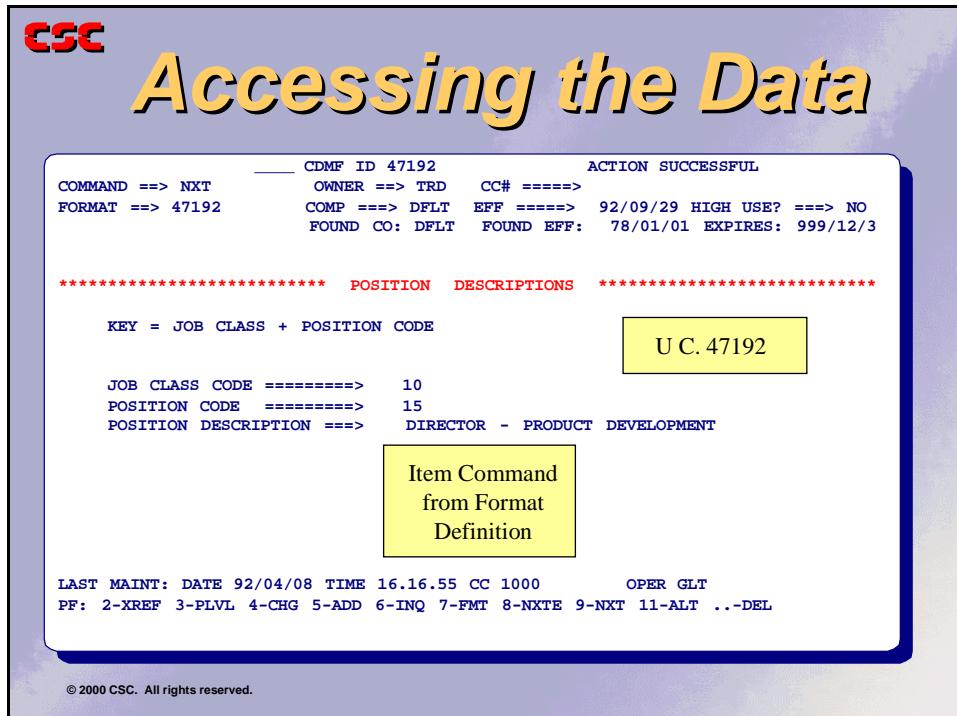


Umbrella Programming

Defining a Format

Accessing the Data

To access the data of format 47192, from a cleared screen, enter U PCD 47192. To access the data from the "CDMF Format Definition Inquiry / Maintenance" screen, enter ITEM in the COMMAND field and press ENTER.



- Items are maintained online
- An individual item is displayed.
- Items are keyed.

Notes:



Commonly Used Actions and Function Keys for CDMF Items

ACTION	PF KEY	DESCRIPTION
XREF	PF2	Display existing cross-references to this item.
CHG	PF4	Change data in an existing item.
ADD	PF5	Add a new item to the file.
INQ	PF6	Retrieve the item matching the item key input on the screen from the data base and present it.
FMT	PF7	Display the format definition for this item.
NXTE	PF8	Functions the same as NXT except that effective date is taken into account. Items will be retrieved only from the effective period first retrieved.
NXT	PF9	Retrieve the item with the next higher key than the one input on the screen and present it.
ALT	PF11	Display the next alternate format item screen.
ENDEFF	---	End the effective period of an existing item. (The item becomes invalid as of the date specified in the effective date field.)
DEL	---	Delete an existing item.

Notes:



Defaulting Concepts

CDMF supports two levels of defaulting in the selection of the items. The type of defaulting to be used is indicated on the custom parameters screen of the format definition. The two types of defaulting are:

- Format level defaulting (default)
- Item level defaulting.

Format Level Defaulting

HOGAN provides a company called DFLT (default) to meet the more common processing requirements.

Should the processing requirements of a company be different from those supplied for company DFLT, a company format may be created comprised of CDMF items specific to that company.

Format level defaulting should be used when there are large numbers of items that are different between companies.

Search criteria for a given format:

csc *Defaulting Concepts*

■ Format Level Defaulting

Action is requested on an item for a specific company
If the format table exists for the specified company
Then ONLY that table is searched:
 If the item row is in the company format table then
 the item is found and returned,
 Else the item is not found,
Else the DFLT format table is searched:
 If the item is defined in the DFLT format table then
 the item is found and returned,
 Else the item is not found.

© 2000 CSC. All rights reserved.



Format Defaulting Examples

Let's examine some examples of search and retrieval with "format" defaulting as illustrated below.

FORMAT 2468		FORMAT 2468		FORMAT 2468		FORMAT 2468	
COMPANY DFLT		COMPANY 12		COMPANY 56		COMPANY 917	
FORMAT		FORMAT		FORMAT		FORMAT	
KEY	DATA	KEY	DATA	KEY	DATA	KEY	DATA
AA	11	AA	21	BB	57	DD	93
BB	22	BB	31	CC	57	NN	92
CC	33	DD	41	HH	34	PP	90
DD	44	FF	51	JJ	39		
				KK	55		

FORMAT	SEARCH VALUES		RESULT OF SEARCH	
	COMPANY	ITEM KEY	COMPANY	DATA
2468	12	AA	12	21
2468	86	CC	DFLT	33
2468	56	AA	NOT FOUND	CONDITION
2468	DFLT	CC	DFLT	33
2468	357	FF	NOT FOUND	CONDITION
2468	917	PP	917	90

Notes:



Umbrella Programming

Defaulting Concepts

Item Level Defaulting

Item level defaulting is best used when there are few differences in the contents of items between companies. Only items that differ from those on the DFLT company need be created.

Using item level defaulting causes CDMF to view the format for a specified company and that of the DFLT company as if they were concatenated.

Search criteria for a given format:

The slide features a red 'CSC' logo at the top left. The title 'Defaulting Concepts' is prominently displayed in large yellow text. Below it, a section titled '■ Item Level Defaulting' is shown. A detailed algorithm for searching an item key is listed:

```
Action is requested on an item for a specific company.  
If the format table exists for the specified company  
then that table is searched:  
  If the item key is in the company format table then  
    the item is found and returned  
  Else the DFLT format table is searched:  
    If the item key is in the DFLT format table then  
      the item is found and returned  
    Else the item is not found.  
Else the DFLT format table is searched:  
  If the item key is in the DFLT format table,  
    the item is found and returned  
  Else the item is not found.
```

At the bottom of the slide, a copyright notice reads: © 2000 CSC. All rights reserved.

Notes:



Item Level Defaulting Examples

Examples of search and retrieval using item level defaulting are illustrated below.

FORMAT 2468		FORMAT 2468		FORMAT 2468		FORMAT 2468	
COMPANY DFLT		COMPANY 12		COMPANY 56		COMPANY 917	
FORMAT		ITEM		ITEM		ITEM	
KEY	DATA	KEY	DATA	KEY	DATA	KEY	DATA
AA	11	AA	21	BB	57	DD	93
BB	22	BB	31	CC	57	NN	92
CC	33	DD	41	HH	34	PP	90
DD	44	FF	51	JJ	39		
				KK	55		

SEARCH VALUES			RESULT OF SEARCH	
FORMAT	COMPANY	ITEM KEY	COMPANY	DATA
2468	12	AA	12	21
2468	86	CC	DFLT	33
2468	56	AA	DFLT	11
2468	357	FF	NOT FOUND CONDITION	
2468	56	KK	56	55
2468	DFLT	AA	DFLT	11

Notes:



Application Access to CDMF Formats

Accessing CDMF—New Technology

The CDMF Application Control Block (ACB) is used to communicate to the CDMF processor. These fields are located at the end of the TCB. To communicate with CDMF:

1. Fill out the following fields in the ACB.

CDMF-ACTION

Action for the request. Action codes are defined in COBOL copybook U48004D and in U48004E for ALC.

CDMF-FORMAT

Format ID (fullword binary) of the formatted being requested.

CDMF-COID

Company ID (halfword binary) of the formatted being requested.

CDMF-EFF-DATE

Optional (if zero defaults to TCB-EFFECTIVE-DATE)

2. Fill out the item key information in the target data group for the format being retrieved. The item key starts after target data group action and result fields.
3. Issue activity 48000, the link activity to CDMF processing.

Returned by the CDMF processor:

Information returned in the following ACB fields:

CDMF-RESULT

Result of the requested action. Result codes are defined in COBOL copybook U48024D and in U48024E for ALC.

CDMF-EXP-DATE

Last date for which the item last retrieved is valid.

CDMF-COID-FOUND

Company ID for which the item last retrieved is found.

CDMF-EFF-DATE-FOUND

Actual effective date of the item last retrieved.

Information returned in the target data group:

The entire data area from the requested item.



CDMF Application Control Block

The CDMF ACB is part of the UTCB.

```
*****
* CDMF APPLICATION CONTROL BLOCK
*****
05 CDMF-CONTROL-BLOCK.
 10 CDMF-ACTION          PIC XX.
 10 CDMF-RESULT          PIC XX.
 10 CDMF-KEY-FIELDS.
    15 CDMF-FORMAT        PIC XXXX.
    15 CDMF-COID          PIC XX.
    15 CDMF-EFF-DATE      PIC S9(7)  COMP-3.
 10 CDMF-EXP-DATE        PIC S9(7)  COMP-3.
 10 CDMF-COID-FOUND     PIC XX.
    88 CDMF-DEFAULT-COID-FOUND   VALUE HIGH-VALUES.
 10 CDMF-EFF-DATE-FOUND  PIC S9(7)  COMP-3.
 10 CDMF-HIGH-USE-FLAG   PIC X.
    88 CDMF-HIGH-USE-ITEM    VALUE 'Y'.
    88 CDMF-NON-PURGEABLE  VALUE 'P'.
*
***** Item ownership is always returned in CDMF-OWMER-APPLICATION.
***** Ownership may be retrieved and updated from data group 48007
***** If this flag is set to a Y. Ownership may be updated from
***** CDMF-OWNER-APPLICATION if this flag is set to a C.
 10 CDMF-OWNER-APP-FLAG  PIC X.
    88 CDMF-OWNER-APP-REQUEST  VALUE 'Y'.
    88 CDMF-OWNER-APP-IN-CTL-BLK  VALUE 'C'.
 10 CDMF-ITEM-LOCATION    PIC X.
    88 CDMF-ITEM-FOUND-IN-TABLE  VALUE 'Y'.
 10 FILLER                PIC X.
 10 CDMF-CC-NO            PIC X(4).
 10 CDMF-LAST-CHANGE-DATA.
    15 CDMF-LAST-CHANGE-DATE  PIC S9(7)  COMP-3.
    15 CDMF-LAST-CHANGE-TIME  PIC S9(7)  COMP-3.
    15 CDMF-LAST-CHANGE-CC-NO  PIC X(4).
    15 CDMF-LAST-CHANGE-SOURCE PIC X(8).
    15 CDMF-LAST-CHANGE-OPER   PIC X(8).
 10 CDMF-SECONDARY-KEY-ID  PIC X(4).
 10 CDMF-SUBSTITUTE-DGID  PIC X(4).
*
***** This field is for internal UMBRELLA use only.
 10 CDMF-RELEASE-CTL-DG-LEN  PIC XX.
*
***** This field is for internal UMBRELLA use only.
 10 CDMF-RELEASE-CTL-FLAGS  PIC X.
* ***** The item application ownership is always returned in this
***** field. This field may only be used in an update when
***** CDMF-OWNER-APP-FLAG is set to a C.
 10 CDMF-OWNER-APPLICATION  PIC X(3).
 10 FILLER                 PIC X(2).
```



Umbrella Programming

Application Access to CDMF Formats

CDMF Action Codes—Copybook U48004D

```
DATA GROUP          COPYBOOK NAME U48004D

*---* START OF U48004D *-----*
* CDMF ACTION LIST FOR 'CDMF CONTROL BLOCK' AND MISCELLANEOUS      *
* CONSTANTS FOR THE 'CDMF CROSS REFERENCE CONTROL BLOCK'.           *
*-----*
01  CDMF-ACTION-LIST.
    05  CDMF-CODE-VALUES.
    10  CDMF-NO-OP.
                                15  FILLER PIC S9(4) COMP VALUE +0000.
    10  CDMF-ADD.
                                15  FILLER PIC S9(4) COMP VALUE +0001.
    10  CDMF-ADD-DUMMY.
                                15  FILLER PIC S9(4) COMP VALUE +0002.
    10  CDMF-REPLACE.
                                15  FILLER PIC S9(4) COMP VALUE +0003.
    10  CDMF-CHANGE.
                                15  FILLER PIC S9(4) COMP VALUE +0003.
    10  CDMF-DELETE.
                                15  FILLER PIC S9(4) COMP VALUE +0004.
    10  CDMF-INQ.
                                15  FILLER PIC S9(4) COMP VALUE +0005.
    10  CDMF-NXT.
                                15  FILLER PIC S9(4) COMP VALUE +0006.
    10  CDMF-NXTE.
                                15  FILLER PIC S9(4) COMP VALUE +0007.
    10  CDMF-KGE.
                                15  FILLER PIC S9(4) COMP VALUE +0014.
    SKIP1
*-----*
*          ADVANCED CDMF RETRIEVAL ACTION CODES                      *
*-----*
10  CDMF-INQ-FROM-TABLE.
    15  FILLER PIC S9(4) COMP VALUE +0008.
10  CDMF-NXT-FROM-TABLE.
    15  FILLER PIC S9(4) COMP VALUE +0009.
10  CDMF-NXTE-FROM-TABLE.
    15  FILLER PIC S9(4) COMP VALUE +0010.
10  CDMF-KGE-FROM-TABLE.
    15  FILLER PIC S9(4) COMP VALUE +0015.
10  CDMF-INQ-FROM-DATABASE.
    15  FILLER PIC S9(4) COMP VALUE +0011.
10  CDMF-NXT-FROM-DATABASE.
    15  FILLER PIC S9(4) COMP VALUE +0012.
10  CDMF-NXTE-FROM-DATABASE.
    15  FILLER PIC S9(4) COMP VALUE +0013.
10  CDMF-KGE-FROM-DATABASE.
    15  FILLER PIC S9(4) COMP VALUE +0016.
    SKIP1
*-----*
*          ADVANCED CDMF LOGGING ACTION CODES                      *
*-----*
10  CDMF-LOG-BEFORE-CHANGE.
    15  FILLER PIC S9(4) COMP VALUE +0017.
10  CDMF-LOG-BEFORE-DELETE.
    15  FILLER PIC S9(4) COMP VALUE +0018.
```



Umbrella Programming

Application Access to CDMF Formats

```
10  CDMF-LOG-AFTER-ADD.  
      15  FILLER PIC S9(4) COMP VALUE +0019.  
10  CDMF-LOG-AFTER-CHANGE.  
      15  FILLER PIC S9(4) COMP VALUE +0020.  
      SKIP1  
*-----*  
*  CONSTANTS FOR THE 'CDMF CROSS REFERENCE CONTROL BLOCK', DATA  *  
*  GROUP 48008, COPYBOOK U48008D                                *  
*  NOTE: THE '66' LEVEL IS USED TO KEEP WORKING STORAGE TO A    *  
*        MINIMUM. THE 'RENAMES' CLAUSE HERE MEANS THAT THE ENTRY HAS *  
*        THE SAME EXACT VALUES AS THE ENTRY IT REDEFINES (RENAMES).  *  
*-----*  
*  
***** VALUES FOR U008-SOURCE-TYPE  
10  XREF-SOURCE-TYPE-HDB          PIC X     VALUE HIGH-VALUES.  
*  
***** VALUES FOR U008-ACTION (ACTION ON SOURCE ENTRY W/IN HOST)  
66  XREF-INQ-SOURCE-ENTRY      RENAMES CDMF-ADD.  
66  XREF-1ST-SOURCE-ENTRY      RENAMES CDMF-ADD-DUMMY.  
66  XREF-NXT-SOURCE-ENTRY      RENAMES CDMF-CHANGE.  
66  XREF-1ST-ENTRY-NXT-HOST   RENAMES CDMF-DELETE.  
66  XREF-KEY-GE-SOURCE-ENTRY   RENAMES CDMF-INQ.  
66  XREF-ADD-SOURCE-ENTRY     RENAMES CDMF-NXT.  
66  XREF-DEL-SOURCE-ENTRY     RENAMES CDMF-NXTE.  
66  XREF-LAST-CALL-DR-PHS2    RENAMES CDMF-INQ-FROM-TABLE.  
*  
***** VALUES FOR U008-RESULT  
66  XREF-ACTION-SUCCESSFUL   RENAMES CDMF-NO-OP.  
66  XREF-SOURCE-ENTRY-NOT-FOUND RENAMES CDMF-ADD.  
66  XREF-HOST-KEY-NOT-FOUND   RENAMES CDMF-ADD-DUMMY.  
66  XREF-END-OF-ENTRIES-FOR-HOST  
                           RENAMES CDMF-REPLACE.  
66  XREF-NO-SUCH-SOURCE-FMT   RENAMES CDMF-DELETE.  
66  XREF-NO-SUCH-HOST-FMT     RENAMES CDMF-INQ.  
66  XREF-DB-NOT-AVAILABLE     RENAMES CDMF-NXT-FROM-TABLE.  
  
***** VALUES FOR U008-FEEDBACK (FEEDBACK FROM SUCCESSFUL UPDATES)  
66  XREF-DEL-HOST-NOT-FOUND   RENAMES CDMF-ADD.  
66  XREF-DEL-NO-XRFS-ON-HOST  RENAMES CDMF-ADD-DUMMY.  
66  XREF-DEL-NO-MATCHING-ENTRY RENAMES CDMF-REPLACE.  
66  XREF-ADD-DUPE-ENTRY      RENAMES CDMF-DELETE.  
66  XREF-ADD-NO-EXP-DATA     RENAMES CDMF-INQ.  
66  XREF-ADD-DUMMY-ADDED     RENAMES CDMF-NXT.  
*  
*----* END OF U48004D  *-----*
```



Umbrella Programming

Application Access to CDMF Formats

CDMF Result Codes Copybook U48024D

```
DATA GROUP           COPYBOOK NAME U48024D

*---* START OF U48024D *---* CDMF RESULT LIST *-----*
*
01  CDMF-RESULT-LIST.
    05  CDMF-RESULT-CODES.
*
*   CDMF-NO-ERRORS
    10 FILLER PIC S9(9) COMP VALUE +00000.
*
*   CDMF-DB-NOT-AVAILABLE
    10 FILLER PIC S9(9) COMP VALUE +00009.
*
*   CDMF-INVALID-ACTION
    10 FILLER PIC S9(9) COMP VALUE +48001.
*
*   CDMF-ITEM-NOT-FOUND
    10 FILLER PIC S9(9) COMP VALUE +48002.
*
*   CDMF-FORMAT-NOT-FOUND
    10 FILLER PIC S9(9) COMP VALUE +48003.
*
*   CDMF-UNABLE-TO-ALLOC-DG
    10 FILLER PIC S9(9) COMP VALUE +48004.
*
*   CDMF-END-OF-FORMAT
    10 FILLER PIC S9(9) COMP VALUE +48005.
*
*   CDMF-DUPE-KEY-ON-ADD
    10 FILLER PIC S9(9) COMP VALUE +48006.
*
*   CDMF-INVALID-EFF-DATE
    10 FILLER PIC S9(9) COMP VALUE +48007.
*
*   CDMF-INVALID-APPL
    10 FILLER PIC S9(9) COMP VALUE +48008.
*
*   CDMF-SKEY-NOT-FOUND
    10 FILLER PIC S9(9) COMP VALUE +48009.
*
*   CDMF-SKEY-READ-ERR
    10 FILLER PIC S9(9) COMP VALUE +48010.
*
*   CDMF-INVALID-CC-NO
    10 FILLER PIC S9(9) COMP VALUE +48011.
*
*   CDMF-INVALID-FAMILY
    10 FILLER PIC S9(9) COMP VALUE +48012.
*
*   CDMF-NO-OWNER-CHANGE
    10 FILLER PIC S9(9) COMP VALUE +48013.
*
*   CDMF-SECURITY-VIOLATION
    10 FILLER PIC S9(9) COMP VALUE +48030.
*
*   CDMF-SECURITY-INACTIVE
    10 FILLER PIC S9(9) COMP VALUE +48031.
*
*   CDMF-DUMMY-REC-FOUND
    10 FILLER PIC S9(9) COMP VALUE +48099.
*
*   CDMF-CCNR-MISSING
    10 FILLER PIC S9(9) COMP VALUE +48371.
*
*   CDMF-CCNR-CLOSED
    10 FILLER PIC S9(9) COMP VALUE +48372.
    05 FILLER
        REDEFINES CDMF-RESULT-CODES.
        10 FILLER PIC XX.
    10 CDMF-NO-ERRORS          PIC XX.
    10 CDMF-DB-NOT-AVAILABLE   PIC XX.
    10 CDMF-INVALID-ACTION    PIC XX.
    10 CDMF-ITEM-NOT-FOUND    PIC XX.
```



Umbrella Programming

Application Access to CDMF Formats

10 CDMF-FORMAT-NOT-FOUND	PIC XX.	
10 CDMF-UNABLE-TO-ALLOC-DG	PIC XX.	10 FILLER PIC XX.
10 CDMF-END-OF-FORMAT	PIC XX.	10 FILLER PIC XX.
10 CDMF-DUPE-KEY-ON-ADD	PIC XX.	10 FILLER PIC XX.
10 CDMF-INVALID-EFF-DATE	PIC XX.	10 FILLER PIC XX.
10 CDMF-INVALID-APPL	PIC XX.	10 FILLER PIC XX.
10 CDMF-SKEY-NOT-FOUND	PIC XX.	10 FILLER PIC XX.
10 CDMF-SKEY-READ-ERR	PIC XX.	10 FILLER PIC XX.
10 CDMF-INVALID-CC-NO	PIC XX.	10 FILLER PIC XX.
10 CDMF-INVALID-FAMILY	PIC XX.	10 FILLER PIC XX.
10 CDMF-NO-OWNER-CHANGE	PIC XX.	10 FILLER PIC XX.
10 CDMF-SECURITY-VIOLATION	PIC XX.	10 FILLER PIC XX.
10 CDMF-SECURITY-INACTIVE	PIC XX.	10 FILLER PIC XX.
10 CDMF-DUMMY-REC-FOUND	PIC XX.	10 FILLER PIC XX.
10 CDMF-CCNR-MISSING	PIC XX.	10 FILLER PIC XX.
10 CDMF-CCNR-CLOSED	PIC XX.	10 FILLER PIC XX.

Notes:



Problem Specifications—CDMF Access



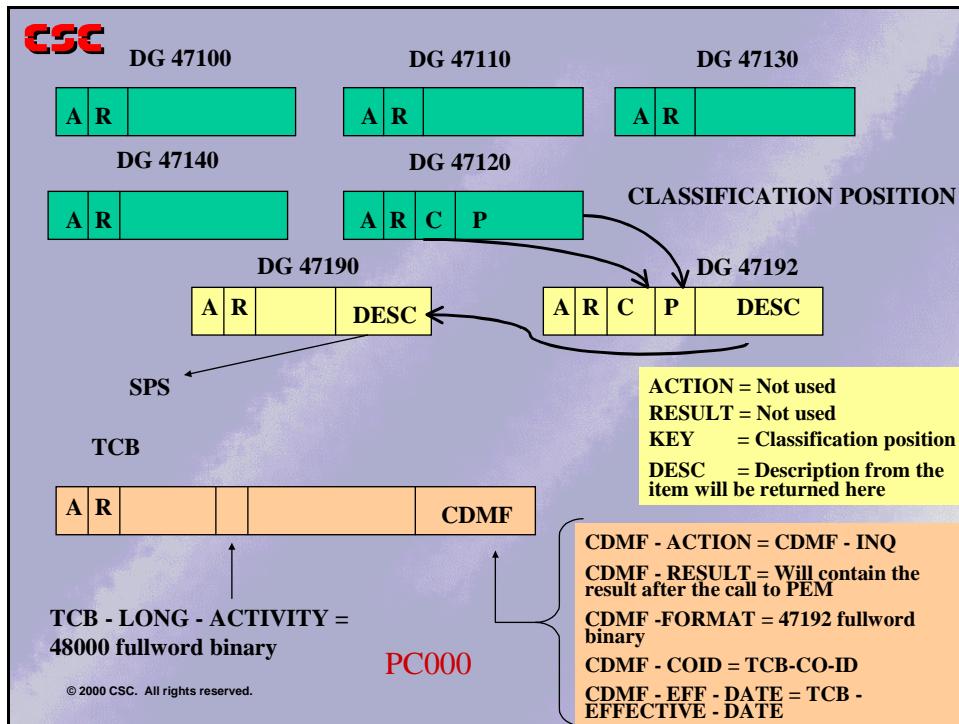
1. The purpose of this phase is to read a CDMF format—47192 to obtain the corresponding position description for the employee's job classification and position. Data group 47120 contains the job classification and position codes needed to build the item key. Data group 47120 is part of the EMP data base. Make sure your data base activity is accessing DG 47120.
2. The item key of the CDMF format 47192 consists of a 2-byte JOB CLASSIFICATION CODE and a 2-byte POSITION CODE.

CDMF action and result code values are defined by the COBOL copybooks U48004D and U48024D respectively. These have been included in your skeleton program.

3. Use activity 48000 to link to CDMF. The link activity is defined in the Process Dictionary. Your copy of the class program contains the binary defined field for the link to CDMF PCD processing.
4. Take the following actions based on the CDMF result code returned after having invoked CDMF.
 - If the CDMF result is not equal to low values, take the following actions based on the CDMF result code:
 - If the result indicates NO FORMAT FOUND, move the message NO FORMAT FOUND to the position description field in data group 47190.
 - If the result indicates NO ITEM FOUND, move the message NO ITEM FOUND to the position description field in data group 47190.
 - If the result indicates other errors, the program should execute a dump and end.
 - The SPS program prints the position description from data group 47190. Your routine will need to move the position description from DG 47192 to DG 47190.
5. Use the owner application and change control created in the Change Control Exercise.



CDMF Access Diagram



Notes:



Umbrella Programming

How Old and New Formats Differ

How Old and New Formats Differ

Items Converted From the PCD System

The following are examples of items converted from the Process Control Data System. Under old technology, the user key length was fixed at 11 bytes. If more than 11 bytes were needed for the item key or items needed to be grouped within user keys, an element key could be set up. The element key is also referred to as the recursive key.

The user key appears in the screen header display. If the PCD contains an element key, it will be identified in the user defined portion of the screen display by key data or similar notation.

Note this example of a converted PCD with its user key.

CSC How Old and New Formats Differ - 2005 Definition

DTS DATE CALC PCD		ACTION SUCCESSFUL				
COMMAND ==> INQ	OWNER ==> DTS	CC# =====>				
FORMAT ==> 2005	COMP ==>	1 EFF =====>	85/01/01 HIGH USE? ==> NO			
USER KEY=> DCB CONTROL FOUND CO: DFLLT	FOUND EFF:	1/01	EXPIRES: 999/12/31			
DCB DATE CALCULATION CONTROL		PCD-ID=2005				
NON-BUSINESS DAYS (MARK WITH X)						
SUN	MON	TUE	WED	THU	FRI	SAT
X						X
U C. 2005						
FIRST MONTH OF FISCAL YEAR 01 (BLANK OR ZERO = FISCAL YEAR SAME AS CALENDAR YEAR)						
LAST MAINT: DATE 85/05/01 TIME 10.53.06 CC				OPER		

© 2000 CSC. All rights reserved.

Notes:



2005 Format Definition

CSC

2005 Format Definition

```
48094 ACTION COMPLETE
----- CDMF FORMAT DEFINITION INQUIRY / MAINTENANCE -----
COMMAND ==> INQ           OWNER ==> DTS CC# ==>
FORMAT ID NUMBER ==>      2005      THIS FORMAT CONTAINS      1 ITEM
COMPANY NUMBER ==>        DFLT

FORMAT ID MNEMONIC => PCDID02005
FORMAT DESCRIPTION => DTS DATE CALCULATION CONTROL      :
*          *          *
CDMF DATABASE ID ==> CDMF3           RELATIONAL DBID ==>
TARGET DATA GROUP ==>      2005           SQL ACTIVITY      ==>
ITEM KEY LENGTH ==>       11
MAP NAME ==>              T58005M
MAP TITLE ==>              DTS DATE CALC PCD      :
SELECT SUBORDINATE FUNCTIONS;
CPRM - CUSTOM PROCESSING PARAMETERS, ACTIVE? Y
XPRM - CROSS REFERENCE PARAMETERS, ACTIVE?
TPRM - CORE TABLING MAINTENANCE, ACTIVE?
HSET - MAINTAIN FORMAT LEVEL HELP

LAST CHG:DATE 92/06/15 TIME 9.35.40 CC#           SRCE UMB130    OPER
PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-XPRM 8-CPRM 9-NXT 10-GEN 11-TPRM ..-ITEM ..-DEL

© 2000 CSC. All rights reserved.
```

Notes:

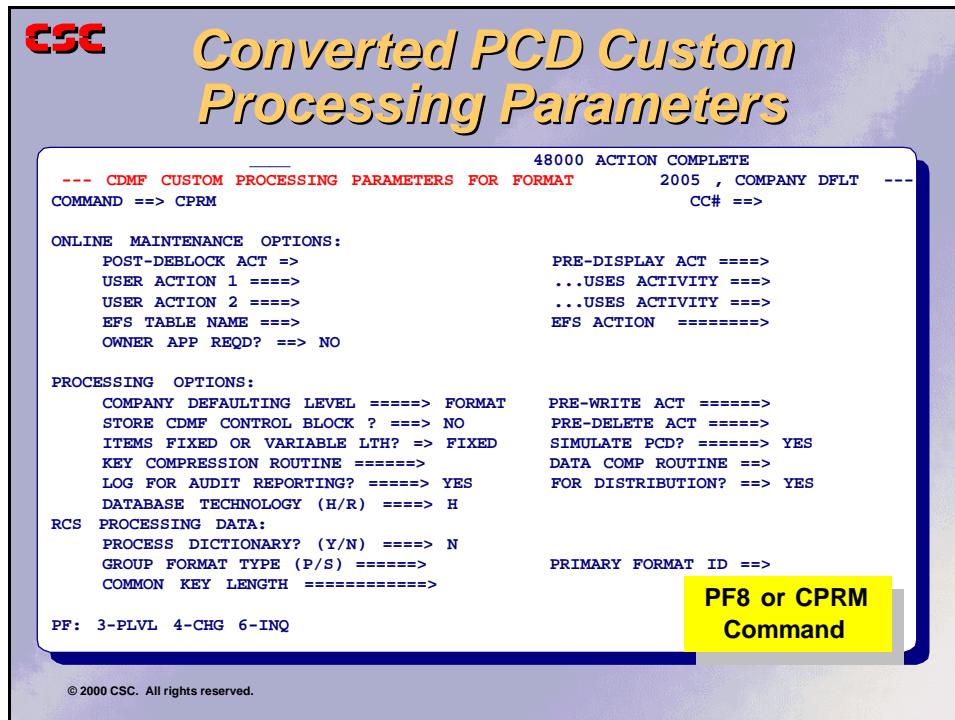


Umbrella Programming

How Old and New Formats Differ

Converted PCD Custom Processing Parameters

For converted PCDs, the SIMULATE PCD flag on the "CDMF Custom Processing Parameters for Format" screen is set to YES.



Notes:



Note: The following shows a converted PCD with both a user key and element key.

CSC

47190 Item Definition

CDMF ID 47190		ACTION SUCCESSFUL	
COMMAND ==> INQ	OWNER ==> TRD	CC# =====>	
FORMAT ==> 47190	COMP ==>	1 EFF =====>	87/01/27 HIGH USE? ==> NO
USER KEY=> 10	FOUND CO: DFLT	FOUND EFF:	78/01/01 EXPIRES: 999/12/31
POSITION DESCRIPTIONS PCD ID 47190			
USER KEY = JOB CLASS		U C.47190	
ELEMENT KEY = POSITION CODE	15		
POSITION DESCRIPTION	DIRECTOR - PRODUCT DEVELOPMENT		
LAST MAINT: DATE 86/03/27 TIME 13.41.54 CC 1012		OPER MTD	

© 2000 CSC. All rights reserved.

To access the "CDMF Format Definition Inquiry / Maintenance" screen, press the PF7 key, or enter **FMT** in the COMMAND field and press ENTER.

Notes:



Umbrella Programming

How Old and New Formats Differ

47190 Format Definition

CSC

47190 Format Definition

```
----- 48094 ACTION COMPLETE
----- CDMF FORMAT DEFINITION INQUIRY / MAINTENANCE -----
COMMAND ==> OWNER ==> TRD CC# ==>
FORMAT ID NUMBER ==> 47190 THIS FORMAT CONTAINS 25 ITEMS
COMPANY NUMBER ==> DFLT

FORMAT ID MNEMONIC ==> PCDID47190
FORMAT DESCRIPTION ==> DATA CONVERTED FROM PCD-ID 47190 :
*
* CDMF DATABASE ID ==> CDMF3 RELATIONAL DBID ==>
TARGET DATA GROUP ==> 47190 SQL ACTIVITY ==>
ITEM KEY LENGTH ==> 13
MAP NAME =====> Z47190M
MAP TITLE =====> CDMF ID 47190 :
SELECT SUBORDINATE FUNCTIONS;
CPRM - CUSTOM PROCESSING PARAMETERS, ACTIVE? Y
XPRM - CROSS REFERENCE PARAMETERS, ACTIVE?
TPRM - CORE TABLING MAINTENANCE, ACTIVE?
HSET - MAINTAIN FORMAT LEVEL HELP

LAST CHG:DATE 89/03/02 TIME 12.29.10 CC# 9 SRCE I001 OPER JLB
PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-XPRM 8-CPRM 9-NXT 10-GEN 11-TPRM ..-ITEM ..-DEL
```

© 2000 CSC. All rights reserved.

PF7 or FMT Command

Notes:



Application Access to PCD Formats

Accessing Simulated PCD—Converted

Due to the large numbers of programs that accessed PCD under the old technology, a means to access the CDMF interface, which simulates the old PCD system, has been provided. With this simulation, existing application programs do not need changes.

In order to simulate PCD:

1. Fill in the fields of data group 1452, the PCD control block as follows:

PCD-ACTION

Action for the request. Action codes are defined in COBOL copybook I57104D and in I57104E for ALC.

PCD-COID

Company ID (halfword binary) of the formatted being requested.

PCD-ID

Format ID (halfword binary) of the formatted being requested.

PCD-USER-KEY

Item key

PCD-EFFECT-DATE

Optional (if zero defaults to TCB-EFFECTIVE-DATE)

2. Fill in the element key in the target data group if the PCD is recursively keyed. The element key is located after the action and result fields in the target data group.
3. Issue activity 1013 (read only) or activity 1014 (read/update) to link to the PCD interface process.

Returned by the PCD processor:

Information returned in data group 1452

PCD-RESULT

Result of the requested action. Result codes are defined in COBOL copybook I57105D and in I57105E for ALC.

PCD FIELDS

Filled in before issuing the activity.

Information returned in the target data group:

The entire data area from the requested item



Umbrella Programming

Application Access to PCD Formats

Data Group 1452 Copybook I57101D

```
DATA GROUP          COPYBOOK NAME I57101D

*****
*           I57101D - PCD CONTROL BLOCK
*****
*           PEM - DG #1452, ALLOCATE ACTIVITY #1530
*****
01  PCD-CONTROL-BLOCK.
    05  PCD-ACTION             PIC XX.
    05  PCD-RESULT              PIC XX.
    05  PCD-INPUT-SECTION.
        10  PCD-COID            PIC XX.
        10  PCD-ID               PIC XX.
        10  PCD-USER-KEY          PIC X(11).
        10  PCD-EFFECT-DATE      PIC S9(7) COMP-3.
        10  PCD-EXPIRE-DATE      PIC S9(7) COMP-3.
        10  FILLER                PIC X.
    05  PCD-OUTPUT-SECTION.
        10  PCD-OUT-COID         PIC XX.
        10  PCD-OUT-ID            PIC XX.
        10  PCD-OUT-USER-KEY      PIC X(11).
        10  PCD-OUT-EFFECT-DATE  PIC S9(7) COMP-3.
        10  PCD-OUT-EXPIRE-DATE  PIC S9(7) COMP-3.
        10  FILLER                PIC X.
    05  PCD-EFFECT-PERIOD.
        10  PCD-FIRST-EFFECT     PIC S9(7) COMP-3.
        10  PCD-LAST-EFFECT      PIC S9(7) COMP-3.
    05  PCD-SWTCH              PIC XX.
    05  PCD-AMSG                PIC X(8).
    05  PCD-RMSG                PIC X(17).
    05  PCD-CCNO                PIC S9(8)  COMP.
    05  FILLER                  PIC X.
    05  PCD-COID-USED          PIC XX.
    05  PCD-USER-FIELD-1        PIC X(40).
    05  PCD-OWNER                PIC X(03).
    05  FILLER                  PIC X(01).
*
*****
```

Notes:



PCD Action Codes Copybook I57104D

```
DATA GROUP          COPYBOOK NAME I57104D

***** START OF I57104D ***** PCD ACTION LIST *****
*
01  PCD-ACTION-LIST.
    05  PCD-ACTION-CODES.
        10  PCD-NO-OPERATION.
                15  FILLER PIC S9(4) COMP VALUE +0000.
        10  PCD-CREATE.
                15  FILLER PIC S9(4) COMP VALUE +0001.
        10  PCD-ADD.
                15  FILLER PIC S9(4) COMP VALUE +0002.
        10  PCD-READ.
                15  FILLER PIC S9(4) COMP VALUE +0003.
        10  PCD-READ-NEXT.
                15  FILLER PIC S9(4) COMP VALUE +0004.
        10  PCD-UPDATE.
                15  FILLER PIC S9(4) COMP VALUE +0005.
        10  PCD-COPY.
                15  FILLER PIC S9(4) COMP VALUE +0006.
        10  PCD-DELETE.
                15  FILLER PIC S9(4) COMP VALUE +0007.
        10  PCD-CLEAR.
                15  FILLER PIC S9(4) COMP VALUE +0008.
        10  PCD-REPLACE.
                15  FILLER PIC S9(4) COMP VALUE +0009.
        10  PCD-ADD-IP.
                15  FILLER PIC S9(4) COMP VALUE +0010.
        10  PCD-DELETE-IP.
                15  FILLER PIC S9(4) COMP VALUE +0011.
        10  PCD-READ-NEXT-SET.
                15  FILLER PIC S9(4) COMP VALUE +0012.
        10  PCD-READ-NEXT-EFFECT-SET.
                15  FILLER PIC S9(4) COMP VALUE +0013.
        10  PCD-READ-NEXT-SET-SEQ.
                15  FILLER PIC S9(4) COMP VALUE +0014.
*
***** END OF I57104D *****
```

Notes:



Umbrella Programming

Application Access to PCD Formats

PCD Result Codes Copybook I57105D

```
DATA GROUP          COPYBOOK NAME I57105D

*****
*           I57105D - PCD RESULT LIST
*****
01  PCD-RESULT-LIST.
    05  PCD-RESULT-CODES.
*   PCD-NO-ERRORS
    10  FILLER      PIC S9(8) COMP VALUE +00000.
*   PCD-INVALID-ACTION
    10  FILLER      PIC S9(8) COMP VALUE +57001.
*   PCD-ACTN-ACTV-ERROR
    10  FILLER      PIC S9(8) COMP VALUE +57002.
*   PCD-NO-GROUP-FOUND
    10  FILLER      PIC S9(8) COMP VALUE +57003.
*   PCD-NO-SET-FOUND
    10  FILLER      PIC S9(8) COMP VALUE +57004.
*   PCD-NO-ELEMENT-FOUND
    10  FILLER      PIC S9(8) COMP VALUE +57005.
*   PCD-DUPLICATE-GROUP
    10  FILLER      PIC S9(8) COMP VALUE +57006.
*   PCD-DUPLICATE-SET
    10  FILLER      PIC S9(8) COMP VALUE +57007.
*   PCD-DUPLICATE-ELEMENT
    10  FILLER      PIC S9(8) COMP VALUE +57008.
*   PCD-END-OF-FILE
    10  FILLER      PIC S9(8) COMP VALUE +57009.
*   PCD-END-OF-GROUP
    10  FILLER      PIC S9(8) COMP VALUE +57010.
*   PCD-END-OF-SET
    10  FILLER      PIC S9(8) COMP VALUE +57011.
*   PCD-INPUT-EXIT-ERROR
    10  FILLER      PIC S9(8) COMP VALUE +57012.
*   PCD-OUTPUT-EXIT-ERROR
    10  FILLER      PIC S9(8) COMP VALUE +57013.
*   PCD-MAINT-EXIT-ERROR
    10  FILLER      PIC S9(8) COMP VALUE +57014.
*   PCD-INVALID-EFF-DATE
    10  FILLER      PIC S9(8) COMP VALUE +57015.
*   PCD-INVALID-EXP-DATE
    10  FILLER      PIC S9(8) COMP VALUE +57016.
*   PCD-INVALID-OUT-EFF-DATE
    10  FILLER      PIC S9(8) COMP VALUE +57017.
*   PCD-INVALID-OUT-EXP-DATE
    10  FILLER      PIC S9(8) COMP VALUE +57018.
*   PCD-INVALID-DATA-GROUP
    10  FILLER      PIC S9(8) COMP VALUE +57019.
*   PCD-INVALID-DISP-ACTV
    10  FILLER      PIC S9(8) COMP VALUE +57020.
*   PCD-INVALID-DBLK-ACTV
    10  FILLER      PIC S9(8) COMP VALUE +57021.
*   PCD-INVALID-DB-ACTV
    10  FILLER      PIC S9(8) COMP VALUE +57022.
*   PCD-MISSING-USER-DATA
    10  FILLER      PIC S9(8) COMP VALUE +57023.
*   PCD-MISSING-CTL-DATA
    10  FILLER      PIC S9(8) COMP VALUE +57024.
```



Umbrella Programming

Application Access to PCD Formats

```
*      PCD-INPUT-SET-NOT-FOUND
      10 FILLER          PIC S9(8) COMP VALUE +57025.
*      PCD-OUTPUT-SET-NOT-FOUND
      10 FILLER          PIC S9(8) COMP VALUE +57026.
*      PCD-INPUT-GROUP-NOT-FOUND
      10 FILLER          PIC S9(8) COMP VALUE +57027.
*      PCD-OUTPUT-GROUP-NOT-FOUND
      10 FILLER          PIC S9(8) COMP VALUE +57028.
*      PCD-UNAUTHORIZED-ACTION
      10 FILLER          PIC S9(8) COMP VALUE +57029.
*      PCD-CC-NO-IS-ZERO-ON-UPDT
      10 FILLER          PIC S9(8) COMP VALUE +57030.
*      PCD-NO-ALTERNATE-WO-PRIMARY
      10 FILLER          PIC S9(8) COMP VALUE +57031.
*      PCD-UNDETERMINED-ERROR
      10 FILLER          PIC S9(8) COMP VALUE +57999.
05 FILLER          REDEFINES PCD-RESULT-CODES.
      10 FILLER.
      15 FILLER          PIC XX.
      15 PCD-NO-ERRORS    PIC XX.
      10 FILLER.
      15 FILLER          PIC XX.
      15 PCD-INVALID-ACTION  PIC XX.
      10 FILLER.
      15 FILLER          PIC XX.
      15 PCD-ACTN-ACTV-ERROR  PIC XX.
      10 FILLER.
      15 FILLER          PIC XX.
      15 PCD-NO-GROUP-FOUND  PIC XX.
      10 FILLER.
      15 FILLER          PIC XX.
      15 PCD-NO-SET-FOUND    PIC XX.
      10 FILLER.
      15 FILLER          PIC XX.
      15 PCD-NO-ELEMENT-FOUND  PIC XX.
      10 FILLER.
      15 FILLER          PIC XX.
      15 PCD-DUPLICATE-GROUP    PIC XX.
      10 FILLER.
      15 FILLER          PIC XX.
      15 PCD-DUPLICATE-SET    PIC XX.
      10 FILLER.
      15 FILLER          PIC XX.
      15 PCD-DUPLICATE-ELEMENT  PIC XX.
      10 FILLER.
      15 FILLER          PIC XX.
      15 PCD-END-OF-FILE    PIC XX.
      10 FILLER.
      15 FILLER          PIC XX.
      15 PCD-END-OF-GROUP    PIC XX.
      10 FILLER.
      15 FILLER          PIC XX.
      15 PCD-END-OF-SET    PIC XX.
      10 FILLER.
      15 FILLER          PIC XX.
      15 PCD-INPUT-EXIT-ERROR  PIC XX.
      10 FILLER.
      15 FILLER          PIC XX.
      15 PCD-OUTPUT-EXIT-ERROR  PIC XX.
```



Umbrella Programming

Application Access to PCD Formats

```
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-MAINT-EXIT-ERROR  PIC XX.  
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-INVALID-EFF-DATE  PIC XX.  
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-INVALID-EXP-DATE  PIC XX.  
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-INVALID-OUT-EFF-DATE  PIC XX.  
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-INVALID-OUT-EXP-DATE  PIC XX.  
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-INVALID-DATA-GROUP  PIC XX.  
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-INVALID-DISP-ACTV  PIC XX.  
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-INVALID-DBLK-ACTV  PIC XX.  
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-INVALID-DB-ACT  PIC XX.  
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-MISSING-USER-DATA  PIC XX.  
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-MISSING-CTL-DATA  PIC XX.  
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-INPUT-SET-NOT-FOUND  PIC XX.  
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-OUTPUT-SET-NOT-FOUND  PIC XX.  
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-INPUT-GROUP-NOT-FOUND  PIC XX.  
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-OUTPUT-GROUP-NOT-FOUND  PIC XX.  
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-UNAUTHORIZED-ACTION  PIC XX.  
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-CC-NO-IS-ZERO-ON-UPDT  PIC XX.  
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-NO-ALT-WO-PRIMARY  PIC XX.  
10  FILLER.  
    15  FILLER          PIC XX.  
    15  PCD-UNDETERMINED-ERROR  PIC XX.
```

*

*



Problem Specifications—PCD Access



1. The purpose of this phase is to read PCD format 47190 to obtain the corresponding position description for the employee's job classification and position. Data group 47120 contains the job classification and position codes needed to build the item key. Data group 47120 is part of the EMP data base. Make sure your data base read activity is reading DG 47120.
2. CDMF format 47190 is a converted PCD. You will need to use data group 1452, the old PCD communication data group, to communicate with CDMF.

Prior to the conversion JOB CLASSIFICATION was the user key and POSITION CODE was the element key.

PCD action and result code values are defined by the COBOL copybooks I57104D and I57105D respectively. These have been included in your skeleton program.

3. Use activity 1013 to link to CDMF. The link activity is defined in the Process Dictionary. Your copy of the class program contains the binary defined field for the link to PCD processing.
4. Take the following actions based on the PCD result code returned after having invoked CDMF.
 - If PCD-RESULT is not equal to low values, take the following actions based on the PCD result code:
 - If PCD-RESULT indicates NO GROUP FOUND move the message NO GROUP FOUND to the position description field.*
 - If PCD-RESULT indicates NO ELEMENT FOUND move the message NO ELEMENT FOUND to the position description field.
 - If PCD-RESULT indicates other errors, the program should execute a dump and end.
 - The SPS program prints the description from the position description field in data group 47190.
 - Use the owner application and change control created in the Change Control Exercise.

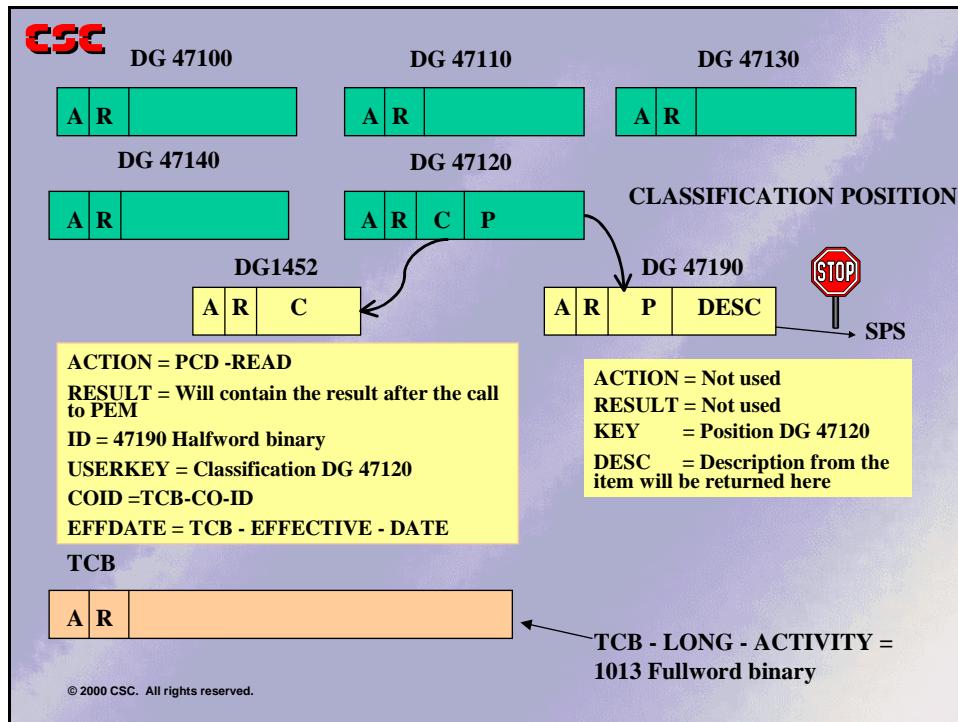
*NO GROUP FOUND is the error message when the PCD (Format) cannot be identified.



Umbrella Programming

Problem Specifications—PCD Access

PCD Diagram



Notes:



Problem Specifications—Format Definition (Optional Exercise)



This exercise will reinforce the tasks needed to create a Format Definition. The format you will create and the data you add to the format will be accessed online and by your program.

In this exercise, you will not be required to add a data group or map definitions. The TARGET DATA GROUP and MAP NAME for format 47192 will be used.

To define the format definition using follow information.

1. The Format ID is 47192.
2. The Company ID is 99xx, where xx is your group number.
3. The Mnemonic should include your group number, as should the description.
4. Use CDMF3 as the data base.
5. The TARGET DATA GROUP ID will be 47192.
6. The length of the key is four bytes.
7. The Map Name is Z47192M for this format.
8. The Map Title should include your group number.
9. Use the owner application and change control created in the Change Control Exercise.
10. At this point you should be able to add items to your format using the Online Item Maintenance processor. From the Format Definition screen, type ITEM into the Command field, and press enter. Alternately, you may access your new CDMF format by entering, from a cleared screen, U C.47192. This will probably display the DFLT company version of 47192. In order to address your CDMF format, change the value of the COMP field to 99xx, where xx is your group number, and press PF6 (INQ). Either option should display the message "48005 END OF FORMAT".
11. Add items to your format using the item keys listed below. You may choose to be creative with the descriptions. Be sure to use Company 99xx and 780101 for your effective date.

JOB CLASS	JOB POSITION	JOB CLASS	JOB POSITION
10	15	40	20
10	60	40	99
20	30	50	11
20	47	50	23
30	30		
30	35		



Umbrella Programming

Problem Specifications—Format Definition (Optional Exercise)

12. Modify your COBOL skeleton program to PERFORM routine PC000(CDMF access), and to bypass routine PC100(PCD access). This can be accomplished by commenting out the PCD PERFORM statement in the main-line section.
13. Modify your execution JCL to test using Company 99xx. The application and function values should remain the same.
14. Submit the execution JCL to test your program against your CDMF format of position descriptions. Remember, the report will print with many more ITEM NOT FOUND items, because you only created a small set of the descriptions used by the Employee System.
15. After checking your report for the new position descriptions from your CDMF format, change your Format Definition allowing the concatenation processing of your CDMF format and the DFLT CDMF format for 47192.
16. Submit the execution JCL to test your program against your format and DFLT format of position descriptions. Now, the report will print with fewer ITEM NOT FOUND because two formats are being searched and combined.

Notes:



Summary

CSC

Summary



- Information usually housed on tables are stored as PCDs
- Major components of CDMF
 - Format (file)
 - Item (record)
 - Target data group (record layout)
- CDMF = collection of files of varying format

© 2000 CSC. All rights reserved.

CSC

Summary

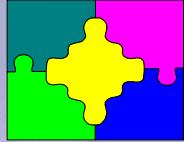
- CDMF key fields
 - Format ID
 - Company ID
 - Item key
 - Effective date
- CDMF format items can be effective dated
- File must be defined online as a format definition
- CDMF actions + Standard Process Dictionary actions plus FMT and NXTE

© 2000 CSC. All rights reserved.





Summary



- Methods for searching file for item
 - Format level
 - Item level
- New CDMF formats accessed through CDMF
- Bridge for accessing old technology
- Simulated PCDs have standard size user key and may have element key
- All HOGAN applications use PCDs
- Umbrella subsystems use PCDs

© 2000 CSC. All rights reserved.



- Information such as interest rates and coded descriptions that would be generally be housed on tables are stored online under the Umbrella System as Process Control Data. Two key features of such data are that they change frequently and need to be accessed by multiple application programs.
- The three major components of CDMF are a format (file), item (record), and target data group (record layout).
- CDMF is a collection of multiple files of varying formats.
- The CDMF key fields are—format ID, company ID, item key, and effective date.
- Items on CDMF formats can be effective dated to form effective series.
- A file must be defined online as a format definition. Various Process Dictionary definitions are required as support items, such as, data group and map definitions.
- Commonly used CDMF actions consist of the Standard Process Dictionary actions plus FMT and NXTE.
- There are two methods for searching a file for an item—format level and



item level.

- Accessing new CDMF formats is through the CDMF Application Control Block (part of the TCB) and link activity 48000.
- A bridge was written so that tables that were formerly PCDs continue to be processed as they were under old technology. They are accessed through PCD control block 1452 and activity 1013 or 1014.
- Simulated PCDs have a standard size user key and may have an element key to form the CDMF item key.
- All HOGAN applications use PCDs.
- Umbrella subsystems such as DTS, SPS, and FPS use PCDs to store process-directing data for date calculations, report generation, and screen flow, respectively.

Notes:



Umbrella Programming

Summary



Condition Code Processing—CCP

15

Purpose



Using the Condition Code
Processing subsystem for
handling user-defined
exceptions

© 2000 CSC. All rights reserved.

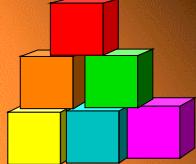
Notes:



Topics



Topics



- ▶ User-defined processing exceptions
- ▶ CDMF format 1301
- ▶ Activities 1350 and 1398
- ▶ TCB-USER-CC
- ▶ Target data group 48551

© 2000 CSC. All rights reserved.

Notes:



Objectives

CSC

Objectives



To become familiar with:

- Use of CCP for user-defined exceptions
- Item key for CDMF format 1301
- Purpose of data fields on format 1301
- Distinction between use of activities 1350 and 1398
- Processing flow of CCP
- Use of TCB-USER-CC within CCP

© 2000 CSC. All rights reserved.

1. To explain the use of CCP for handling user-defined exceptions
2. To explain the item key for CDMF format 1301
3. To describe the purpose of the data fields on format 1301
4. To distinguish between the use of activities 1350 and 1398
5. To illustrate processing flow of CCP
6. To define the use of TCB-USER-CC within CCP to explain how control data group 48551 is used.

Notes:



Condition Code Processing Capabilities



Condition Code Processing

- Error Conditions
- Non-error condition

Offers:

- Standard approach to MSG Handling
- Online maintained via CDMF
- MSGs stored external to application program

© 2000 CSC. All rights reserved.

User-Defined Exceptions

Condition Code Processing (CCP) gives a standard approach for identifying and handling user-defined exceptions. Through parameters maintained online, one can externally control application logic flow for exceptional situations. For example, suppose that a program encounters invalid data. Rather than hard-code an error-handling routine into the program, a condition code may be set and control passed to the Condition Code Processing system.

There are two advantages of this approach. The processing of errors can be changed without recoding, recompiling, and relinking the application program. In addition, the condition code can be passed as a parameter among application programs in a calling chain.

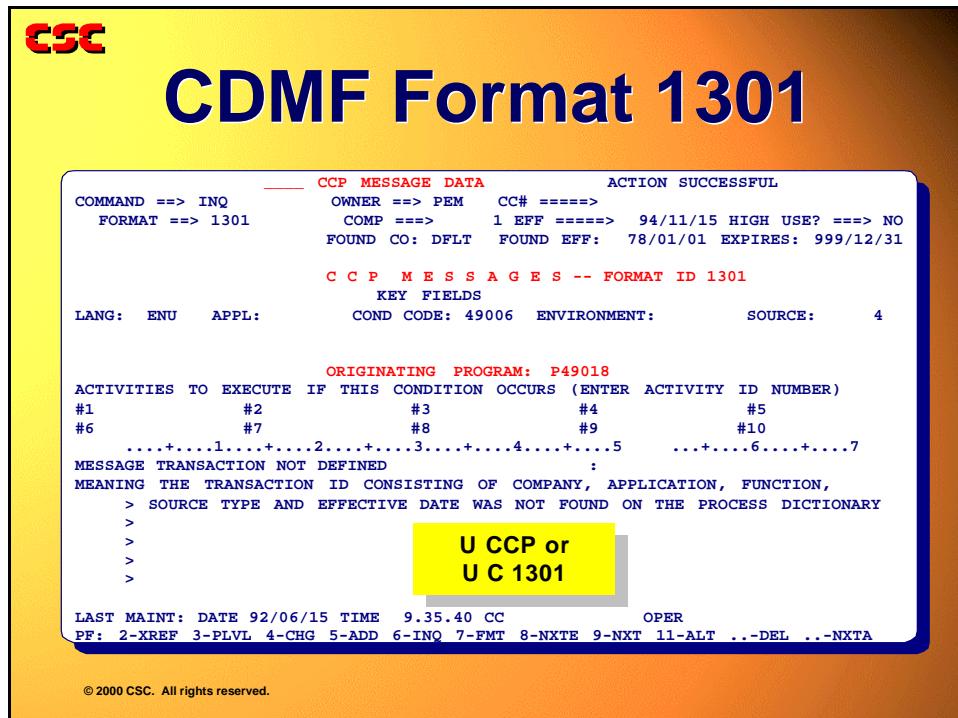
Notes:



CDMF Format 1301

Condition codes are maintained on the CDMF format 1301. This format was a PCD under the old technology. It has been converted so that it is now a native CDMF format.

There are two important pieces of data stored on CDMF 1301. A message field is provided. This message can be displayed online or printed on a report. In addition, activities for processing the errors (printing a warning message, issuing a dump, abending, and so on) are listed on the entry.



Notes:



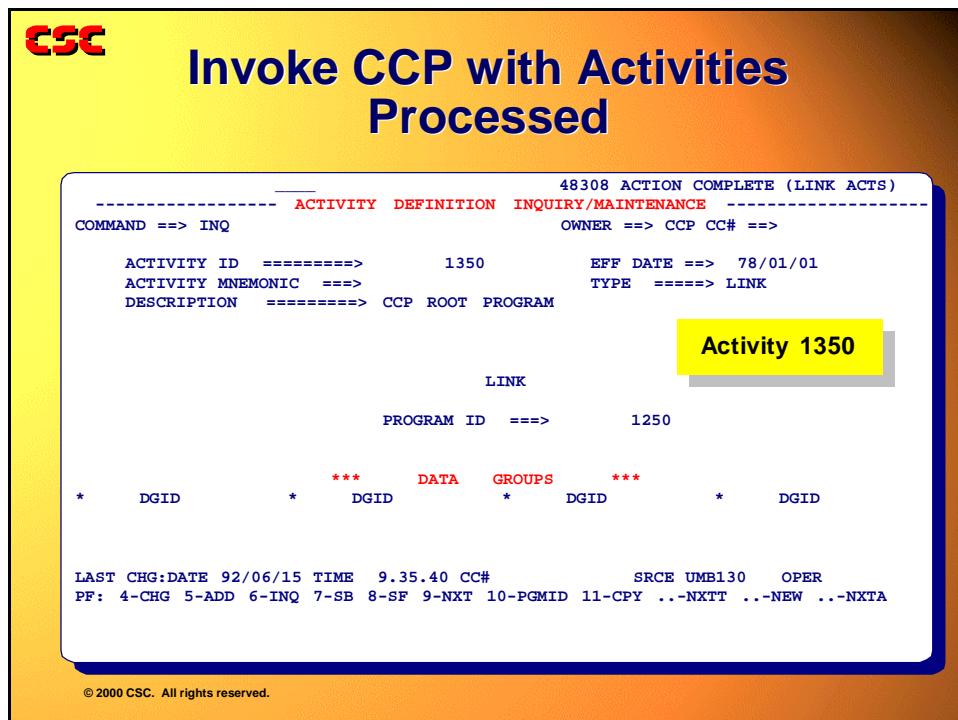
Umbrella Programming

Condition Code Processing Capabilities

Invoking CCP

There are basically two approaches for invoking CCP. An application program can link directly to CCP at the time of the exception. Alternatively, the application program can set a condition code that is passed back up the calling chain to a program that centrally processes exceptions by linking to CCP.

When activity 1350 is used, all activities are executed in the sequence in which they appear on the CDMF 1301 entry. An activity such as dump and end processing can be issued.



Notes:



When processing is invoked using activity 1398, the activities are not executed.

CSC

Invoke CCP with Activities Ignored

```
48308 ACTION COMPLETE (LINK ACTS)
----- ACTIVITY DEFINITION INQUIRY/MAINTENANCE -----
COMMAND ==> INQ OWNER ==> CCP CC# ==>
ACTIVITY ID ======> 1398     EFF DATE ==> 78/01/01
ACTIVITY MNEMONIC ==>           TYPE =====> LINK
DESCRIPTION ======> CCP RETURN DESCRIPT'N

LINK

PROGRAM ID ===> 1250

*      ***      DATA      GROUPS      ***
*      DGID      *      DGID      *      DGID      *      DGID

LAST CHG:DATE 92/06/15 TIME 9.35.40 CC# SRCE UMB130 OPER
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-PGMID 11-CPY ..-NXTT ..-NEW ..-NXTA
```

© 2000 CSC. All rights reserved.

Notes:



Umbrella Programming

Condition Code Processing Capabilities

CCP Program Definition

Both methods of calling Condition Code Processing cause the message and meaning associated with the condition code to be placed in data group 48551.

The screenshot shows a terminal window titled "CCP Program Definition" with a yellow header and orange background. The title bar has "CSC" in red. The main area displays a program definition record with fields like PROGRAM ID, EFF DATE, LANGUAGE, and OWNER. It also shows data groups used by the program and authorized activities. At the bottom, there's a copyright notice and a footer with a globe icon.

ACTION SUCCESSFUL

----- PROGRAM DEFINITION INQUIRY/MAINTENANCE -----

COMMAND ==> INQ OWNER ==> CCP CC# ==>

PROGRAM ID ==> 1250 EFF DATE ==> 78/01/01 PEM TECHNOLOGY => FULLWORD
LINKNAME ==> T58550 LANGUAGE ==> COBOL SOURCE NAME ==> T58550
DESCRIPTION => CCP ROOT STATUS ==> PROD
HIGH, MED, OR LOW USAGE? ==> LOW USED ONLINE, BATCH OR BOTH? ==> BOTH

----- DATA GROUPS USED BY PROGRAM -----

* PP ---DGID---	* PP ---DGID---	* PP ---DGID---
1 48551	2 1301	3 1302
4 1031	5 1320	6 3210

----- AUTHORIZED ACTIVITIES -----

* -ACTIVITY- * -ACTIVITY- * -ACTIVITY- * -ACTIVITY- * -ACTIVITY- * -ACTIVITY-
-- ALL --

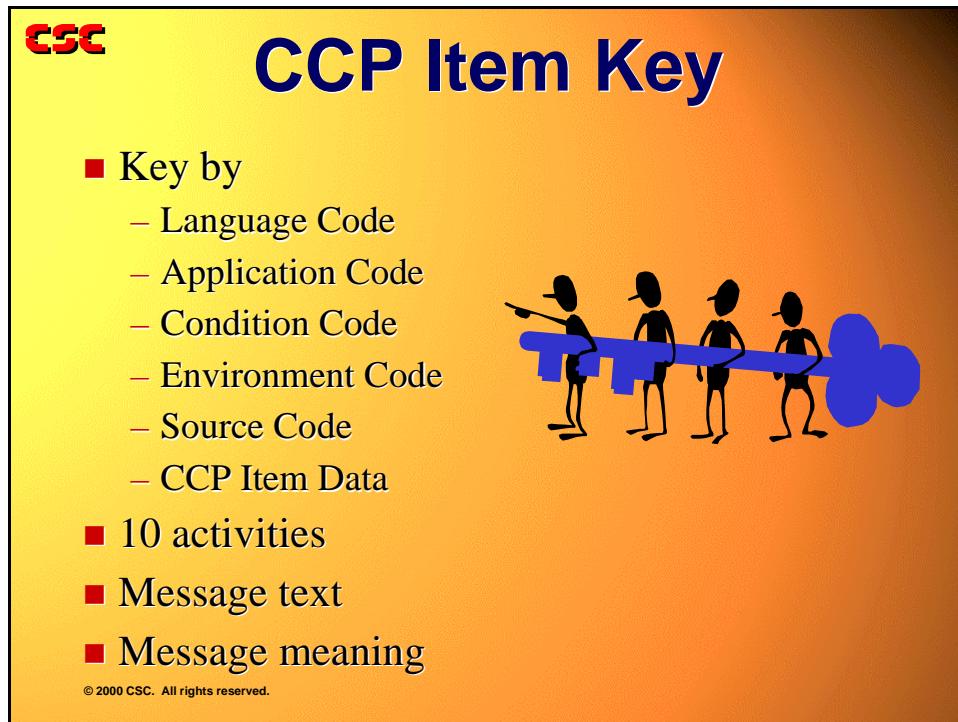
LAST CHG:DATE 92/06/15 TIME 9.35.40 CC# SRCE UMB130 OPER
PF: 2-XREF 3-PLVL 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-BYLINK 11-BYID ...ACTS

© 2000 CSC. All rights reserved.

Notes:



CCP Item Key



The slide features a yellow-to-orange gradient background. In the top left corner is the red CSC logo. The title "CCP Item Key" is centered in large blue letters. Below the title is a bulleted list of key components. To the right of the list is a graphic of four stylized black figures holding a large blue puzzle piece.

- Key by
 - Language Code
 - Application Code
 - Condition Code
 - Environment Code
 - Source Code
 - CCP Item Data
- 10 activities
- Message text
- Message meaning

© 2000 CSC. All rights reserved.

CDMF format 1301 has a five-part item key.

- LANGUAGE CODE
- APPLICATION CODE
- CONDITION CODE
- ENVIRONMENT CODE
- SOURCE CODE.

Language Code is a 3 character valid ISO language code value. The application code is a two digit number assigned to a specific application system, such as, 88 for UES. Environment code is not currently used. Source code is 3 for online and 4 for batch. If you are processing online and there is no condition code defined with a source of 3, CCP will use the message defined for source 4. The application code and the environment code will both default to binary zeros if they are not set by the application program.

The application programmer or designer sets up a condition code that must be a unique number within the application code grouping.

The originating program field is for documentation.



Umbrella Programming

Condition Code Processing Capabilities

Note that up to 10 activities can be issued from a 1301 entry. Any activity defined to the Process Dictionary can be issued. Most frequently the activity would be a link or a dump. Remember that if activities are issued from the exception entry on CDMF 1301, the link activity to CCP that must be used is activity 1350.

The message field provides for an explanation of the exception. It is picked up by FPS and PEM formatted dumps. There are six additional lines to expand on the meaning of the message. These lines can be used on help screens.

Notes:



CDMF Format 1301

Format Item Maintenance

The purpose of this section is to discuss a typical CDMF 1301 format entry and review the data stored on it. Below is a sample entry.

CSC

CDMF Format 1301

```
CCP MESSAGE DATA ACTION SUCCESSFUL
COMMAND ==> INQ OWNER ==> UES CC# =====>
FORMAT ==> 1301 COMP ==> 1 EFF =====> 92/10/13 HIGH USE? ==> NO
          FOUND CO: DFLT FOUND EFF: 78/01/01 EXPIRES: 999/12/31

C C P M E S S A G E S -- FORMAT ID 1301
KEY FIELDS
LANG: ENU APPL: 88 COND CODE: 8304 ENVIRONMENT: SOURCE: 4

ORIGINATING PROGRAM: U88420
ACTIVITIES TO EXECUTE IF THIS CONDITION OCCURS (ENTER ACTIVITY ID NUMBER)
#1      #2      #3      #4      #5
#6      #7      #8      #9      #10
.....+....1.....+....2.....+....3.....+....4.....+....5 .....+....6.....+....7
MESSAGE THOT NOT FOUND :
MEANING COULDNT FIND THOT FROM THINKER NAME ENTERED
>
>
>
>
>

LAST MAINT: DATE 92/06/15 TIME 9.35.40 CC OPER
PF: 2-XREF 3-PLVL 4-CHG 5-ADD 6-INQ 7-FMT 8-NXTE 9-NXT 11-ALT ..-DEL ..-NXTA
```

© 2000 CSC. All rights reserved.

Notes:



Umbrella Programming

CDMF Format 1301

Format Definition for CCP

The screenshot shows a terminal window with the CSC logo at the top left. The title bar reads "Format Definition for CCP". The main area displays the format definition for CDMF Format 1301. The output includes:

```
48094 ACTION COMPLETE
----- CDMF FORMAT DEFINITION INQUIRY / MAINTENANCE -----
COMMAND ==> INQ OWNER ==> CCP CC# ==>
FORMAT ID NUMBER ==> 1301 THIS FORMAT CONTAINS 14504 ITEMS
COMPANY NUMBER ==> DFLT

FORMAT ID MNEMONIC => PCDID01301
FORMAT DESCRIPTION => CONDITION CODE PROCESSING DATA :
*
CDMF DATABASE ID ==> CDMF3 RELATIONAL DBID ====
TARGET DATA GROUP ==> 48551 SQL ACTIVITY ====
ITEM KEY LENGTH ==> 11
MAP NAME =====> T58550M
MAP TITLE =====> CCP MESSAGE DATA :

SELECT SUBORDINATE FUNCTIONS;
CPRM - CUSTOM PROCESSING PARAMETERS, ACTIVE? Y
XPRM - CROSS REFERENCE PARAMETERS, ACTIVE? Y
TPRM - CORE TABLING MAINTENANCE, ACTIVE? Y
HSET - MAINTAIN FORMAT LEVEL HELP

LAST CHG:DATE 92/06/15 TIME 9.35.40 CC# SRCE UMB130 OPER
PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-XPRM 8-CPRM 9-NXT 10-GEN 11-TPRM ..-ITEM ..-DEL
```

At the bottom left of the terminal window, it says "© 2000 CSC. All rights reserved."

Target Data Group

The first important data on the format definition for CDMF format 1301 is the target data group. Under the new technology, data group 48551 is the target data group.

Data group 1301 had been the target data group associated with the old technology PCD. To accommodate application programs written under the old technology, PEM assumes that any condition code that has an application code of binary zeros was created under old technology. It copies the information in data group 48551 to data group 1301 so that application programs that reference data group 1301 can continue to do so.

Programmers writing new applications under the new technology should reference data group 48551 rather than 1301. If the application code is not zero, there will be no entries in data group 1301.



Processing Flow

Fields Set by Application Program

If CCP used normal PCD processing, an application programmer would code the access to a CDMF format as covered in the previous section of this class. Because CCP is heavily used throughout Hogan, it has fewer requirements for setting key fields and actually accessing CDMF 1301—and the rules are different.

The application program sets up two fields in the TCB that will be used by the CCP system. These fields are:

- TCB-USER-CC
- TCB-USER-CC-APP

CCP picks up TCB-SOURCE-TYPE. TCB-USER-ENVMT is available for future use.

Notes:



Umbrella Programming

Processing Flow

TRANSACTION DEFINITION

```
APPLICATION ID ======>
FUNCTION ID ======>
SOURCE ID ======>
COMPANY ID LIST ======>
EFFECTIVE DATE ======>
```

```
***** START OF P49000D ***** TCB ****
```

```
* DATA GROUP 00001 (USER TRANSACTION CONTROL BLOCK) *
```

```
01 TRANSACTION-CONTROL-BLOCK.
 05 FILLER          PIC XXXX.
 05 TCB-TRANS-NO    PIC XXXX.
 05 TCB-CO-ID       PIC XX.
 05 TCB-APPL-ID     PIC XX.
 05 TCB-FUNC-ID     PIC XX.
 05 TCB-SOURCE-TYPE.

  .
 05 FILLER          REDEFINES TCB-USER-DATA.
===== 10 TCB-USER-CC      PIC XX.
        10 TCB-USER-RESULT  PIC XX.
        10 TCB-USER-ENVMT  PIC XX.

  .
 05 TCB-LONG-ACTIVITY-N   PIC S9(09) COMP.
 05 TCB-LONG-ACTIVITY    REDEFINES TCB-LONG-ACTIVITY-N.

  .
 05 TCB-LONG-DGID-N      PIC S9(09) COMP.
 05 TCB-LONG-DGID        REDEFINES TCB-LONG-DGID-N.

  .
===== 05 TCB-USER-CC-APP  PIC XX.
  .
```

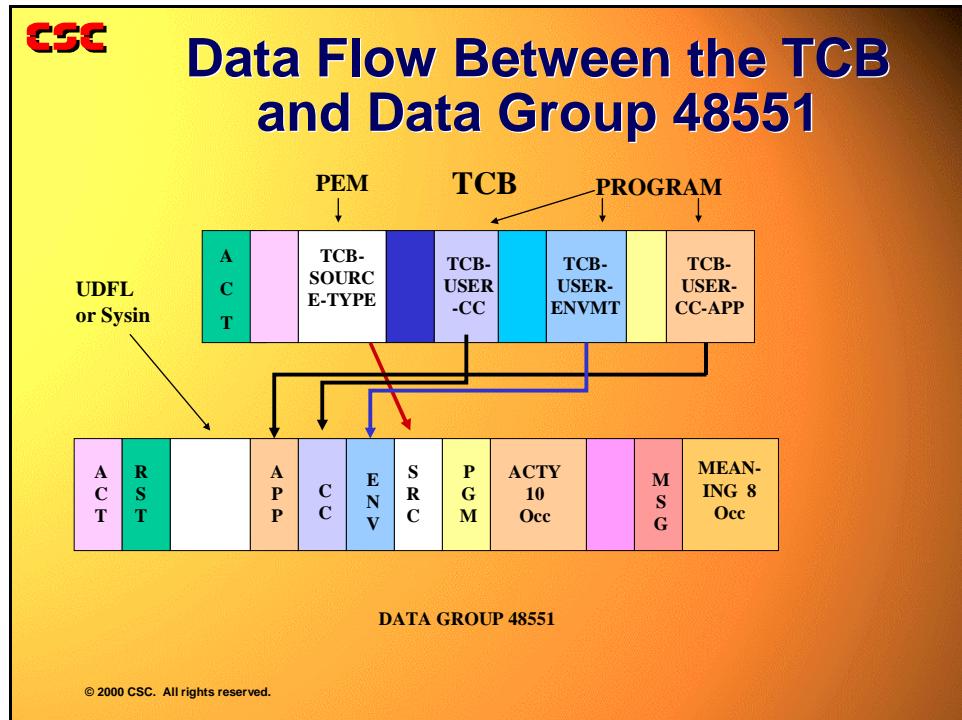
Notes:



Process Flow

Either activity 1350 or activity 1398 is issued to link to CCP. The key fields set in the TCB are moved to data group 48551. Then CCP is able to read the correct CDMF 1301 entry. Activities may be executed. The message and meaning lines can be accessed in fields in data group 48551.

Data Flow Between the TCB and Data Group 48551



Notes:



Umbrella Programming

Processing Flow

Copybook for Data Group 48551

```
DATA GROUP          COPYBOOK NAME U48551D

***** START OF U48551D **** CCP-CDMF-FORMAT-48551 ****
*
*   DATA GROUP NUMBER = 48551  (PCD ID 1301)
*
*   THIS DATA GROUP IS THE MASTER CDMF RECORD FOR CCP.
*   THE KEY IS APPLICATION ID, CONDITION CODE, ENVIRONMENT,
*   SOURCE.
*
*****
*   01  CCP-CDMF-MASTER.
    05  CCP-ACTION-48551          PIC XX.
    05  CCP-RESULT-48551         PIC XX.
    05  CCP-KEY.
        10  CCP-KEY-ISO-CODE      PIC X(03).
        10  CCP-KEY-APPLICATION    PIC XX.
        10  CCP-KEY-APPLICATION-N REDEFINES CCP-KEY-APPLICATION
                                      PIC S9(4) COMP.
        10  CCP-KEY-COND-CODE     PIC XX.
        10  CCP-KEY-COND-CODE-N  REDEFINES CCP-KEY-COND-CODE
                                      PIC S9(4) COMP.
        10  CCP-KEY-ENVIRONMENT   PIC XX.
        10  CCP-KEY-ENVIRONMENT-N REDEFINES CCP-KEY-ENVIRONMENT
                                      PIC S9(4) COMP.
        10  CCP-KEY-SOURCE        PIC XX.
        10  CCP-KEY-SOURCE-TYPE-N REDEFINES CCP-KEY-SOURCE
                                      PIC S9(4) COMP.
            88  SOURCE-IS-BATCH    VALUE +0004.
    05  CCP-ORIGINATING-PROGRAM  PIC X(8).
    05  CCP-ACTIVITIES OCCURS 10 TIMES.
        10  CCP-NEXT-ACTIVITY    PIC XXXX.
        10  CCP-NEXT-ACTIVITY-HW REDEFINES CCP-NEXT-ACTIVITY.
            15  CCP-NEXT-ACT-HI-ORDER  PIC XX.
            15  CCP-NEXT-ACT-LO-ORDER  PIC XX.
    05  CCP-MESSAGE             PIC X(40).
    05  CCP-MEANING.
        10  CCP-MEANING-LINES    OCCURS 8 TIMES
                                      PIC X(70).
        05  FILLER               PIC X(37).
*
***** END OF U48551D *****
```





Problem Specifications—CCP

Create CDMF 1301 Entry - Phase I

Create an entry on CDMF 1301 for condition code 99xx (where xx is your group number). The key is language ENU, application 99, condition code 99xx, environment 0, and source 4. This entry should issue activity 14 and contain a message that you define.

Insert logic into your version of the skeleton program to handle the following situation:

```
If employee number (EMP-KEY-ID) 12346 is found,  
issue the condition code you defined on CDMF 1301.
```

The TCB data element defined by the COBOL data name TCB-USER-CC is used to pass the condition code to CCP. It is a binary field. You will also need to set the application field (99) for CCP in the TCB, TCB-USER-CC-APP.

Note: You should not change TCB-SOURCE-TYPE as it is a system-maintained field.

Determine the activity you need to use to link to CCP. Examine your program to see if the activity has been defined. If necessary, modify the program and program definition.

After CCP has been invoked, place the message for that condition code in EMP-POS-DESCRIPTION. This message will be contained in data group 48551, which you need to add to your program definition, Linkage Section, and USING statement. Therefore, you will need to perform the following steps:

1. Add DG 48551 to your program definition.
2. Include the copybook, module U48551D, in the Linkage Section of your program. A copy of the copybook is included on the previous page.
3. Add CCP-CDMF-MASTER to your USING statement.

Make sure that these three additions are in the same position relative to the other data groups.

Because your CDMF 1301 entry also issues activity 14, a dump will be generated by PEM after your report is printed.

Remove Activity 14 from CDMF 1301 Entry - Phase II

After you run the job successfully and review the dump, change your condition code by removing activity 14. In subsequent executions of the problem, you will not want to cause a dump.



Summary

CSC

Summary



- Condition Code Processing gives standard approach to user-defined processing exceptions
- Condition Codes contained on CDMF format 1301
- Item key for format 1301
 - Language Code
 - Application Code
 - Condition Code
 - Environment Code
 - Source Code

© 2000 CSC. All rights reserved.

} Halfwords

CSC

Summary



- Link activities to CCP
 - 1350 if activities issued from 1301 entry
 - 1398 if no activities executed
- CCP set up internally when application program links to CCP
- Data accessed by application from appropriate fields of data group 48551

© 2000 CSC. All rights reserved.



Umbrella Programming

Summary

- Condition Code Processing gives a standard approach for handling user-defined processing exceptions.
- Condition codes are contained on CDMF format 1301.
- The item key for format 1301 consists of: Language Code, Application Code, Condition Code, Environment Code, and Source Code.
- There are two link activities to CCP: 1350 if activities are issued from the 1301 entry and 1398 if no activities are to be executed.
- The control data group for CCP is set up internally when an application program links to CCP. Key fields within data group 48551 are moved from TCB fields, including TCB-USER-CC, which are set by the application program.
- Data can then be accessed by the application from the appropriate fields of data group 48551.

Notes:



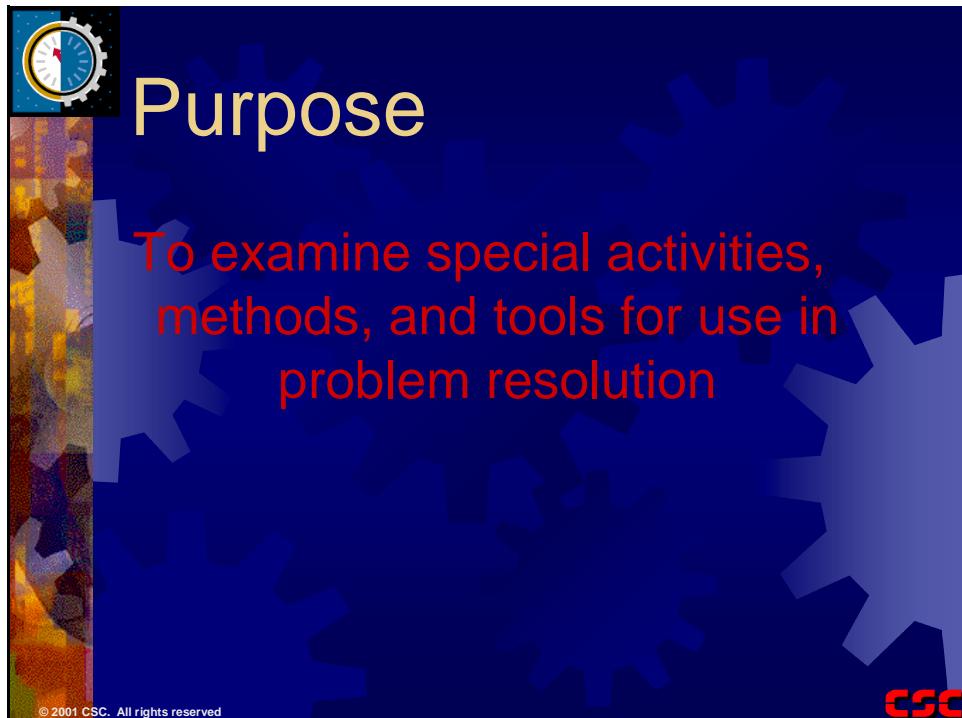
Umbrella Programming

Summary



Testing, Debugging, Dumps Under Umbrella 16

Purpose



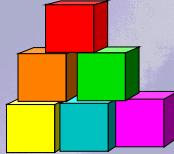
Notes:



Topics



Topics



- Umbrella System Messages and Codes
- PEM Formatted Dumps
- PEM Formatted Dump Options
- IMS Dump Options
- Dump Option Keywords
- Common Error Messages
- System Memory Access Retrieval Tool
- Activity Breakpoint Monitor (ABM)
- Dump example
- PEM dumps in batch

© 2000 CSC. All rights reserved

7/10/00

Notes:



Objectives

CSC

Objectives



To become familiar with:

- Two sources for looking up a PEM message
- PEM message on a SMART header screen (CIS)
- Key parts of a PEM formatted dump
- PEM formatted dump options set up in batch, CICS and IMS
- Dump option keywords and what they represent
- SMART commands and symbolic addresses
- PEM formatted dump example
- Research of several PEM formatted dumps

© 2000 CSC. All rights reserved

1. To identify two sources for looking up a PEM message
2. To identify the PEM message on a SMART header screen (CICS)
3. To list key parts of a PEM formatted dump
4. To explain how PEM formatted dump options can be set up in batch, CICS, and IMS
5. To list the dump option keywords and describe what they represent
6. To discuss the SMART commands and symbolic addresses
7. To walk through an example PEM formatted dump.
8. To research several PEM formatted dumps.



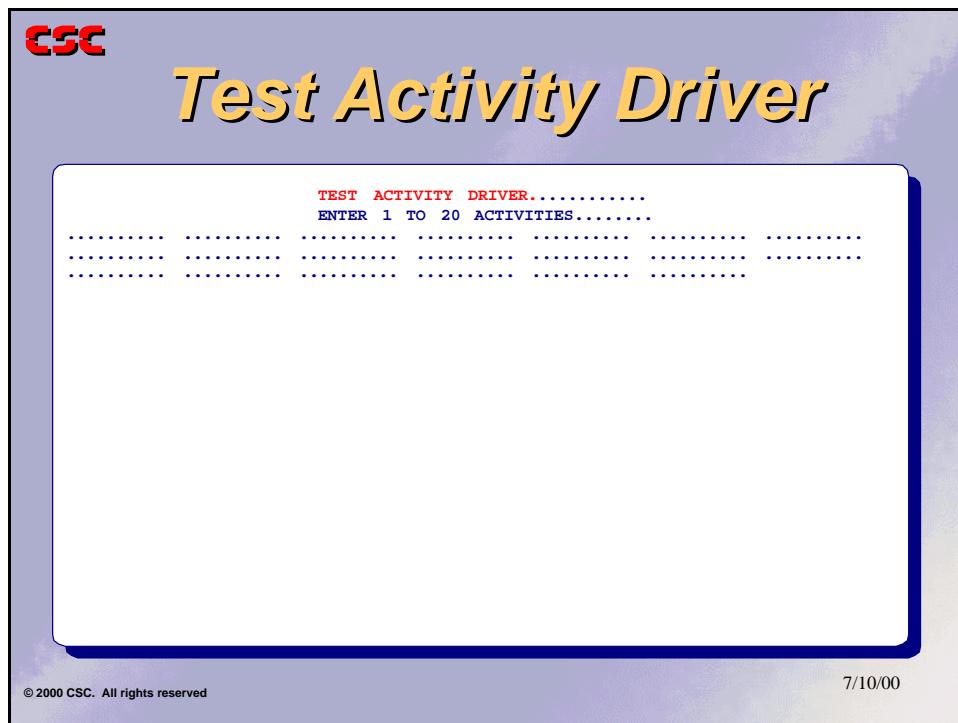
Testing

Activity Driver

The activity driver transaction can be used to execute 1 to 20 activities that require no data group or TCB parameters. It is normally used during testing or debugging so that selected activities may be executed to test a transaction. It may be used both online and batch.

Online Operation

The activity driver can be invoked online through TEST. Obtain the proper screen by issuing the TEST transaction. The following panel image will appear.



Enter the activities to be executed in the order you desire and press ENTER. Note that, in an IMS/DC environment that uses response mode, a data communications display is necessary to unlock the terminal. Activity number 911 can be used to display back the screen if no display activity will be included in the list of activities.



Batch Operation

Prior to CDMF technology, the activity driver was critical for testing. The transaction definitions were maintained in a table that had to be reassembled when a new definition or change was needed. The activity driver provided a programmer with a means for testing the transaction before modifying the table.

The batch activity driver is issued by SYSIN cards in the following format:

```
//SYSIN DD *
 1 49 904
 #905@ ACT1 @ ACT2 @ ACT3 @1@%
```

Up to 20 activity ID's can be listed. No spaces are included around the activity number because blanks are not valid input for numeric fields.

** NOTE ** If you do not end the activity list with an activity that will either end the program or the transaction, the activity driver program will issue a read activity for an additional 20 activities. This will produce unwanted results in any environment. Therefore, it is recommended that activity 1 always be the last activity in the list.

Notes:



Testing, Debugging, and Dump Activities

Dump Activity

The DUMP Activity allows the application program to get a PEM formatted dump of selected areas of storage. The data that is formatted in these dumps is controlled by the dump options set at PEM generation time. There is a special transaction available that allows you to override these options.

Several processing options can be specified by the Dump activities.

Predefined activities for dump

- ACTIVITY 14—FULL DUMP DISP=WAIT.
- ACTIVITY 15—FULL DUMP DISP=EOT.
- ACTIVITY 16—TCB DUMP ONLY DISP=WAIT.
- ACTIVITY 17—USER SUPPLIED DATA GROUP ID, DISP=WAIT.
- ACTIVITY 18—USER SUPPLIED DATA GROUP ID, DISP=END.

CSC *Testing, Debugging and Dump Activity*

48305 ACTION COMPLETE (DUMP ACTS)

ACTIVITY	DEFINITION	INQUIRY/MAINTENANCE
COMMAND ==> INQ		OWNER ==> PEM CC# ==>
ACTIVITY ID ======>	15	EFF DATE ==> 78/01/01
ACTIVITY MNEMONIC ==>		TYPE =====> DUMP
DESCRIPTION ======>	DUMP AND END TRANS.	

Dump Activity

DUMP
DISPOSITION ==> EOT
SPECIAL OPTIONS: ALL CONTROL BLOCKS & DATAGROUPS ==> Y
TCB & ACTIVITY TRACE ONLY =====> N
USER SUPPLIED DATAGROUP ONLY ==> N
*** DATA GROUPS ***
* DGID * DGID * DGID * DGID

LAST CHG:DATE 92/06/15 TIME 9.35.40 CC# SRCE UMB130 OPER
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 11-CPY ..-NXTT ..-NEW ..-NXTA ..-DEL

© 2000 CSC. All rights reserved



Exception Activity

The exception activity enables the user program to log exception messages using the internal PEM mechanism for exceptions. An MVS condition code can also be set as the job step condition code in OS/VS batch environments. A data group can be specified to contain a message for the exception. Data base rollback can be requested. The PEM mechanism for exceptions is different for each environment.

For online processing, messages are displayed to the terminal that originated the transaction and logged either to the online journal file or the master terminal. In batch processing, messages are written on an exceptions report.

Data Base Rollback Request

If coded on the exception activity, a rollback request causes data base updates to be backed out. In addition, it sets the job step condition code and prints a message.

Note: The rollback of data base updates will only take place in environments that support rollbacks, such as IMS/DC, BMP, or CICS with DTB support. PEM does not log before/after DB images for rollback use.

Delivered Exception Activities

- Activity 22—CC=00,DG=0,DISP=WAIT, user-supplied CC only.
- Activity 23—CC=00,DG=9,DISP=WAIT, user-supplied CC and MSG in data group 9.
- Activity 24—CC=12,DG=9,DISP=EOT,ROLLBACK=YES MSG in data group 9 and EOT.
- Activity 25—CC=00,DG=9,DISP=END, user-supplied CC and MSG in data group 9. End program.
- Activity 26—CC=00,DG=9,DISP=EOT. Same as activity 25 with EOT.
- Activity 27—CC=12,DG=0,DISP=WAIT. Set CC=12.
- Activity 30—CC=00,DG=0,DISP=WAIT,ROLLBACK=YES. Rollback only.
- Activity 33—CC=00,DG=9,DISP=EOT,ROLLBACK=YES. Rollback only.
- Activity 35—This link activity invokes a program that fills in the message number of data group 9 with the value from the Transaction Control Block field, TCB\$USCC (ASSEMBLER) or TCB-USER-CC (COBOL). The message text is then obtained from Condition Code Processing PCD 1301. Exception activity 39 is issued to abend with the exception message.
- Activity 39—CC=12,DG=9,DISP=ABND, MSG in data group 9 and abend.

Note: Because the name of the program issuing this activity is displayed on the front of the message, the message description on PCD 1301 should not exceed 35 characters.



Umbrella Programming

Testing, Debugging, and Dump Activities

Exception Activity

CSC

Exception Activity

```
48307 ACTION COMPLETE (EXCEP ACTS)
----- ACTIVITY DEFINITION INQUIRY/MAINTENANCE -----
COMMAND ==> INQ OWNER ==> PEM CC# ==>
ACTIVITY ID =====> 23 EFF DATE ==> 78/01/01
ACTIVITY MNEMONIC ==> TYPE =====> EXCEP
DESCRIPTION =====> EXCPN, CC=0, DG=009

EXCEPTION

CONDITION CODE ==> DATA GROUP =====> 9
DISPOSITION =====> WAIT DB ROLLBACK? ==> N

LAST CHG:DATE 92/06/15 TIME 9.35.40 CC# SRCE UMB130 OPER
PF: 4-CHG 5-ADD 6-INQ 9-NXT 10-DGID 11-CPY ..-NXTT ..-NEW ..-NXTA ..-DEL
```

© 2000 CSC. All rights reserved

7/10/00

CSC

Exception Activity - 2

```
48307 ACTION COMPLETE (EXCEP ACTS)
----- ACTIVITY DEFINITION INQUIRY/MAINTENANCE -----
COMMAND ==> INQ OWNER ==> PEM CC# ==>
ACTIVITY ID =====> 24 EFF DATE ==> 78/01/01
ACTIVITY MNEMONIC ==> TYPE =====> EXCEP
DESCRIPTION =====> EXCPN, CC=12, DG=9

EXCEPTION

CONDITION CODE ==> 12 DATA GROUP =====> 9
DISPOSITION =====> EOT DB ROLLBACK? ==> Y

LAST CHG:DATE 92/06/15 TIME 9.35.40 CC# SRCE UMB130 OPER
PF: 4-CHG 5-ADD 6-INQ 9-NXT 10-DGID 11-CPY ..-NXTT ..-NEW ..-NXTA ..-DEL
```

© 2000 CSC. All rights reserved

7/10/00



Exception Activity Vs Dump Activity

CSC

Exception Activity vs. Dump Activity

- **Activity 35 = dump and die**
- **Activity 15 = ends a transaction**
- **Activity 35 = link activity**
 - TCB-USER-CC with PCD 1301 indicates specific abend error
 - Activity 35 = causes PCD 1301 to be read
- **Activities 14 and 15 = cause dump to file**
- **Activities 12 or 35 cause smart screen**

© 2000 CSC. All rights reserved

The preferred approach to end a transaction and cause a dump (dump and die) is to issue activity 35. Activity 15 is a dump activity that ends a transaction. While activity 15 can be used in testing, it should never be used in production because it does not set a return code for the operator.

Activity 35 is a link activity to a program that issues the exception activity 39. TCB-USER-CC is set to a condition code defined to PCD 1301 to indicate the specific abending error. Activity 35 causes PCD 1301 to be read, the message to be moved into data group 9, and the response code to the operator to be set to 12 for the step.

Activities 14 and 15 are supported online and cause a dump to the dump file. In the CICS environment, this eliminates invoking a SMART screen. Use either activity 12 or 35 to cause a SMART screen.

Notes:



Dumps

Dumps: Friend or Fiend?

Every data processor has generated a dump. Some are easier to solve than others. PEM creates a formatted dump that eases the solution of an abend.

PEM will generate a dump on an abend condition or when a dump activity is issued. The dump is identified with labels for important components. An error message is printed.

How to Approach the Abend

Knowing what caused the problem helps solve it. The first thing to do is to read the PEM message. Many problems can be solved without looking at any HEX output. Review the message to help identify where the problem occurred.

PEM messages are identified with a five digit number in the 49000 range. There are two sources for looking up the message. Umbrella System Messages and Codes Manual provides a listing of the messages and an explanation of their meanings. The messages are also stored in an online table, CDMF format 1301.

Dump Methodology

1. Examine Exception Message Number is CCP 1301 CCP Key
2. Common Error Message Section
3. Activity Trace Table
4. Process Dictionary Displays
5. Source Code - Use Activity Trace
6. Dump Diving
7. More eyes after 1 hour
8. 1-800-99Hogan

Website – www.BANKING.CSC.com

© 2001 CSC. All rights reserved

csc



Common Error Messages

The following messages relate to common problems encountered when first working with the Umbrella System:

PROBLEM	POSSIBLE CAUSES
49005 LOAD FAILED FOR Z999913	COBOL program not compiled. Linkname on program definition is different from the name used to linkedit the program to the LOADLIB. The JCL points to wrong LOADLIB.
49006 TRANSACTION NOT DEFINED	The SYSIN CO/APP/FUNC invalid. Transaction definition was not defined.
49008 ACTIVITY NOT AUTHORIZED	Activity not on the program definition. Activity ID loaded to TCB-LONG-ACTIVITY contains the wrong value. Activity ID defined as halfword (fullword program).
49017 PROGRAM NOT DEFINED 00000047913 850731	The program ID specified on the link activity has not been defined. The program ID specified on the link activity is wrong.
49049 DG-47130 NOT IN DB- EMP	The order of data groups on data base activity is not the same as the data base definition. The data group is not part of the data base. Only data groups from the same data base may be specified on the HDB activity. The program ID specified on the link activity is wrong.
49057 ACTIVITY NOT DEFINED	The activity definition for the activity issued has not been defined. The activity ID issued is wrong.
49088 ACTION 007 CONFLICTS WITH USE. DG47110	Positional data groups are non-keyed and cannot be accessed with a KEY action code. Sequential actions cannot be used with random uses.



Umbrella Programming

Common Error Messages

PROBLEM	POSSIBLE CAUSES
S0C4 PROTECTION EXCEPTION (IF PSW IS WITHIN THE APPLICATION PROGRAM)	Data group referenced in COBOL program but not defined in the program definition. Dynamic data group referenced prior to allocation. The order of the data groups in the program definition, COBOL linkage section and the Using Statement are not the same. Data group length in COBOL program is greater than the data group definition. Data base access involving the last activity issued has not completed successfully.
S0C7 DATA EXCEPTION	The order of the data groups in the program definition, COBOL linkage section and the Using Statement are not the same. Data group not field initialized. Access to data in data group with DG-RESULT not DGR-OK. Logic error.

Notes:



Dump Options

Option Keywords—Batch

Several special options are shown below.

KEYWORD	DESCRIPTION
DUMP	SYSUDUMP
NOPDUMP	NO PEM dump, suppress PEM abend recovery support. SYSUDUMP/partition dump automatic.
TEST	Test mode allows some exceptions to continue.
MON	Start Umbrella monitor.
=XXXXX	Force an abend when exception XXXXX occurs. Only one exception can be active at a time for this option.
<pre>//PEMPARM DD * DUMP, TEST or MON or DUMP, MON /*</pre>	

PEM Formatted Dump Options

PEM produces a formatted dump when an abend or dump activity occurs. The dump process automatically prints a dump header. The application-related data includes a formatted user transaction control block (TCB), a list of recent activities, the registers at abend, and the contents of all allocated data groups.

This dump will also format PEM control blocks, based on parameters carried in the PEM vector table (PVT). Normally, most of these control blocks are not needed to solve problems. To simplify and shorten the dumps produced, PEM is distributed with options set to dump only the ITCB, UPCB, and PVT. The default dump options are set when PEM is generated.

It is sometimes necessary to change the dump options only for the duration of a transaction or job. In this situation, a transaction may be run to change the default dump options. For batch environments, include this transaction:

```
1 49 707
#7@ OPT1 @ OPT2 @ OPT3 @ ... @%
```



Umbrella Programming

Dump Options

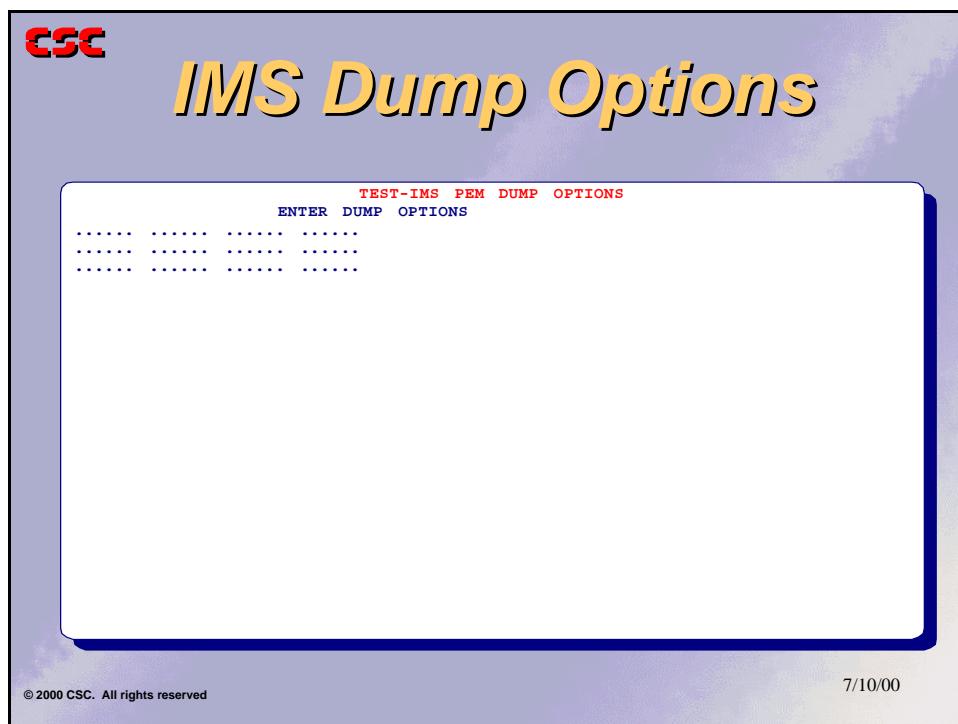
Online Dump Options

Using the PEMR transaction, you can turn the monitor utility on in an online region. During installation, default dump options are set. TS06 can change the options.

To obtain the input screen from a cleared screen, enter:

TS06

The dump options screen appears as follows:



Enter the desired dump options and press ENTER.

Options MON, USAGE, NOMON, and NOUSAGE may be set from this screen.

CAUTION:

The monitor tracks the executing activities, the EXCPs, the time used by the activities, and the issuer of the activities. This information is written to a sequential file that becomes input for batch reports. Monitor should be used during low-volume, controlled periods. The prompt for IMS is TS06 or /FOR P49707MM. If monitor is to be activated, the monitor data set must be added to the message region.



Umbrella Programming

Dump Options

The keywords turn on a given option; to turn off an option add NO to the keyword (for example, NOPVT prevents printing of the PVT in the dump). Unless the combination of options used resets the default options, those options will remain in effect. The reset is additive, not a complete replacement.

PEM is distributed with options set to dump only the ITCB, UPCB, and PVT. Some examples of the other options and descriptions are listed below.

<u>KEYWORD</u>	<u>DESCRIPTION</u>
ACTS	Activity definition index and tables.
ALL	Set all options.
CDMF	All Common Data Management Facility indexes, tables, and work areas hierarchical data base company table.
DBWA	Data base work areas.
LOAD	Load module table.
PCB	DL/I PCD list.
PEM	PEM load module.
PGMS	Program definition index and tables.
PVT	PEM vector table.
SORT	Sort activity work area.
TRAC	Activity trace table.
UPCB	User Program Control Block.

Notes:



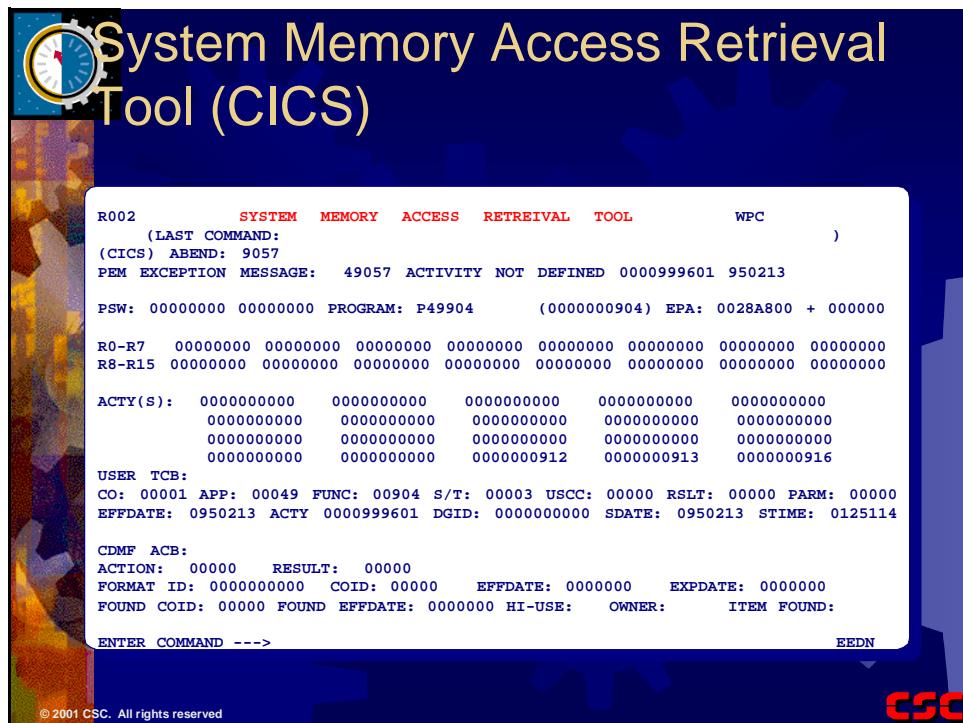
Umbrella Programming

Dump Options

System Memory Access Retrieval Tool (CICS)

Included in the CICS version of the Umbrella System is the System Memory Access Retrieval Tool (SMART). SMART can run stand alone or as an abend exit of a PEM transaction. The SMART header display contains the last command, CICS abend, PSW, abending program ID, registers, the last 10 executed activities, formatted TCB and PEM exception message. Using a facility called PEM Formatted System Dump or PFSD for short, you can send the dump to a sequential file for later printing in batch.

The following is an example of a SMART header screen which is generated with the Tran Code “Kore.”



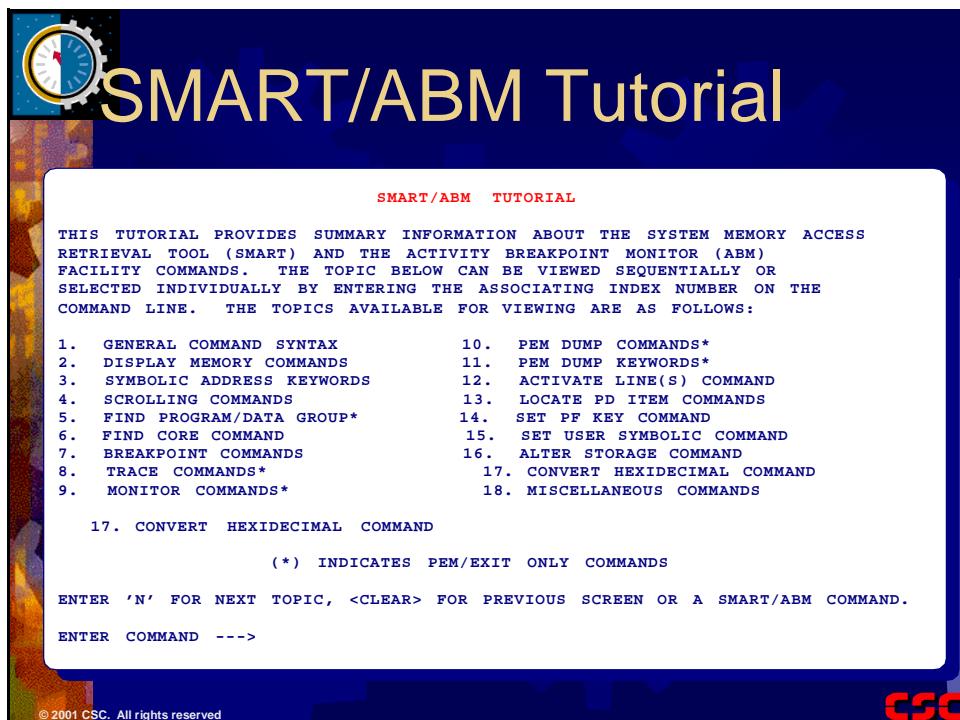
Notes:



System Memory Access Retrieval Tool

SMART/ABM Tutorial

The following screen is displayed using the SMART Command of H or HELP. The main menu contains a list of SMART command summary topics that can be selected specifically or sequentially as listed. Each topic provides a summary descriptions of SMART commands and facilities with command syntax and examples. Specific topic viewing is initiated by entering the topic number, from the menu, on the COMMAND line. To view the next sequential topic, enter N on the COMMAND line, from any of the tutorial screens, and press ENTER. The tutorial can be terminated from any screen by pressing the CLEAR key. This will cause a return to the screen prior to entering the tutorial.



Notes:



SMART Commands and General Syntax



Smart Commands and General Syntax

■ Operation <OPERAND LIST>

- Explicit Address
- Current Location
- Symbolic Address
- Combined Format

© 2000 CSC. All rights reserved

Aside from the initial SMART screens, the data and/or screens displayed are controlled by operator entered commands. The actual commands are discussed in detail later in this section, but the basic command syntax is:

OPERATION <operand list>

where OPERATION is a single or multiple character command code and the OPERANDLIST is one or more command operands required to execute the command.

The delimiter that separates the OPERATION from to OPERAND LIST varies from command to command, however, in most cases a blank, comma (,) or equal sign (=) can be used. Operands within the OPERAND LIST are generally separated by a comma or blank. The commands definitions will define any specific requirements.

The previous command can be repeated by simply pressing ENTER without entering any data on the COMMAND line. This can be useful when tracing a series of pointers that are always at the same offset. For instance, the command I*+4 is entered to trace a series of control blocks.

Most commands require hexadecimal addresses for operands. These addresses may be entered in several ways:

EXPLICIT ADDRESS A 1- to 8-character hexadecimal values representing a valid storage address. Simple arithmetic can also be used with valid values.



Examples:

104
AD8944
1D6+E90
E-4+103

CURRENT LOCATION * (asterisk). Use the current display address.

SYMBOLIC ADDRESS A predefined or user-defined symbol that equates to an explicit storage address. These symbols are used in place of explicit addresses to point to commonly used storage, control blocks, and/or registers.

COMBINED FORMAT A combination of any of the other address formats. Notice that only one symbolic address can be used in a given command and should appear between parentheses.

Examples:

104(PVT)
*+AD8
13DC(ITCB)+1468-C

SMART PF Keys

PF1 => HELP	PF13 => HELP
PF2 => ?	PF14 => ?
PF3 => END	PF15 => END
PF4 => B=SKIP;END	PF16 => B=SKIP;END
PF5 => FD=>	PF17 => FD=>
PF6 => ZAP	PF18 => ZAP
PF7 => @=SB>	PF19 => @=SB>
PF8 => @=SF>	PF20 => @=SF>
PF9 => A=1,10	PF21 => A=11,20
PF10 => P=DUMP;END	PF22 => P=DUMP;END
PF11 => @=ORIGIN	PF23 => @=ORIGIN
PF12 => RETREIVE	PF24 => RETREIVE

Notes:



Umbrella Programming

System Memory Access Retrieval Tool

Symbolic Addresses

Symbolic addresses are used to reference storage areas, within SMART storage display commands, without actually entering the explicit address. The symbols are stored in an in-core table with their associating storage addresses. Within the DISPLAY command, the symbol (once found in the table) is replaced by the associating explicit address before processing. This allows the operator to display commonly used storage areas without concern for their explicit addresses. Considering most storage areas and control blocks addresses change from SMART session to SMART session, this facility simplifies displaying these area considerably.

There are two types of symbolic addresses: predefined and user-defined. Predefined symbolics are setup once SMART is invoked. User-defined symbolics can be added, once a SMART session has begun, to allow individually customized symbols to be used in the same manner. The symbolics predefined in the symbol table depend upon how SMART was invoked. For example, if SMART is invoked as a PEM abend exit, most control blocks and storage areas associated with PEM transactions will be present in-core and will be defined to the symbol table (such as, the TCB and the Activity Trace Table). However, if SMART is invoked by the stand-alone transaction (KORE), none of these PEM storage area and control block will be present. Consequently, the uses of these symbolics will cause the BASE SYMBOL NOT DEFINED message to appear.

Note: When in doubt as to whether a symbol is valid, the symbol table can be displayed. If the symbol appears with no associating address, the symbol is not valid.

No matter how SMART is invoked, some symbolic addresses are always available. These symbols point to common OS and CICS control blocks and SMART work areas. The symbols are:

CVT	Communication Vector Table
EIB	CICS Execute Interface Block (for the SMART Module)
EIS	CICS Dynamic Storage Area (for the SMART Module)
* PVT	PEM VECTOR TABLE
* PEM\$xxxx	PEM Control Blocks (chained off the PVT).

Note: Asterisk (*) denotes PEM storage area symbols that will be present if at least one PEM transaction has executed in the CICS region. This also includes all Process Dictionary data and index table symbolics not prefixed with PEM\$.

The following symbols, CSA, DCT, FCT, PCT, PPT, TACB, TCA, TCT, TCTTE, TCTUA, and TRT, available in previous releases, are no longer valid.

The contents and uses of these control blocks are defined in various CICS publications. Consult your CICS system programmer for additional information.



Umbrella Programming

System Memory Access Retrieval Tool

The following is a complete table of all the predefined symbols:

SYMBOL	DESCRIPTION
ACB	CDMF Application Control Block
ACTS	Activity definition index
CDMF\$PMA	CDMF Pool Management Area
CDMF\$STAT	CDMF Formats Statistics Table
CDMF\$TOT	CDMF Table of Tables
CVT	Communication Vector Table
DG	Data group (variable, reset by FD= command)
DGIX	Allocated data group index
DGS	Data group definition index
DBWA	Data base work area
EIB	CICS Execute Interface Block (for the SMART Module)
EIS	CICS Dynamic Storage Area (for the SMART Module)
EVT	PEM Error Vector Table
ICB	CDMF internal control block
ITCB	PEM internal TCB
MAPS	Map definition index
PEM	PEM load module
PEM\$ALST	Activity definition index
PEM\$APP	Transaction code conversion table
PEM\$CSIX	Common Storage Control Area pointer
PEM\$DBIX	HDB index
PEM\$DGNX	Temporary data group index
PEM\$DVCT	DCT



Umbrella Programming

System Memory Access Retrieval Tool

PEM\$ENQC	Enqueue Chain
PEM\$HCOL	HDB company list table
LOADLIST	PEM load list
PEM\$MAPL	Map index
PEM\$PDG	Permanent data group index
PEM\$PDGC	PERM DGS Index for CSA
PEM\$PLST	Program definition index
PEM\$PSA	Permanent Extended Storage
PEM\$SAML	Sequential data base index
PEM\$SCVT	MVS Subsystem Communication Table entry
PEM\$SITE	Permanent site control data group
PEM\$TBLS	Tabling Statistics Site Permanent DG
PEM\$TCOL	Transactions/maps company list table
PEM\$TLST	Transaction definition index
PGM	Load address of abending program (variable, reset by FP= command)
PGMS	Program definition index
PSB	IMS PSB address list
SYMBOL	DESCRIPTION
PSW	PSW at entry to abend (available only after CICS ASxx abends)
PVT	PEM Program Vector Table
REGS	Registers at entry to abend if SMART is invoked as an abend exit. Otherwise, the registers of the user program.
RSA	Register save area chain



Umbrella Programming

System Memory Access Retrieval Tool

Rnn	Where nn is 00 to 15. When used with the display storage command, this will be a pointer to the register value in storage. When used with the indirect display command, this will be the pointer to the storage area referenced by the register.
SQLCA	SQL communication area
SQLMSG	SQL translated exception message
SWA	Sequential data base work area
TAI	PEM Task Abend Information area
TIN	PEM BMS input/output mapping work area
TRAN	Transaction definition table
UPCB	Current PEM User Program Control Block
UTCB	User Transaction Control Block
UTIN	CICS original terminal I/O area

Notes:

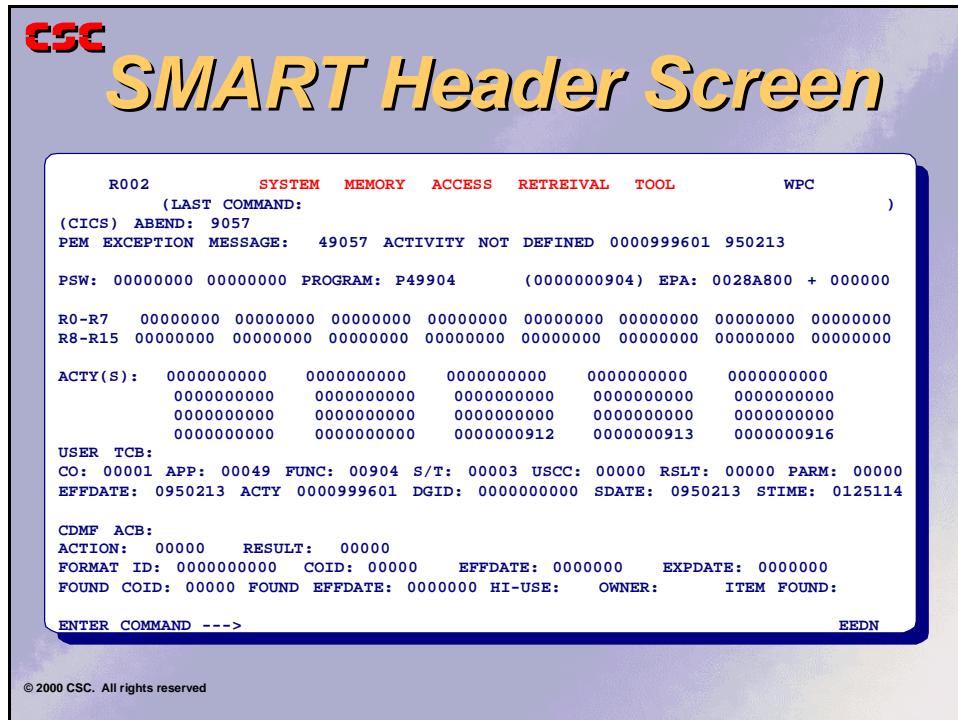


Umbrella Programming

System Memory Access Retrieval Tool

SMART Dump

SMART Header Screen



Notes:



SMART Storage Display Screen

The screenshot displays the "SMART Storage Display" interface. At the top, the CSC logo is visible. Below it, the title "SMART Storage Display" is centered in a large, stylized font. The main area contains a table with columns: LAST, COMMAND:, SYSTEM, MEMORY, ACCESS, RETREIVAL, TOOL, and WPC. The table lists numerous memory access entries, such as command D(PEM) with various parameters like address ranges and data values. At the bottom of the table, there is a prompt "ENTER COMMAND ---->" and the identifier "EEDN". A copyright notice at the bottom left reads "© 2000 CSC. All rights reserved".

LAST	COMMAND:	SYSTEM	MEMORY	ACCESS	RETREIVAL	TOOL	WPC
00224028	+00000	90ECD00C	18AF189A	4A90A014	4510A016	*5.....*
00224038	+00010	0B0C0000	100058F0	92F805EF	50D01004	*0.8..&...*
00224048	+00020	18F1BF1F	D0184780	A030D207	F05C1000	*	.1.....K.0*..*
00224058	+00030	18DF5880	D05C47F0	A05A1FD7	F4F9F0F0	**.0!.P4900*
00224068	+00040	F4C34040	E4D4C2F2	F0F04040	F1F161F0	*	4C UMB200 11/0*
00224078	+00050	F561F9F4	40F1F24B	F0F647F0	A40CE4D4	*	5/94 12.06.0..UM*
00224088	+00060	C2C6C9E7	404040C3	D6D7E8D9	C9C7C8E3	*	BFIX COPYRIGHT*
00224098	+00070	404DC35D	40C8D6C7	C1D540E2	E8E2E8C5	*	(C) HOGAN SYSTE*
002240A8	+00080	D4E26B40	C9D5C34B	6B40F1F9	F9F44040	*	MS, INC., 1994*
002240B8	+00090	C1D3D340	D9C9C7C8	E3E240D9	C5E2C5D9	*	ALL RIGHTS RESER*
002240C8	+000A0	E5C5C440	D3C9C3C5	D5E2C5C4	40D4C1E3	*	VED LICENSED MAT*
002240D8	+000B0	C5D9C9C1	D3E260D7	D9D6D7C5	D9E3E840	*	ERIALS-PROPERTY*
002240E8	+000C0	D6C640C8	D6C7C1D5	40404040	00000000	*	OF HOGAN*
002240F8	+000D0	002265A8	002272F0	00228308	00224028	*0.....*
00224108	+000E0	00228AA0	00228BD8	00229078	00229618	*Q.....*
00224118	+000F0	00229C20	0022CC50	0022DCF8	00000000	*&...8....*
00224128	+00100	0022E808	0022F330	00230550	00231118	*	..Y...3....&...*
00224138	+00110	00000000	00231370	00232528	00000000	**
00224148	+00120	00238540	00238858	00266560	00239C88	*-....*
00224158	+00130	0023BA80	0023BC88	0023C070	0023C2E0	*B.*

ENTER COMMAND ----> EEDN

© 2000 CSC. All rights reserved

Activity Breakpoint Monitor (ABM)

The Activity Breakpoint Monitor can halt execution of any PEM transaction before, after, or both before and after any or all activities. User commands are provided to select any subset of activities to breakpoint. When a breakpoint is triggered, an ABM screen is presented to the user with all ABM and SMART commands available. Ending an ABM session (with the END command) resumes the transaction at the breakpoint.

This facility also allows trace reports and monitor files to be written from CICSPEM.

Notes:



Umbrella Programming

System Memory Access Retrieval Tool

Using ABM

The best way to start using ABM is as follows:

1. Enter KORE.
2. Enter B=FIRST to ensure that the first activity issued by PEM has a breakpoint set and transfer control to the ABM.

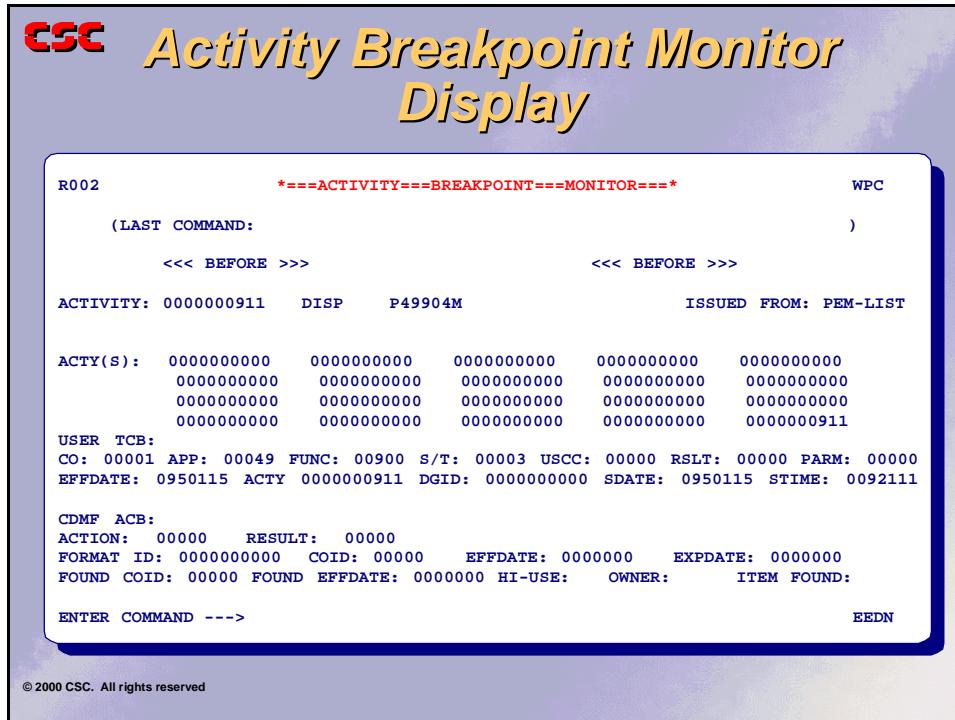
You may also set special PF keys at this time.

3. END the KORE session and enter the Umbrella transaction to be breakpointed.
4. An ABM screen is displayed BEFORE the first PEM activity. At this point you may:
 - Set or reset the breakpoints to a smaller set of activities
 - Enter B=SKIP (default value of PF4/16) to skip all other breakpoints in this transaction (for example, a prompt transaction)
 - Start a Trace report (this automatically skips breakpoints)
 - Start PEM writing a monitor file (breakpoints skipped)
 - Start both Trace and Monitor
 - Enter any SMART commands to examine or alter storage
 - Enter END to end the ABM session.
5. After ending the ABM session, the next session starts. Each activity being breakpointed generates an ABM session. They continue until the transaction ends.

Note: When a DC DISP activity is breakpointed, the screen is displayed for three seconds, then the next ABM session (for example, AFTER display) starts. When the transaction ends, the screen displayed by the DC DISP activity will be redisplayed.



Activity Breakpoint Monitor Display



Notes:



Umbrella Programming

System Memory Access Retrieval Tool

PEM Formatted Dumps

A PEM formatted dump reduces debugging time. It generates less output. Because each data group is dumped separately there is no need to dump all program storage.

The order of the dump is as follows:

1. SYSIN STATEMENTS TO PEM.

This page contains the SYSIN cards that are source to PEM. These cards identify the PEM transaction that is executing.

2. PEM EXCEPTION REPORT

The header page prints abending time, date, job name, error number, error description, company number, application, function, source type, abending program ID and abending activity ID.

3. FORMATTED DUMP SUMMARY

This page contains summary data for the dump. The PSW at abend is followed by the registers. Next is the abending program ID with offset and abending-while-executing information. The formatted TCB is next. The CDMF Application Control Block has information on the last format (file) processed.

4. TRACE TABLE

The Trace Table lists the last 20 activities executed. The most current is at the bottom of the list. By pairing link activities to end activities, you can follow the activities issued within each of the programs.

5. INTERNAL PEM CONTROL BLOCKS

The Internal Transaction Control Block (ITCB) is in dump format and contains information based on the dump options directly related to PEM.

6. USER PROGRAM CONTROL BLOCKS

Each program executed by the transaction has its own UPCB. The first UPCB listed is the last active program at time of abend. The control block contains the registers of the last call to PEM, address lists, and the program parameter list beginning at offset 000070. The program parameter list contains the positional parameter number followed by the address of the location of the parameter. Positional parameter 00 is the address of the location of the UTCB.

7. DATA GROUP STORAGE

Each data group is printed separately and listed in numerical data group ID sequence.



8. PEM FORMATTED DUMP INDEX

Index cross-references all items in a PEM formatted dump with the page and hex address location within the dump.

Example of a PEM Formatted Dump

The suggested steps for debugging a PEM formatted dump are to examine the parts of the dump in the following order:

1. PEM Exception Report (Header)
2. Formatted Dump Summary
3. Trace Table
4. User PCB
5. Data Group Fields

The instructor will walk through the following example of a PEM formatted dump. The key parts of the dump will be discussed.

```
1 99 9933
10:03:42 WPC$$DMP 49008 ACTIVITY NOT AUTHORIZED
00001 00099 09933 00004 0000999413 0000999513
P E M FORMATTED STORAGE DUMP EXCEP TION DATE=98/02/23 TIME=10:03:42 PAGE 0002
EXCEPTION MESSAGE: 49008 ACTIVITY NOT AUTHORIZED

FORMATTED USER TCB --
CO: 00001 APP: 00099 FUNC: 09933 S/T: 04 TIME: 100342 DATE: 0980223 SOURCE: WPC$$DMP OPER:
ACT: 00000 RSLT: 00000 USER: 000000000000 DG: 05610 EFF DATE: 0980223 DEST: PARM: 00400
XACT: 0000999513 XDG: 0000005610
SQL FIELDS -- ACTN: 00000 RSLT: 00000 D2CC:

CDMF APPLICATION CONTROL BLOCK
ACT: 00005 RSLT: 00000
KEY FIELDS --
FORMAT ID: 0000002015 COID: 00001 EFF DATE: 0980223 EXP DATE: 0990101
FOUND COID: DFLT FOUND EFF DATE: 0980101
FLAGS ----- HI USE: N OWNER APP: ITEM FOUND: N
LAST CHANGED -- DATE: 0950211 TIME: 153316 CHG CNTL NUM: 0000000001 SOURCE: UMB200E6 OPER:
CHANGE CONTROL NUMBER: 0000000000 SECONDARY KEY: SUB DGID: 0000000000
P E M FORMATTED STORAGE DUMP EXCEP TION DATE=98/02/23 TIME=10:03:42 PAGE 0003
ACTIVITY TRACE TABLE
ACTY ID TYPE
0000000001 END
0000000006 WORK AREA DG ID = 0000000000
0000048000 LINK PGM ID= 0000048000
```



Umbrella Programming

System Memory Access Retrieval Tool

```
0000000001 END
0000048000 LINK PGM ID= 0000048000
0000000001 END
0000048000 LINK PGM ID= 0000048000
ACT: 00005 RSLT: 48002 FMT: 0000048312 CO: DFLT EFF: 0980223 FOUND CO: XXXXX EFF: XXXXXXXX
0000000001 END
0000048000 LINK PGM ID= 0000048000
ACT: 00005 RSLT: 00000 FMT: 0000048620 CO: DFLT EFF: 0980223 FOUND CO: DFLT EFF: 0780101
0000000001 END
0000048000 LINK PGM ID= 0000048000
ACT: 00005 RSLT: 00000 FMT: 0000048926 CO: DFLT EFF: 0980223 FOUND CO: DFLT EFF: 0780101
0000000001 END
0000048000 LINK PGM ID= 0000048000
ACT: 00005 RSLT: 00000 FMT: 0000048512 CO: DFLT EFF: 0980223 FOUND CO: DFLT EFF: 0780101
0000000001 END
0000048000 LINK PGM ID= 0000048000
ACT: 00005 RSLT: 48002 FMT: 0000048312 CO: DFLT EFF: 0980223 FOUND CO: XXXXX EFF: XXXXXXXX
0000000001 END
0000000001 END
0000999413 LINK PGM ID= 0000999413
0000047921 LINK PGM ID= 0000001850
0000000004 WORK AREA DG ID = 0000000000
P E M FORMATTED STORAGE DUMP EXCEP TION DATE=98/02/23 TIME=10:03:42 PAGE 0004
0000000004 WORK AREA DG ID = 0000000000
0000000004 WORK AREA DG ID = 0000000000
0000001260 LINK PGM ID= 0000001860
0000001287 LINK PGM ID= 0000001280
0000048995 LINK PGM ID= 0000048995
0000000001 END
0000048000 LINK PGM ID= 0000048000
ACT: 00014 RSLT: 00000 FMT: 0000002350 CO: DFLT EFF: 0780101 FOUND CO: DFLT EFF: 0780101
0000000001 END
0000000001 END
0000001287 DYNAMIC ACTIVITY
0000048995 LINK PGM ID= 0000048995
0000000001 END
0000048000 LINK PGM ID= 0000048000
ACT: 00014 RSLT: 00000 FMT: 0000002350 CO: DFLT EFF: 0780101 FOUND CO: DFLT EFF: 0780101
0000000001 END
0000000001 END
0000000004 DYNAMIC ACTIVITY
0000000004 DYNAMIC ACTIVITY
0000000001 END
0000000001 DYNAMIC ACTIVITY
0000001900 LINK PGM ID= 0000001600
0000001013 LINK PGM ID= 0000048000
ACT: 00005 RSLT: 00000 FMT: 0000002005 CO: 00001 EFF: 0980223 FOUND CO: DFLT EFF: 0880101
0000000001 END
0000048000 LINK PGM ID= 0000048000
ACT: 00005 RSLT: 00000 FMT: 0000002015 CO: 00001 EFF: 0980223 FOUND CO: DFLT EFF: 0980101
0000000001 END
0000000001 END
```



Umbrella Programming

System Memory Access Retrieval Tool

```

00000001900 LINK PGM ID= 00000001600
P E M FORMATTED STORAGE DUMP E X C E P T I O N DATE=98/02/23 TIME=10:03:42 PAGE 0005
0000000001 END

0000999613 DATA BASE DB ID = EMP
P E M FORMATTED STORAGE DUMP E X C E P T I O N DATE=98/02/23 TIME=10:03:42 PAGE 0006
INTERNAL TRANSACTION CONTROL BLOCK
000F6010 000000 0000B500 00006140 00320032 0004C058 0004C040 0004C108 0004C010 11000FF8 *...../.....A.....8*
000F6030 000020 00064A70 000F6010 00000000 00000000 0004CC98 00EC0F57 00010063 26CD0004 *.....-.....*.
000F6050 000040 0100342F 0980223F E6D7C35B C45C4D47 40404040 40404040 40404040 40404040 *.....WPC$SMP *.
000F6070 000060 40404040 0005B250 0B808878 000100D7 C5D4C4C2 C4C5C600 0005D008 00000000 *.....PEMDBDEF *.
000F6090 000080 057C057C 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
000F60B0 0000A0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
000F60D0 0000C0 0005C028 00000000 00000000 00000000 00000000 0B801824 0B801848 *.....*.
000F60F0 0000E0 D7E2C2C5 C4E4C140 00000000 00000000 C24000000 F0404040 40404040 *PSUEDUA.....B.....0 *.
000F6110 000100 C3C8C1D5 C7C54C03 D6D5E3D9 D6D340D5 E4D4C2C5 D97A4040 FOFOPFO FOFOFOFO *CHANGE CONTROL NUMBER. 00000000*.
000F6130 000120 F0F04040 40404040 40404040 40404040 40404082 C5C3D655 C4C1D9E8 40D2C5E8 *00 SECONDARY KEY*.
000F6150 000140 7A404040 40404040 40404040 40404040 40E2B4C2 40C4C7C9 C47A4040 FOFOPFOFO *.....SUB DGID. 0000*.
000F6170 000160 FOFOPFO FOFOPD6 D7C5D97A 40404040 40404040 40404000 00000000 00000000 *000000 OPER. ....*.
000F6190 000180 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....WPC$SDMPSTEP1 JS01*.
000F61B0 0001A0 F0404040 00000000 00010000 00000000 00000000 00000000 00000000 *0.....*.
000F61D0 0001C0 00000000 00000000 00000000 00000000 00000000 0005C168 00000000 00000000 *.....A.....*.
000F61F0 0001E0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
SAME
000F6250 000240 00000000 00000D93 00000001 00007CC8 00000000 000018D8 000001F0 000001F0 *.....H.....Q.....0 *.
000F6270 000260 00000D93 0B80F242 80000000 00000000 0000000C 0000000C 40404040 40404040 40404040 *.....2.....*.
000F6290 000280 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
000F62B0 0002A0 00000000 00404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.
000F62D0 0002C0 40404040 40404040 40404040 40404040 40404040 40404000 00000000 00000008 *.....*.
000F62F0 0002E0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
000F6310 000300 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *00000000 *.
000F6330 000320 00000000 00000000 00000000 00000000 0004C240 00004000 00000000 00000001 *.....B.....*.
000F6350 000340 00000000 00000000 00000000 00000000 00000000 00063820 00000000 00000000 *.....*.
000F6370 000360 F0F0F7C2 00005FA8 00057C8 8000B976 0019330 000F4059 00000000 00000000 *007B.....H.....*.
000F6390 000380 00000001 8000B5E6 00000005 00000000 00000000 00000000 00000000 00000000 *.....M.....*.
000F63B0 0003A0 00007A68 0B8504E8 00000000 00000000 00000000 00000000 00000000 00000000 *.....Y.....*.
000F63D0 0003C0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
000F63F0 0003E0 00000040 008DF954 008DF930 FD000008 00005FF8 FD000000 008E1E30 00007000 *.....9.....9.....8 *.
000F6410 000400 00006000 00007A68 00EC732A 00000000 00000000 00000000 00000000 *.....*.
000F6430 000420 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
000F6450 000440 00000000 00000000 00000000 00000000 00000000 0005A8D0 00000000 00000000 *.....*.
000F6470 000460 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
SAME
000F64D0 0004C0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
000F64F0 0004E0 00000000 BCDAFFF7 0780101F 00000005 D0085FF8 00000000 00000000 0780101C *.....8.....*.
000F6510 000500 40096052 1C014384 5C000000 00E4D4C2 C3FLF640 40404040 40404040 40400000 *.....*.UMBC16 ..*.
000F6530 000520 0003F5E4 800401F6 80040EB0 80042F58 00000000 00000000 0B830A00 400000BC *..5U..6.....*.
000F6550 000540 BFFF7F00 00000003 800001D7 C4C64040 40404000 00000000 00000000 00000000 *.....PDF ..*.
000F6570 000560 00BB8520 000000BB 86800000 00008000 00000000 00000000 00000000 00000008 *.....*.
000F6590 000580 00000000 0004F1E8 0004DF08 0004E0D8 00830A00 00000012 0000101F 0000BA3 *.....1Y.....8 *.
000F65B0 0005A0 FFFF7D7C4 C6404040 4040FFFF EFE1FFFF FFFF787F EFE10000 00000000 00000000 *..PDF .....8 *.
000F65D0 0005C0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
SAME
000F66B0 0006A0 00200000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
000F66D0 0006C0 00000000 00000000 00063418 00000000 0000825C 000081DB 00000000 00000005 *.....*.
000F66F0 0006E0 00000000 BCDAFFF7 0780101F FF000005 D0085FF8 00000000 00000000 0780101F *.....8.....*.
000F6710 000700 00096052 1C014384 5C000000 00E4D4C2 C3FLF640 40404040 40404040 40400000 *.....*.UMBC16 ..*.
000F6730 000720 0003F5E4 800401F6 80040DB2 80042F58 00000000 00000000 0B830A00 600000BC *..5U..6.....*.
000F6750 000740 BFFF7F00 00000003 800001D7 C4C64040 40404000 00000000 00000000 00000000 *.....PDF ..*.
000F6770 000760 00BB8520 000000BB 86800000 00080000 00000000 00000000 00000000 00000000 *.....*.

P E M FORMATTED STORAGE DUMP E X C E P T I O N DATE=98/02/23 TIME=10:03:42 PAGE 0007
000F6790 000780 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
SAME
000F67F0 0007E0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
000F6810 000800 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....ILP.....*.
000F6830 000820 00000BC9 D3D7F1F3 F04040C4 C6D34040 40404000 00008130 00000000 0004D100 *..ILP130 DFL ..J.*.
000F6850 000840 0004D104 0004D106 0004D129 0004D12D 0004D12E 0004D150 0004D152 00000000 *..J..J..J..J..J..J..*.
000F6870 000860 00000000 00000000 00005D134 00005E5CC F87FFEF1 00007FAB 0004D106 00000000 *.....J..J..8.....J..*.
000F6890 000880 00000000 00000000 00000000 00404040 40404000 000678F0 0000678F0 000012D4 C6D7D4C1 *.....0..MFPM*.
000F68B0 0008A0 40000000 00000000 0B8014C0 00000000 00000000 00000000 00000000 00000000 *.....*.
000F68D0 0008C0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
000F68F0 0008E0 00000000 00000000 C4E30000 000100BB 00007A68 00EC0F57 00010063 26CD0004 *.....DT.....*.
000F6910 000900 00000000 00000000 000015E9 00000000 00000000 00000000 00000000 00000000 *.....WPC$SDMP *.
000F6930 000920 40404040 40404040 40404040 40404040 40404000 00000000 00000000 00000000 *.....*.
000F6950 000940 223FC240 000F4059 000015EA 00000000 00005000 0000007DF 00010980 223C0990 *..B.....*.
000F6970 000960 101CFFFF 0980101F D500D500 00000000 00000000 00000000 00000000 00000000 *.....N.N..UMB2*.
000F6990 000980 F0FOCSF6 40404040 40404040 00000000 00000000 00000000 00000000 00000000 *00E6 ..DT..*.
000F69B0 0009A0 000C0054 00010000 F140F9F9 00F9F9F3 F3404040 40404040 40404040 40404040 *.....1 99 9933 *.
000F69D0 0009C0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.
000F69F0 0009E0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....DT.....*.
000F6A10 000AA0 00020000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
000F6A30 000A20 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.
SAME
000F6A80 40404040 40404040 40404040 00000000 C4E30000 000A0548 00000000 00000000 *.....DT.....*.
000F6AB0 000AA0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
SAME
000F6B70 000B60 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....Y..*.
000F6B90 000B80 00000000 00000000 5880E0000 0B80E0000 00000000 00000000 00000000 41E0E000 *.....*.
000F6B90 000BA0 0B80E0000 00000000 00000000 00000000 00000000 40404040 40404040 40404040 *.....*.
000F6BD0 40404040 40404040 40404040 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
000F6BF0 000BE0 00000000 00000000 00000000 00000000 00000000 40404040 4040C5D5 E4404040 *.....ENU ..*
```



Umbrella Programming

System Memory Access Retrieval Tool

000F6C10	000C00	40404040	40404040	40404040	40404040	40404040	40404040	40404040	40404040	*
000F6C30	000C20	40000000	00000000	00000000	0005C5D5	E4F04040	40400000	00CC5D5E4	C5D5E400	* NENU0 *
000F6C50	000C40	00000000	00CC5D5E4	00000000	00000000	40404000	00000000	00000000	00000000	* ENU *
000F6C70	000C60	00000000	00000000	00000000	F0000000	00000000	00000000	00000000	00000000	* 0 *
000F6C90	000C80	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	* *
		SAME								
000F6EB0	000EA0	00000000	0980223F	0004C120	00000000	00000000	00000000	00000000	0000057C8	* A H*
000F6ED0	000EC0	00000000	00000000	00000000	40000000	00000000	00000000	00000000	00000000	* *
000F6EF0	000EE0	00000000	00000000	00000000	40404040	4040405D5	E4000000	00000000	00000000	* ENU *
000F6F10	000FF0	00000000	00000000	00000000	00000000	D5C5D5E4	00000000	00069630	D4D6D5C9 E3D6D940	* MONITOR *
000F6F30	000F20	00000000	00000000	00000000	00000000	00000000	00000000	5050C9C2	C1E9F3F3	* IBAZ33 *
000F6F50	000F40	00100000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	* *
000F6F70	000F60	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	* *
000F6FB0	000F80	00000000	00000000	00000000	00000000	00000000	00000000	00008000	00000000	* *
000F6FB0	000FA0	00000000	00000000	00000000	00000000	E7E7E7E7	E7E7E7E7	000D5680	000D56D0	* XXXXXXXX *
000F6FD0	000FC0	000D77F0	000D5678	07F908C5	D4D74040	40404000	40404040	40404004	E5E2C1D4	* .. 0 9. EMP VSAM*

PEM --- COBOL II/COBOL 370 CONTROL BLOCK

00069630	000000	D7C5D440	C3D6C2D6	D340C3D6	D5E3D9D6	D340C2D3	D6C3D240	C2C1E3C3	C8D7C5D4	* PEM COBOL CONTROL BLOCK BATCHPEM*
00069650	000020	00000000	00000000	00000000	00000000	00000000	00011268	00069630	00000474	* *
00069670	000040	0005B2C0	0B804154	00000030	0005B2F0	00000000	0B80189C	0B80404D	0005B250	* 0. M. .
00069690	000060	8B850468	00011238	00007A68	00E732A	00000000	00000000	00000000	000COPF20	* *
00069680	000080	00069630	00064B80	000CC358	00000008	0006140	8B00B56E	0006140	0B8018FE	* C. . . / . . / . *
00069690	0000A0	0B804232	0002288	8B859148	0010990	0007A68	0B8504E8	00000000	00000000	* BY. . . Y. . *

P	E	M	FORMATTED	STORAGE	DUMP	E X C E P T I O N			DATE=98/02/23		TIME=10:03:42	PAGE 0008
000696F0	0000C0	00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000	*	SAME	0B852B48	0B853160	0B853180	00008358	*	- - - - -	*
00069730	000100	00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000	0B852B48	0B853160	0B853180	00008358	*	* Y	*	
00069750	000120	00000000 0B8531B8 00000000 0B80189C	00000000 00000000 00000000 00000000	0B8040D4	0B004BE04	00008358	0B852B48	*	* M	*	
00069770	000140	0B852B2E2 0B852B40 00000000 00000000	00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000	0B8002D4	00069630	00000000	*	* S	M	*
00069790	000160	00000018 00000000 00006140 8000856E	00000000 00000000 00000000 00000000	00008358	0B801814	0B80423C	000CC2E8	*	*	/ /	BY*	
000697B0	000180	0B8591FC 000115D8 00007A68 00006140	00000000 00000000 00000000 00000000	00000000	00000000	00000000	00000000	*	* Q	/ /	*
000697D0	0001A0	00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000	00000000	00000000	00000000	00000000	*	*	*
000697F0	0001C0	00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000	00000000	00000000	00005828	000057C8	*	*	H*
00069810	0001E0	00000000 0B853028 000057C8 00000000	00000000 00000000 00000000 00000000	800112CC	0B852B66	00000000	80011830	*	* H	*	
00069830	000200	00005B260 00000000 00000000 00000000	00000000 00000000 00000000 00000000	00000000	00000000	00000000	00000000	*	*	- - - - -	*	
00069850	000220	00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000	00000000	00000000	00000000	00000000	*	*	*
00069930	000300	00005FA8 0B853028 80FD64E8 8B8527C0	00000474 00069974 0B804154 00000030	*	SAME Y	*	*	* Y	*	
00069950	000320	00005B2F0 00000000 0B80189C 0B8040D4	00005B250 0B850468 00011238 00007A68	*	* O	M	*	* M	*	
00069970	000340	00EC732A 80008358 0B853238 00000000	00000000 00000000 00000000 00000000	*	*	*	*	*
00069990	000360	00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000	00000000	00000000	00000000	00000000	*	*	*
00069A70	000440	00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000	00000000	00000000	00000000	00000000	*	*	*
00069A90	000460	00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000	00000000	00000000	00000000	00000000	*	*	*

PEM --- REGISTER SAVEAREA CHAIN

P49019B UMB201 03/19/96 18.05

00007DC8 000000 F0F0F7C2 0005F5A8 000057C8 8000B976 00019330 000F4059 00000000 00000000 *007B.....H..... *
00007DE8 000020 00000001 8000B56E 00000005 00000000 0B8040D4 0005B250 0B80410C 000EB500 *.....M..... *
00007E08 000040 00007A68 0B8B504E8 00000000 00000000 00000000 00000000 00000000 00000000 *.....Y..... *

P49055B UMB201 03/19/96 18.17

000057C8 000000 F0F1F9C2 00007DC8 00005828 800194B2 0003DAA0 00000000 00000008 0001952C *019B..H.....*
000057E8 000020 00000001 00006140 00000005 00000000 0B8040D4 0005B250 0B80410C 00019330 *....../.....M..*
00005808 000040 00007A60 0B88504E8 00000000 00000000 00000000 00000000 00000000 00000000 *....Y.....*

K P

00005828 000000 F0F5FC52 00057C8 00058888 80030F20 00032B64 00000000 000310F0 000312C8 *055B..H.....0..H*
00005848 000020 00000001 000F73A0 00000005 00000000 BB8040D4 005B250 BB80410C 0030DA0 *.....M.....*
00005868 000040 00007A68 00006140 00000000 00000000 00000000 00000000 00000000 00000000 *...../. .*

```

PEM --- PROGRAM DEFINITION INDEX

0B802808 000000 00000008 0B802F80 000007F8 0B804008 00000000 00000000 00000000 00000000 *.....8....*.
0B802828 000020 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
          SAME
0B802F68 000760 00000000 00000000 00000000 00000000 00000000 00000000 00000050 9211130C *.....*.
0B802F88 000780 0000000C 0B8041CC 00000640 9211130C 0000000C 0B80423C 0000073A 9211130C *.....*.
0B802FA8 0007A0 0000000C 0B804154 00000744 9211130C 0000000C 0B80418C 0000BB80 9211130C *.....*.
0B802FC8 0007C0 0000000C 0B804070 0000BBDC 9211130C 0000000C 0B804014 0000BF63 9211130C *.....*.
0B802F88 0007B0 0000000C 0B804208 000F3FF5 9211130C 0000000C 0B8040D4 *.....5.....M

```

P E M FORMATTED STORAGE DUMP E X C E P T I O N DATE=98/02/23 TIME=10:03:42 PAGE 0009

PEM --- PROGRAM DEFINITION TABLE

0B804008	000000	00000000	0B804288	0001078	000BBDC	E4F4F8F0	F9F24040	00064A8A	48409211	*	U48092
0B804028	000020	130C0000	000C0000	0001000E	000BB80	0000BBD0	0000BBDA	0000BBDB	0000BC00	*	*	*	*	*	
0B804048	000040	0000BC1	0000BDEC	0000BE32	0000BE56	0000BF1E	0000009	0000BCB6	0000BCB8	*	..A.	*	*	*	
0B804068	000060	0000BB80	0000BBC2	0000BB80	E4F4F8F0	F0F04040	00064AC0	54C09211	130C0000	*	..B.	..	U48000	*	*
0B804088	000080	000C0000	00000011	0000BB80	0000BB89	0000BB86	00000000	00000000	00000000	*	*	*	*	*	
0B8040A8	0000A0	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	*	*	*	*	*	
0B8040C8	0000C0	00000000	00000000	0000BBC2	000F3F5	E9F9F9F9	F4F1F340	00064AE0	88209211	*	*	B.	52999413	*	
0B8040E8	0000E0	130C0000	000C0000	000C000C	00003F5	00000576	0000076C	0000BAED	0000B831	*	*	5.	*	*	
0B804108	000100	0000BB32	0000BB80	000F40BD	000F40D1	000F40E5	000F40F9	000F4199	000F7FC	*	*	J.	V.	9.	
0B804128	000120	0000B806	0000B810	0000B81A	0000BB24	000007D0	000005AC	0000B856	0000B858	*	*	*	*	*	
0B804148	000140	0000BDA7	00033515	00077551	0000737A	C9F5F4F2	F0F04040	00064B00	54C09211	*	*	154200	*
0B804168	000160	130C0000	000C0000	00000006	00000900	00000407	00000000	000004A4	0000BBDA	*	*	*	*	*	
0B804188	000180	00000E11	00000744	C9F5F4F2	F4F84040	00064B20	58409211	130C0000	000C0000	*	*	154248	*
0B8041A8	0001A0	00000008	00000900	00000407	00000000	00000901	00000902	00000903	0000BBDA	*	*	*	*	*	
0B8041C8	0001C0	0000092F	00000500	C9F5F4F1	F2F84040	00064B40	88209211	130C0000	000C0000	*	*	154128	*



Umbrella Programming

System Memory Access Retrieval Tool

```

0B8041E8 0001E0 00020005 0000BB80 0000BF63 00000901 00000902 0000092E 0000BB87 0000BBDA *.....*.
0B804208 000200 0000BF63 E4F4F8F9 F9F54040 00064B60 48409211 130C0000 000C0000 00040001 *.....048995 .....*.
0B804228 000220 00000001 00000011 00000023 0000BB80 0000BDBA 00000640 E3F5F8F0 F0F14040 *.....T58001 *.
0B804248 000240 00064B80 4C209211 130C0000 000C0000 00030008 000003F5 0000076D 0000BB80 *.....5.....*.
0B804268 000260 000007D0 000005AC 000007D5 000007DF 000007D6 000007D1 00000D93 0000BBDA *.....N.....O.....J.....*.

```

PEM --- ALLOCATED DATA GROUP INDEX

```

0004CC98 000000 00000030 0000003E 0004CD90 0004D090 00000000 00000000 00000000 00000000 *.....*.
0004CCB8 000020 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
SAME
0004CD78 0000E0 00000000 00000000 00000000 00000000 00000000 00000000 00000009 003600E3 *.....T*.
0004CD98 000100 0004D0A8 080D014 0000004A 048400E3 000C53C0 0B80EBAE 00000407 01F400B3 *.....T.....4..T*.
0004CDB8 000120 000C51A8 0B80EB6C 00005AC 008A00E3 0005B8B0 0B80E5C6 000007D0 0AA200E3 *.....T.....VF.....T*.
0004CDC8 000140 0005B7E8 0B80E43C 000007D1 02BC00E3 000CC860 0B80F230 000007D5 000000E3 *.....Y.....U.....J.....T.....H-.....2.....N.....*.
0004CDF8 000160 000CC3A0 0B80ED6A 000007D6 0000B0E3 000CC830 0B80F21E 000007DF 042000E3 *.....C.....O.....T.....H.....2.....*.
0004CE18 000180 000CC3D8 0B80EDBC 000008FD 004000E3 000C8890 0B80EBD2 00000900 309C00E3 *..CQ.....T.....K.....*.
0004CE38 0001A0 000C20E8 0B80EB5A 00000901 001900E3 000C89A8 0B80EC3C 00000902 0FD300E3 *.....Y.....T.....L.....T*.
0004CE58 0001C0 000C89E8 0B80ECE7 00000903 018D00E3 000C99E0 0B80ECC8 00000906 0F0000E3 *.....Y.....=.....T.....H.....0.....T*.
0004CE78 0001E0 000CB118 0B80ED46 00000928 0FEB00E3 000CA068 0B80ED34 0000092F 040000E3 *.....T.....T.....T*.
0004CE98 000200 000CB9B8 0B80ED22 00000D93 18CA80D7 000D5728 0B80F242 00000E11 300000E3 *.....P.....2.....T*.
0004CEB8 000220 000C5868 0B80EBC0 000015EA 019000E3 000CC130 0B80ED58 00003515 020400E3 *.....T.....A.....T*.
0004CED8 000240 0005BCF0 0B80E846 0000B7FC 003200E3 0005B340 0B80E05A 0000B806 00C800E3 *.....0.....Y.....T.....H.....T*.
0004CEF8 000260 0005B398 0B80E048 0000B810 009600E3 0005B488 0B80E106 0000B81A 012C00E3 *.....T.....*.
0004CF18 000280 0005B548 0B80E158 0000B824 012C00E3 0005B698 0B80E2CA 0000B856 002E00E3 *.....T.....S.....T*.
0004CF38 0002A0 0005B960 0B80E730 0000B858 003000E3 0005B988 0B80E752 0000B880 072400E3 *.....X.....T.....X.....T*.
0004CF58 0002C0 0005A990 0B80E036 0000B885 1C0000E3 0004D0F0 0B80D026 0000B886 03F900E3 *.....T.....0.....9.....T*.
0004CF78 0002E0 0004ED00 0B80D038 0000B887 00C600E3 0004F108 0B80D04A 0000B889 271000E3 *.....F.T.....1.....T*.
0004CF98 000300 0004F1E0 0B80D056 0000BBA3 014000E3 00051900 0B80D06E 0000B8C2 015000E3 *.....I.....*.....T.....B.....T*.
0004CFB8 000320 0005B0D8 0B80E048 0000BBD8 00A480D7 0005C0B0 0B80D092 0000B8DC 005400E3 *.....Q.....P.....T*.
0004CFD8 000340 00051A08 0B80D080 0000BC6 0FA0000E3 00053FD8 0B80DF66 0000BBC8 296800E3 *.....T.....Q.....T*.
0004CFF8 000360 00054FA0 0B80DF78 0000BEC0 066800E3 00051A80 0B80D13C 0000BCC1 004000E3 *.....T.....J.....A.....T*.
0004D018 000380 00052110 0B80DC16 0000BCC2 018000E3 0005A718 0B80DF9C 0000B8D80 2DC000E3 *.....B.....T.....T*.
0004D038 0003A0 00057930 0B80DF8A 0000BDA7 02BC00E3 0005B5A0 0B80E77C 0000B8DC 01B4000E3 *.....T.....X.....T*.
0004D058 0003C0 000521E8 0B80DC78 0000BE56 00F200E3 0005B2350 0B80DD82 0000BF1E 1B4600E3 *.....Y.....2.....T.....T*.
0004D078 0003E0 00052468 0B80DF54 00077511 007300E3 0005BF18 0B80EA78 00000000 00000000 *.....T.....*.

```

P E M FORMATTED STORAGE DUMP

E X C E P T I O N

DATE=98/02/23 TIME=10:03:42 PAGE 0010

PEM --- PERMANENT DATA GROUP INDEX - PRIVATE

```

000841A8 000000 00000002 0000003E 00084580 000845A0 00000000 00000000 00000000 00000000 *.....*.
000841C8 000020 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
SAME
00084568 0003C0 00000000 00000000 00000000 00000000 00000000 00000000 00000D93 18CA80D7 *.....P*.
00084588 0003E0 000D5728 0B80F242 0000BBDB 00A480D7 0005C0B0 0B80D092 *.....2.....P.....*.

```

USER TRANSACTION CONTROL BLOCK

```

00008358 000000 00007A68 0EC0FE57 00010063 26CD0004 00000000 00000000 000015EA 01900100 *.....*.
00008378 000020 342F0980 223F6D7 C35B58C4 D4D74040 40404040 40404040 40404040 40404040 *.....WPC$SDMP *.
00008398 000040 40400000 00000000 00000000 00400080 223FC240 000F4059 000015EA 00000000 *.....B.....*.
000083B8 000060 00050000 000007DF 00010980 223C0990 101CFFFF 0980101F D500D500 00000000 *.....N.N.....*.
000083D8 000080 0950211C 0153316C 00000001 E4D4C2F2 F0FC05F6 40404040 40404040 00000000 *.....UMB20056 *.
000083F8 0000A0 00000000 00000000 00000000 00000000 *.....*.

```

USER TRANSACTION CONTROL BLOCK EXTENSION

```

000085E0 000000 00007A68 0EC0FE57 00010063 26CD0004 00000000 00000000 000015EA 01900100 *.....Y.....*.
00008600 000020 41E0E000 0B0E0000 00000000 00000000 00000000 00000000 00004040 00004040 *.....*.
00008620 000040 40404040 40404040 40404040 40404040 40404040 00000000 00000000 00000000 *.....*.
00008640 000060 00000000 00000000 00000000 00000000 00000000 00000000 40404040 00000000 *.....*.
00008660 000080 4040C5D5 E4404040 40404040 40404040 40404040 40404040 40404040 00000000 *.....ENU.....*.
00008680 0000A0 40404040 40404040 40000000 00000000 00000000 0005C5D5 E4F04040 404040000 *.....NENU.....*.
000086A0 0000C0 0CC5D5E4 C5D5E400 000C0000 0CC5D5E4 00000000 00000040 40404000 000C0000 *.....ENU.....ENU.....*.
000086C0 0000E0 00000000 00000000 00000000 00000000 00000000 F0 *.....0.....*.

```

UPCB PROGRAM ID=z999413 ENTRYPOINT=8B850468 COMPILED=02/06/98 13.25.44

```

0005B250 000000 00000000 00000000 000004000 00000000 00000000 00000000 00000000 00000000 *.....UPCB.....M*.
0005B270 000020 0005B250 0B8533A8 0B8534A0 0005BCF8 0005B348 0005B3A0 0005B490 0005B550 *.....8.....*.
0005B290 000040 0005B6A0 00008358 0B8534F8 0B850510 0B8508CE 0B8504E8 0B8040D4 00000000 *.....8.....Y.....M.....*.
0005B2B0 000060 000COE80 8B850468 00000000 0005B2F4 *.....4.....*.

```

PARAMETER LIST - A(DATAGROUP)

```

0005B2C0 000000 00008358 0005B348 0005B3A0 0005B490 0005B550 0005B6A0 0005B7F0 0005B8B8 *.....0.....*.
0005B2E0 000020 0005B968 0005B9C0 0005BA18 0005BCF8 8005BF20 *.....8.....*.

```

ACTIVITY LIST - A(ACTIVITY DEFINITION)

```

0005B2F4 000000 00000000 00000000 0B8018FE 00000000 0B8018AC 00000000 00000000 0B80191E *.....*.
0005B314 000020 00000000 00000000 00000000 00000000 *.....*.

```

DATA GROUP ID=0000000000

```

0004D0B0 000000 00000000 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.
0004D0D0 000020 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.

```

P E M FORMATTED STORAGE DUMP E X C E P T I O N DATE=98/02/23 TIME=10:03:42 PAGE 0011

DATA GROUP ID=00000000074

```

000C53C8 000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
SAME
000C5828 000460 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.
000C5848 000480 00000000 *.....*.

```

DATA GROUP ID=00000001031



Umbrella Programming

System Memory Access Retrieval Tool

```

000C51B0 000000 00000000 00404040 40404040 40404040 40404040 40404040 40404040 *.....
000C51D0 000020 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....
SAME
000C5270 0000C0 40404040 40404040 40404040 40404040 BB310000 0000000F 00008358 000C20F0 *.....
000C5290 0000E0 0005C1B0 0005B8B8 00000000 000C2050 00000000 800821A2 00082078 00000001 *.....
000C52B0 000100 000C20B0 000820C2 0008358 000C20F0 000C2F80 0B8018AC 800C5870 000C51B0 *.....
000C52D0 000120 00082078 000B2B31 00083078 00082078 00000000 00000000 00000000 *.....
000C52F0 000140 00000000 00000000 00000000 00000000 0B8018BC 00000000 0B8018DE 00008358 *.....
000C5310 000160 800C5314 0B801814 00000000 00000000 40404040 40404040 40404040 40404040 *.....
000C5330 000180 40404040 40404040 40404040 40404040 00000000 00000000 00000000 00000000 *.....
000C5350 0001A0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....
000C5370 0001C0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....
000C5390 0001E0 00000000 00000000 00000000 00000000 00000000 *.....
DATA GROUP ID=0000001452
0005B888 000000 00000000 00000000 40404040 40404040 40404000 00000F00 00000F00 00000000 *.....
0005B8D8 000020 40404040 40404040 40404000 00000F00 00000F00 00000F00 00000F00 00004040 *.....
0005B8F8 000040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....
0005B918 000060 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....
0005B938 000080 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....
DATA GROUP ID=0000002000
0005B7F0 000000 00040000 0980309C 0980309C C4000001 0C40D4C1 D9C3C840 F96B40F1 F9F9F840 *.....
0005B810 000020 40404040 40404040 40404000 00000F00 00000F00 00000F00 00000F00 00004040 *.....
0005B830 000040 00000F00 00000F00 00000F00 00000F00 00000F00 00000F00 00000F00 00000F00 *.....
0005B850 000060 00000F00 00000FP2 40404040 4040C240 40404040 40404040 40404040 40404040 *.....
0005B870 000080 40404040 40404040 40404040 40404040 40404040 40400000 0F404040 40404040 *.....
0005B890 0000A0 000F *.....
DATA GROUP ID=0000002001
000CC868 000000 00000000 0010004 F0F9F8F0 F3F0F9F0 F0F9F8F0 F6F80072 021C0000 002C000D *.....
000CC888 000020 0000004C F2000000 00024C00 04F0F0F9 F9F80400 40404040 40D4C1D9 C3C840F9 *.....
000CC8A8 000040 6840F1F9 F9F84040 40404040 40404040 40404000 40C40000 010C4040 F0007200 *, 1998 .. D... O... *.
000CC8C8 000060 7C007202 1CF2F2F2P2 F2400000 010C0000 001C0000 001C0000 00000000 00004040 *.....
000CC8E8 000080 40404040 40404040 40000000 0F000000 0F000000 00040404 40404040 40404040 *.....
000CC908 0000A0 40000000 0F000000 000F0000 000F0000 000F0000 40404040 40404040 40404040 *.....
000CC928 0000C0 40404040 40404040 40404040 40000000 00000000 40404040 40404040 40404040 *.....
000CC948 0000E0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....
000CC968 000100 010C4040 40400980 223CF0F9 F8F0F5F2 F5F0F9F8 F0F3F0F9 F0F9F8F0 F1F0F100 *.. ....098052509803090980101.*.
P E M FORMATTED STORAGE DUMP E X C E P T I O N DATE=98/02/23 TIME=10:03:42 PAGE 0012
000CC988 000120 72021C40 40404040 40404040 40404040 40404040 002C0002 C6C64040 C540C640 *.....
000CC9A8 000140 4040C540 C6404040 C540C640 4040C5C5 0008B700 1C001C40 001C001C 00001C00 * E F E F EE .X. .....
000CC9C8 000160 1C40001C 001C4000 1C001C00 001C001C E7002C00 0C002C40 404040F0 F00004F0 *.....
000CC9C8 000180 F1F0F1F0 F9F8F1F0 28000009 80223C09 80309C4 0000010C 000122B8 F0F8F8F0 *19100.....
000CCA08 0001A0 F1F0F1F9 F9F8F1F2 F3F14040 40404040 40404040 40404040 40404040 40404040 *1019991231 USAUS*.
000CCA28 0001C0 C1404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *A *.
000CCA48 0001E0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....
SAME
000CCA8 000280 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....
000CCB08 0002A0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....
DATA GROUP ID=0000002005
000CC3A8 000000 00000000 E7400040 4000E7F0 F1 *.....
DATA GROUP ID=0000002006
000CC838 000000 00000000 40404040 40404040 *.....
DATA GROUP ID=0000002015
000CC3E0 000000 00000000 00050000 000007DE 00010980 223C0981 231CFFF 0980101F D500D500 *.....
000CC400 000020 00000000 0950211C 0153316C 00000001 E4D4C2F2 F0F0C5F6 40404040 40404040 *.....
000CC420 000040 00000000 00000000 00000000 00000000 0000FF0 F1F0F1F5 D5C5E640 E8C5C1D9 *.....
000CC440 000060 E240C4C1 E8404040 40404040 40404040 F0F2F1F6 P2D7D9C5 E2C9C4C5 D5E37DE2 *S DAY 02162PRESIDENT.S*.
000CC460 000080 402C29D9 E3C8C4C1 E8404040 40F0F5F2 F5F2D4C5 D4D6D9C9 C1D340C4 C1E84040 * BIRTHDAY 05252MEMORIAL DAY *.
000CC480 0000A0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....
000CC4A0 0000C0 40404040 404040F0 F9F0F7F2 D3C1C2D6 D940C4C1 E8404040 40404040 40404040 *.....
000CC4C0 0000E0 40404040 F1F0F1F2 F2C3D6D3 E4D4C2E4 E240C4C1 E8404040 40404040 40404040 *.....
000CC4E0 000100 40F1F1F1 F1F0F1F2 E3C5D9C1 D5E240C4 C1E84040 40404040 40404040 4040F1F1 *.....
000CC500 000120 F2F6F5E3 C8C1D5D2 E2C7C9E5 C9D5C740 C4C1E840 40404040 4040F1F2 F2F2F5F6 *265THANKSGIVING DAY 12256*.
000CC520 000140 C3C8D9C9 E2E3D4C1 E240C4C1 E8404040 40404040 40404040 40404040 40404040 *CHRISTMAS DAY *.
000CC540 000160 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....
SAME
000CC7E0 000400 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....
000CC800 000420 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....
DATA GROUP ID=0000002301
000C8898 000000 00000000 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....
000C88B8 000020 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....
DATA GROUP ID=0000002304
000C20F0 000000 00000000 000CB2F0 00000000 00000000 00000000 00000000 00000000 0001BB31 *.....
000C2110 000020 40404040 40404000 002C000C 00000000 00000000 00000000 00000000 00000000 *.....
000C2130 000040 00000000 00000000 00000000 00000000 000C5140 00000000 000C8908 *.....
000C2150 000060 00000000 900D2B3A 000D2AE0 000C5284 000C8968 0B8041CC 00008358 000C20F0 *.....
000C2170 000080 000C2F80 0B8018CE 0B80418C 000C51B0 000D2AE0 00010990 000D3AE0 000D2AE0 *.....
000C2190 0000A0 600D3282 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....
P E M FORMATTED STORAGE DUMP E X C E P T I O N DATE=98/02/23 TIME=10:03:42 PAGE 0013

```



Umbrella Programming

System Memory Access Retrieval Tool

```

000C21B0 0000C0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0*
000C21D0 0000E0 00000000 00000000 60082188 00000000 00000000 00000000 00000000 00000000 *.....*.0*
000C21F0 000100 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0*
    SAME
000C2230 000140 0001BB31 40404040 40400000 000C8B76 000C8968 000CB180 00008358 000C20F0 *.....*.0*
000C2250 000160 000C2F80 000CB180 000C89B0 000CB120 000CB178 00010990 000D3AE0 000D2AE0 *.....*.0*
000C2270 000180 000C2CC 0A0D2E2C 000C8AD8 00000000 00000000 00000000 00000000 00000000 *.....*.Q...
000C2290 0001A0 00008358 000C2F0 000C2F80 000CB180 600486A 000CB120 000CB178 00010990 *.....*.0...
000C2280 0001C0 000D3AE0 000D2AE0 000C22CC A00D49BE 000C8AD8 00000000 00000000 00082DE0 *.....*.Q...
000C22D0 0001E0 000C8968 000C2F29 00008358 000C20F0 000C2F80 000CB198 000CB2C0 000CB2DC *.....*.0...
000C22F0 000200 000CB178 00010990 000D3AE0 000D2AE0 000CB180 A00D332A 00000000 0003AE0 *.....*.0...
000C2310 000220 000D2AE0 00000000 00000000 00000000 00000000 00000000 00000000 00083A46 *.....*.0...
000C2330 000240 00000000 00000000 00000000 00000000 00015EA 01900000 000CC138 00000000 *.....*.A...
000C2350 000260 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
    SAME
000C2570 000480 00000000 00000000 00000000 00000000 000C2F80 000BB31 00000000 00000000 *.....*.0...
000C2590 0004A0 00000000 000C5284 000C2B00 00000000 00000000 00000000 00000000 000C8898 *.....*.0...
000C2580 0004C0 0005B8B8 00000000 00000000 00000000 00082700 00082732 000827B2 000827BC *.....*.0...
000C25D0 0004E0 000827C6 000828Z6 00082858 000828DA 00082902 00082D0E 00082D0E 000830FC *.....*.F...
000C25F0 000500 0008312A 00083140 00083226 00083234 000832CA 0008369E 000836AB 000836FA *.....*.0...
000C2610 000520 00083718 00083778 000837TA 00083800 000839D4 00083A2C 00083A46 00000000 *.....*.M...
000C2630 000540 00000000 00000000 00083AE2 01000000 0000092D 00000000 01000000 00000000 *.....*.S...
000C2650 000560 00000000 010CB120 00000906 0FF00000 00000000 00000000 00000000 00000000 *.....*.0...
000C2670 000580 00000000 00000908 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2690 0005A0 00000000 0000090B 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C26B0 0005C0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C26D0 0005E0 00000000 00000910 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C26F0 000600 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2710 000620 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2730 000640 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2750 000660 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2770 000680 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2790 0006A0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C27B0 0006C0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C27D0 0006E0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C27F0 000700 00000A64 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2810 000720 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2830 000740 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2850 000760 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2870 000780 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2890 0007A0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C28B0 0007C0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C28D0 0007E0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C28F0 000800 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2910 000820 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2930 000840 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2950 000860 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2970 000880 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2990 0008A0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C29B0 0008C0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C29D0 0008E0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C29F0 000900 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2A10 000920 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2A30 000940 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2A50 000960 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...

```

P E M FORMATTED STORAGE DUMP E X C E P T I O N DATE=98/02/23 TIME=10:03:42 PAGE 0014

```

000C2A70 000980 00000000 00000000 0000082A 00000000 00000000 0000082B 00000000 00000000 *.....*.0...
000C2A90 0009A0 0000082C 00000000 00000000 0000082D 00000000 00000000 0000082E 00000000 *.....*.0...
000C2A80 0009C0 00000000 00000000 0000082F 00000000 00000000 00000830 00000000 00000000 *.....*.0...
000C2A8D 0009E0 00000000 00000000 00000832 00000000 00000000 00000833 00000000 00000000 *.....*.0...
000C2A9F 000A00 00000000 00000000 00000568 00000000 00000000 00000000 0000056F 00000000 *.....*.0...
000C2B10 000A20 00000000 00000000 00000571 00000000 00000000 00000572 00000000 00000000 *.....*.0...
000C2B30 000A40 00000000 00000000 00000574 00000000 00000000 00000000 00000575 00000000 *.....*.0...
000C2B50 000A60 00000576 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2B70 000A80 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2B90 000AA0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2B80 000AC0 00000A6F 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2B9D 000AE0 00000000 00000000 00000A72 00000000 00000000 00000000 00000A8C 00000000 *.....*.0...
000C2BF0 000B00 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000A8F 00000000 *.....*.0...
000C2C10 000B20 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2C30 000B40 00000000 00000000 00000A93 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2C50 000B60 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2C70 000B80 00000A98 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2C90 000B90 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2C80 000BC0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2CD0 000BE0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2CF0 000C00 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2D10 000C20 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2D30 000C40 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2D50 000C60 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.P...
000C2D70 000C80 D7000000 00000000 00000000 00000000 00000000 00000000 00000000 08FCF4F7 F9F2F140 40404040 40098022 *P.....*.47921 ...
000C2D90 000CA0 3F0022B8 08FEF4F7 F9F2F140 40404040 40098022 3F000000 00000000 00000000 *.....*.47921 ...
000C2DB0 000CC0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
    SAME
000C2E50 000D60 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2E70 000D80 00000000 40000000 00400000 00004000 00000000 00000000 40000000 00400000 *.....*.0...
000C2E90 000D00 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2EB0 000DC0 40000000 00400000 00004000 00004000 00000000 40000000 00400000 00004000 *.....*.0...
000C2ED0 000DE0 00000000 00000000 40000000 00400000 00000000 00000000 00000000 40000000 *.....*.0...
000C2EF0 000E00 00400000 00004000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2F10 000E20 00000000 000001F4 5CF04040 40404040 00000000 00000000 40E6F0F5 40404040 *.....*.4*0 .....W05 ...
000C2F30 000E40 40000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.P.VE ...
000C2F50 000E60 40404040 40404040 40404040 40000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2F70 000E80 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2F90 000EA0 000CB120 000CC110 00000000 00000000 00330000 03E8B22B8 22B80000 00000000 *.....*.A....Y...
000C2FB0 000EC0 00000000 0000C5298 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...
000C2FD0 000EE0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.0...

```



Umbrella Programming

System Memory Access Retrieval Tool

P	E	M	FORMATTED STORAGE DUMP	E X C E P T I O N	DATE=98/02/23	TIME=10:03:42	PAGE 0015
000C5150	003060	SAME	00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000	*.....	*	
000C5170	003080		00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000	*.....	*	
DATA GROUP ID=0000002305							
000C89B0	000000	00050000 22B8F4F7 F9F2F140 40404040	40921989 8C000000 00	*.....47921	*	
DATA GROUP ID=0000002306							
000C89F0	000000	00010000 FF000000 00000E8 000C8B76	000C8AD8 00000000 00000000 0960521F	*.....Y.....Q.....-	*	
000C8A10	000020	0145622F 40404040 40404040 00000000	00000000 00000000 E4D4C2C3 00000000	*.....UMBC.....*	*	
P E M FORMATTED STORAGE DUMP E X C E P T I O N DATE=98/02/23 TIME=10:03:42 PAGE 0015							
000C8A30	000040	00000000 00000000 00000000 00000064	0000E200 00000AC5 00000000 40000000	*.....S...E.....*	*	
000C8A50	000060	00400000 00004000 0000040 00000000	40000000 00400000 00004000 00000040	*.....	*	
000C8A70	000080	00000000 40000000 00400000 00004000	00000040 00000000 40000000 04000000	*.....	*	
000C8A90	0000A0	00004000 0000040 00000000 40000000	00400000 00004000 00000000 40000000	*.....	*	
000C8AB0	0000C0	40000000 00400000 00004000 00000040	00000000 40000000 04000000 00004000	*.....	*	
000C8AD0	0000E0	00000040 00000000 00000000 0645CF0	40404040 40400000 00000000 000000E6	*.....*0W*		
000C8AF0	000100	F0F14040 40404000 0015EA00 000008D7	0002E5C5 40404040 40404040 40404040	*01.....P..VE	*	
000C8B10	000120	40404040 40404040 40404040 40404040	40404000 00000000 C65CF040 40404040	*.....H*0	*	
000C8B30	000140	40000000 00000000 0040E6F0 F2404040	40400000 15EA0008 0008D700 02E5C540	*.....W02.....P..VE	*	
000C8B50	000160	40404040 40404040 40404040 40404040	40404040 40404040 40404040 40400000	*.....	*	
000C8B70	000180	000A012C 5CF04040 40404040 00000000	00000000 406E6F0F3 40404040 40000015	*.....*0W03*	
000C8B90	0001A0	EA001000 08D70002 E5C54040 40404040	40404040 40404040 40404040 40404040	*.....P..VE	*	
000C8BB0	0001C0	40404040 40404040 40000000 A01905C	F0404040 40404000 00000000 00000040	*.....*0	*	
000C8BD0	0001E0	E5F0F440 40404040 00015EA0 00180008	D70002E5 C5404040 40404040 40404040	*W04.....P..VE	*	
000C8BF0	000200	40404040 40404040 40404040 40404040	40404040 00000000 01F45CF0 40404040	*.....4*0	*	
000C8C10	000220	40400000 00000000 00000046 F0F54040	40404000 0015EA00 20008D7 0002E5C5	*.....W05.....P..VE*	*	
000C8C30	000240	40404040 40404040 40404040 40404040	40404040 40404040 40404040 404040FF	*.....	*	
000C8C50	000260	FFFFFF00 00000000 00000000 00000000	00000000 00000000 00000000 00000000	*.....	*	
000C8C70	000280	00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000	*.....	*	
SAME							
000C9990	000FA0	00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000	*.....	*	
000C99B0	000FC0	00000000 00000000 00000000 00000000	00000000	*.....	*	
DATA GROUP ID=0000002307							
000C99E8	000000	00000000 00004040 40404040 40404040	40000000 0F004040 000C8C4F 000C9974	*.....	*	
000C9A08	000020	FFFFFFFFFF 0000000A 40404040 40404040	40404040 40404040 40404040 40404040	*.....	*	
000C9A28	000040	40404040 40404000 00082DE0 000CB968	000CB2F0 00008358 000C20FO 000C2F80	*.....0.....0.....*	*	
000C9A48	000060	000CB198 000CB2F0 000CB2DC 000CB178	00010990 000D4D48 000D2AE0 000CB1B0	*.....0.....(.....*	*	
000C9A68	000080	600D3282 000C99E8 00000000 00000000	0000000F 40404040 00000000 00000000	*-.....Y.....	*	
000C9A88	0000A0	00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000	*.....	*	
SAME							
000C9B48	000160	00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000	*.....	*	
000C9B68	000180	00000000 00000000 00000000 00	00000000	*.....	*	
DATA GROUP ID=0000002310							
000CB120	000000	00000000 000CB190 000CC110 000CB198	00000000 000C5298 00000000 600822EE	*.....A.....	*	
000CB140	000020	22082700 000CB200 000CB2B0 600822AA	0008358 000C20F0 000C2F80 000CB120	*.....0.....0.....*	*	
000CB160	000040	000CB178 000CB180 000CB190 0000B831	00083078 00082078 00000000 00000000	*.....	*	
000CB180	000060	22082700 000CB200 820827B2 00000000	00000000 00000000 00000000 000CB304	*.....	*	
000CB1A0	000080	000CC0BC 000CB200 00000000 00000000	0000000A 000CB130 00000000 60082E54	*.....	*	
000CB1C0	0000A0	00082DE0 00082F70 000C2B0B 00000000	0008358 000C2F00 000C2F80 000CB198	*.....0.....*	*	
000CB1E0	0000C0	000CB2C0 000CB180 000CB190 000B831	00083078 00082078 00000000 00000000	*.....	*	
000CB200	0000E0	000CB230 00082F70 000CB198 00640000	00000000 07000003 000CB22C 07D00007	*.....P.....P.....P.....*	*	
000CB220	000100	000CC138 00730006 40000000 0000000F	000CB260 00082F70 00000000 00C80000	*.A.....-.....H.*	*	
000CB240	000120	00000000 D7000003 000CB25C D7000007	000CC140 00730006 40000000 0000000F	*.P.....*P.....A.....*	*	
000CB260	000140	000CB290 00082F70 00000000 012C0000	00000000 07000003 000CB28C D7000007	*.....P.....P.....P.....*	*	
000CB280	000160	000CC148 00730006 40000000 0000000F	000CB2C0 00082F70 00000000 01900000	*.A.....	*	
000CB2A0	000180	00000000 D7000003 000CB2BC D7000007	000CC150 00730006 40000000 0000000F	*.P.....P.....A.....*	*	
000CB2C0	0001A0	000CB2F0 00082F70 00000000 01F40000	00000000 D7000003 000CB2EC D7000007	*.0.....4.....P.....P.....*	*	
000CB2E0	0001C0	000CC158 00730006 40000000 0000000F	00000000 0083A46 00000000 00000000	*.A.....	*	
000CB300	0001E0	00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000	*.....	*	
P E M FORMATTED STORAGE DUMP E X C E P T I O N DATE=98/02/23 TIME=10:03:42 PAGE 0016							
SAME							
000CC0E0	000FC0	00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000	*.....	*	
000CC100	000FE0	00000000 00000000 00000000 00000000	00000000	*.....	*	
DATA GROUP ID=0000002350							
000CA070	000000	000E00000 05D5E422 B8F4F7F9 F2F14040	40404040 9219898C 00000000 FF000000	*.....ENU..47921	*	
000CA090	000020	00000008 00000000 00000000 00000000	00000000 0960521F 0145622F 40404040	*.....Y.....	*	
000CA0B0	000040	40404040 00000000 00000000 00000000	E4D4C2C3 00000000 00000000 00000000	*.....UMBC.....*	*	
000CA0D0	000060	00000000 00000004 0000E200 000000AC5	00000000 04000000 00000004 00000000	*.....S...E.....*	*	
000CA0F0	000080	00000004 00000000 00000000 00400000	00004000 00000004 00000000 00000000	*.....	*	
000CA110	0000A0	00400000 00040000 00000040 00000000	40000000 04000000 00004000 00000040	*.....	*	
000CA130	0000C0	00000000 40000000 00400000 00000000	00000040 00000000 40000000 04000000	*.....	*	
000CA150	0000E0	00000000 00000004 00000000 40000000	00400000 00000004 00000000 00000000	*.....	*	
000CA170	000100	00000000 00645CF0 40404040 40400000	00000000 0000406E F0F14040 40404000	*.....*0W01	*	
000CA190	000120	0015EA00 000008D7 002E5C5 40404040	40404040 40404040 40404040 40404040	*.....P..VE	*	
000CA1B0	000140	40404040 40404040 40404000 0000A000	C85CF040 40404040 40000000 00000000	*.....H*0	*	
000CA1D0	000160	004086F0 F2404040 40400000 15EA0008	0008D700 02E5C540 40404040 40404040	*. W02.....P..VE	*	
000CA1F0	000180	40404040 40404040 40404040 40404040	40404040 40400000 000A012C 5CF04040	*.....*0	*	
000CA210	0001A0	40404040 00000000 00000000 00645CF0	40404040 40000015 EA01000 08D70002	*.....W03.....P..*	*	
000CA230	0001C0	E5C54040 40404040 40404040 40404040	40404040 40404040 40404040 40404040	*VE	*	
000CA250	0001E0	40000000 0A01905C F0404040 40404000	00000000 00000000 E6F0F40 40404040	*.....*0W04	*	
000CA270	000200	000015EA 00180008 D70002E5 C5404040	40404040 40404040 40404040 40404040	*.....P..VE	*	
000CA290	000220	40404040 40404040 40404040 0000A000	01F45CF0 40404040 40400000 00000000	*.....4*0	*	
000CA2B0	000240	000040E6 F0F54040 40404000 015EA00	200008D7 0002E5C5 40404040 40404040	*.. W05.....P..VE	*	
000CA2D0	000260	40404040 40404040 40404040 40404040	40404040 40404040 40404040 40404040	*.....	*	
000CA2F0	000280	00000000 00000000 00000000 00000000	00000000 00000000 00000000 00000000	*.....	*	
SAME							



Umbrella Programming

System Memory Access Retrieval Tool

```

000CB030 000FC0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.*  

000CB050 000FE0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.*  

DATA GROUP ID=0000002351  

000C9BA0 000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.*  

    SAME  

000C9F80 0003E0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.*  

DATA GROUP ID=0000003475 (PERM - PVT)  

000D5730 000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.*  

    SAME  

000D65F0 000EC0 00000000 00000000 00000000 FFFF0028 00000980 223C0980 309CC400 00010C00 *.....D....*  

000D6610 000EE0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.*  

    SAME  

000D6D70 001640 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000001 *.....*.*  

000D6D90 001660 F0000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *0.....*.*  

000D6DB0 001680 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.*  

    SAME  

000D6FD0 0018A0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.*  

000D6FF0 0018C0 00000000 00000000 0000 00000000 00000000 00000000 00000000 00000000 *.....*.*  

P E M FORMATTED STORAGE DUMP          E X C E P T I O N          DATE=98/02/23      TIME=10:03:42      PAGE 0017  

DATA GROUP ID=0000003601  

000C5870 000000 00000000 00000000 00000000 00000000 00000000 00F00000 00000000 00000000 *.....0.....*  

000C5890 000020 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.*  

    SAME  

000C8850 002FE0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.*  

DATA GROUP ID=0000005610  

000CC138 000000 00000000 0000000F 00000000 0000000F 00000000 0000000F 00000000 0000000F *.....*.*  

000CC158 000020 00000000 0000000F 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.*  

000CC178 000040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.*  

    SAME  

000CC298 000160 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.*  

000CC2B8 000180 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.*  

DATA GROUP ID=0000013589  

0005BCF8 000000 00200000 00010000 0001111F C3C5C1E2 C1D94040 40404040 404040E3 C9C2C5D9 *.....CEASAR      TIBER*  

0005BD18 000020 C9E4E240 40404040 4040F1F0 F0F2F0F3 F3000062 680C0009 28998C40 40404040 *IUS      1002033.....*  

0005BD38 000040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.*  

    SAME  

0005BD88 0000C0 40404040 40404040 00000040 40404040 40404040 40404040 400F0000 00004040 *.....*.*  

0005BDD8 000080 40404040 40404040 00000000 00000000 00000000 00000000 00004040 00004040 *.....*.*  

0005BDP8 000100 40404040 40404040 40404040 4040400F 0F404040 40404040 40404040 40404040 *.....*.*  

0005BE18 000120 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.*  

    SAME  

0005BE58 000160 40404040 40404040 40404040 40404040 40404040 40404040 40000000 00404040 40404040 *.....*.*  

0005BE78 000180 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.*  

    SAME  

0005BEB8 0001C0 40404040 40404040 40404040 40404040 40404040 40404040 40400000 00000000 00000000 *.....*.*  

0005BED8 0001E0 00000000 00000000 00000000 00000000 00000000 00004040 40404040 40404040 *.....*.*  

0005BEF8 000200 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.*  

DATA GROUP ID=0000047100  

0005B348 000000 00000000 00010000 0001111F 40404040 40404040 40404040 40404040 40404040 *.....*.*  

0005B368 000020 40404040 40404040 40404040 40404040 4040 40404040 40404040 40404040 *.....*.*  

DATA GROUP ID=0000047110  

0005B3A0 000000 00000000 C3C5C1E2 C1D94040 40404040 404040E3 C9C2C5D9 C9E4E240 40404040 *.....CEASAR      TIBERIUS *  

0005B3C0 000020 4040B5C9 C140C4D6 D3B6D9D6 E2C14040 40404040 40404040 40404040 40404040 *.....VIA DOLOROSA *  

0005B3E0 000040 D9D6D4C5 40404040 40404040 40404040 40404040 40404000 00001F40 40404040 *.....ROME *  

0005B400 000060 C340D400 00000F00 0F000F00 00000000 0F404040 40404040 40404040 40404040 *C M.....*  

0005B420 000080 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.*  

0005B440 0000A0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.*  

0005B460 0000C0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.*  

P E M FORMATTED STORAGE DUMP          E X C E P T I O N          DATE=98/02/23      TIME=10:03:42      PAGE 0018  

DATA GROUP ID=0000047120  

0005B490 000000 00000000 0761219F 0000000F 40404040 40404040 40404040 40404040 40404040 *.....*.*  

0005B4B0 000020 40404040 40404040 4040F1F0 F0F8F0F0 F2F0F3F3 40404040 40404040 40404040 *.....1008002033 *  

0005B4D0 000040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.*  

0005B4F0 000060 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.*  

0005B510 000080 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.*  

DATA GROUP ID=0000047130  

0005B550 000000 00000000 0000000F 0000000F 00006268 0C000012 536C0000 04293C00 0000000F *.....*.*  

0005B570 000020 00000010 0C000045 751C40F1 00000010 0C404000 0000000F 40400000 00000F40 *.....1.....*.*  

0005B590 000040 40000000 000F4040 00000000 0F404000 0000000F 40400000 00000F40 40000000 *.....*.*  

0005B5B0 000060 000F4040 00000000 0F404000 0000000F F0F1004 0C000001 567C4040 00000F00 *.....01.....*.*  

0005B5D0 000080 0000000F 40400000 0F000000 000F4040 00000F00 0000000F 40400000 0F000000 *.....*.*  

0005B5F0 0000A0 000F4040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.*  

0005B610 0000C0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.*  

    SAME  

0005B650 000100 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.*  

0005B670 000120 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*.*  

DATA GROUP ID=0000047140

```



Umbrella Programming

System Memory Access Retrieval Tool

```

0005B6A0 000000 00000000 000000F 000000F 00092899 8C000000 000F0000 00000F00 0000000F *.....*. *
0005B6C0 000020 00000000 0F000145 678C4040 00000000 0F404000 000000F 40400000 00000F40 *.....*. *
0005B6D0 000040 40000000 000F4040 00000000 0F404000 000000F 40400000 00000F40 40000000 *.....*. *
0005B700 000060 000F4040 00000000 0F404000 0000000F 40400000 0000000F 000F4040 00000F00 *..*. *
0005B720 000080 000000F 40400000 0F000000 000F4040 00000F00 0000000F 40400000 0F000000 *.....*. *
0005B740 0000A0 000F4040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*. *
0005B760 0000C0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*. *
          SAME
0005B780 000100 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*. *
0005B7C0 000120 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*. *

DATA GROUP ID=0000047190
0005B968 000000 00000000 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *....*. *
0005B988 000020 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *....*. *

DATA GROUP ID=0000047192
0005B9C0 000000 00000000 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *....*. *
0005B9E0 000020 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *....*. *

DATA GROUP ID=0000048000
0005A998 000000 0010C040 000007DF 000CC430 000CC3E0 042D0000 00000000 F0F9F9F1 F2F3F100 *....D..C.....0991231.*.
0005A9B8 000020 00000000 00000000 00000000 00000000 000007DF FFFF8040 C3C4D4C6 F3404040 *.....CDMF3*.
0005A9D8 000040 000007DF C4E3E203 00000000 00000789 00000000 00000000 00050000 000007DF *....DTS.....*.
0005A9F8 000060 0010980 223C0000 0F40C3D6 D5E3D9D6 D3404040 40409219 898C0000 00000000 *.....CONTROL.....*.

P E M FORMATTED STORAGE DUMP           E X C E P T I O N           DATE=98/02/23      TIME=10:03:42      PAGE 0019
0005AA18 000080 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
          SAME
0005AAD8 000140 00000000 00000000 00000000 00000000 00000000 00000000 00000FFF 0980101F *.....*. *
0005AAP8 000160 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
          SAME
0005AC18 000280 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
0005AC38 0002A0 101F0000 0000C3E0 042D0000 00000000 00000990 101C4009 5021C001 53136C00 *....C.....*.
0005AC58 0002C0 000001E4 D4C2F2F0 F0C5F640 40404040 40404000 00000000 00000004 C3C240C3 *....UMB200E6.....DCB C*.
0005AC78 0002E0 D6D5B3D9 D6D3000C C3A800D0 000CC3AC 00000000 00000000 00000000 00000000 *ONTROL..C.....C. ....*.
0005AC98 000300 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
          SAME
0005AE58 000440 00000000 00000000 00000000 00000000 00000000 5007CF4A 6007CF9A 6007D01A *.....*. *
0005AE78 0004E0 00000000 4007CEC0 4007CF2F 4007D03E 7007D06A 00000000 4007E1E6 6007E1F4 *.....*.2.....W..4*.
0005AE98 000500 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
0005AEB8 000520 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
0005AED8 000540 00000000 00000000 00000000 00000000 00000000 5007E88C 5007E236 00000000 *.....Y..S.....*.
0005AEF8 000560 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
0005AF18 000580 00000000 00000000 00000000 00000000 4007E50E 00000000 00000000 00000000 *.....V.....*.
0005AF38 0005A0 00000000 00000000 4007CE50 4007D078 50081FAE 5007D07E 00000000 00000000 *.....*. *
0005AF58 0005C0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
          SAME
0005B088 000700 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
0005B0B8 000720 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *

DATA GROUP ID=0000048005
0004D0F8 000000 00000000 016A0000 0000BCDA FFFFC6C2 C4E24040 4040FFFF FFFFFFFF FFFF0000 *.....FBDS.....*.
0004D118 000020 00000000 00000000 00000000 00000000 00F87FEF E1004010 0780101C 0960521C *.....*.8.....*. *
0004D138 000040 0143845C 00000000 E4D4C2C3 F1F64040 40404040 40404040 00000000 000CF100 *....*.UMBC16.....1.*.
0004D158 000060 00000000 00100000 000000BC 00000000 00000000 00000000 00000000 00000000 *.....*. *
0004D178 000080 00000000 00000000 00000000 00000000 08000190 1980050 00010000 D5D5C8D6 *.....1.....NNH0*.
0004D198 0000A0 C7D5C6C2 C4C40000 10004040 40404040 4040C6C2 C4E24040 40400000 00000000 *GNFBDD....FBDS.....*.
0004D1B8 0000C0 00000000 00000000 00000000 00000000 07F90000 00000000 00000000 00000000 00000000 *.....*.9.....*. *
0004D1D8 0000E0 40400000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
0004D1F8 000100 00000009 C7200000 00000100 000018040 00284D00 04019200 000000E2 F0F6F4F0 *....G.....(.....S0640*.
0004D218 000120 F8F0F0D2 F0F6F4F0 F8F0F001 8E018E00 00000000 00000000 00000000 00000000 *800K0640800.....*.
0004D238 000140 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
0004D258 000160 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*.H.*.
0004D278 000180 28404040 40404040 40404040 40404040 40000000 00000000 00000000 00000000 00000000 *.....*. *
0004D298 0001A0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
0004D2B8 0001C0 00000000 00000000 00000000 00000000 B81A0000 B7FC0202 01012040 02800000 *.....*. *
0004D2D8 0001E0 00012C00 F200F240 40404040 40404040 40404040 40404000 00000000 00000000 *.....2.2.....*.
0004D2F8 000200 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
0004D318 000220 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
0004D338 000240 20400028 00000001 2C021A02 1A404040 40404040 40404040 40404040 40000000 *.....*. *
0004D358 000260 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
0004D378 000280 00000000 00000000 00000000 00000000 00000000 00000000 00000000 B8100000 *.....*. *
0004D398 0002A0 B7FC0204 01012040 02800000 00009603 42034240 40404040 40404040 40404040 *.....*. *
0004D3B8 0002C0 40404000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
0004D3D8 0002E0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
0004D3F8 000300 00000000 00004040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*. *
0004D418 000320 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*. *
          SAME
0004D518 000420 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*. *

P E M FORMATTED STORAGE DUMP           E X C E P T I O N           DATE=98/02/23      TIME=10:03:42      PAGE 0020
0004D538 000440 00280000 0005402 1E0000C3 C4E2E2C5 C7F84040 40404040 40404000 50005000 *.....CDSSEG8.....*.
0004D558 000460 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
0004D578 000480 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
0004D598 0004A0 0290109 40400028 00000000 54026E00 00C3C4E2 E2C5C7F9 40404040 40404040 *.....CDSSEG9.....*.
0004D5B8 0004C0 4005000 5000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
0004D5D8 0004E0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
0004D618 000520 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*. *
          SAME
0004D678 000580 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 40404040 *.....*. *
0004D698 0005A0 40404040 400001C5 D584F040 40404040 40404040 40404040 40404040 40404040 *.....ENU0.....*.
0004D6B8 0005C0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*. *
```



Umbrella Programming

System Memory Access Retrieval Tool

SAME

```

0004D718 000620 40404040 40404040 40404040 40404040 40404040 40404040 00180050 3000006F * .....*
0004D738 000640 D5404001 01404040 40404040 40000000 00D7F4F9 F0F7FC03 40000000 00404040 *N .. P49070C .....*
0004D758 000660 40404040 40000000 00404040 40404040 40400000 0000D7F4 F9F0F7F0 C9400000 * .....P49070I ..*
0004D778 000680 00004040 40404040 40400000 0000F040 00180050 3000006F D5404001 01404040 *.. ..0 .....N ..*
0004D798 0006A0 40404040 40000000 00D7F4F9 F0F7FC03 40000000 00E5F8F1 F1F9F0C3 40000000 * .. P49070C ..V81190C ..*
0004D7B8 0006C0 00404040 40404040 40400000 0000D7F4 F9F0F7F0 C9400000 0000E5F8 F1F9F0F0 * .. P49070I ..V81190*
0004D7D8 0006E0 C9400000 0000F140 00180050 E080006F E8404006 0ED7F4F9 F1F1F9C3 40000000 *I ..1 .....Y ..P49119C ..*
0004D7F8 000700 00404040 40404040 40000000 00404040 40404040 40400000 0000F240 * .. .....* ..*
0004D818 000720 40400000 00004040 40404040 40400000 00004040 40404040 40400000 0000F240 * .. .....* ..2 *
0004D838 000740 009B0084 2000006F D5404000 04040400 40404040 40400000 00404040 40404040 * .. ..N .. .....*
0004D858 000760 40000000 004AF2F3 F0F4FBC3 40000000 00004040 40404040 40400000 00004040 * .. M23048C ..* ..*
0004D878 000780 40404040 40400000 0000D4F2 F3F0F4F8 C9400000 0000F940 00FF0084 3000006F * .. M23048I ..9 .....*
0004D898 0007A0 D5404001 01404040 40404040 40000000 00404040 40404040 40000000 00404040 *N .. ..* ..*
0004D8B8 0007C0 40404040 40000000 00404040 40404040 40400000 00004040 40404040 40400000 * .. ..* ..*
0004D8D8 0007E0 00004040 4040C500 04C10001 C300C600 04C20001 C300C700 04C30001 C300C800 * .. E..A..C.F..B..C.G..C..C.H.* ..
0004D8F8 000800 04C40006 C300C400 04CA0015 C300C500 04DF0001 C300C600 04E00001 C300C700 *..D..C.D..C.E..C.F..C.G.* ..
0004D918 000820 04E10001 C300C800 04E20006 C300C400 04E80015 C300C500 04FD0001 C300C600 *..C.H..S..C.D..Y..C.E..C.F.* ..
0004D938 000840 04FE0001 C300C700 04FF0001 C300C800 05000006 C300C400 05060015 C300C500 *..C.G..C.H..C.D..C.E.* ..
0004D958 000860 051B0001 C300C600 051C0001 C300C700 051D0001 C300C800 051E0006 C300C400 *..C.F..C.G..C.H..C.D.* ..
0004D978 000880 05240015 C300C500 05390001 C300C600 053A0001 C300C700 053B0001 C300C800 *..C.E..C.F..C.G..C.H.* ..
0004D998 0008A0 053C0006 C300C400 05420015 C300C500 05570001 C300C600 05580001 C300C700 *..C.D..C.E..C.F..C.G.* ..
0004D9B8 0008C0 05590001 C300C800 055A0006 C300C400 05600015 C300C500 05750001 C300C600 *..C.H..C.D..C.E..C.F.* ..
0004D9D8 0008E0 05760001 C300C700 05770001 C300C800 05780006 C300C400 057E0015 C300C500 *..C.G..C.H..C.D.=C.E.* ..
0004D9F8 000900 05930001 C300C600 05940001 C300C700 05950001 C300C800 05960001 C300C900 *..C.F..C.G..C.H..C.I.* ..
0004DA18 000920 05970007 C016400 05980002 C016300 05A00002 C016800 05A20003 C016B00 *..C..B..B..B..C..* ..
0004DA38 000940 05A50001 C016C00 05A60003 C016B00 05A90001 C016C00 05AA0003 C016B00 *..C..C..C..C..C..* ..
0004DA58 000960 05AD0001 C016C00 05A80003 C016B00 05B10001 C016C00 05B20003 C016B00 *..C..C..C..C..C..C..* ..
0004DA78 000980 05B50001 C016C00 05B60003 C016B00 05B90001 C016C00 05BA0003 C016B00 *..C..C..C..C..C..C..* ..
0004DA98 0009A0 05BD0001 C016C00 05BE0003 C016B00 05C10001 C016C00 05C20003 C016B00 *..C..C..C..A..C..B..C..* ..
0004DAB8 0009C0 05C50001 C016C00 05C60003 C016B00 05C90001 C016C00 05CA0003 C016B00 *..E..C..F..C..I..C..C..* ..
0004DAD8 0009E0 05CD0001 C016C00 05CE0003 C016B00 05D10001 C016C00 05D20003 C016B00 *..C..C..C..J..C..K..C..* ..
0004DAF8 000A00 05D50001 C016C00 05D60003 C016B00 05D90001 C016C00 05DA0003 C016B00 *..N..C..O..C..R..C..C..* ..
0004DB18 000A20 05DD0001 C016C00 05DE0003 C016B00 05E10001 C016C00 05E20003 C016B00 *..C..C..C..C..S..C..* ..
0004DB38 000A40 05E50001 C016C00 05E60003 C016B00 05E90001 C016C00 05EA0003 C016B00 *..V..C..W..C..Z..C..* ..
0004DB58 000A60 05ED0001 C016C00 05EE0003 C016B00 05F10001 C016C00 05F20003 C016B00 *..C..C..1..C..2..C..* ..
0004DB78 000A80 05F50001 C016C00 05F60003 C016B00 05F90001 C016C00 05FA0003 C016B00 *..5..C..6..C..9..C..C..* ..
0004DB98 000A90 05FD0001 C016C00 05FE0003 C016B00 06010001 C016C00 06020003 C016B00 *..C..C..C..C..C..C..* ..
0004DBB8 000AC0 06050001 C016C00 06060003 C016B00 06090001 C016C00 060A0003 C016B00 *..C..C..C..C..C..C..* ..
0004DBD8 000AE0 060D0001 C016C00 060E0003 C016B00 06110001 C016C00 06120003 C016B00 *..C..C..C..C..C..C..* ..
0004DBF8 000B00 06150001 C016C00 06160003 C016B00 06190001 C016C00 061A0003 C016B00 *..C..C..C..C..C..C..* ..
0004DC18 000B20 061D0001 C016C00 061E0003 C016B00 06210001 C016C00 06220003 C016B00 *..C..C..C..C..C..C..* ..
0004DC38 000B40 06250001 C016C00 06260003 C016B00 06290001 C016C00 062A0003 C016B00 *..C..C..C..C..C..C..* ..

```

P	E	M	FORMATTED STORAGE DUMP	E	X	C	E	P	T	I	O	N	DATE=98/02/23	TIME=10:03:42	PAGE 0021
0004DC58	000B60	062D0001	C3016C00 062E0003 C3016B00	06310001	C3016C00	06320036	C3015D00	*....C.....C.....C.....C.....C.....*							
0004DC78	000B80	00000000	00000000 00000000 00000000	00000000	00000000	00000000	00000000	*.....*							
0004ECD8	001BE0	00000000	00000000 00000000 00000000	00000000	00000000	00000000	00000000	*.....*							

DATA GROUP ID=0000048006

0004ED08	000000	00000000	00000000 00000000 00000000	00000000	00000000	00000000	00000000	*.....*
		SAME						
0004FOC8	0003C0	00000000	00000000 00000000 00000000	00000000	00000000	00000000	00000000	*.....*
0004FOE8	0003E0	00000000	00000000 00000000 00000000	00000000	00000000	00		*.....*

DATA GROUP ID=0000048007

0004F110	000000	00000000	00000000 00000000 00000000	00000000	00000000	00000000	00000000	*.....*
		SAME						
0004F1B0	0000AO	00000000	00000000 00000000 00000000	00000000	00000000	00000000	00000000	*.....*
0004F1D0	0000CO	00000000	00000000 00000000 00000000	00000000	00000000			*.....*

DATA GROUP ID=0000048009

0004F1E8	000000	00000000	00B80000 00000000 00000000	00000000	00000000	00000000	00000000	*.....*
0004F208	000020	00000000	00000000 00000000 00000000	00000000	00000000	00000000	00000000	*.....*
0004F228	000040	00000000	00000000 00000000 00000000	00000000	00000000	00000000	00000000	*.....*
0004F248	000060	40404000	0007DFC4 E3E20003 E3F5F8F0	F1F5D440	80000040	00000000	00000000	* .. DTS .. T58015M ..*
0004F268	000080	00000000	40404040 40404040 D5C5E640	40400000	077C4040	40404040	00000000	* .. NEW ..*
0004F288	0000A0	00000789	40404000 40404040 40404040	40404040	40404040	00000000	00000000	*.....*
0004F2A8	0000C0	00000000	00000000 00000000 00000000	00000000	00000000	00000000	00000000	*.....*
		SAME						
000518C8	0026E0	00000000	00000000 00000000 00000000	00000000	00000000	00000000	00000000	*.....*
000518E8	002700	00000000	00000000 00000000 00000000	00000000	00000000			*.....*

DATA GROUP ID=0000048035

00051908	000000	00000000	C3C4D4C6 F3404040 00000000	00002800				* .. CDMF3 ..*
----------	--------	----------	----------------------------	----------	--	--	--	----------------

DATA GROUP ID=0000048066

0005B0E0	000000	0005B0EC	0005B20C 0005B18C 0004C098	0005BB82	0000BCB8	FFFF0980	223C9991	*.....*
0005B100	000020	231FFFFF	0780101F 00000000 0004C0C8	000E0000	0000092E	FFFF0780	101C9991	* .. H ..*
0005B120	000040	231FFFFF	0780101F 00000000 0004C0E0	000E0000	0000092E	FFFF0780	101C9991	* ..*
0005B140	000060	231FFFFF	0780101F 00000000 0004C100	00050000	000007D5	0010980	223C9991	* .. A .. N ..*
0005B160	000080	231FFFFF	0780101F 00000000 0004C040	00050000	000007DF	0010980	223C0981	* ..*
0005B180	0000A0	231FFFFF	0780101F 00000000 0004C070	0005BB82	0000BCB8	FFFF0980	223C9991	* ..*
0005B1A0	0000C0	231FFFFF	0780101F 00000000 0004C078	00050000	0000BDEC	FFFF0980	223C9991	* ..*
0005B1C0	0000E0	231FFFFF	0780101F 00000000 0004C080	00050000	0000BF1E	FFFF0980	223C9991	* ..*
0005B1E0	0000100	231FFFFF	0780101F 00000000 0004C088	00050000	0000BD80	FFFF0980	223C9991	* ..*
0005B200	000120	231FFFFF	0780101F 00000000 0004C090	00050000	0000BD80	FFFF0980	223C9991	* ..*
0005B220	000140	231FFFFF	0780101F 00000000 00000000					* ..*

P E M FORMATTED STORAGE DUMP E X C E P T I O N DATE=98/02/23 TIME=10:03:42 PAGE 0022

DATA GROUP ID=0000048091 (PERM - PVT)



Umbrella Programming

System Memory Access Retrieval Tool

```

0005C0B8 000000 00000000 0980223C 0100342C 0000000F 0000000F 0000000F 0000000F 40404040 *.....*. *
0005C0D8 000020 40404040 40404040 40404040 40404040 00000000 00000000 00000000 00000000 *.....*. *
0005C0F8 000040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*. *
0005C138 SAME
0005C138 000080 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*. *
0005C158 000000 40404040
DATA GROUP ID=0000048092
00051A10 000000 00000000 4007AE12 4007AE18 6007AE1E 4007AE24 4007AE2A 4007AE30 4007AE36 *.....*. *
00051A30 000020 4007AE3C 4007AE42 4007AE48 6007BE22 4007BFEC 4007AE4E 00000000 00000000 *.....*. *
00051A50 000040 00000000 00000000 00000000 00000000 00000000 *.....*. *
DATA GROUP ID=0000048310
00053FE0 000000 00642D0 07900000 00160000 40400018 0050E080 006FE8C2 400E0E40 40404040 *.....YB..*. *
00054000 000020 40404000 00000000 40404040 40404000 00000040 40404040 40404000 000000D4 *.....*.M*
00054020 000040 40404040 40404040 00000000 40404040 40404040 40404040 40404040 *.....*. *
00054040 000060 00000000 C1400018 0050E080 006FE840 400E0E40 40404040 40404000 00000040 *.....A.....Y..*. *
00054060 000080 40404040 40404040 00000040 40404040 40404000 000000D4 40404040 40404040 *.....M..*. *
00054080 000000 00000000 40404040 40404040 00000000 40404040 40404040 00000000 40404040 40404040 *.....B..*. *
000540A0 0000C0 0050A000 006FD540 400E0040 40404040 40404000 00000040 40404040 40404000 40404000 *.....N..*. *
000540C0 000080 00000000 40404040 40404000 000000C1 40404040 40404040 40404040 00000000 40404040 *.....A..*. *
000540E0 000100 40404040 00000000 40404040 40404040 00000000 C3400018 00503000 006FD540 *.....C.....N*. *
00054100 000120 40010140 40404040 40404000 000000D7 F4F9F0F7 F0C34000 00000040 40404040 *.....P49070C..*. *
00054120 000140 40404000 00000040 40404040 40404040 00000000 D7F4F9F0 F7F0C940 00000000 *.....P49070I..*. *
00054140 000160 40404040 40404040 00000000 C7400018 00503000 006FD540 40010140 40404040 *.....G.....N..*. *
00054160 000180 40404000 000000D7 F4F9F0F7 F0C34000 00000040 40404040 40404000 00000000 40404040 40404040 *.....P49070C..*. *
00054180 0001A0 40404040 40404040 00000000 D7F4F9F0 F7F0C940 00000000 40404040 40404040 40404040 *.....P49070I..*. *
000541A0 0001C0 00000000 D140000C 00281080 006FD540 40020240 40404040 40404000 000000D7 *.....J.....N..*.P*
000541C0 0001E0 F4F9F0F7 F1C34000 00000040 40404040 40404000 000000D3 40404040 40404040 *49071C..*.L*. *
000541E0 000220 00000000 D7F4F9F0 F7F1C940 00000000 40404040 40404040 00000000 D2400018 *.....P49071I..*.K..*. *
00054200 000220 00501080 006FD540 40303400 40404040 40404000 000000D7 F4F9F0F7 F1C34000 *.....N..*.P49071C..*. *
00054220 000240 00000040 40404040 40404000 000000D4 40404040 40404040 00000000 D7F4F9F0 *.....M..*.P490* *
00054240 000260 F7F1C940 00000000 40404040 40404040 00000000 D6400018 00503000 006FD540 *71I ..O.....N*. *
00054260 000280 40010140 40404040 40404000 00000040 40404040 40404000 00000000 40404040 00000000 *.....N1663* *
00054280 0002A0 F1404000 00000000 40404040 40404040 00000000 40404040 40404040 00000000 *1 ..*. *
000542A0 0002C0 D5F1P6F6 F3F1L4040 00000000 D8400018 00503000 006FD540 40010140 40404040 *N16631 ..Q.....N..*. *
000542C0 0002E0 40404000 000000D7 F4F9F0F7 F0C34000 000000D4 F2F1F0F4 F7E74000 00000040 *.....P49070C..*.M21047X..*. *
000542E0 000300 40404040 40404040 00000000 D7F4F9F0 F7F0C940 00000000 D4F2F1F0 F4F7E740 *.....P49070I..*.M21047X* *
00054300 000320 00000000 D9400018 0050E080 006FD540 4000C040 40404040 00000000 D7F4F9F0 *.....R.....N..*. *
00054320 000340 40404040 40404000 000000D4 F2F2F0F4 F7E74000 000000D4 40404040 40404040 *.....M22047X..*.M *
00054340 000360 00000000 40404040 40404040 00000000 D4F2F2F0 F4F7E7400 00000000 E240000C *.....M22047X..*.S..*. *
00054360 000380 00281000 006FD540 40010140 40404040 40404000 000000D7 F4F9F0F7 F0C34000 *.....N..*.P49070C..*. *
00054380 0003A0 00000040 40404040 40404040 00000040 40404040 40404040 00000000 D7F4F9F0 *.....P490* *
000543A0 0003C0 F7F0C940 00000000 40404040 40404040 00000000 E3400000 00283000 006FD540 *70I ..T.....N*. *
000543C0 0003E0 40010140 40404040 40404000 000000D7 F4F9F0F7 F0C34000 000000E7 F7F8F0F4 *..P49070C..*.X7804* *
000543E0 000400 F0C34000 00000040 40404040 40404040 00000000 D7F4F9F0 F7F0C940 00000000 *OC ..P49070I..*. *
00054400 000420 40404040 40404040 00000000 E4000108 00506080 006FD540 40060400 40404040 *.....U.....N..*. *
00054420 000440 40404000 00000000 40404040 40404000 00000000 40404040 40404000 000000D4 *.....M..*. *
00054440 000460 40404040 40404040 00000000 40404040 40404040 00000000 D4F1F5F0 F4F0E740 *.....M15040X* *
P E M FORMATTED STORAGE DUMP E X C E P T I O N DATE=98/02/23 TIME=10:03:42 PAGE 0023
00054460 000480 00000000 E5400018 00503000 006FD540 40010140 40404040 40404000 000000D7 *....V.....N..*.P*
00054480 0004A0 F4F9F0F7 F0C34000 000000D4 F2F1F0F4 F5E74000 00000040 40404040 40404040 *49070C..*.M21045X..*. *
000544A0 0004C0 00000000 D7F4F9F0 F7F0C940 00000000 D4F2F1F0 F4F5E740 00000000 E6400018 *.....P49070I..*.M21045X..*.W..*. *
000544C0 0004E0 00503000 006FD540 40010140 40404040 40404000 000000D7 F4F9F0F7 F0C34000 *....N..*.P49070C..*. *
000544E0 000500 000000D4 F2F3F0F4 F5C34000 00000040 40404040 40404000 00000000 D7F4F9F0 *.....M23045C..*.P490* *
00054500 000520 F7F0C940 00000000 D4F2F1F0 F4F5C940 00000000 E7400018 00503000 006FD540 *70I ..M23045I ..X.....N*. *
00054520 000540 40010140 40404040 40404000 000000D7 F4F9F0F7 F0C34000 00000040 40404040 *.....P49070C..*. *
00054540 000560 40404000 00000000 40404040 40404040 00000000 D7F4F9F0 F7F0C940 00000000 *.....P49070I..*. *
00054560 000580 D4F1P5F0 F4F3E740 00000000 E8400018 00503000 006FD540 40010140 40404040 *M15043X ..Y.....N..*. *
00054580 0005A0 40404000 000000D7 F4F9F0F7 F0C34000 000000D4 F2P3F0F4 F7C34000 00000040 *.....P49070C..*.M23047C..*. *
000545A0 0005C0 40404040 40404040 00000000 D7F4F9F0 F7F0C940 00000000 D4F2F3F0 F4F7C940 *.....P49070I..*.M23047I..*. *
000545C0 0005E0 00000000 E9400018 00503000 006FD540 40010140 40404040 40404000 000000D7 *....Z.....N..*.P*
000545E0 000600 F4F9F0F7 F0C34000 00000040 40404040 00000000 D4F40040 40404040 00000000 *49070C ..*. *
00054600 000620 00000000 D7F4F9F0 F7F0C940 00000000 40404040 40404040 00000000 F400018 *.....P49070I..*. *
00054620 000640 00503000 006FD540 40010140 40404040 40404000 000000D7 F4F9F0F7 F0C34000 *....N..*.P49070C..*. *
00054640 000660 000000E5 F8F1F1F9 F0C34000 00000040 40404040 40404040 00000000 D7F4F9F0 *.....P490* *
00054660 000680 F7F0C940 00000000 E5F8F1F1 F9F0C940 00000000 F1400018 0050E080 006FE840 *70I ..V81190I ..1 ..Y* *
00054680 0006A0 4060ED7 F4F9F1F1 F9C34000 00000040 40404040 40404040 00000040 40404040 40404040 *..P49119C ..*. *
000546A0 0006C0 40404000 00000040 40404040 40404040 00000000 40404040 40404040 00000000 *.....P490* *
000546C0 0006E0 40404040 40404040 00000000 F240009B 00842000 006FD540 40000040 40404040 *.....2 ..N..*. *
000546E0 000700 40404000 00000000 40404040 40404040 00000000 D2P3F0F4 F8C34000 00000000 *.....M23048C..*. *
00054700 000720 40404040 40404040 00000000 40404040 40404040 00000000 D4F2P3F0 F4F8C940 *.....M23048I..*. *
00054720 000740 00000000 F94000FF 00843000 006FD540 40010140 40404040 40404040 00000040 *.....9 ..N..*. *
00054740 000760 40404040 40404040 00000040 40404040 40404040 40404040 40404040 40404040 *.....P490* *
00054760 000780 00000000 40404040 40404040 00000000 40404040 40404040 00000000 40404040 40404040 *.....P490* *
00054780 0007A0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....P490* *
000547E0 SAME
00054F60 000F80 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....P490* *
00054F80 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....P490* *
00054F88 000000 005CDF8 00F80000 D9C5D300 C3C7F1C4 C5D74040 0030001 0060007 000B0015 *...8..REL.CG1DEP..*. *
00054F8C 000020 FFFF6DC4 E2C6D740 40400001 FDE9FDE9 D6C4E2D5 C6D74040 0020001 FDE8FDEA *..ODSPF ..Z.ZODSNFP ..Y..*. *
00054F88 000040 FFFF6D9 C4E7F4040 40400001 000DFFFF E3D9C4F0 F1404040 0010001 001E3D9 *..TRDX ..Z.TRD01 ..TR*..*. *
00055008 000060 C4F0F240 40400001 0020002 E3D9C4F0 F3404040 0010003 003E3D9 C4F0F440 *D02 ..TRD03 ..TRD04 * *
00055028 000080 40400001 00400004 E3D9C4F0 F5404040 0010005 005E3D9 C4F0F640 40400001 *..TRD05 ..TRD06 ..*. *
00055048 0000A0 00060006 E3D9C4F0 F7404040 0010007 007E3D9 C4F0F840 40400001 0080008 *..TRD07 ..TRD08 ..*. *
00055068 0000C0 E3D9C4F0 F9404040 0010009 009E3D9 C4F1F040 40400001 00AA000A E3D9C4F1 *TRD09 ..TRD10 ..TRD1* *
00055088 0000E0 F1404040 001000B 000BE3D9 C4F1F240 40400001 000C000C 00000000 00000000 *1 ..TRD12 ..*. *
000550A8 000100 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....P490* *
0005578E 002940 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....P490* *
00057908 002960 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....P490* *

```



Umbrella Programming

System Memory Access Retrieval Tool

DATA GROUP ID=0000048320

```

00051A88 000000 00000000 00050000 0000BCCC FFFF0980 223C9991 231FFFFF 0780101F D540D540 *.....N N *
00051A88 000020 00000000 0980220C 0191615C 00000000 D7E9D45B 5BD9C5C7 40404040 40404040 *...../*...PZMS$REG *
00051AC8 000040 40404040 00000000 000000E4 D4C20000 40404040 40404040 40404040 *.....UMB. *
00051A88 000060 40404040 40404040 40404040 40404040 E3D9D5C7 40404040 40404040 40404040 *.....TRNG *
00051B08 000080 40404040 40404040 40404040 40404040 404040C3 C9C3E2E5 E2C1D4D3 E5F34040 *.....CICSVSAMLV3 *
00051B28 0000A0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....SAME

```

P E M FORMATTED STORAGE DUMP E X C E P T I O N DATE=98/02/23 TIME=10:03:42 PAGE 0024

```

00051B88 000100 40404040 40404040 40404040 40404040 40404040 40404040 40054040 40404040 *.....*
00051B88 000120 40404040 40404040 40404040 40404040 4BFL2E4 E2C44040 40C5D5E4 E740404B *.....12USD ..ENUX ..
00051BC8 000140 40404040 40404040 00046140 40404040 40404040 40404040 E2B8E2D7 D9C9D5E3 *.....SYSPRINT*
00051B88 000160 D7C5D4C4 D4D74040 40404040 40404040 D5E8D5D5 E8404040 E8D540D5 40404040 *.....PMDMP NYNNY YN N *
00051C08 000180 E8D5E8E8 40404040 40404040 40404040 405D5D5D D6D5E8D6 40404040 40404040 *.....YNYY NNNONYO *
00051C28 0001A0 D5D5F040 40404040 40404040 40404040 C4C2F240 50C5CSF1 D5404040 E4E2C140 *.....NNO DB2 .E1N USA *
00051C48 0001C0 40404E42 C1404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....USA E1 *
00051C68 0001E0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*
00051C88 000200 40404040 40404040 40404040 40404040 4040C3C1 C4404040 206B2020 206B2020 *.....CAD ..
00051C88 000220 206B2020 206B2020 214B2020 604BF200 4AE7D740 4040206B 2020206B 2020206B *.....-2.MXP ..,..,..
00051CC8 000240 2020206B 2020214B 2020604B F200E4E2 5B404040 204B2020 204B2020 204B2020 *.....-2.USS ..,..,..
00051C88 000260 204B2020 215B2020 605BF200 E4E2C440 4040206B 2020206B 2020206B 2020206B *.....$-$2.USD ..,..,..
00051D08 000280 2020214B 2020604B F2000000 00000000 00000000 00000000 00000000 00000000 *.....-2. ....,..
00051D28 0002A0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*
00051C88 SAME *.....*
00052008 000580 00000000 00000000 00000000 00000000 00000000 00004040 40404040 40404040 *.....*
00052028 0005A0 0001C5D5 E4F04040 40404040 40404040 40404040 40404040 40404040 40404040 *.....ENU0 *
00052048 0005C0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*
00052048 SAME *.....*
000520A8 000620 40404040 40404040 40404040 40404040 40400000 00000000 00000000 00000000 *.....*
000520C8 000640 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*
000520E8 000660 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*

```

DATA GROUP ID=0000048321

```

00052118 000000 00000000 0005BB82 0000BCCC FFFF0980 223C9991 231FFFFF 0780101F D540D540 *.....A.....N N *
00052138 000020 00000000 0960325C 0161202C 00011445 E4D4C2F2 F0F14040 40404040 40404040 *.....-.*./....UMB201 *
00052158 000040 40404040 00000000 000000E4 D4C20000 40404040 40404040 40404040 *.....UMB. *
00052178 000060 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*
00052198 000080 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*
000521B8 0000A0 40404040 40404040 40404040 40404040 40404040 *.....*

```

DATA GROUP ID=0000048322

```

0005A720 000000 00000000 00050000 0000BCCC2 FFFF0980 223C9991 231FFFFF 0780101F D540D540 *.....B.....N N *
0005A740 000020 00000000 0960325C 0161202C 00011445 E4D4C2F2 F0F14040 40404040 40404040 *.....-.*./....UMB201 *
0005A760 000040 40404040 00000000 000000E4 D4C20000 E4D4C200 00000000 00000000 00000000 *.....UMB.UMB. ....,..
0005A780 000060 F9F6F0F3 40404040 4F2F0F1F0 F0F0F000 00000000 00000000 00000000 00000000 *9603 2010000 *.....*
0005A7A0 000080 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*
0005A880 000160 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*

```

DATA GROUP ID=0000048512

```

00057938 000000 00005020 00380000 D9D7C740 E4C5F0F1 00020001 00001F00 003FFFFFF 00001F00 *.....RPG UE01. ....,..
00057958 000020 003FE4C5 F0F20020 00010000 4FF9999F FFF0F000 4F99999F 00000000 00000000 *.....UE02. ....,..
00057978 000040 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*
0005A6D8 002DA0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*

```

P E M FORMATTED STORAGE DUMP E X C E P T I O N DATE=98/02/23 TIME=10:03:42 PAGE 0025

DATA GROUP ID=0000048551

```

0005BA18 000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*
0005BC98 000280 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*
0005BCB8 0002A0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 *.....*

```

DATA GROUP ID=0000048620

```

000521F0 000000 00000000 00050000 0000BDEC FFFF0980 223C9991 231FFFFF 0780101F D540D540 *.....N N *
00052210 000020 00000000 0980220C 0191554C 00000000 D7E9D45B 5BD9C5C7 40404040 40404040 *.....PZMS$REG *
00052230 000040 40404040 00000000 D4C20000 40404040 40404040 40404040 0980220C 0191554C *.....UMB. ....,..
00052250 000060 C5C5F1D5 C5D7D9E5 D7D9E540 D6D540D5 D6400000 000FC5C5 F1E54B40 40404040 *EE1NEPRVPRV ON NO ..EE1V. *
00052270 000080 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*
00052290 0000A0 40404040 40C5D7D9 E5D5D640 40404040 40404040 40404040 40000010 00000020 *.....EPRVNO. ....,..
000522B0 0000C0 00000010 00000020 00000001 00000020 00000010 00000020 00000010 00000020 *.....*
000522D0 0000E0 00000010 00000000 00000001 00000001 40404040 40404040 40404040 40404040 *.....*
000522F0 000100 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*
00052310 000120 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....*

```

DATA GROUP ID=0000048726

```

00052358 000000 00000000 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....
00052378 000020 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....
00052418 0000C0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....
00052438 0000E0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....

```

DATA GROUP ID=0000048926

```

00052470 000000 00069C30 03B60000 40404009 60521C01 43943C00 0004C1C6 D900F0F1 D1C1D5D1 *.....-....AFR.01JANJ*
00052490 000020 C1D5E4C1 D9C9C540 40404000 08F0F2C6 C5C2C6C5 C2D9E4C1 D9C9C540 40400009 *ANUARIE ..02FEBFEBRUARIE ..
000524B0 000040 F0F3D4C1 D9D4C1C1 D9E34040 40404040 400005F0 F4C1D7D9 C1D7D9C9 D3404040 *03NARMAART ..04APRAPRIL *
000524D0 000060 40404040 0005F0F5 D4C5C9D4 C5C94040 40404040 40404000 403F0F6D1 E4D5D1B4 *..05MEIMEI ..06JUNJUJU*
000524F0 000080 D5C9C540 40404040 40400005 F0F7D1E4 D3D1B4D3 C9C54040 40404040 400005F0 *NIE ..07JULJULIE ..0*

```



Umbrella Programming

System Memory Access Retrieval Tool

```

00052510 0000A0 F8C1E4C7 C1E4C7E4 E2E3E4E2 40404040 0008F0F9 E2C5D7E2 C5D7E3C5 D4C2C5D9 *8AUGAUGUSTUS ..09SEPSEPTEMBER*
00052530 0000C0 40404000 09F1F0D6 D2E3D6D2 E3D6C2C5 D9404040 40400007 F1F1D5D6 E5D5D6E5 * ..10OKTOKTOBER ..11NOVNOV*
00052550 0000E0 C5D4C2C5 D9404040 400008F1 F2C4C5E2 C4C5E2C5 D4C2C5D9 40404040 0008C5D5 *EMBER ..12DESEDEMBER ..EN*
00052570 000100 C700F0F1 D1C1D5D1 C1D5E4C1 D9E8A4040 40404000 07F0F2C6 C5C2C6C5 C2D9E4C1 *G.01JANJANUARY ..02FEBFEBRUAR*
00052590 000120 D9E8A4040 40400008 F0F3D4C1 D9D4C1D9 C3C8A4040 40404040 400005F0 F4C1D7D9 *RY ..03MARMARCH ..04APR*
000525B0 000140 C1D7D9C9 D3404040 40404040 40005F05 D4C1E8D4 C1E8A4040 40404040 40404000 *APRIL ..05MAYMAY ..*
000525D0 000160 03F0F6D1 E4D5D1E4 D5C5C4040 40404040 40400004 F0F7D1E4 D3D1E4D3 E8A404040 *.06JUNJUNE ..07JULJULY *
000525F0 000180 40404040 400004F0 8C1E4C7 C1E4C7E4 E2E34040 40404040 0006F0F9 E2C5D7E2 * ..08AUGAUGUST ..09SEPS*
00052610 0001A0 C5D7E3C5 D4C2C5D9 40404000 09F1F0D6 C3E3D6C3 E3D6C2C5 D9404040 40400007 *EPTEMBER ..10OCTOCTOBER ..
00052630 0001C0 F1F1D5D6 E5D5D6E5 C5D4C2C5 D9404040 400008F1 F2C4C5C3 C4C5C3C5 D4C2C5D9 *11NOVNOVEMBER ..12DECEDECEMBER*
00052650 0001E0 40404000 008C5D5 E400F0F1 D1C1D5D1 C1D5E4C1 D9E8A4040 40404000 07F0F2C6 * ..ENU.01JANJANUARY ..02F*
00052670 000200 C5C2C6C5 C2D9E4C1 D9E8A4040 40400008 F0F3D4C1 D9D4C1D9 C3C8A4040 40404040 *EBFEBRUARY ..03MARMARCH *
00052690 000220 400005F0 F4C1D7D9 C1D7D9C9 D3404040 40404040 0005F0F5 D4C1E8D4 C1E8A4040 *.04APRAPRIL ..05MAYMAY *
000526B0 000240 40404040 40404000 03F0F6D1 E4D5D1E4 D5C54040 40404040 40404004 F0F7D1E4 * ..06JUNJUNE ..07JU*
000526D0 000260 D3D1E4D3 E8A404040 40404040 400004F0 F8C1E4C7 C1E4C7E4 E2E34040 40404040 *LJULY ..08AUGAUGUST *
000526F0 000280 0006F0F9 E2C5D7E2 C5D7E3C5 D4C2C5D9 40404000 09F1F0D6 C3E3D6C3 E3D6C2C5 *..09SEPSEPTEMBER ..10OCTOCTOBER*
00052710 0002A0 D9404040 40400007 F1F1D5D6 E5D5D6E5 C5D4C2C5 D9404040 400008F1 F2C4C5C3 *R ..11NOVNOVEMBER ..12DEC*

```

P	E	M	FORMATTED STORAGE DUMP	E X C E P T I O N	DATE=98/02/23	TIME=10:03:42	PAGE
00052730	0002C0	C4C5C3C5 D4C2C5D9	40404040 0008C5E2	D700F0F1 C5D5C5C5 D5C5D9D6 40404040	*DECEMBER ..ESP.01ENEENERO *		0026
00052750	0002E0	40404000 05F0F2C2	C5C2C6C5 C2D9C5D9	D6404040 40400007 F0F3D4C1 D9D4C1D9	* ..02BEFEBRERO ..03MAR*		
00052770	000300	E9D64040 40404040	400005F0 F4C1C2D9	C1C2D9C9 D3404040 40404040 0005F0F5	*ZO ..04ABRABRIL ..05*		
00052790	000320	D4C1E8D4 C1E8D640	40404040 40404040	04F0F6D1 E4D5D1E4 D5C9D640 40404040	*MAYMAYO ..06JUNJUNIO *		
000527B0	000340	40400005 F0F7D1E4	D3D1E4D3 C9D64040	40404040 400005F0 F8C1C7D6 C1C7D6E2	* ..07JULJULIO ..08AGOAGOS*		
000527D0	000360	E3D64040 40404040	0006F0F9 E2C5D7E2	C5D7E3C9 C5D4C2D9 C5404000 C0AF1F0D6	*TO ..09SEPSEPTIEMBRE ..100*		
000527F0	000380	C3E3D6C3 E3C4C2D9	C5404040 40400007	F1F1D5D6 E5D5D6E6 C95C4D2 C95C54040	*CTOCTUBRE ..11NOVNOVIEMBRE *		
00052810	0003A0	400009F1 F2C4C9C3	C4C9C3C9 C5D4C2D9	C5404040 00090000 00000000 00000000	* ..12IDCDICICIEMBRE ..*		
00052830	0003C0	00000000 00000000	00000000 00000000	00000000 00000000 00000000 00000000	* ..*		
00052840	0003E0	SAME			* ..*		
00053F90	001B20	00000000 00000000	00000000 00000000	00000000 00000000 00000000 00000000	* ..*		
00053FB0	001B40	00000000 0000			* ..*		

DATA GROUP ID=0000488721

0005BF20	000000	00000000 40404040	40400000 40404040	40404040 40404040 40000040 40404040	*.... *		
0005BF40	000020	40404040 40404040	40404040 40404040	40404040 40404040 40404040 40404040	* *		
0005BF60	000040	40404000 00404040	40404040 40404040	00000000 00000000 000F0000 000000F00	* *		
0005BF80	000060	00000000 00000000	00000000 00000000	00000000	* *		

P E M FORMATTED STORAGE DUMP E X C E P T I O N DATE=98/02/23 TIME=10:03:42 PAGE 0027

P E M F O R M A T T E D D U M P I N D E X

CONTROL BLOCK	@ADDRESS	PAGE
ACTIVITY TRACE TABLE	2	
INTERNAL TRANSACTION CONTROL BLOCK	000F6010	5
PEM --- COBOL II/COBOL 370 CONTROL BLOCK	00069630	7
PEM --- REGISTER SAVEAREA CHAIN		8
PEM --- PROGRAM DEFINITION INDEX	0B802808	8
PEM --- PROGRAM DEFINITION TABLE	0B804008	9
PEM --- ALLOCATED DATA GROUP INDEX	0004CC98	9
PEM --- PERMANENT DATA GROUP INDEX - PRIVATE	000841A8	10
USER TRANSACTION CONTROL BLOCK	0008358	10
USER TRANSACTION CONTROL BLOCK EXTENSION	000895E0	10
UPCB PROGRAM ID=Z999413 ENTRYPOINT=8B850468 COMPILED=02/06/98 13.25.44	0005B250	10
DATA GROUP ID=0000000009	0004D0B0	10
DATA GROUP ID=0000000074	000C53C8	11
DATA GROUP ID=00000001031	000C51B0	11
DATA GROUP ID=00000001452	0005B8B8	11
DATA GROUP ID=00000002000	0005B7F0	11
DATA GROUP ID=00000002001	000CC868	11
DATA GROUP ID=00000002005	000CC3A8	12
DATA GROUP ID=00000002006	000CC838	12
DATA GROUP ID=00000002015	000CC3E0	12
DATA GROUP ID=00000002301	000C8898	12
DATA GROUP ID=00000002304	000C20F0	12
DATA GROUP ID=00000002305	000C89B0	14
DATA GROUP ID=00000002306	000C89F0	14
DATA GROUP ID=00000002307	000C99E8	15
DATA GROUP ID=00000002310	000CB120	15
DATA GROUP ID=00000002350	000CA070	16
DATA GROUP ID=00000002351	000C9BA0	16
DATA GROUP ID=00000003475 (PERM - PVT).	000D5730	16
DATA GROUP ID=00000003601	000C5870	17
DATA GROUP ID=00000005610	000CC138	17
DATA GROUP ID=0000013589	0005BCF8	17
DATA GROUP ID=0000047100	0005B348	17
DATA GROUP ID=0000047110	0005B3A0	17
DATA GROUP ID=0000047120	0005B490	18
DATA GROUP ID=0000047130	0005B550	18
DATA GROUP ID=0000047140	0005B6A0	18
DATA GROUP ID=0000047190	0005B968	18
DATA GROUP ID=0000047192	0005B9C0	18
DATA GROUP ID=0000048000	0005A998	18
DATA GROUP ID=0000048005	0004D0F8	19
DATA GROUP ID=0000048006	0004ED08	21
DATA GROUP ID=0000048007	0004F110	21
DATA GROUP ID=0000048009	0004F1E8	21
DATA GROUP ID=0000048035	00051908	21
DATA GROUP ID=0000048066	0005B0E0	21
DATA GROUP ID=0000048091 (PERM - PVT).	0005C0B8	22
DATA GROUP ID=0000048092	00051A10	22
DATA GROUP ID=0000048310	00053FE0	22
DATA GROUP ID=0000048312	00054FA8	23

P E M FORMATTED STORAGE DUMP E X C E P T I O N DATE=98/02/23 TIME=10:03:42 PAGE 0028



Umbrella Programming

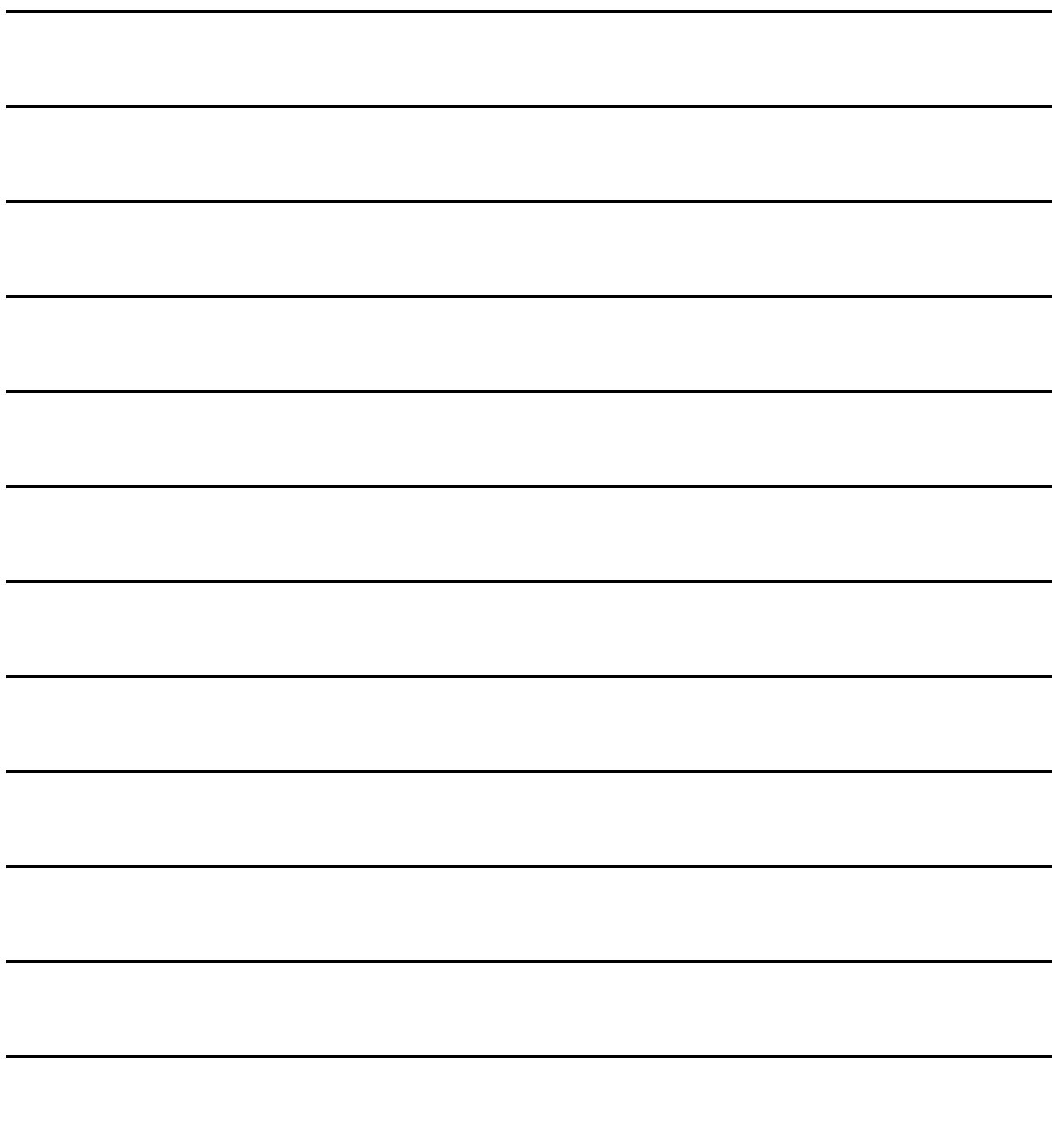
System Memory Access Retrieval Tool

***** END OF PEM FORMATTED STORAGE DUMP *****

***** END OF PEM JOB *****

.GS SCAN ON

Notes:



PEM Formatted Dump Exercises



You are being supplied with a manual that contains four example PEM formatted dumps. The purpose of these exercises is for you to practice debugging some PEM dumps. The examples are designed to present you with a variety of causes for abends.

You need to independently research each exercise and attempt to discover the cause of the problem.

The instructor will discuss each exercise along with any questions you encounter in your research.

Notes:



Summary

CSC

Summary



- Many dumps debugged by reading header page
- PEM messages errors stored on format 1301
- PEM message retrieved on SMART header screen
- PEM provides HEX data
- Debugging a PEM formatted dump
 - PEM Exception Report
 - Formatted Dump Summary
 - Trace Table
 - User PCB
 - Data Group Fields

© 2000 CSC. All rights reserved

7/10/00

CSC

Summary



- Keywords used to change dump options
- Numerous SMART commands and addresses for aiding in debugging
- Data available through error messages
- Practice in debugging is best

© 2000 CSC. All rights reserved

7/10/00



Umbrella Programming

Summary

- Many dumps within the Hogan System can be debugged by reading the header page. There are a few problems that occur frequently.
- The messages for PEM errors are stored on CDMF format 1301. The assigned numbers for PEM are the 49000 range.
- In CICS, the PEM message is retrieved on the SMART header screen.
- PEM provides HEX data in a readable and organized format.
- The suggested steps for debugging a PEM formatted dump are:
 1. PEM Exception Report.
 2. Formatted Dump Summary.
 3. Trace Table.
 4. User PCB.
 5. Data Group Fields.
- Dump option keywords are used to change dump options.
- In CICS environments, SMART provides a look at storage for solving online dumps. There are numerous SMART commands and symbolic addresses for aiding in debugging.
- Valuable data is available and easy to access in PEM formatted dumps through PEM error messages, the formatted TCB, trace table, UPCB, and data group storage.
- Practice in debugging PEM formatted dumps is the best means for learning how to research abends.

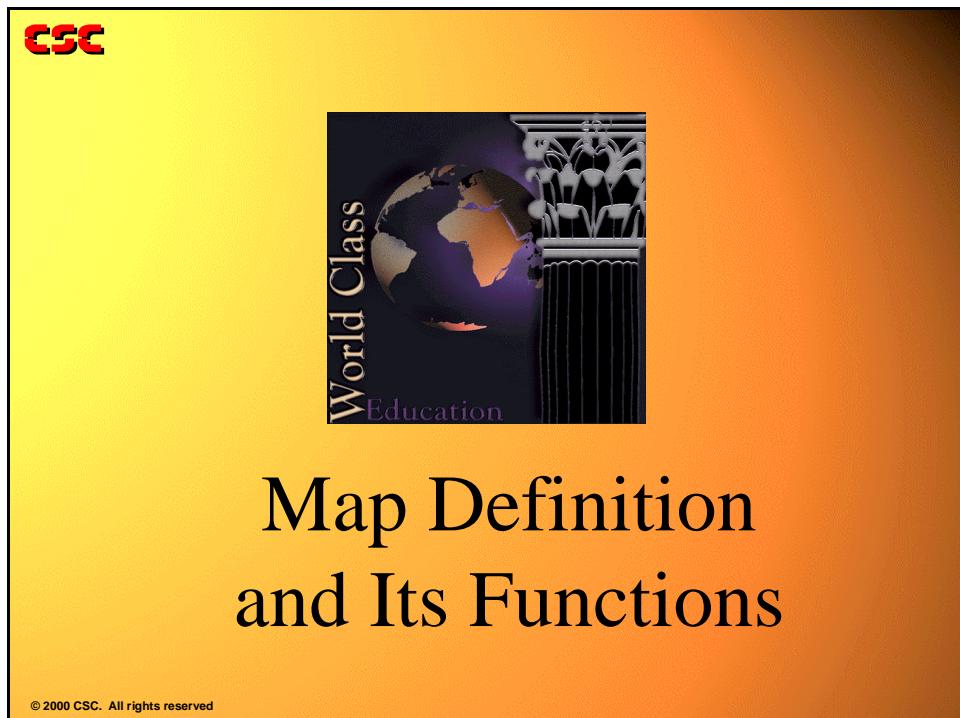
Notes:



Mapping Support Facilities

17

Purpose



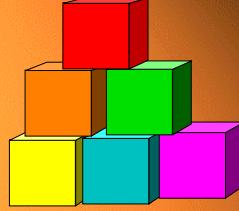
Notes:



Topics

CSC

Topics



- Work Area Activity
- Data Communications Activity
- Mapping maintenance
- Online transaction requirements
- Map assembly

© 2000 CSC. All rights reserved

Objectives

CSC

Objectives



Learn to:

- Create a Work Area activity
- Create Data Communications activities
- Create a Map Definition
- Run the map documentation/assembly process

© 2000 CSC. All rights reserved



Introduction

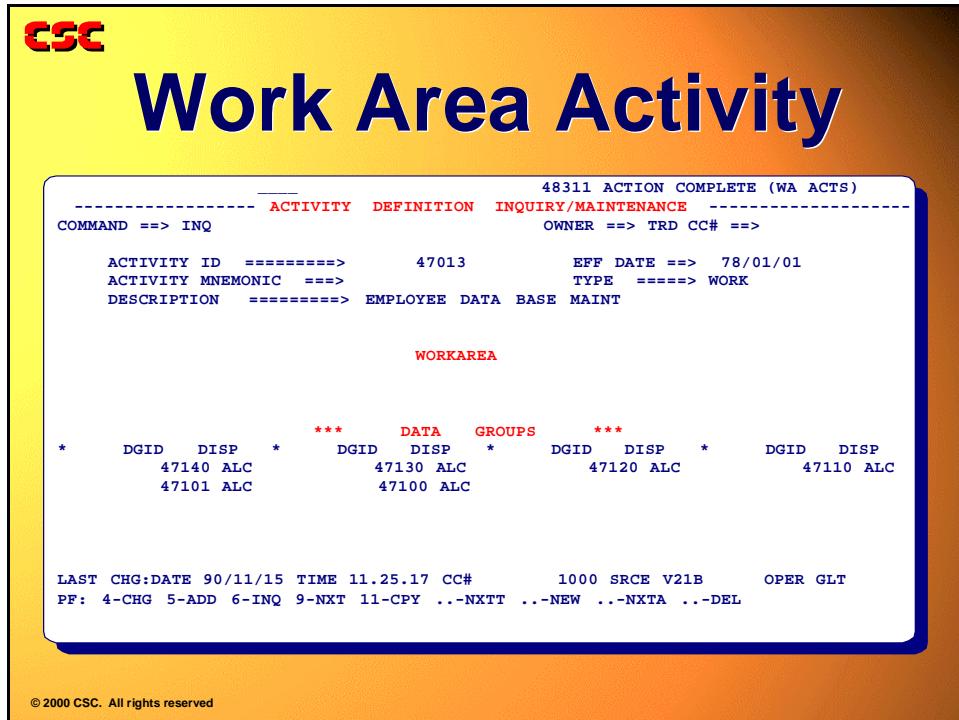
Maps are definitions of screen images that are stored on CDMF, and used by online to format, present, and receive real time data.

There are several additional co-requisite Process Dictionary items required to correctly process a map. Let's begin by examining work area results.

Work Area Activity Definition

Work Area Activities allow the application to allocate, initialize, and release storage areas for data groups. All data groups must have a definition in the Process Dictionary. PEM uses this definition to allocate space for the data group and to initialize the space if so directed. PEM will allocate all the data groups listed in a Work Area Activity in a single area of storage.

A Work Area Activity is required prior to a Data Communications Activity with a service of DBLK (deblock) in order to dynamically establish working storage for the map. An abend will occur if a Work Area Activity has not allocated the data group before the deblocking occurs.



The screenshot shows a CSC software interface with a yellow header bar containing the CSC logo and the title "Work Area Activity". Below the header is a command-line interface window with the following text:

```
48311 ACTION COMPLETE (WA ACTS)
----- ACTIVITY DEFINITION INQUIRY/MAINTENANCE -----
COMMAND ==> INQ OWNER ==> TRD CC# ==>
ACTIVITY ID =====> 47013 EFF DATE ==> 78/01/01
ACTIVITY MNEMONIC ==> TYPE =====> WORK
DESCRIPTION =====> EMPLOYEE DATA BASE MAINT

WORKAREA

*      DGID      DISP    *      ***      DATA      GROUPS      ***
*      DGID      DISP    *      DGID      DISP    *      DGID      DISP
47140 ALC          47130 ALC          47120 ALC          47110 ALC
47101 ALC          47100 ALC

LAST CHG:DATE 90/11/15 TIME 11.25.17 CC#      1000 SRCE V21B      OPER GLT
PF: 4-CHG 5-ADD 6-INQ 9-NXT 11-CPY ..-NXTT ..-NEW ..-NXTA ..-DEL
```

At the bottom left of the interface window, there is a copyright notice: "© 2000 CSC. All rights reserved".



Umbrella Programming

Introduction



Work Allocate Disposition Values

- ALC – If not already allocated, Allocate and Initialize per DG Definition.
- ANO – If not already allocated, Allocate without Initializing.
- REL – Release storage
- INT – Force Re-Initialization

© 2001 CSC. All rights reserved

Notes:



Data Communications Activity

Data Communications Activities provide the application program access to the teleprocessing network and the batch data communications simulator. Two map related services may be performed by Data Communications Activities. Deblock (DBLK) puts data into Data Groups. Display (DISP) displays the contents of Data Groups.

Notes:



Umbrella Programming

Data Communications Activity

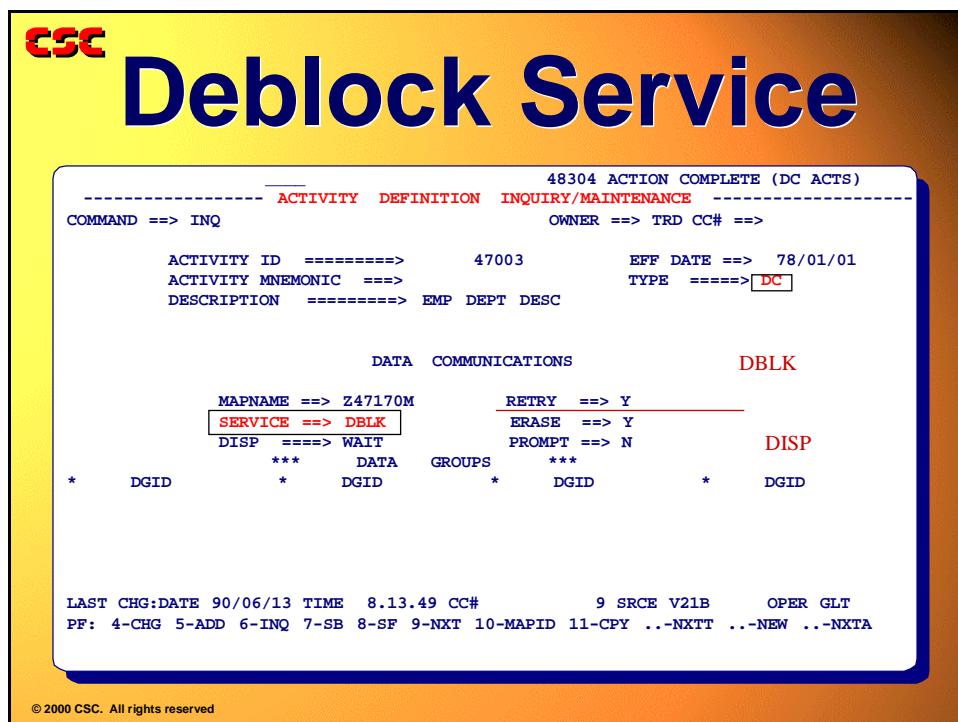
Deblock Service

The DEBLOCK service causes PEM to update selected fields in the Data Groups based on input from the screen (if the transaction is online) or SYSIN (if the transaction is batch). During the deblock operation the following edits are performed against the input data:

1. Required field not entered.
2. Non-numeric data in a numeric field, excluding edit mask characters.
3. Numeric data exceeds the field size, that is, overflow.

When the retry option is requested on the DC Activity Definition, PEM displays ? in all the fields found in error and ends the transaction. If retry is not requested, the data is passed as entered to the application program.

Multiple deblock operations may be performed against the same data. This is typically done when key information is required for a Data Base access prior to deblocking the data that may have been changed. When multiple deblock operations are performed, the maps used must describe the fields in the input starting from the first field, that is, the first map may describe the first 10 fields. The second map may describe the first 30 fields. The third may describe the first 50 fields. Each map must begin with field one and continue through field n, where n is the last field you wish to deblock.



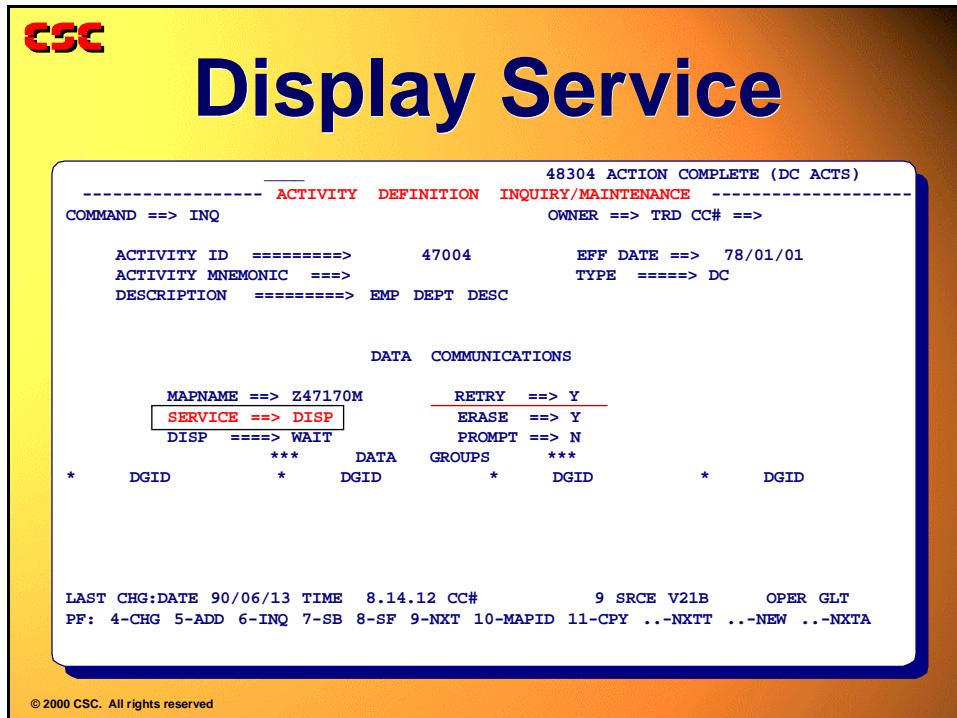
Display Service

The DISPLAY service is the reverse of the deblock service. It causes PEM to display selected fields from the data groups to the screen. The fields that are displayed are indicated in the Map Definition.

Display has an optional erase feature. It causes the screen to be erased prior to displaying the data. When erase=no is chosen, the application program should ensure that the map to be displayed is compatible with the current format on a screen otherwise, when the new map is displayed, it could contain characters from the old map. This garbage data can prevent the proper execution of the current application.

A prompt is the display of only the literals on a screen. It results when prompt=yes is set.

The application program can route output to a destination other than the terminal that originated the transaction. This is done by placing the Terminal ID in the TCB-DESTINATION field, and the device type code in the TCB-DEVICE-TYPE field prior to the activity request.



Umbrella Programming

Map Definition Maintenance

Map Definition Maintenance

This section will outline the procedure for defining and maintaining maps. After an overview of the required steps, it will walk through each basic step and define its requirements in more detail.

The following steps are required to define a map to the Umbrella System:

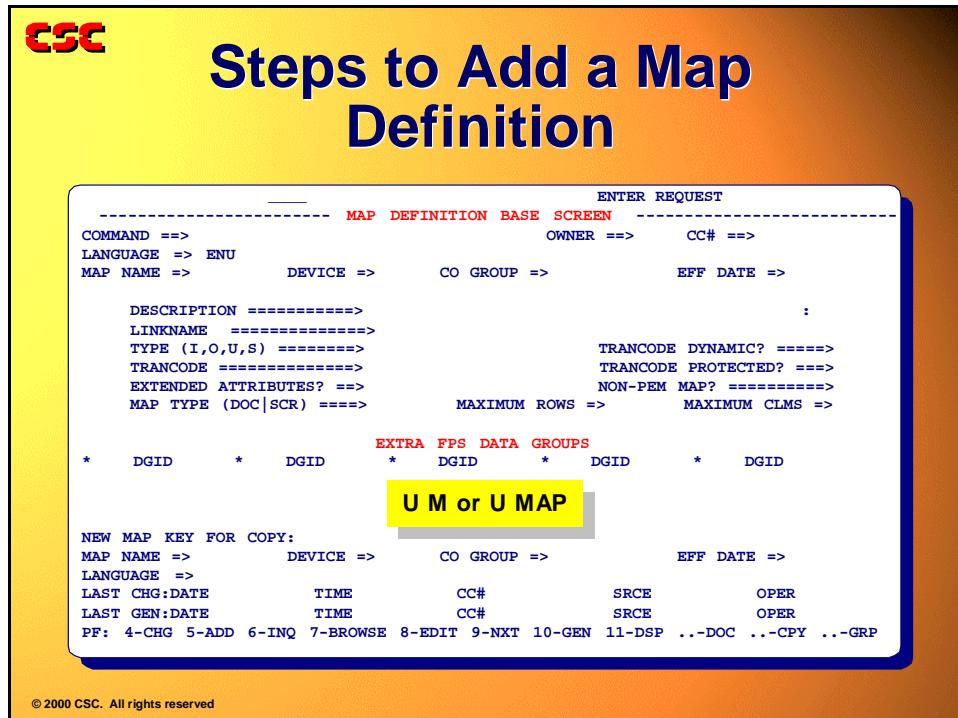
1. CICS users: Add the map name in the CICS PPT or RDO MAP entry.
2. Define the base information necessary to identify the map, including language, mapname, device, title, and linkname. This is done on the "Map Definition Base Screen".
3. Draw a layout of the map on the screen, identifying literals and modifiable fields. This is done on the design screen ("Map Definition Screen"). To display the design screen, press PF8 or enter EDIT into the COMMAND field.
4. Describe the characteristics of the fields identified in the previous step. This includes assigning attributes to all fields, identifying the data elements for the modifiable fields and assigning the edit mask. This is done on the map field definition maintenance screen. This screen is displayed by pressing PF9 or entering FDM into the COMMAND field.
5. If the map is to display repetitive fields, define the fields as groups on the "Map Definition Group Screen". This screen is displayed by entering GRP into the COMMAND field.
6. If the map is to display extended attributes or make use of cursor positioning by the application program, define these on the supplemental "Map Field Definition Maintenance" screen. This screen is displayed by pressing PF6 or entering FDM2 into the COMMAND field.
7. Save the map definition.
8. Generate the map from the base screen by pressing PF10 or entering GEN into the COMMAND field.
9. Run the CICS/IMS map assembly.
10. CICS users: Newcopy the map using the map link name.



Steps to Add a Map Definition

Start with the "Map Definition Base Screen".

Enter U M or U MAP.



Notes:

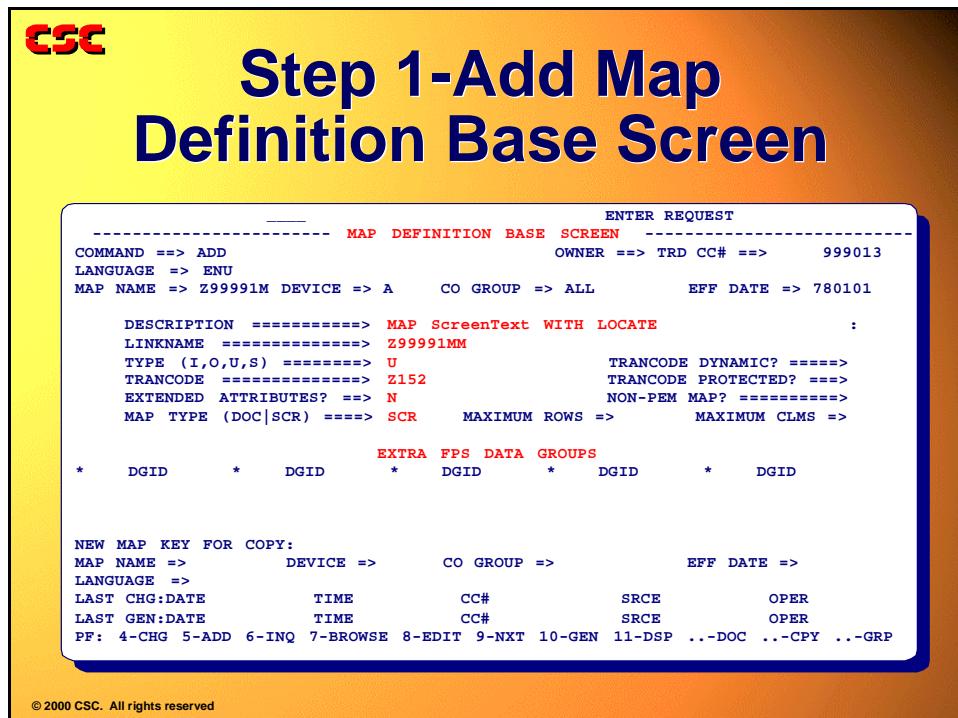


Umbrella Programming

Steps to Add a Map Definition

Step 1—Add Map Definition Base Screen

Enter the base information for the map. Enter the command ADD or press PF5. Because this is an update access, the owner and change control number must be entered.

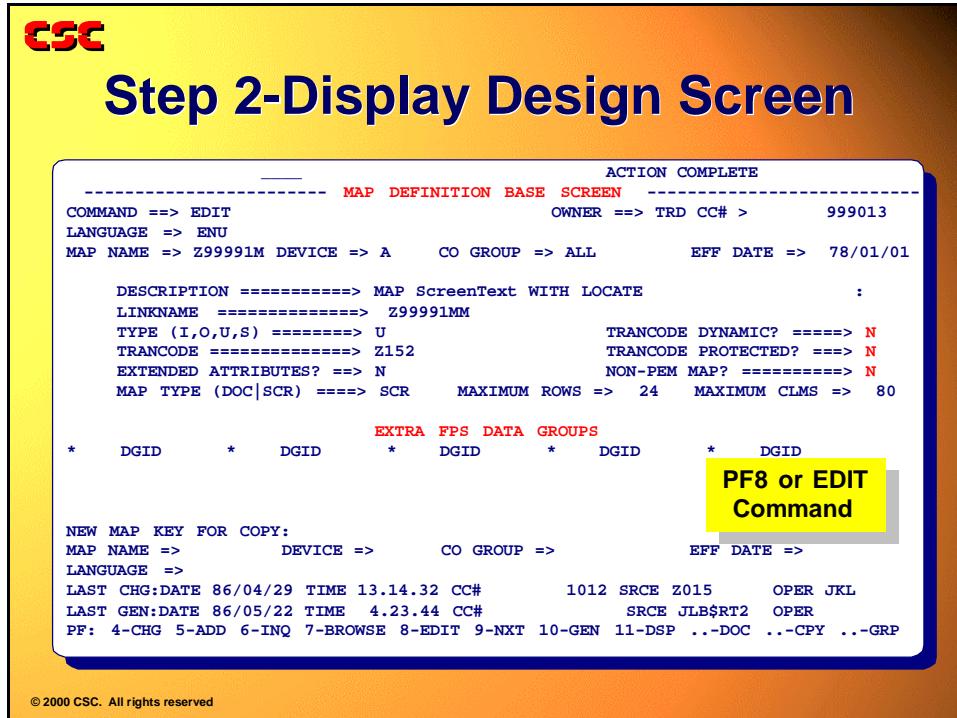


Notes:



Step 2—Displaying the Design Screen

To display the design screen, press PF8 or enter EDIT into the COMMAND field.



Notes:



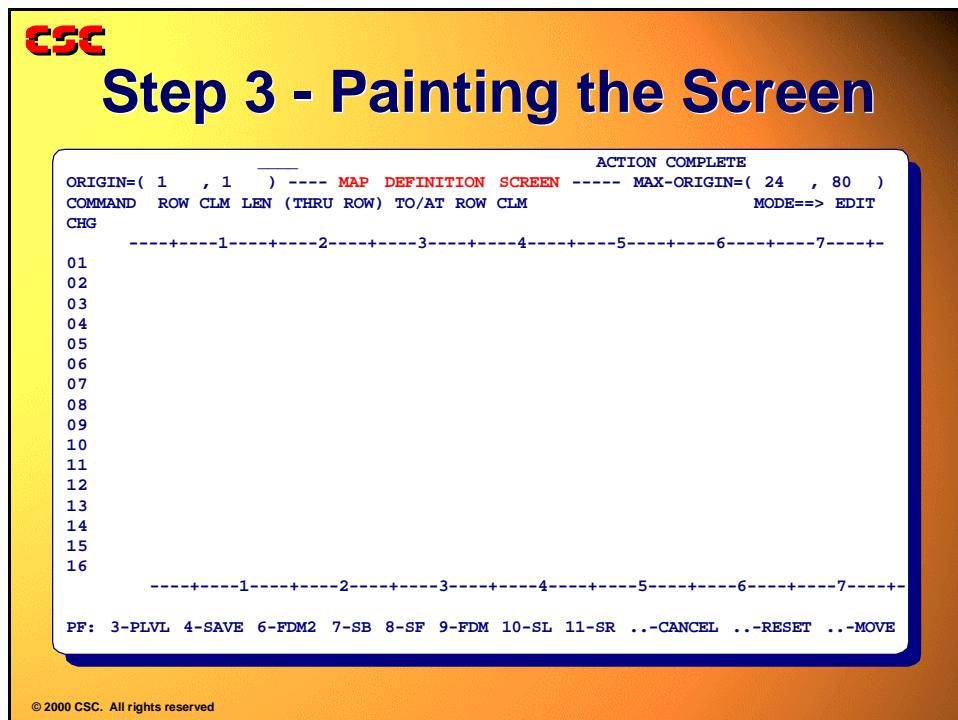
Umbrella Programming

Steps to Add a Map Definition

Step 3—Painting the Screen

To paint the screen:

- Enter the literals and modifiable fields.
- Indicate the alphanumeric modifiable fields with Xs and the numeric modifiable fields with their edit pattern.



Notes:



CSC

Painting the Screen

```
ACTION COMPLETE
ORIGIN=( 1 , 1 ) ----- MAP DEFINITION SCREEN ----- MAX-ORIGIN=( 24 , 80)
COMMAND ROW CLM LEN (THRU ROW) TO/AT ROW CLM MODE==> EDIT
CHG
-----+---1---+---2---+---3---+---4---+---5---+---6---+---7---+-
01
02
03           "MAPPING SAMPLE EXAMPLE"
04
05
06     "EMPLOYEE COMPANY ID"      ZZZS9      "EMPLOYEE ID"      ZZZZZZZZS9
07
08           "EMPLOYEE NAME"
09
10
11           xxxxxxxxxxxxxxxx      xxxxxxxxxxxxxxxx
12
13
14
15
16
-----+---1---+---2---+---3---+---4---+---5---+---6---+---7---+-
PF: 3-PLVL 4-SAVE 6-FDM2 7-SB 8-SF 9-FDM 10-SL 11-SR ..-CANCEL ..-RESET ..-MOVE
```

© 2000 CSC. All rights reserved

Notes:



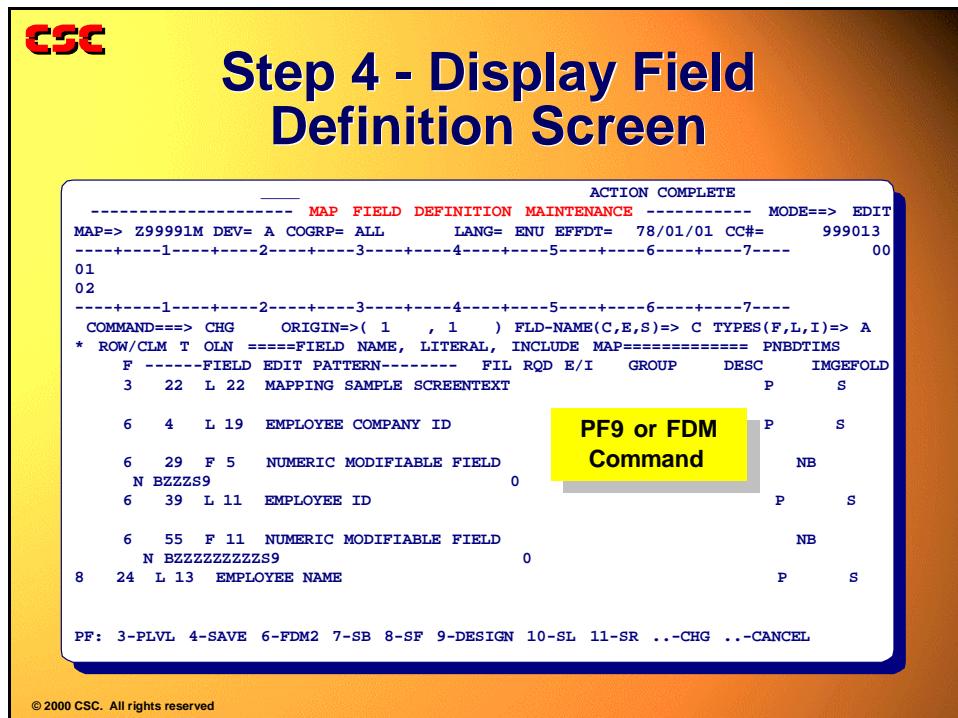
Umbrella Programming

Steps to Add a Map Definition

Step 4—Display the Field Definition Screen

To display the field definition screen, press PF9 or enter FDM in the COMMAND field.

To identify the data fields, change the modifiable fields to actual data group fields.



The screen can be divided into four areas, as follows.

- The key area: Row 3
- The panel image area: Row 4 through Row 8
- The command area: Row 9
- The field definition area: Row 10 through Row 23.

Three rows of the map being defined appear in the panel image area in a format like that of the "Map Definition Screen" design area. The first field in the field definition area is the first field in the second line of the panel image area.

The Command Area Fields:

ORIGIN

This is the row, column of the field definition that is displayed in the first row of the field definition area. This field may be used to position the displays.



FLD-NAME (C,E,S)

This indicates the format type in which the field name is displayed. This field defaults to C.

Valid values are:

- C COBOL name
- E Element name
- S Self-defining name.

TYPES (F,L,I)

This defines the types of fields that are to be displayed in the field definition area of the display. Valid values are:

- A All fields and includes
- F Modifiable fields only
- I Includes only
- L Literal or text fields only.

The Field Definition Area of the display consists of six sets of recurring fields, one per field or include. Each field definition requires two rows on the display. The top row of column headings references the first row of each field definition; the bottom row references the second.

Enter or verify values in the following fields:

* (action)

The action to be performed on this field definition. Default is no change. Valid values are:

- A Add a new field definition.
- C Change the field definition.
- D Delete the field definition.
- L Change the field definition and exit to the data group Data Element Locate Facility.

ROW/CLM

This is the starting row/column of the field being defined.

If this is an include map field (type I), the row specified is the first row in which the included map is displayed.



Umbrella Programming

Steps to Add a Map Definition

T

This indicates which type of field is being defined. The following are valid types:

- F Modifiable field
- I Include map
- L Literal or text field.

OLN

The output or display length for this field. For numeric modifiable fields, this value may be less than the length of the edit pattern, so that only the rightmost digits may be displayed.

FIELD NAME, LITERAL, INCLUDE MAP

For a TYPE = F field, this is the ASSEMBLER, COBOL, or Data Dictionary identifier of the data element to be displayed in this field.

For a TYPE = L field, this is the text or literal.

For a TYPE = I field, this is the name of the include map.

If a modifiable field name is changed or added, the map maintenance program attempts to determine the field's location. If data group definition does not know where that field is, a "Data Group Element Locate Screen" appears with an error message: NO MATCHING ELEMENTS.

Before generation of the map, these fields must be located so the data group and element identifier can be resolved.

PNBDTIMS

This field defines the normal attributes for modifiable or literal fields. An attribute is selected by entering a nonblank character beneath the letter representing that attribute.

The letters PNBDTIMS represent the following attributes:

- P (Protected) The terminal operator is not allowed to enter data in this field.
- N (Numeric) The terminal operator may enter only numeric data: 0 through 9, -(minus),
- B (Bright) Data is displayed at high intensity.
- D (Dark) Data is invisible. (Dark and bright are mutually exclusive.)
- T (Pen Detectable) This field is light pen detectable.
- I (Insert Cursor) Put the cursor in this field location. If multiple fields are



coded with this attribute, the cursor is placed in the last one.

- M (Modified Data Tag On) The modified data tag (MDT) is on for this field. This causes the value contained in the field to be transmitted as though the operator altered the field when ENTER is pressed.
- S (Skip) Causes the cursor to skip over this field automatically.

F (For TYPE=F entries only.) Enter the format of the field:

- A Alpha, left justified
- C Character, left justified
- L Numeric, left justified
- N Numeric, right justified
- Q Signed numeric, left justified
- R Character, right justified
- S Signed numeric, right justified
- Blank Use format specified on data group definition.

FIELD EDIT PATTERN

An Edit Pattern must be entered for numeric (packed or binary data) fields. Numeric data maintained in display (zoned decimal) format is displayed as character data, and no editing is done.

FIL

Define the fill character with which the TP monitor is to fill the field if it has been only partially entered. For CICS, the only valid values are a blank and zero. If other than these values are entered here, the Map Assembly program for CICS substitutes blank or zero depending on the format of the field.

RQD

Y indicates that this field is always required to be entered.

CAUTION: Any nonblank character entered defaults to Y. A blank indicates not required.

E/I

Error/Ignore. When the user program requests an error display activity, the Umbrella System Mapping System fills each field that contains high values (X'FF') with the Device Suffix Error Code (defined in the Device Characteristics Table Entry). If high values are valid for this field, override the error code fill indicator by entering Y.



Umbrella Programming

Steps to Add a Map Definition

CAUTION: Any nonblank character entered defaults to Y. A blank indicates high values are not valid.

GROUP

This field connects the field being defined with a repetitive field group.

DESC

The 1- to 8-character field description for modifiable fields. The description is used for informational purposes only.

IMGEFOLD

The Umbrella System Image Application Folder name. See "Folder Definition Inquiry/Maintenance Base".

Numeric Field Edit Patterns

An Edit Pattern must be entered for numeric (packed or binary data) fields. Numeric data maintained in display (zoned decimal) format is displayed as character data, and no editing is performed.

Edit patterns for maps are taken from the following places:

- Dynamically modifiable edit pattern (described later)
- Map field definition.

There are two types of edit patterns:

- Standard edit patterns
- User coded edit patterns.

The following are standard edit patterns:

NONE

There is no editing.

\$ An edit pattern is automatically generated at map GEN time to edit a monetary element. Note that PEM is designed to "display blank when zero" when processing "\$" as an edit mask.

\$\$ An edit pattern is automatically generated at map GEN time to edit a monetary element. Note this causes PEM to display significant digits for monetary edit masks.

The following are characteristics of this edit pattern:

- Generated for the specific element size



- Has two decimal places
- Significance is forced at the decimal point
- Has a trailing dash if negative (ZZS.99-).

/ An edit pattern is automatically generated at map GEN time to edit a number (counter) element. The following are characteristics of this edit pattern:

- Generated for the specific element size
- Significance is forced before the last digit
- Negative numbers are not indicated.

.nn Indicates a date edit pattern. The editing logic flips the standard packed date format of CYYMMDD to the appropriate format, if necessary. The .nn can be:

- .00 The site default.
- .01 MM-DD-YY
- .02 MM/DD/YY
- .03 MM DD YY
- .04 MMDDYY
- .11 CYY-MM-DD
- .12 CYY/MM/DD
- .13 CYY MM DD
- .14 CYYMMDD
- .21 DD-MM-YY
- .22 DD/MM/YY
- .23 DD MM YY
- .24 DDMMYY.
- .50 Causes this field to be edited based upon the settings on the site control "System Site Editing Models" screen. Editing is done by PEM Generalized Editing facility, which needs to be active within PEM if this option is chosen.

Note: Note: SPS has additional options. See "SPS Format Definition" in the chapter "Scheduled Processing System (SPS)" of the Umbrella System Support Systems Programming Guide.

BIN-FW

This field is transmitted as a signed binary length 4 fullword. This is a special edit type for some devices that are processed by terminal input/output exits. The output length must be equal to 4.

Self-Defining Edit Mask.

The is a user-coded edit pattern, and must conform to the following rules:

11. The first character is a fill character. If not present, one is added.



Umbrella Programming

Steps to Add a Map Definition

12. The edit pattern must contain an odd number of replacement digit positions. If not, Z is added at the left.
13. The replacement digits (Z, 9, S) in the edit pattern are aligned right to left with the digits in the numeric field being displayed.
14. Syntax characters:
 - B = Blank
 - S = Significant-start digit position (after this point, display leading zeroes)
 - Z = Zero-suppress digit position (do not display leading zeroes)
 - 9 = Nonsuppress digit position (display all values).
 - Special Symbols S, *, /, %, -, ., , (comma), 0, DR, and CR..

Example:

BZZZ,ZZZ,ZS9.99- 12345678 displays as 123,456.78

12 displays as 0.12

-1234 displays as 12.34-

BS9999 12345678 displays as 45678

12 displays as 00012

-1234 displays as 01234

Notes:



Step 5—Identify the Modifiable Fields

Using the Locate Command

CSC

Step 5 - Identify Modifiable Fields Using the Locate Command

```
----- MAP FIELD DEFINITION MAINTENANCE ----- MODE=> EDIT
MAP=> Z99991M DEV= A COGRP= ALL LANG= ENU EFFDT= 78/01/01 CC#= 999013
      -3-----4-----5-----6-----7-----+---8
05
06 ZZZS9 "EMPLOYEE ID" ZZZZZZZZS9
07
      -3-----4-----5-----6-----7-----+---8
COMMAND==> CHG ORIGIN=>( 6 , 29 ) FLD-NAME(C,E,S)=> C TYPES(F,L,I)=> F
* ROW/CLM T OLN =====FIELD NAME, LITERAL, INCLUDE MAP===== PNBDTIMS
      F -----FIELD EDIT PATTERN----- FIL RQD E/I GROUP DESC IMGEFOLD
      3 22 L 22 MAPPING SAMPLE ScreenText          P   S
      6 4 L 19 EMPLOYEE COMPANY ID          P   S
      L 6 29 F 5 47100          NB
      N BZZZS9          0
      6 39 L 11 EMPLOYEE ID          P   S
      6 55 F 11 NUMERIC MODIFIABLE FIELD          NB
      N BZZZZZZZS9          0
      8 24 L 13 EMPLOYEE NAME          P   S
PF: 3-PLVL 4-SAVE 6-FDM2 7-SB 8-SF 9-DESIGN 10-SL 11-SR ...CHG ...CANCEL

© 2000 CSC. All rights reserved
```

Notes:



Umbrella Programming

Steps to Add a Map Definition

Data Group Element Locate Screen

CSC

Data Group Element Locate Screen

48999 SELECT THE CORRECT ELEMENT
----- DATA GROUP ELEMENT LOCATE SCREEN -----
ENTER COMMAND ==> CHG NEXT SEQ # ==> 1
ELEMENT ALIAS ==> 47100
SELECT CRITERIA: DATA GROUP ==> 47100 EFFECTIVE DATE ==> 78/01/01
LANGUAGE ==> (A-ALC,C-COBOL,E-ELEMENT,S-SELF-DEFINING)

*	SEQ	DATA.GROUP	EFF.DAT	EID	LANG	ELEMENT.ALIAS.NAME	LNGTH	DISP	TYP
	1	47100	780101	276	S	EMP-KEY-GROUP	E001F020	50	C
	2	47100	780101	280	S	EMP-KEY-GROUP-MOVE	E001F024	46	4 C
S	3	47100	780101	277	S	EMP-CO-ID	E001F021	2	4 C
	4	47100	780101	278	S	EMP-KEY-ID	E001F022	6	6 P
	5	47100	780101	279	S	EMP-FILLER	E001F023	38	12 C

A NON-BLANK IN THE "*" FIELD AND BLANKS IN THE COMMAND FIELD RESULTS IN SELECTING THE FIELD AND RETURNING TO THE PRIOR SCREEN.

PF:

© 2000 CSC. All rights reserved

Using the Element ID

CSC

Step 5 - Identify Modifiable Fields Using the Element ID

ACTION COMPLETE
----- MAP FIELD DEFINITION MAINTENANCE ----- MODE==> EDIT
MAP=> Z99991M DEV= A COGRP= ALL LANG= ENU EFFDT= 78/01/01 CC#= 999013
-4-----5-----6-----7-----8-----9
05
06 "EMPLOYEE ID" ZZZZZZZZS9
07
-4-----5-----6-----7-----8-----9
COMMAND==> CHG ORIGIN=>(6 , 39) FLD-NAME(C,E,S)=> E TYPES(F,L,I)=> A
* ROW/CLM T OLN =====FIELD NAME, LITERAL, INCLUDE MAP===== PNBDTIMS
F -----FIELD EDIT PATTERN----- FIL RQD E/I GROUP DESC IMGEFOLD
6 39 L 11 EMPLOYEE ID P S
C 6 55 F 11 TQE47002 NB
N BZZZZZZZS9 0 P S
8 24 L 13 EMPLOYEE NAME
10 14 F 15 ALPHANUMERIC MODIFIABLE FIELD B C
10 32 F 15 ALPHANUMERIC MODIFIABLE FIELD B C

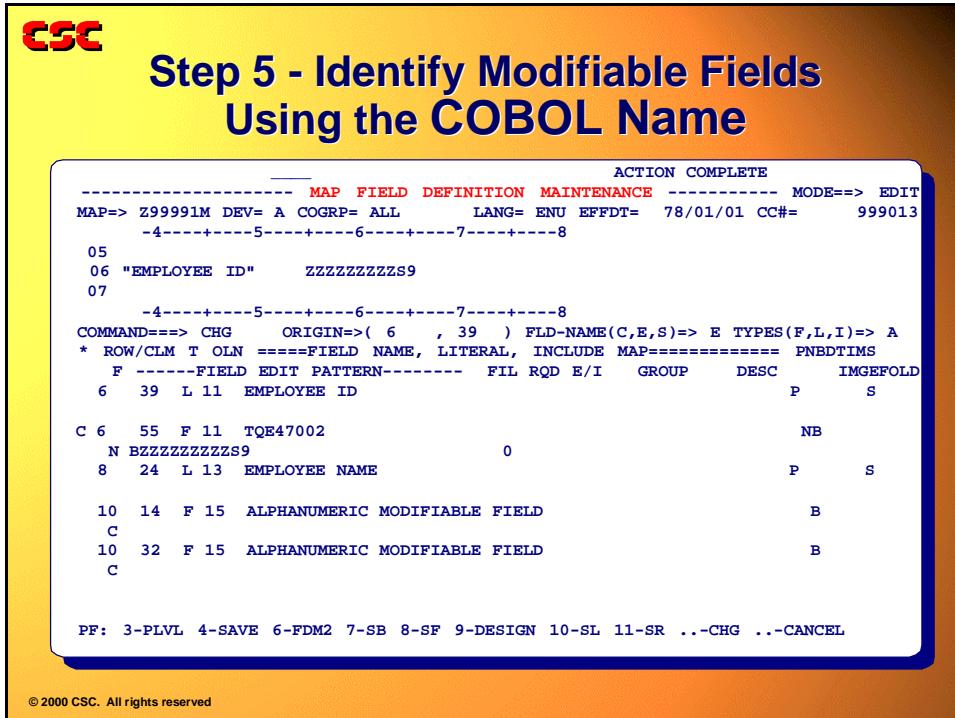
PF: 3-PLVL 4-SAVE 6-FDM2 7-SB 8-SF 9-DESIGN 10-SL 11-SR ...CHG ...CANCEL

© 2000 CSC. All rights reserved



Using the COBOL Name

Note: Not available on converted data groups.



Notes:



Umbrella Programming

Steps to Add a Map Definition

Step 6—Assign Attributes to Data Fields

ACTION COMPLETE

MAP FIELD DEFINITION MAINTENANCE MODE==> EDIT
MAP=> Z99991M DEV= A COGRP= ALL LANG= ENU EFFDT= 78/01/01 CC#= 999013
-----+-----+-----+-----+-----+-----+-----+
05
06 "EMPLOYEE COMPANY ID" ZZZS9 "EMPLOYEE ID" ZZZZZZZZS9
07
-----+-----+-----+-----+-----+-----+-----+
COMMAND==> CHG ORIGIN=>(6 , 4) FLD-NAME(C,E,S)=> C TYPES(F,L,I)=> A
* ROW/CLM T OLN =====FIELD NAME, LITERAL, INCLUDE MAP===== PNBDTIMS
F -----FIELD EDIT PATTERN----- FIL RQD E/I GROUP DESC INGEFOLD
6 4 L 19 EMPLOYEE COMPANY ID P S

C 6 29 F 5 EMP-CO-ID NB IM
N BZZZS9 0 P S
6 39 L 11 EMPLOYEE ID

C 6 55 F 11 EMP-KEY-ID NB M
N BZZZZZZZS9 0 P S
8 24 L 13 EMPLOYEE NAME

10 14 F 15 EMP-F-NAME B
C
PF: 3-PLVL 4-SAVE 6-FDM2 7-SB 8-SF 9-DESIGN 10-SL 11-SR ..-CHG ..-CANCEL

© 2000 CSC. All rights reserved

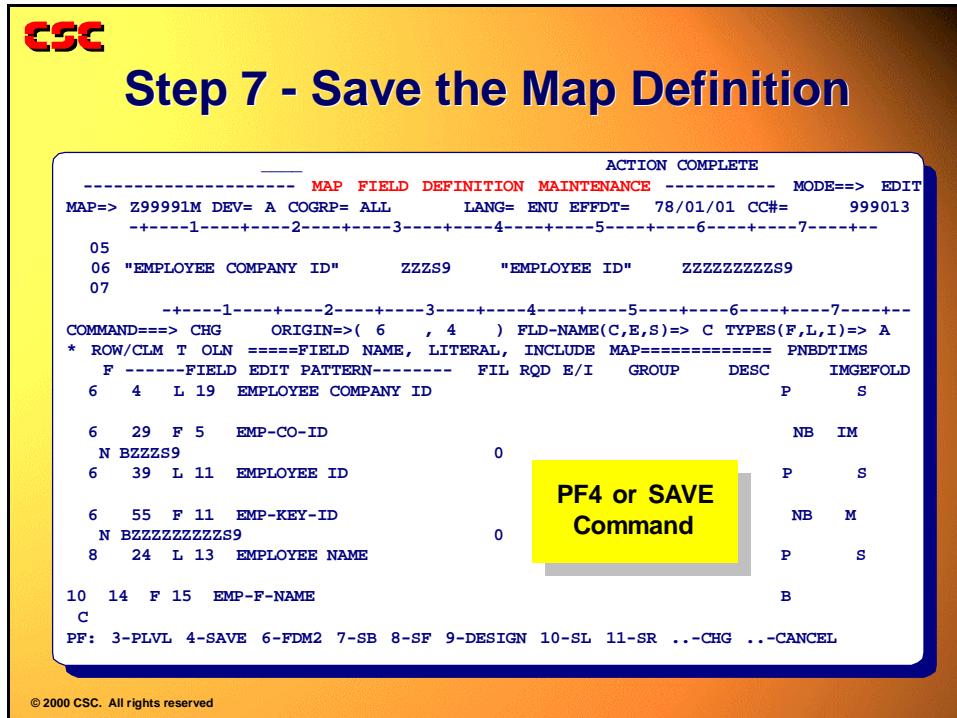
The available attributes are:

- | | |
|----------|--|
| P | Field is protected |
| N | Field is numeric |
| B | Field is bright |
| D | Field is dark |
| T | Field is pen detectable |
| I | Insert cursor here |
| M | Modified data tag. The modified data tag causes the value in the field to be transmitted to core even if it has not changed. This attribute must be turned on if you plan to be able to execute a sequential read from your map. |
| S | Field is to be skipped. |



Step 7—Save the Map Definition

To save the screen, press PF4 or enter SAVE in the COMMAND field of the "Map Field Definition Maintenance" screen.



Notes:

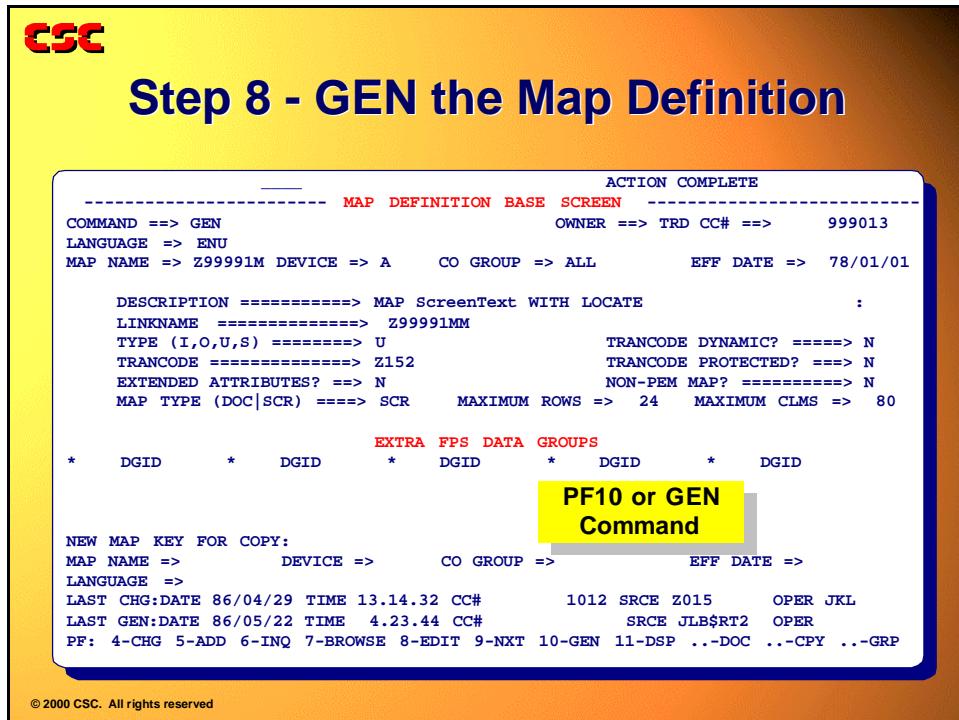


Umbrella Programming

Steps to Add a Map Definition

Step 8—GEN the Map Definition

Generate the map by pressing PF10 or entering GEN in the COMMAND field.



Notes:



Step 9 -Assemble the Map

Batch Map Documenter

The first transaction (1 49 315) generates a batch map documentation report. Two formats of the report are available.

Full format gives the screen image showing the attribute bytes and all variable field locations. Also listed are the field definitions for the map. Diagnostics are produced for error conditions such as overlapping fields, or nested includes.

Screen format gives the screen image only. Attribute bytes and protected fields are not underscored.

This transaction also produces a report of all daughter jobs submitted to the internal reader, as well as a list of the maps contained in the daughter jobs.

```
//SYSIN      DD   *
 1 49 315
#318@@FROMMAP@D@EFFDATE@THRUMAP@_@@OPTION@COGROUP@ISOLANG@%
```

Where:

FROMMAP	1 to 7-character starting map name. If the FROMMAP and THRUMAP names are not the same, (that is, a range of maps is being documented), all maps with a linkname of NONE will be skipped.)
D	A 1-character device code (normally A).
EFFDATE	Effective date. CYYMMDD format (optional).
THRUMAP	1 to 7-character ending map name.
-	The fill character to be placed in the variable field locations. If the printer does not have an underscore (_) character, a period (.) is suggested.
OPTION	FULL or SCREEN.
COGROUP	1 to 8-character company group name. Default is every company group.
ISOLANG	3-character ISO Language code. Default is ENU. This parameter must be specified if other than ENU is desired.



Umbrella Programming

Steps to Add a Map Definition

Batch Map ASSEMBLER

The batch map ASSEMBLER transaction (1 49 303) produces map source statements for the TP environment. Output of the job is written to the MAPSOUT file that is normally the internal reader. The name of the JCL member we are using for the map assembly is CICS. It is housed on the CDMF file. We are using an optional parameter to replace the job card in that JCL member with our own job card.

```
//SYSIN      DD   *
1 49 303
#319@@FROMMAP@D@EFFDATE@THRUMAP@TPMON@BBBBB@WWWWW@AAAAAA@
COGROUP@NBRMAP@JCLMEMB@ OVERRIDE@TYPE@LANGUAGE@ISOLANG@%
```

Where:

FROMMAP	1 to 7-character starting map name.
D	A 1-character device code (normally A).
EFFDATE	Effective date. CYYMMDD format (optional).
THRUMAP	Ending map name. Default is FROMMAP.
TPMON	ID of TP monitor. Enter CICS or IMS.
BBBBB	Optional activity executed before each map. Default is 48808.
WWWWW	Optional activity executed to write map output. Default is 322.
AAAAA	Optional activity executed after each map. Default is 48808.
COGROUP	The 1 to 8-character company group name. The default is every company group.
NBRMAP	The number of maps to be assembled in each job generated. The default is one.
JCLMEMB	The name of the JCL member used to generate the assembly JCL. The default is the system specified (CICS or IMS).
OVERRIDE	Specify if there are override JCL statements in this job stream. If there are override JCL statements, they must immediately follow this parameter statement and must be terminated by an end statement. The format of the override statement is:

NNNNN DATA



Umbrella Programming

Steps to Add a Map Definition

Where:

NNNNN is the sequence number of the job statement.

DATA is the job statement to be generated.

Values for this parameter are:

N No override (default)

Y Overrides follow

A Override all JCL. (Use only the override JCL statements.)

TYPE If the environment is CICS, this value overrides the TYPE parameter on the DFHMSD macro.

LANGUAGE If the environment is CICS and the TYPE Parameter is DSECT, then this parameter overrides the LANG parameter on the DFHMSD macro.

ISOLANG 3-character ISO Language code. Default is ENU. This parameter must be specified if other than ENU is desired.

Notes:



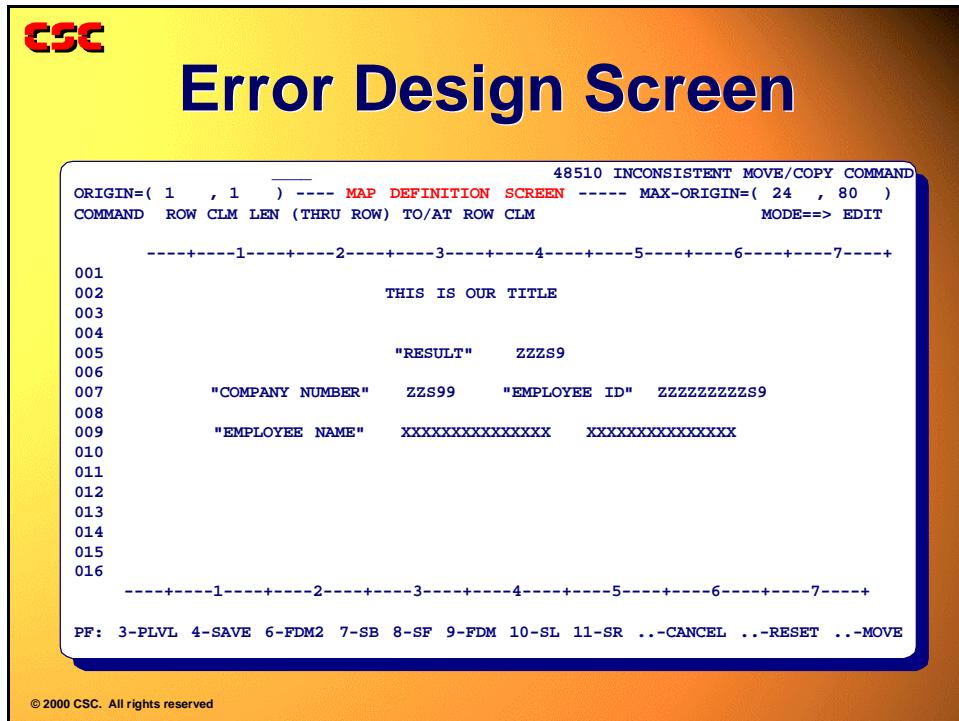
Umbrella Programming

Steps to Add a Map Definition

Map Editor Errors

The Mapping Support System editor will display error messages during your edit session when coding errors are encountered.

Example of Error During Edit Session



The screenshot shows a terminal window titled "Error Design Screen" with the CSC logo at the top left. The window contains the following text:

```
48510 INCONSISTENT MOVE/COPY COMMAND
ORIGIN=( 1 , 1 ) ---- MAP DEFINITION SCREEN ----- MAX-ORIGIN=( 24 , 80 )
COMMAND ROW CLM LEN (THRU ROW) TO/AT ROW CLM MODE==> EDIT

-----+---1---+---2---+---3---+---4---+---5---+---6---+---7---+
001
002      THIS IS OUR TITLE
003
004
005          "RESULT"      ZZZS9
006
007      "COMPANY NUMBER"   ZZZ99      "EMPLOYEE ID"   ZZZZZZZZZS9
008
009      "EMPLOYEE NAME"    XXXXXXXXXXXXXXXX   XXXXXXXXXXXXXXXX
010
011
012
013
014
015
016      -----+---1---+---2---+---3---+---4---+---5---+---6---+---7---+
PF: 3-PLVL 4-SAVE 6-FDM2 7-SB 8-SF 9-FDM 10-SL 11-SR ..-CANCEL ..-RESET ..-MOVE
```

At the bottom of the window, it says "© 2000 CSC. All rights reserved".

Literals must be enclosed in DOUBLE quotes.

Notes:



Example of Error During Edit Session

CSC **Error During Edit Session**

```
48519 FLDS OVERLAP:009/030 & 009/044
----- MAP FIELD DEFINITION MAINTENANCE ----- MODE==> EDIT
MAP=> Z99901M DEV= A COGRP= ALL LANG= ENU EFFDT= 78/01/01 CC#= 999013
-----3-----4-----5-----6-----7-----8
06
07 ZZZS9 "EMPLOYEE ID" ZZZZZZZZZS9
08
-----3-----4-----5-----6-----7-----8
COMMAND==> CHG ORIGIN=>( 7 , 26 ) FLD-NAME(C,E,S)=> C TYPES(F,L,I)=> A
* ROW/CLM T OLN =====FIELD NAME, LITERAL, INCLUDE MAP===== PNBDTIMS
F -----FIELD EDIT PATTERN----- FIL RQD E/I GROUP DESC IMGEFOLD
7 26 F 5 EMP-CO-ID NB
N BZZZS9 0 P S
7 37 L 11 EMPLOYEE ID
7 51 F 11 EMP-KEY-ID NB IM
N BZZZZZZZZS9 0 P S
9 9 L 13 EMPLOYEE NAME
C 9 30 F 15 EMP-F-NAME P B S
9 44 F 15 EMP-L-NAME P B S
C
PF: 3-PLVL 4-SAVE 6-FDM2 7-SB 8-SF 9-DESIGN 10-SL 11-SR ...CHG ...CANCEL
```

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

Steps to Add a Map Definition

Example of Error During Map GEN

Edit checks are also made at map GEN time.

CSC

Error During Map GEN

```
48816 Z99901MA 007/026 ELE UNKNOWN
----- MAP DEFINITION BASE SCREEN -----
COMMAND ==> GEN OWNER ==> ZZA CC# ==> 999013
LANGUAGE => ENU
MAP NAME => Z99901M DEVICE => A CO GROUP => ALL EFF DATE => 78/01/01
DESCRIPTION =====> UMB PGMRERS CLASS :
LINKNAME =====> Z99901MM
TYPE (I,O,U,S) =====> U TRANCODE DYNAMIC? =====> N
TRANCODE =====> 201W TRANCODE PROTECTED? ===> N
EXTENDED ATTRIBUTES? ==> N NON-PFM MAP?=====> N
MAP TYPE (DOC|SCR) ==> SCR MAXIMUM ROWS => 24 MAXIMUM CLMS => 80
EXTRA FPS DATA GROUPS
* DGID * DGID * DGID * DGID * DGID * DGID

NEW MAP KEY FOR COPY:
MAP NAME => DEVICE => CO GROUP => EFF DATE =>
LANGUAGE =>
LAST CHG:DATE 92/08/18 TIME 7.47.53 CC# 999013 SRCE F10F OPER GLT
LAST GEN:DATE TIME CC# SRCE OPER
PF: 4-CHG 5-ADD 6-INQ 7-BROWSE 8-EDIT 9-NXT 10-GEN 11-DSP ..-DOC ...-CPY ...-GRP
```

© 2000 CSC. All rights reserved

All MODIFIABLE FIELDS must be related to fields prior to the GEN.

CSC

007/026 ELE Unknown

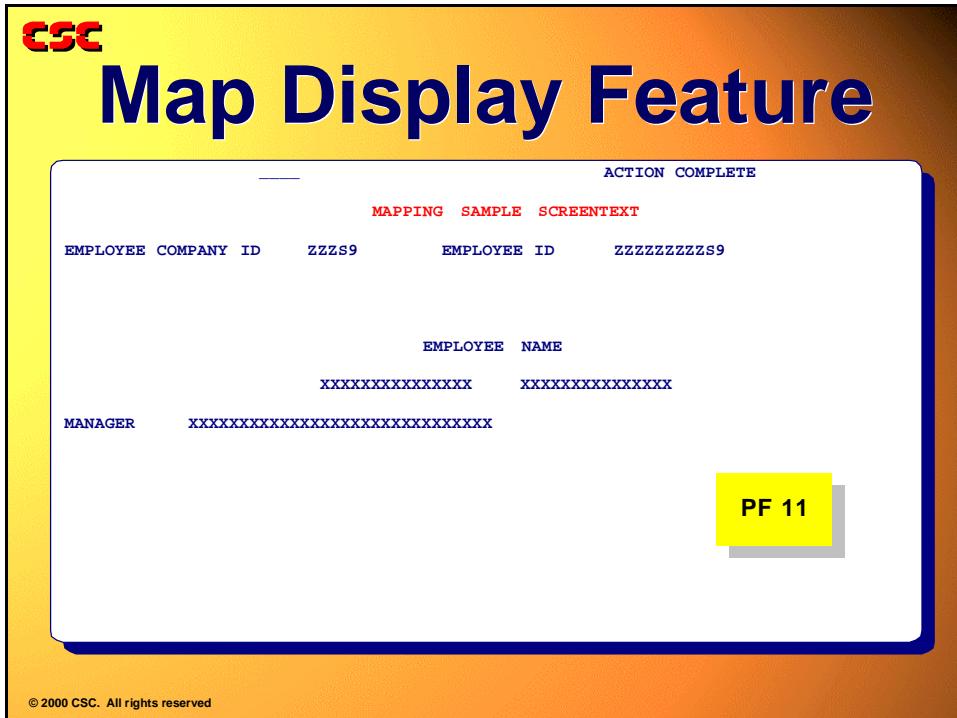
```
ACTION COMPLETE
----- MAP FIELD DEFINITION MAINTENANCE ----- MODE==> EDIT
MAP=> Z99901M DEV= A COGRP= ALL LANG= ENU EFFDT= 78/01/01 CC#= 999013
-----+-----+-----+-----+-----+-----+-----+-----+
00
01
02 "THIS IS OUR TITLE"
-----+-----+-----+-----+-----+-----+-----+-----+
COMMAND==> CHG ORIGIN=>( 1 , 1 ) FLD-NAME(C,E,S)=> C TYPES(F,L,I)=> A
* ROW/CLM T OLN =====FIELD NAME, LITERAL, INCLUDE MAP===== PNBDTIMS
F -----FIELD EDIT PATTERN----- FIL RQD E/I GROUP DESC IMGEFOLD
2 25 L 17 THIS IS OUR TITLE P S
5 27 L 6 RESULT P S
5 38 F 5 EMP-RESULT PNB N
BZZZS9 0
7 9 L 14 COMPANY NUMBER P S
7 26 F 5 NUMERIC MODIFIABLE FIELD NB
N BZZZS9 0
7 37 L 11 EMPLOYEE ID P S
PF: 3-PLVL 4-SAVE 6-FDM2 7-SB 8-SF 9-DESIGN 10-SL 11-SR ..-CHG ..-CANCEL
```

© 2000 CSC. All rights reserved



The Map Display Feature

The PF11 or DSP in the COMMAND field of the "Map Definition Base Screen" will display a sample of the map.



Notes:

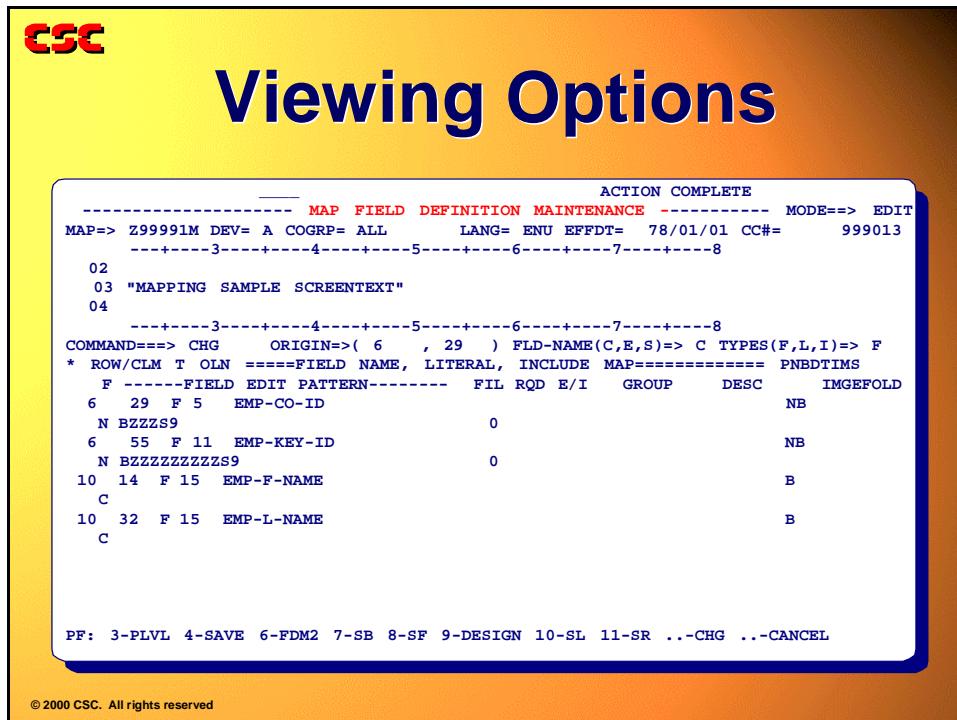


Umbrella Programming

Steps to Add a Map Definition

Viewing Options of Map Field Definition Maintenance

The following shows how to view options on the Map Field Definition Maintenance Screen variable fields.



Notes:



Problem Specifications—Map Definition



Create a Map Definition to display the following information from the Employee data base. Make only the COMPANY NUMBER and EMPLOYEE ID fields modifiable, using the Modified Data Tag attribute. The remaining fields should be protected.

FROM DATA GROUP	DDID	COBOL NAME
DATA GROUP RESULT	47100	TQE47000 EMP-RESULT
COMPANY NUMBER	47100	TQE47001 EMP-CO-ID
EMPLOYEE ID	47100	TQE47002 EMP-KEY-ID
EMPLOYEE FIRST NAME	47110	TQE47102 EMP-F-NAME
EMPLOYEE LAST NAME	47110	TQE47101 EMP-L-NAME
CURRENT EARNINGS	47130	TQE47303 EMP-C-EARN-TOT
YEAR-TO-DATE EARNINGS	47140	TQE47403 EMP-Y-EARN-TOT

Data Group Result will contain the value of 7 on a NOT FOUND and the value of 1 on END DATA.

Notes:



Umbrella Programming

Problem Specifications—Map Definition

Suggested Steps for Problem Solution

1. Base information for Map Definition:

MAP NAME: Z999xxM, where xx is your group number.

DEVICE: A

LINKNAME: Z999xxMs (LOAD MODULE NAME), where xx is your group number, and where s is the map suffix (A=MFS,M=BMS).

MAP TYPE: U

TRANCODE: ZxxW, where xx is your group number.

DESCRIPTION: Include your group number in the Map Description field and in the title of the map you design.

2. Design your screen layout
3. Save and GEN your map
4. Submit the JCL to Documentation and Assemble your map.

Notes:



Map Documentation/Assembly JCL

JCL to generate map documentation and assemble the map has been set up for you.
The instructor will tell you the location of the JCL.

Below is a copy of the JCL to generate map documentation and assemble the map.

```
MODULE NAME ZUPCXXMD

//ZUP{J}MD JOB (HOGN,{B},BEF),'MAP DOCUMENTOR',MSGCLASS=9,
//                  TIME=(00,04),REGION=4M,NOTIFY=&SYSUID
///*
//P$ $$LIB JCLLIB ORDER=( {TL}.PROCLIB)
//*
//JS010      EXEC HOGNBPEM,
//              TEMPLIB=' {L}.TESTLIB',
//              VSAM=' {V}'
//*-----*
//* DOCUMENT MAP Z999\\\MA AND SUBMIT FOR ASSEMBLY *
//*-----*
//SYSIN      DD *
1 49 315
#318--Z999\\\MA--Z999\\\M--SCREEN%
1 49 315
#318--Z999\\\MA--Z999\\\M--FULL%
1 49 303
#319--Z999\\\MA--Z999\\\MCICS--{N}CICS-Y%
100 //ZUP{J}MA JOB (HOGN,{B},BEF),'CICS MAP Z999\\\MA',MSGCLASS=9,
101 //                  TIME=(00,04),REGION=4M,NOTIFY=&SYSUID
END
//*-----*
```

Notes:



Summary

CSC

Summary



© 2000 CSC. All rights reserved

- Process Dictionary houses required information to enable PEM to perform an inquiry function
- Activities should be defined prior to transaction definition
- Screens can be designed and changed quickly
- Standard JCL is delivered
- In CICS, new copy of map required for each change

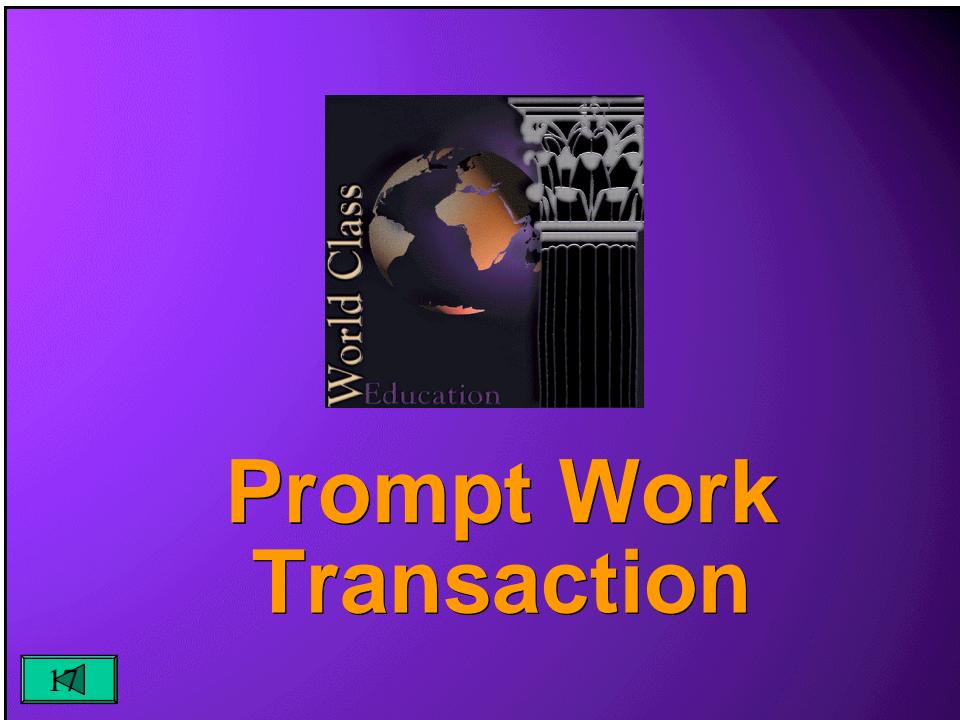
- The Process Dictionary houses the required information to enable PEM to perform an inquiry function on a data base. It is not necessary to invoke a program for an inquiry-only function.
- Activities should be defined prior to the transaction definition.
- Screens can be designed and changed quickly once you become familiar with the mapping facility.
- Standard JCL (for map documentation and assembly) is delivered.
- In CICS, a new copy of a map (or program) is required each time that a change is made.

Notes:



Prompt/Work Transactions

18



Notes:



Purpose



There must be a starting place for everything. This chapter explores the starting point for all Hogan Processing - The Transaction Definition

© 2000 CSC. All rights reserved

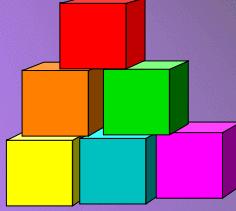
Notes:



Topics

CSC

Topics



- Transaction Definitions
- Conversational vs.
nonconversational vs. pseudo
transactions

© 2000 CSC. All rights reserved

Objectives

CSC

Objectives



- Trace processing flow through a hypothetical transaction
- Follow an online walk though of the UEPR prompt transaction
- Follow an online walk though of the UEQS work transaction
- Identify key Process Dictionary entries used by transaction
- Create an online function by coding
 - Two transaction definitions
 - Several activities
 - A map
 - To test the function

© 2000 CSC. All rights reserved



Umbrella Programming

Example Transaction

Example Transaction

The Transaction Definition indicates the activities to execute for this transaction.

From a cleared screen, enter U TXN UEPR to display the Transaction Definition for the UES Quotations System prompt. PEM executes this single activity and ends the transaction.

CSC

Transaction Definition

ACTION SUCCESSFUL	
----- UMBRELLA TRANSACTION DEFINITION INQUIRY/MAINTENANCE -----	
COMMAND ==> INQ	OWNER ==> UES CC# ==>
APPLICATION ID ======>	88
FUNCTION ID ======>	8010
SOURCE ID ======>	3
COMPANY ID LIST ======>	ALL
EFFECTIVE DATE ======>	78/01/01
TRANSACTION CODE ======>	UEPR
TRANSACTION DESCRIPTION ==>	UES QUOTATIONS SYSTEM PROMPT
DL/I PSB NAME ======>	PSBUESA
DB2 PLAN NAME ======>	
APPC: REMOTE PEM ==>	CONVERSATION ==>
* ACTIVITY * TRANSACTION ACTIVITIES *	
488301 ACTIVITY * ACTIVITY * ACTIVITY *	
LAST CHG:DATE 92/06/15 TIME 9:35:40 CC# SRCE UMB130 OPER	
PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-ACTS 11-TRAN ..-DEL ..-NEW	

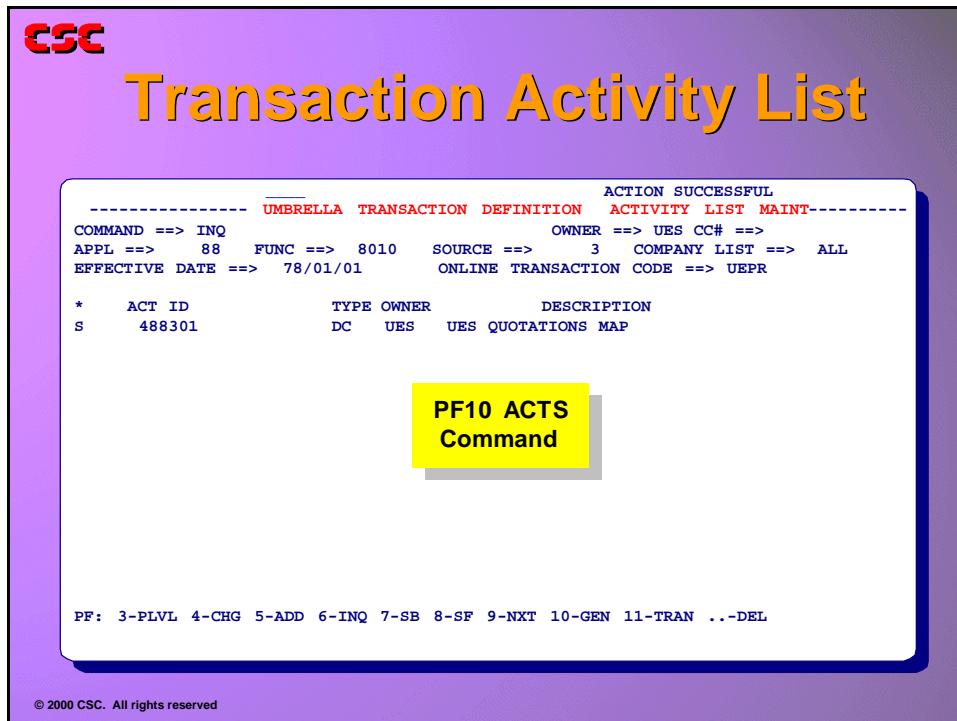
© 2000 CSC. All rights reserved

Notes:



Transaction Activity List Screen

Press PF10 to view the listing and brief description of the activity that makes up the transaction.



Press PF3 to return to the Transaction Definition. To view the entire definition of activity ID 488301, enter INQ in the COMMAND field and S in the action (*) field prior to activity 488301 and press ENTER.

Notes:



Umbrella Programming

Transaction Activity List Screen

Activity Definition 488301

The type of activity is Data Communications (DC). The service is display (DISP). The name of the map associated with this activity is U88201M.

CSC Activity Definition ID 488301

48304 ACTION COMPLETE (DC ACTS)			
ACTIVITY DEFINITION		INQUIRY/MAINTENANCE	
COMMAND ==> INQ		OWNER ==> UES CC# ==>	
ACTIVITY ID ======>	488301	EFF DATE ==>	78/01/01
ACTIVITY MNEMONIC ==>		TYPE ======>	DC
DESCRIPTION ======>	UES QUOTATIONS MAP		
DATA COMMUNICATIONS			
MAPNAME ==> U88201M	RETRY ==> Y		
SERVICE ==> DISP	ERASE ==> Y		
DISP =====> WAIT	PROMPT ==> N		
* DGID	*** DATA GROUPS ***		
	* DGID * DGID *		
LAST CHG:DATE 92/06/15 TIME 9.35.40 CC# SRCE UMB130 OPER			
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-MAPID 11-CPY ..-NXTT ..-NEW ..-NXTA			

© 2000 CSC. All rights reserved

PF10 (MAPID) displays the "Map Definition Base Screen."

Notes:



Map Definition U88201M

The Data Communications activity points to this Map Definition. Two important pieces of information this screen provides are the MAP LINKNAME and the TRANCODE.

CSC

Map Definition U88201M

ACTION COMPLETE										
MAP DEFINITION BASE SCREEN					OWNER ==> UES CC# ==>					
COMMAND ==> INQ	LANGUAGE ==> ENU	MAP NAME => U88201M	DEVICE => A	CO GROUP => ALL	EFF DATE =>	78/01/01				
DESCRIPTION =====> CLIENT ORDERS/INQUIRY ON THOTS-DB :										
LINKNAME =====> U88201MM										
TYPE (I,O,U,S) =====> U					TRANCODE DYNAMIC? =====> N					
TRANCODE =====> UEQS					TRANCODE PROTECTED? =====> N					
EXTENDED ATTRIBUTES? ==> Y					NON-PEM MAP?=====> N					
MAP TYPE (DOC SCR) =====> SCR					MAXIMUM ROWS =>	24	MAXIMUM CLMS => 80			
DGID	*	DGID	*	DGID	*	DGID	*	DGID	*	
EXTRA FPS DATA GROUPS										
NEW MAP KEY FOR COPY:										
MAP NAME =>	DEVICE =>	CO GROUP =>	EFF DATE =>							
LANGUAGE ==>										
LAST CHG:DATE 86/01/17 TIME 2.34.12 CC# SRCE JMMXREF OPER										
LAST GEN:DATE 86/03/14 TIME 1.50.55 CC# SRCE JMMMAPGN OPER										
PF: 4-CHG 5-ADD 6-INQ 7-BROWSE 8-EDIT 9-NXT 10-GEN 11-DSP ...-DOC ...-CPY ...-GRP										

PF10 or MAPID Command

© 2000 CSC. All rights reserved

Notes:

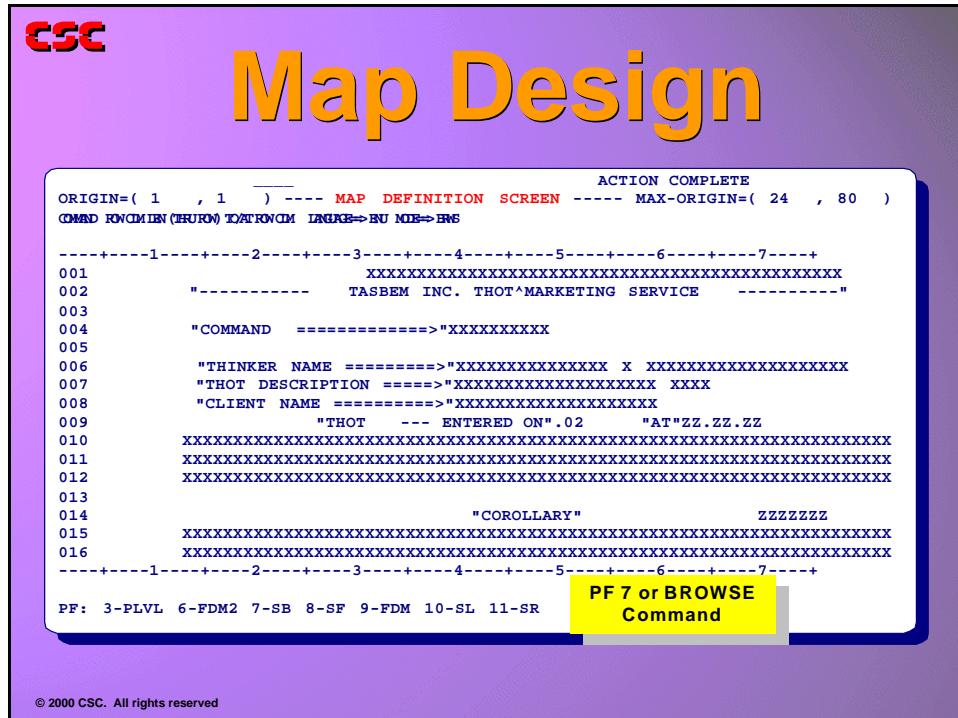


Umbrella Programming

Transaction Activity List Screen

Map Design Screen

From the "Map Definition Base Screen", press PF7 to browse through the "Map Definition Screen."

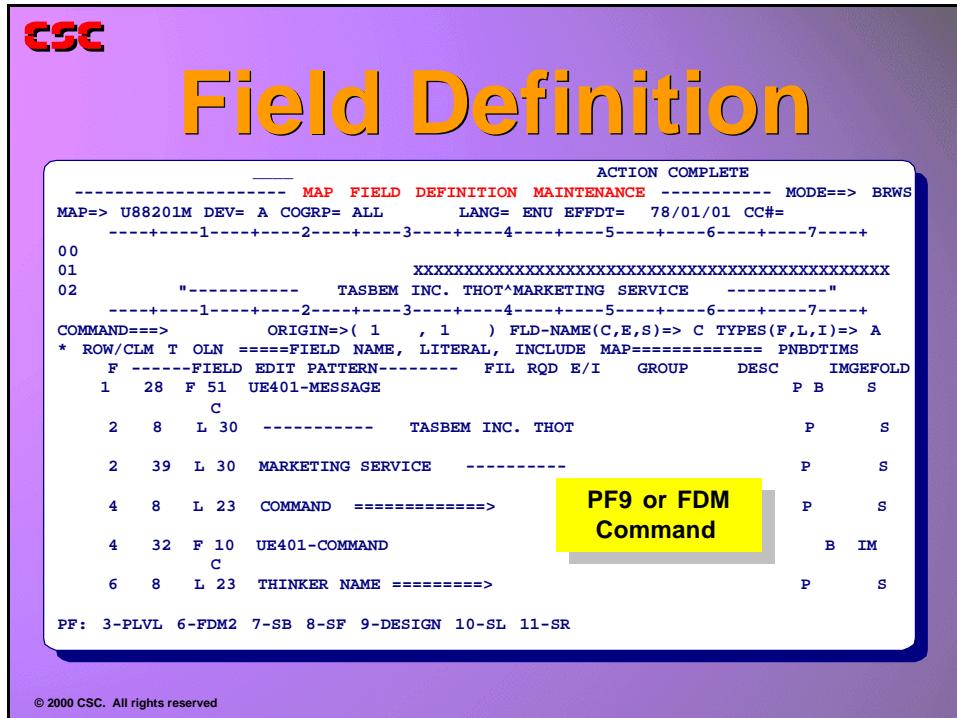


Notes:



Field Definition Screen

From the "Map Definition Screen" press PF9 to display the "Map Field Definition Maintenance" (FDM) screen.



Notes:

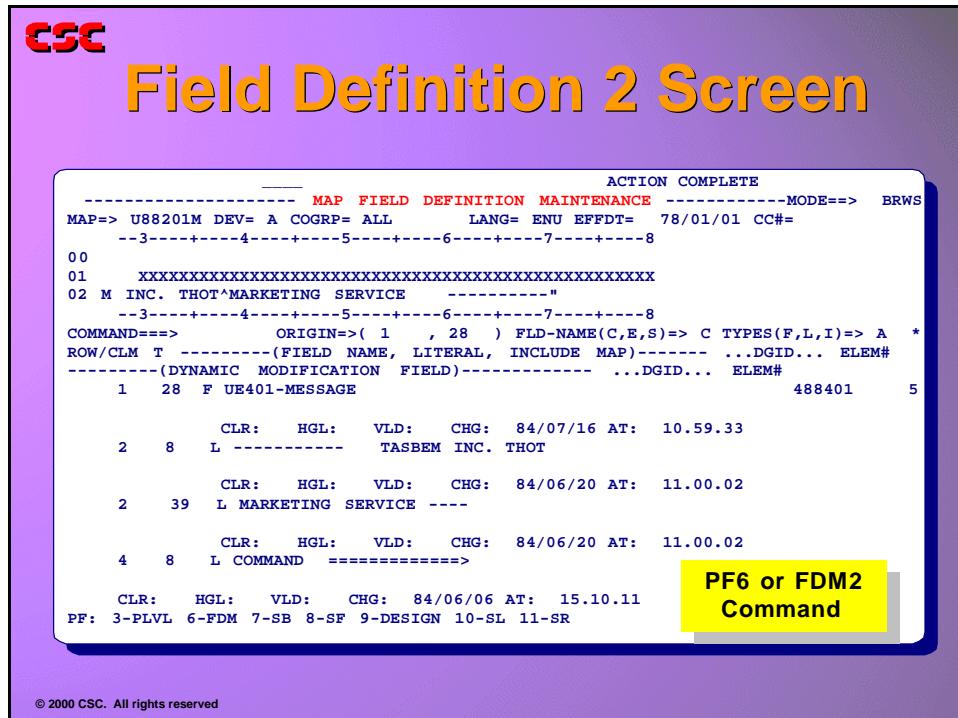


Umbrella Programming

Transaction Activity List Screen

Field Definition 2 Screen

Fields on the map refer to elements defined to a data group. To view the data group to which the map fields belong, enter FDM2 or PF6 key. The fields reference data group 488401.



Notes:



Data Group Definition 488401

To view the Data Group Definition for data group ID 488401. Key I in the COMMAND field, to increment your conversation, and press ENTER. Now key DG 488401 in the COMMAND field.

CSC

Data Group 488401

ACTION COMPLETE	
PROCESS DICTIONARY DATA GROUP DEFINITION SETUP	
COMMAND ==> INQ	OWNER ==> UES CC# ==>
SPECIFY/VERIFY PARAMETERS BELOW:	
DATA GROUP ID NUMBER ==> 488401	DATA DICTIONARY ID ==> UES488401
EFFECTIVE DATE ==> 78/01/01	
DATA GROUP DESCRIPTION ==> UES QUOTATIONS INQUIRY PGM WORK DG	
LENGTH OF DATA GROUP ==> 900	
INITIALIZE (YES,NO,FLD) ==> FLD	
INIT. CHARACTER IF "YES" ==> X' 00 '	
DATA GROUP TYPE ==> TEMPORARY	CSA ELIGIBLE ==>
COPY BOOK NAME: COBOL ==> U88401D	ALC ==> NONE
NEW DATA GROUP KEY FOR COPY:	
DATA GROUP ID==>	ALIAS==>
	EFFECTIVE DATE==>
LAST CHG:DATE 86/09/25 TIME 15.21.26 CC# 39776 SRCE Z005 OPER PRD	
LAST GEN:DATE 86/09/25 TIME 15.21.26 CC# 39776 SRCE Z005 OPER PRD	
PF: 2-XREF 4-CHG 5-ADD 6-INQ 7-BROWSE 8-EDIT 9-NXT 10-GEN 11-CPY ..-DEL	

• I or INCR in Command
• = D.488401 in Command

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

Transaction Activity List Screen

Data Group Fields

To browse the Data Group Definition by pressing PF7.

The screenshot shows a terminal window with the CSC logo at the top left. The title bar reads "Data Group Fields". Below the title is a command line interface with the following text:

```
ACTION COMPLETE
----- DATA GROUP DEFINITION INQUIRY / MAINTENANCE -----
COMMAND INPUT ==> HP                                     CC# ==> 0
DATA GROUP I.D.==> 0000488401 EFF DATE==> 78/01/01 COLS: 9 80 MODE ==> BRWS
***** -1---+---2---+---3---+---4---+---5---+---6---+---7---+---8
000001 1 UE401-WORK-DG.
000003 05 UE401-ACTION          PIC XX.
000004 05 UE401-RESULT         PIC XX.
000005 05 UE401-SCREEN.
000007 10 UE401-MESSAGE        PIC X(30).
000009 10 UE401-COMMAND        PIC X(10).
000011 10 UE401-THINKER-NAME.
000013 15 UE401-THINKER-SURNAME PIC X(20).
000015 15 UE401-THINKER-FIRST-NAME PIC X(15).
000017 15 UE401-THINKER-MIDDLE-INITIAL PIC X.
000019 10 UE401-THOT-DESCRIPTION PIC X(020).
000021 10 UE401-THOT-CODE       PIC X(004).
000023 10 UE401-CLIENT-NAME    PIC X(020).
000025 10 UE401-THOT.
000027 15 UE401-THOT-1         PIC X(70).
000029 15 UE401-THOT-2         PIC X(70).
000031 15 UE401-THOT-3         PIC X(70).
***** -1---+---2---+---3---+---4---+---5---+---6---+---7---+---8
PF: 4-SAVE 5-HP 6-BROWSE 7-SB 8-SF 9-EDIT 10-SL 11-SR ...EID ..-LTYP ..-CANCEL
```

A yellow callout box points to the "PF7 or BROWSE Command" text in the bottom right corner of the command area. At the bottom left of the screen, there is a copyright notice: "© 2000 CSC. All rights reserved".

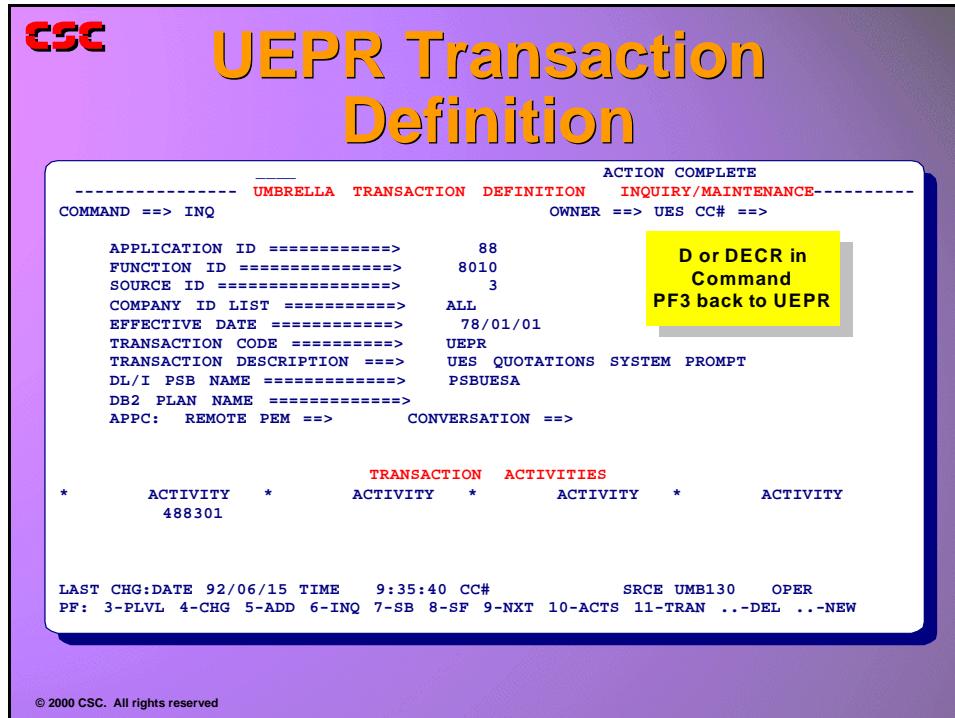
Key D in the COMMAND field to decrement your conversation to your prior conversation.

Notes:



UEPR Transaction Definition

If we were to PF3 back up through our conversation we would redisplay the UEPR Transaction Definition.



At execution time PEM would locate the UEPR Transaction Definition and process all of the activities in the list. After the last activity is processed, PEM would free storage associated with this transaction. The UEPR Transaction Definition contains a DC Display activity.

UEPR Prompt Transaction Display

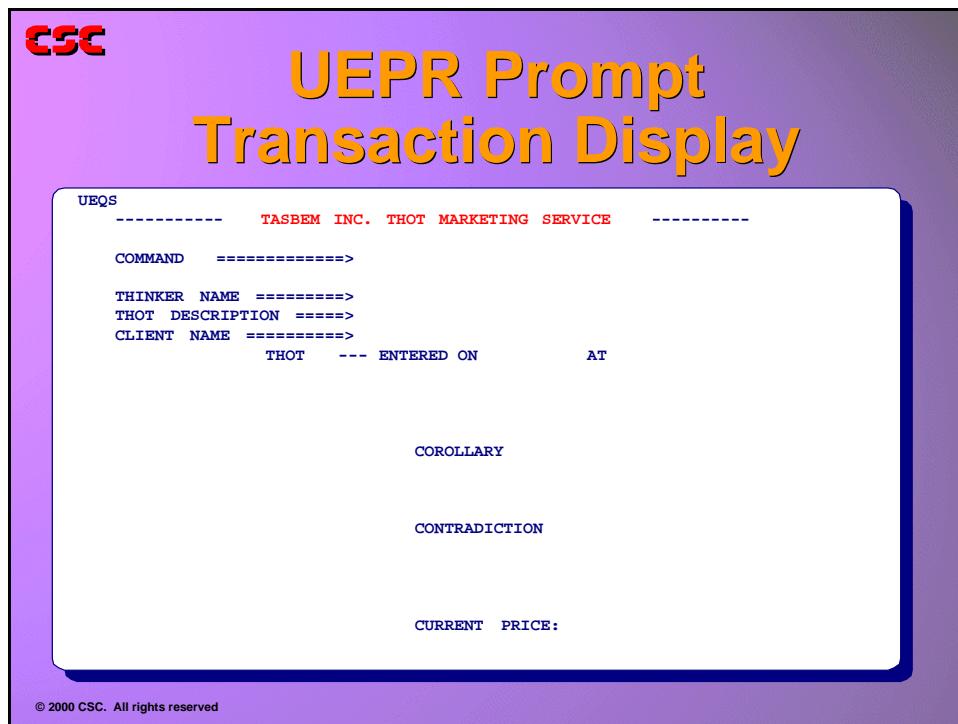
The purpose of the PROMPT transaction is to display the initial formatted screen. With the display of the TASBEM INC. THOT MARKETING SERVICE Screen, the transaction invoked by UEPR is complete.

UEQS is the transaction code formatted on the screen in the upper left corner. If the ENTER key were pressed the UEQS transaction would be executed.



Umbrella Programming

Transaction Activity List Screen



Notes:



Transaction Flow Example



Using the terminal, follow the processing flow of the transaction invoked by UEQS as if you were PEM.

List the Process Dictionary entries in the order they would be accessed during processing.

Locate the Process Dictionary entries as if you were PEM by following the suggested steps below.

1. Locate the Transaction Definition (as illustrated in the online transaction walk through for UEPR.)
 2. Locate each of the Activity Definitions one by one as PEM would. If an activity points to another Process Dictionary Definition, locate that definition. It could be a Map Definition, a Program Definition, or several Data Group Definitions.
-
-
-
-
-
-
-
-
-
-
-
-
-
-
-
-
-
-
-
-
-
-
-

PLEASE DO NOT READ THE NEXT PAGE UNTIL INSTRUCTED TO DO SO!!!



Umbrella Programming

Transaction Flow Example

This page is intentionally left blank.



Online Transaction Flow Exercise Solution

The Process Dictionary definitions would be accessed in the following order:

TRANSACTION DEFINITION	ID 88 8011 3 ALL (UEQS)
WA ACTIVITY	ID 488601
DATA GROUP DEFINITION	ID 488401, 488311, 488310, AND SO ON
DC ACTIVITY (DBLK)	ID 488201
MAP DEFINITION	ID U88201M
LK ACTIVITY	ID 488401
PROGRAM DEFINITION	ID 488401 (LINKNAME U88401)
- OTHER ACTIVITIES ISSUED BY PROGRAM U88401	
LK ACTIVITY	ID 1398
PROGRAM DEFINITION	ID 1250 (LINKNAME T58550)
LK ACTIVITY	ID 48000
PROGRAM DEFINITION	ID 48000 (LINKNAME U48000)
LK ACTIVITY	ID 488510
PROGRAM DEFINITION	ID 488510 (LINKNAME U88510)
--- OTHER DATA GROUPS AND ACTIVITIES ISSUED BY PROGRAM U88510	
DC ACTIVITY (DISP)	ID 488301
MAP DEFINITION	ID U88201M

Notes:



Problem Specifications—Prompt/Work Transaction



Our user wants online (inquiry only) access to data from the EMP Data Base. You will need to build the supporting Process Dictionary entries to support the users request.

Notes:



Sequential Database Access - Phase I



1. Define the supporting Process Dictionary entries to process the map you defined in a prior exercise.
 - 9991xx, where xx is your group number, Work Area (WA) to allocate the Data Groups referenced by the Map Definition.
 - 9992xx, where xx is your group number, Data Communications (DC) to deblock the data from the screen into the data groups.
 - 9993xx, where xx is your group number, Data Communications (DC) to display the map.
 - 9996yy, where yy is your group number + 20, Data Base (HDB) to sequentially read the Data Base EMP.
2. Define the two PEM transactions on the Process Dictionary. Determine the activities required for each.

PROMPT TRANSACTION	WORK TRANSACTION
APPLICATION ID 99	APPLICATION ID 99
FUNCTION ID 99xx	FUNCTION ID 99yy
SOURCE ID 3	SOURCE ID 3
COMPANY ID ALL	COMPANY ID ALL
TRANCODE ZxxP	TRANCODE ZxxW

Where xx is your group number.
Where yy is your group number + 20.

3. Test the inquiry function using the following steps:
 - Clear screen.
 - Enter transaction code ZxxP, where xx is your group number.
 - After your map is displayed, press the ENTER key. The first employee should be displayed. Note the result code.
 - Press the ENTER key. The next employee should be displayed. Note the result code.
 - Change the company number to 80.
 - Continue to press the ENTER key until the result code = 1 (EOF).

Note: Data Group Result will contain the value of 7 on a NOT FOUND and the value of 1 on END DATA.



Umbrella Programming

Problem Specifications—Prompt/Work Transaction

Note: The Link Activity is not used in this exercise.



Random Database Access - Phase II

1. After successfully testing the function as defined, make necessary changes required to inquire against the data base randomly.
2. Assume the operator enters the data base key fields of company number and employee number.
3. The company number and employee number for testing your transaction are as follows:

COMPANY:	1	
EMPLOYEE NUMBERS:	1111	TIBERIUS CAESAR
	12346	CAESAR CLAUDIUS
	878	(not on file)
	1010101010	BO DEREK

4. Test the inquiry function using the following steps:
 1. Clear screen.
 2. Enter transaction code ZxxP, where xx is your group number.
 3. After your map is displayed, enter a company number and employee number from the above list, and press the ENTER key. Note the result code. If successful, the exact employee requested should be displayed.
 4. Continue with the remaining test cases.

Note: Data Group Result will contain the value of 7 on a NOT FOUND and the value of 1 on END DATA.

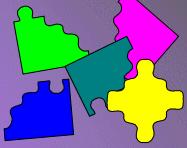
Notes:



Summary



Summary



- High volume online transactions can be run as a pair of CICS or IMS transactions (prompt,work)
- Begin with the Process Dictionary transaction definition to trace flow
- Identify the work activities from the transaction definition
- Trace flow of activities to walk through the work flow and identify entries
- Umbrella screens provide flexibility via internal cross-reference facilities



© 2001 CSC. All rights reserved

►9

- The Umbrella Example System (UES) contains models of the use of all Umbrella subsystems.
- Under Hogan, high volume online transactions can be efficiently run as a pair of CICS or IMS transactions (a prompt and a work transaction).
- To trace work flow under Hogan both in batch and online environments, begin with the Process Dictionary transaction definition.
- From the transaction definition, you can identify the activities that make up the work.
- By tracing the flow of activities, you can walk through the work flow and identify the other Process Dictionary entries invoked, such as, maps, programs, data bases, data groups, data fields.
- The Umbrella screens are designed to provide flexibility in accessing the various and numerous screens housing the Process Dictionary parameters.



Executing a Program Online

19

Purpose



To invoke a program
online you must
issue a Link Activity

© 2000 CSC. All rights reserved

Notes:



Objectives



The slide features a light purple background with a faint white cloud graphic. In the top left corner is the red CSC logo. To its right, the word "Objective" is written in large, bold, yellow letters with a black shadow. In the top right corner is a red and white bullseye target with three yellow arrows hitting the center. At the bottom left, there is a small copyright notice: "© 2000 CSC. All rights reserved".

To be able to invoke your program online

Invoking a Program Online

Invoking a program under the Umbrella in an online environment is no different from invoking a program in batch. The link activity is used. The COBOL program we have been coding has only executed in the batch environment, SO FAR. In the JCL SYSIN you provided PEM with the information needed to locate the Transaction Definition in the Process Dictionary. Your batch Transaction Definition specified a link activity to your COBOL program.

In the WORK Transaction Definition, coded in the last exercise, we did not execute any link activities. Our online transaction displayed our map for both sequential and random accesses to the EMP Data Base.

Notes:



Problem Specifications—Execute Program Online



Link to Application Program - Phase I

- Change the WORK Transaction to execute Link activity 9994xx, where xx is your group number, instead of HDB activity 9996yy, where yy is your group number + 20. Your activities should now be WA, DC, LK, and DC.
- From a cleared screen, key the PROMPT ZxxP, where xx is your group number, Transaction Code and press ENTER. The Prompt Transaction caused your map to be displayed just as it did in the last lab exercise.
- Now press the ENTER key again. This will cause the ZxxW, where xx is your group number, to be executed. However, this time the work transaction contains the link activity to your program. It is your program that is reading the EMP Data Base.
- Data group result contains the value of 7 on a NOT FOUND and the value of 1 on END DATA.

Notes:



Umbrella Programming

Problem Specifications—Execute Program Online

Enhance Map/Random Access - Phase II

Not only does your COBOL program read the EMP Data Base it also links to Date Services and CDMF. With all of that extra information accessible it would be nice to display some additional information on our map.

FROM DATA GROUP	DDID	COBOL NAME
10 BUSINESS DAYS FROM TODAY	2000	THE48006 DCB-RAW-DATE-AREA
JOB CLASS	47120	TQE47206 JOB-STAT-CLASS
JOB POSITION CODE	47120	TQE47207 JOB-STAT-POSITION
POSITION DESCRIPTION	47190	TQE47902 EMP-POS-DESCRIPTION

1. Modify your screen layout. Follow all steps required when a map is changed.
2. In the real world an online inquiry would access the data base randomly. To make our exercise more like the real world, keep the sequential access for batch processing but for the online processing access the data base randomly. Follow all steps required when a program is changed.
 - Change the BA000-MAIN-LINE section to test if the program is executing in batch or online by checking the TCB-SOURCE field. This field, TCB-BTCH, is located in the UTCB.

```
IF TCB-BTCH
    PERFORM RD000-READ-EMP-DB-SEQUENTIAL THRU RD099-EXIT
ELSE
    PERFORM RD100-READ-EMP-DB-RANDOM THRU RD199-EXIT.
```

- Add code in the RD100-READ-EMP-DB-RANDOM paragraph to issue HDB activity 9996yy, where yy is your group number plus 20, to access the EMP Data Base randomly. This activity was created in a prior exercise.
- Test for the following three conditions after accessing the data base.

Successful, exit

Not found, exit

Any other error, abend



Summary



Summary



- Several steps are required to change a screen
- Programs are invoked from a transaction by a link activity
- Data groups referenced by a map must be allocated prior to a data communication deblock activity

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

Summary



Function Processing System—FPS

20

Purpose



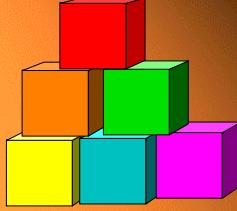
Notes:



Topics

CSC

Topics



- Features of FPS
- Environment flow
- Application control with FPS
- Information needed for online processing
- FPS Maintenance Screens
- FPS Usage of CDMF Formats

© 2000 CSC. All rights reserved

Objectives

CSC

Objectives



To become familiar with:

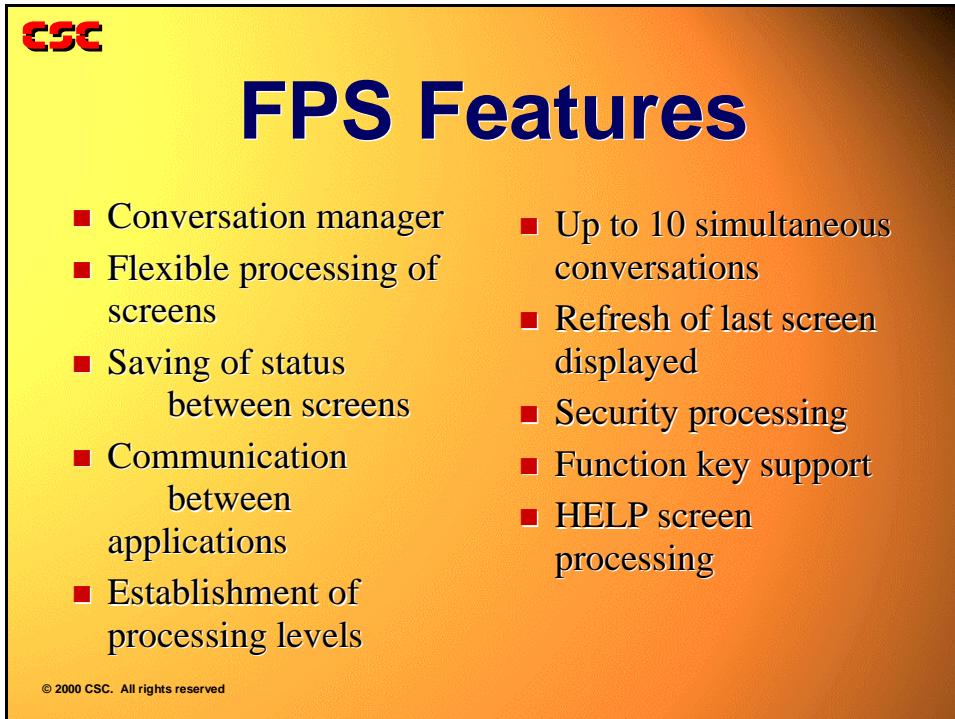
- Key features of FPS
- Process flow under FPS
- How application is in control with FPS
- Information FPS needs to do online processing
- Screens involved in FPS maintenance
- CDMF Format associated with each FPS component

© 2000 CSC. All rights reserved



FPS Features

FPS is our online conversation manager software. It enables the processing of the features listed here.



The slide has a yellow-to-orange gradient background. In the top left corner is the CSC logo. The title 'FPS Features' is in large blue bold letters. Below the title is a bulleted list of nine features. At the bottom left is a small copyright notice.

CSC

FPS Features

- Conversation manager
- Flexible processing of screens
- Saving of status between screens
- Communication between applications
- Establishment of processing levels
- Up to 10 simultaneous conversations
- Refresh of last screen displayed
- Security processing
- Function key support
- HELP screen processing

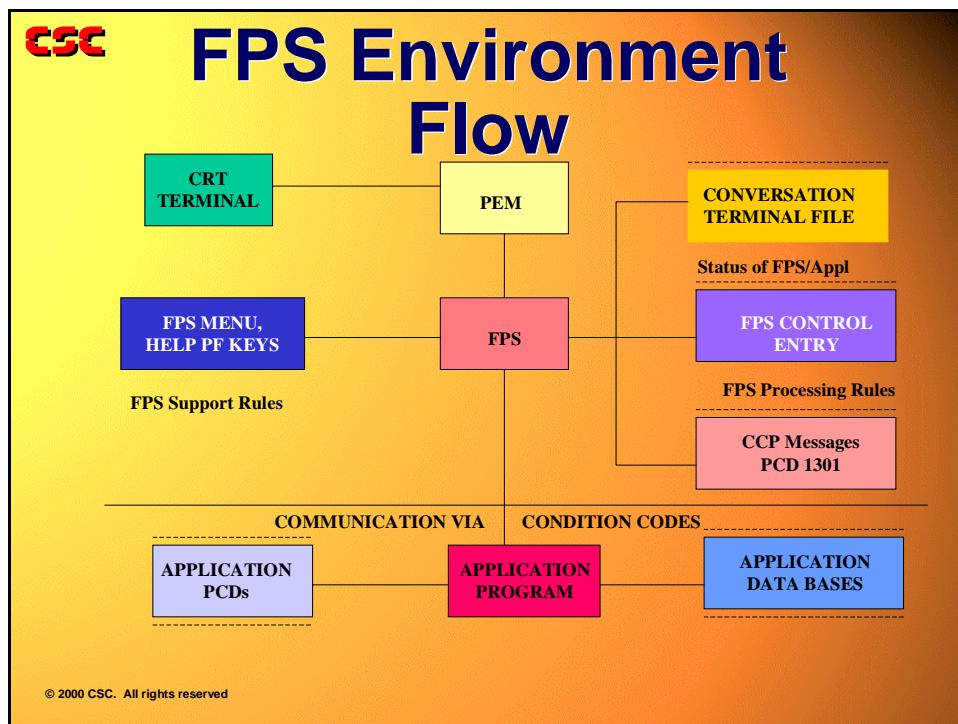
© 2000 CSC. All rights reserved

Notes:



Basics of FPS

FPS Environment Flow



Notes:



Application Control

Application systems that do online processing through the Function Processing System (FPS) control the screen flow and sequencing for the operator. The processing is driven by the business function, not a set of transaction codes.

Two important concepts to understand about FPS are conversation layer and processing level. A conversation is the utilization of FPS to perform a business task within an application. For example, Hogan's Customer Information System (CIS) is an FPS application. If a teller needs to make a name change for a customer, an FPS conversation is initiated. The teller could then be asked to locate account information for another customer. The teller could start up a second conversation within the CIS application. FPS allows an operator 10 simultaneous conversations per terminal and application.

Within a conversation layer, there can be multiple processing levels. A change in processing level can be needed to walk through the steps to accomplish a task. Most importantly, there is usually a need to save data from one step to the next. Changes in processing levels signal FPS to save data. When a customer name change is needed, other screens that contain the name (such as the account information screen) are presented to the operator for updating. These processing levels are set up for the operator by the design of the screen flow (scenario).

Notes:



Umbrella Programming

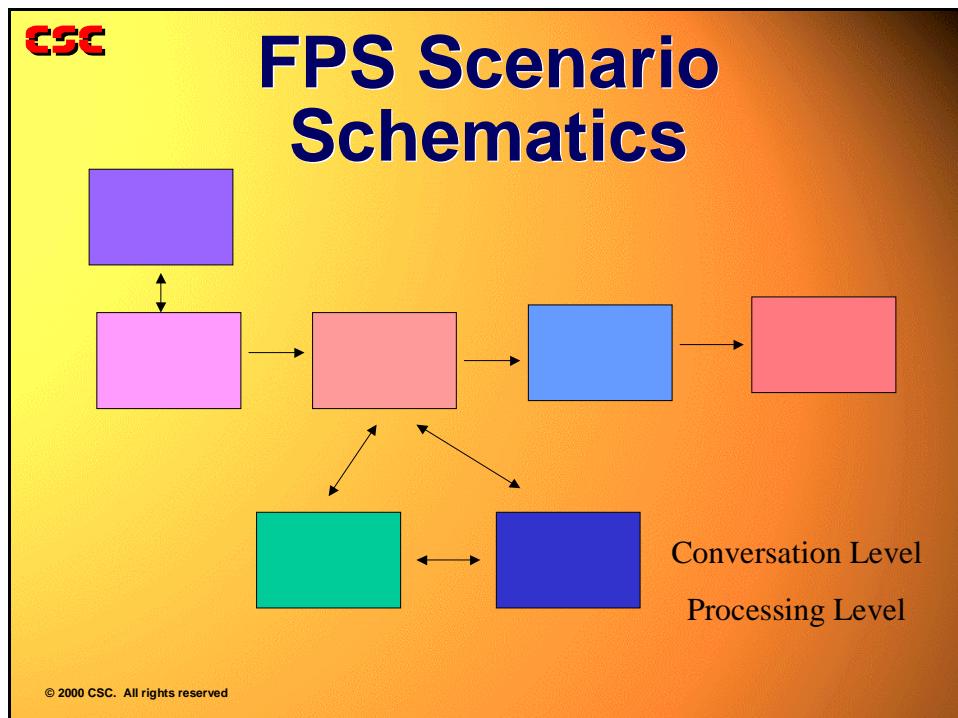
Basics of FPS

FPS Scenario Schematics

FPS is a driver module that uses CDMF formats (PCDs) and COBOL programs for online processing. Every screen is associated with a FPS Control record.

The following schematic illustrates a means of diagramming FPS scenarios. Each box represents a screen or FPS Control record. You will note that conversations may flow down or across. Downward flow indicates a change in processing level.

Because scenarios can be quite complicated, the use of such diagrams is strongly recommended.



Notes:



Information Needed by FPS

There are four pieces of information that have to be available. They are:

1. The screen to display
2. The program to process the screen
3. The screen that appears next
4. The data resources to save before displaying the next screen.

Data is saved across screens by FPS. It uses a file, the Conversational Terminal File (CTF) to save the data. The remaining information is provided to FPS on CDMF. FPS Conversational Control records are used to process the flow of screens (scenarios).

FPS Application Definition Requirements

To define an application to FPS, there are four requirements.

1. Each application must have at least three transactions. These are:
 - Prompt
 - Work
 - Reset

For example, these transactions for the DEM system are DEMO, DEMT, and DEMR, respectively.

2. The application must be defined in the Application Table.
3. There must be a FPS Control record for the main menu.
4. There must be a FPS operator/password entry to access the application.

Notes:



Umbrella Programming

Basics of FPS

Application Programs Running Under FPS

FPS links to COBOL programs to perform the business logic associated with the application. There are several requirements for COBOL programs running under FPS.

- They must interrogate the FPS command field to check the operator request.
- They cannot issue data communication activities.
- They must set a condition code for each possible outcome.
- They must return control to PEM.

Application Maps Running Under FPS

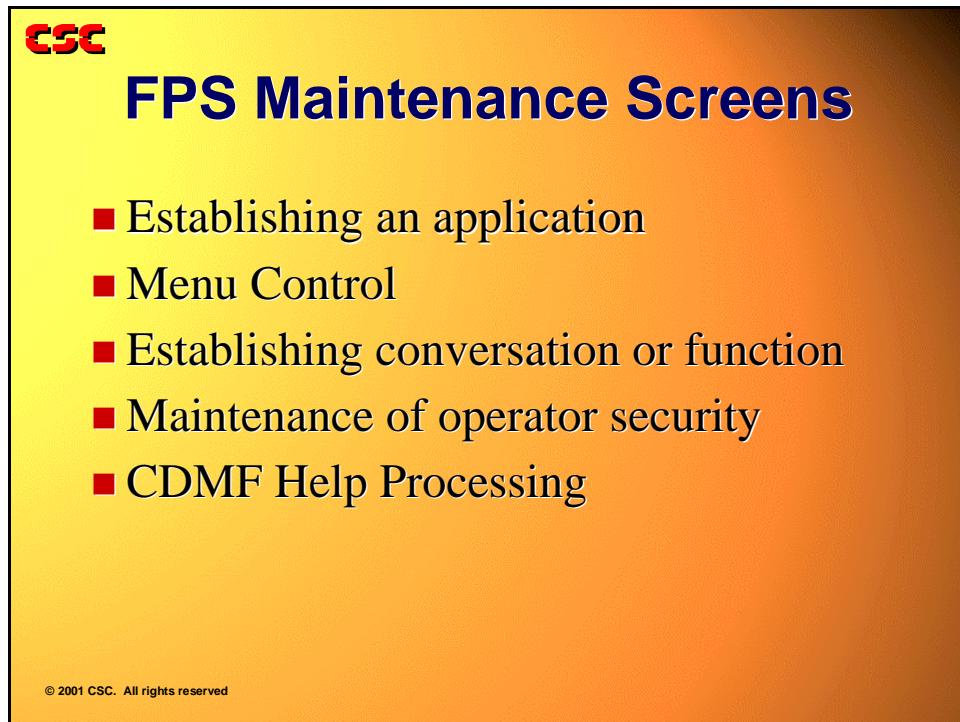
- The maps must contain the FPS header include maps.

Notes:



FPS Maintenance Screens

As with all Umbrella systems, FPS processing is controlled by parameters stored on CDMF and maintainable on-line through the Umbrella Maintenance Facility. The on-line maintenance facility consists of five components:



The screenshot shows a yellow-tinted menu screen from CSC's FPS Maintenance Screens. At the top left is the CSC logo. The main title "FPS Maintenance Screens" is centered in large blue font. Below it is a bulleted list of five items, each preceded by a red square bullet point. The list includes: "Establishing an application", "Menu Control", "Establishing conversation or function", "Maintenance of operator security", and "CDMF Help Processing". At the bottom left of the screen, there is a small copyright notice: "© 2001 CSC. All rights reserved".

- Establishing an application
- Menu Control
- Establishing conversation or function
- Maintenance of operator security
- CDMF Help Processing

© 2001 CSC. All rights reserved

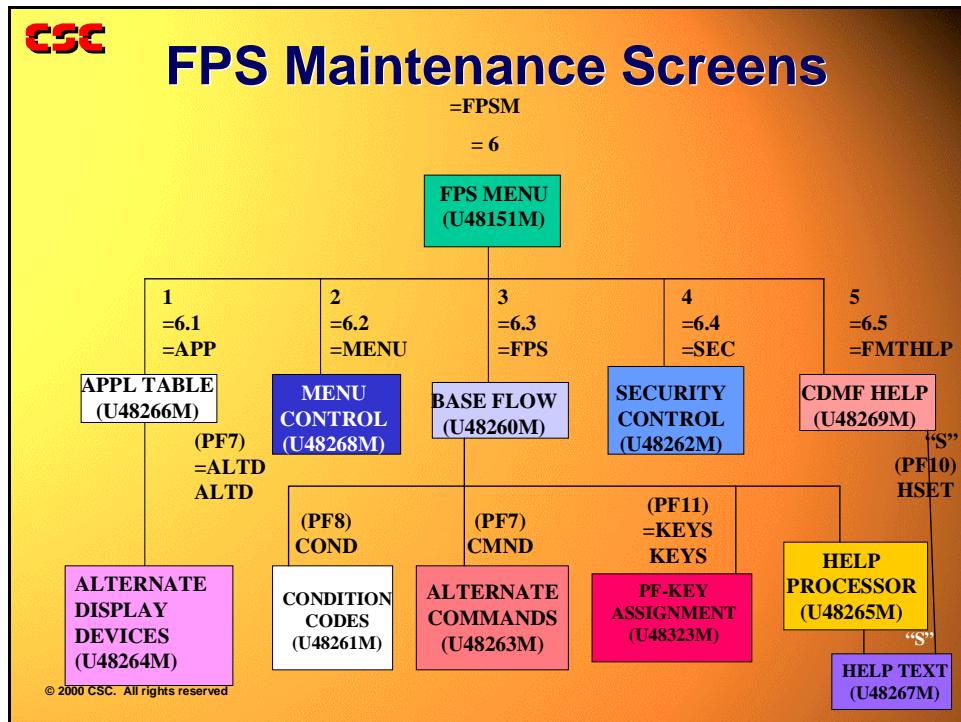
The following chart outlines the components of the conversation for FPS maintenance. Each box represents an on-line screen. Prompts from that screen are shown on the line leading to the box. Program function (PF) keys are shown in parenthesis. Unless otherwise indicated, all prompts are entered in the command field for the screen. Exceptions, such as prompts which must be entered on a specific line, are shown within double quotes. Any command which is preceded by an equal sign can also be entered from a cleared screen using the U transaction. Otherwise, the screen can only be accessed from its parent.

This class will limit its discussions to the first three (3) screens, Application Definition, Menu Control, and Conversation Control.



Umbrella Programming

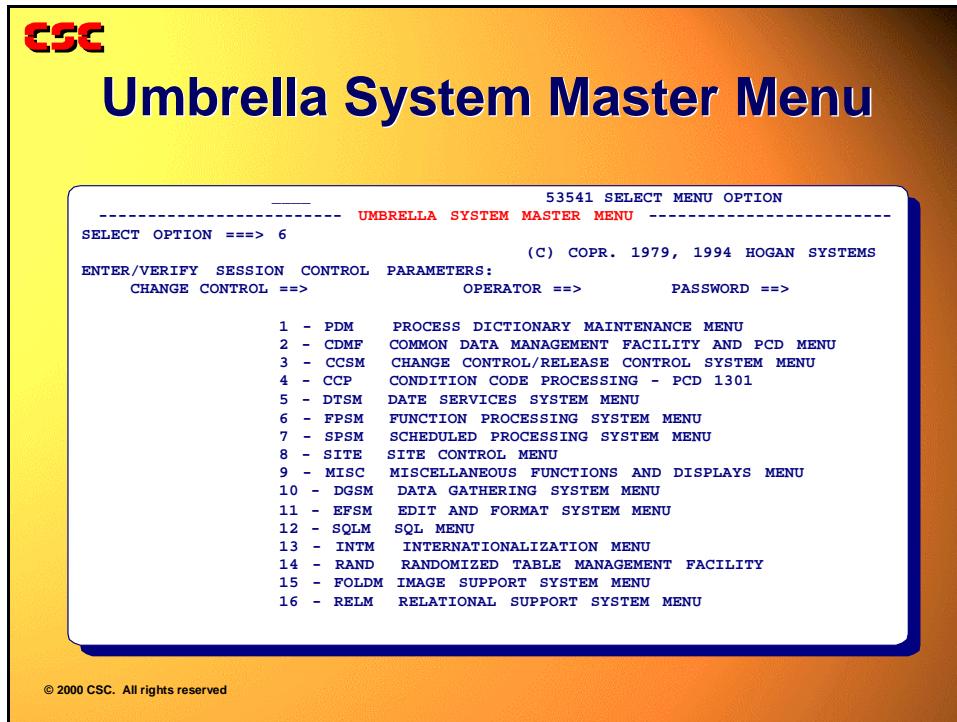
FPS Maintenance Screens



Notes:



These options can be accessed through the Umbrella System Master Menu.



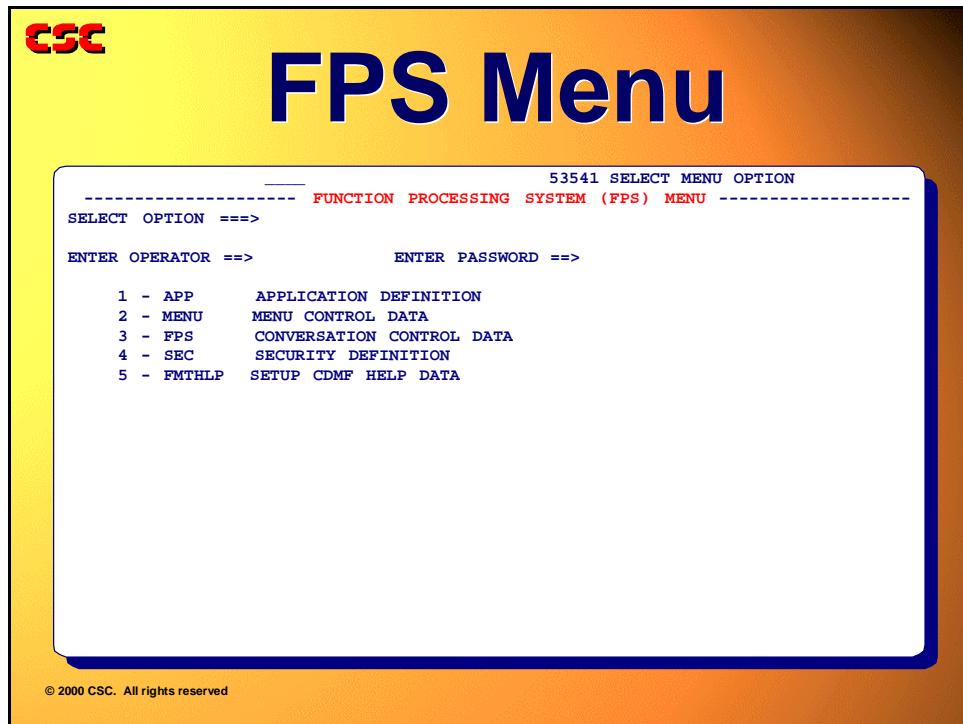
Key 6 or FPSM in the SELECT OPTION field and press ENTER to display the "Function Processing System (FPS) Menu" screen.

Notes:



Umbrella Programming

FPS Maintenance Screens



The direct prompts for these five functions are:

1. APP
2. MENU
3. FPS
4. SEC
5. FMTHLP

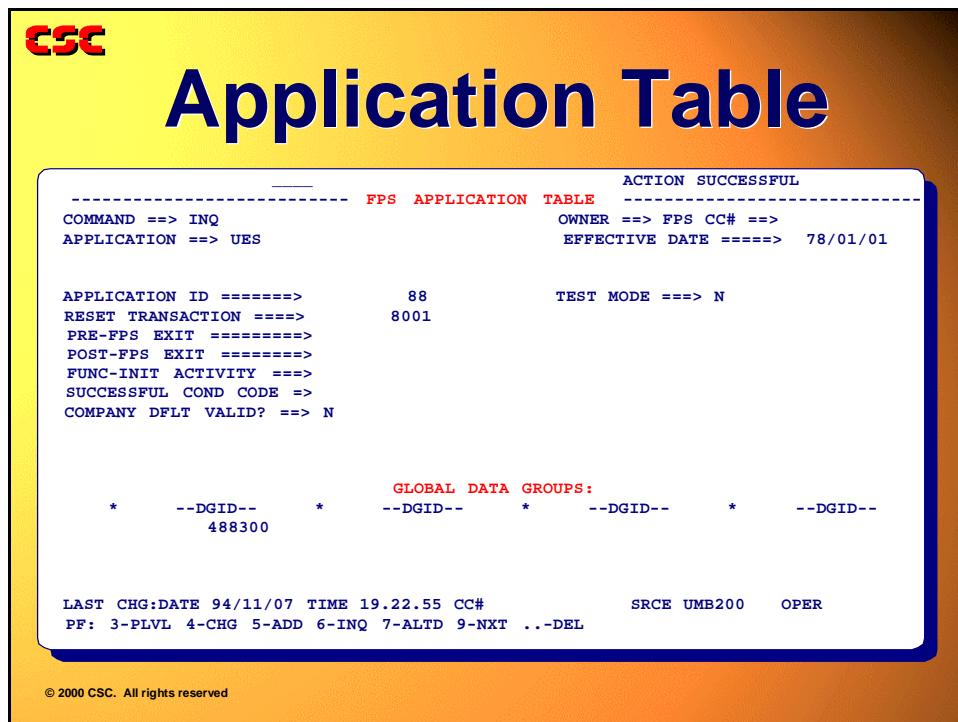
Notes:



Application Table Screen

The Application ID is used by the FPS Supervisor in determining the transaction to be processed. Any activities common to the application, rather than the conversation, are to be set-up on the "Application Definition" Screen.

The "Application Definition" Screen looks like this:



The "Application Definition" Screen can be displayed using one of the following techniques:

1. From the Function Processing System Menu, enter 1 in the SELECT OPTION field and press ENTER.
2. From the UMBRELLA System Master Menu, enter 6.1 in the SELECT OPTION field and press ENTER.
3. From any UMBRELLA System screen, enter =APP.xxx in the SELECT OPTION field and press ENTER.
4. From a cleared screen key in:

U APP.xxx

Where xxx is the application keyword. If a value for xxx is not entered, the first APP component on file is displayed.



Umbrella Programming

FPS Maintenance Screens

Menu Processor Control Data Screen

The starting point of all on-line systems is the main menu. Each application which runs under the FPS system must have a MENU function defined on the Conversation Control record. The key must be in the following format:

xxxMENU000 - where xxx = application code

The FPS Supervisor automatically builds this key when a PROMPT or a RESET transaction is entered. This entry is responsible for DISPLAYING and DEBLOCKING the master menu for the application xxx.

Although there is more than one way to process the menu, the most straightforward method is through the use of the MENU PROCESSOR. Using this method, the Conversation Control entry with a function of MENU must have a LINK activity of 13563 for program I53563, the Menu Processor program. There must also be a Menu entry (PCD 1686) with the same user key as the Conversation Control entry, xxxMENU000. The Menu entry contains a list of prompt values and the associated user key of the Conversation Control record to begin a specific function.

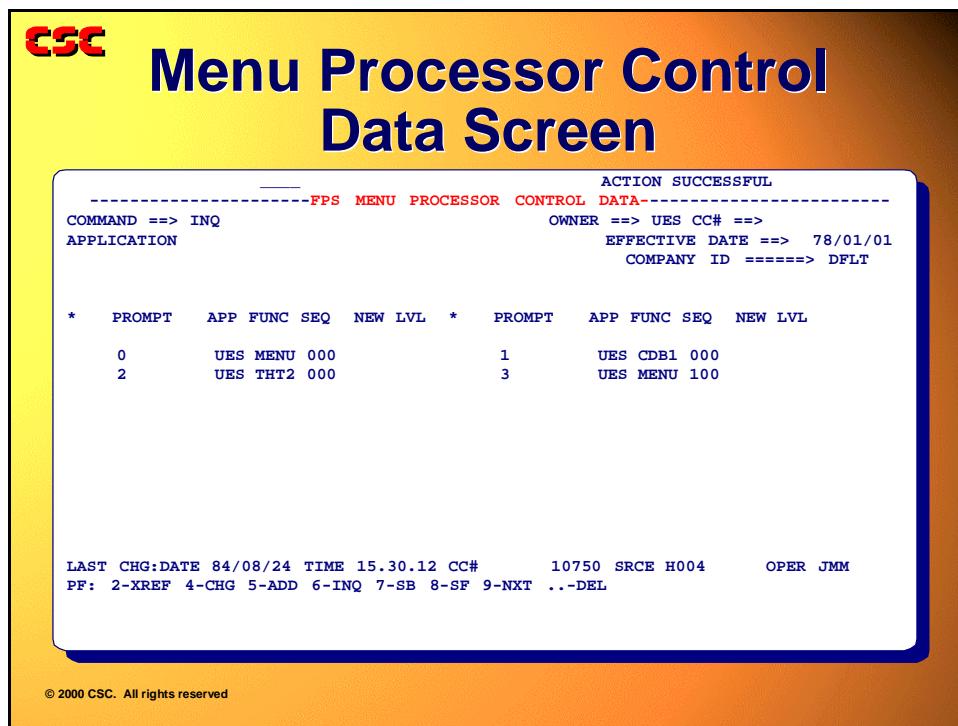
When FPS invokes the Menu Processor program, the Menu entry is retrieved and the list of prompt values is interrogated to determine the correct Conversation Control record needed to begin the execution of the function requested.

FPS expects menu prompts to be placed in the next command field or the FPS action field. In addition, any command with an = (equal sign) in position one invokes the menu processor to determine the correct Conversation Control entry to execute.

Notes:



The Menu Processing Maintenance screen looks like this:



The Menu Processing Maintenance screen can be displayed using one of the following techniques:

1. From the Function Processing System Menu enter 2 in the SELECT OPTION field and press ENTER.
2. From the UMBRELLA System Master Menu, enter 6.2 in the SELECT OPTION field and press ENTER.
3. From any UMBRELLA System screen, enter =MENU.xxxx yyyy sss in the SELECT OPTION field and press ENTER.
4. From a cleared screen key in:

U MENU.xxxx yyyy sss

Where xxx is the application mnemonic.

yyyy is the function desired.

sss is the sequence number of the menu.

If a value for xxx yyyy sss is not entered, the first MENU component on file is displayed.



Umbrella Programming

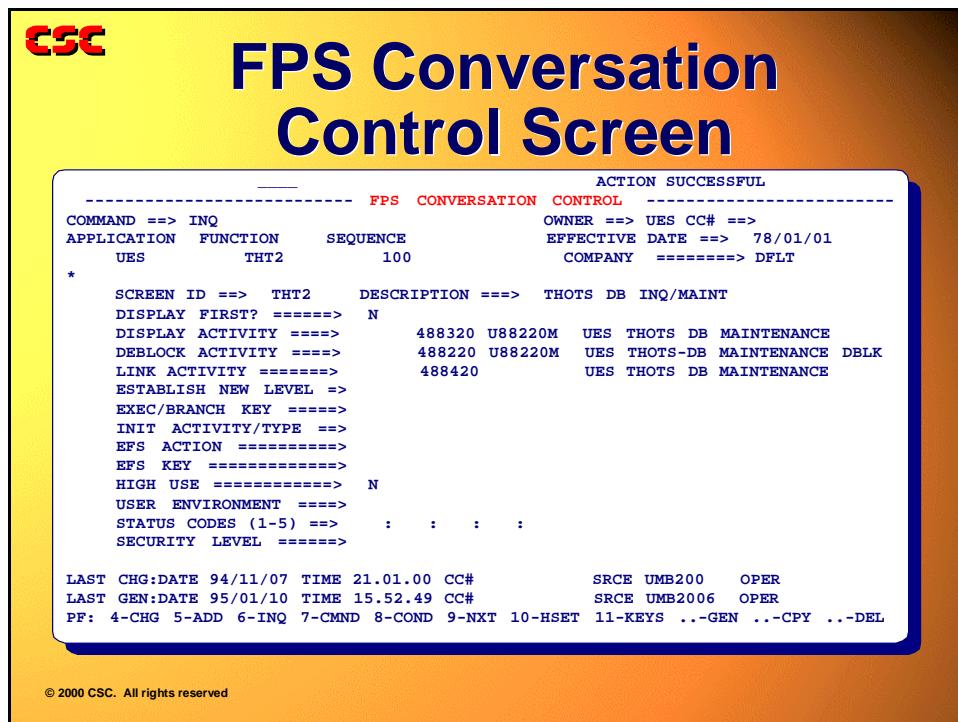
FPS Maintenance Screens

Conversation Control Screen

The flow of logic executed by FPS is maintained on this screen. FPS determines the correct Conversation Control entry from the Menu Processor. This entry is then retrieved and interrogated for Displaying and Deblocking the correct maps, as well as, executing the Link activity of the correct program for massaging the data properly. This entry also directs FPS to the NEXT entry to be executed in the flow of the conversation.

Due to the amount of information contained on the Conversation Control record, the on-line maintenance is contained on two screens. The first contains the basic information and the second contains all the condition codes for that particular entry.

The FPS Basic Conversation Control screen looks like this:



There are two simple forms Conversation Control Entry. The first is called the Basic Logic Entry, or BL. The view in the example screen shows a typical BL entry. It is called a Basic Logic Entry because it contains Activity Numbers that cause external application logic processing to occur, and because it contains a Condition Code table which determines the continuing flow of the transaction logic.

The second type of entry is called a Branch Execute Entry, or BE. A BE unconditionally redirects the flow of an FPS transaction to another Conversation Control Entry. An example of this sub type appears later in this chapter.

The FPS Basic Conversation Control screen can be displayed using one of the following techniques:



Umbrella Programming

FPS Maintenance Screens

1. From the Function Processing System Menu enter 3 in the SELECT OPTION field and press ENTER.
2. From the UMBRELLA System Master Menu, enter 6.3 in the SELECT OPTION field and press ENTER.
3. From any UMBRELLA System screen, enter =FPS.xyyyssss in the SELECT OPTION field and press ENTER.
4. From a cleared screen key in:

U FPS .xxxxyyysss

Where xxx is the application mnemonic.

yyyy is the function (e.g., CONV).

sss is the sequence number desired (e.g., 000).

If a value for xxxxssss is not entered, the first FPS component on file is displayed.

Notes:



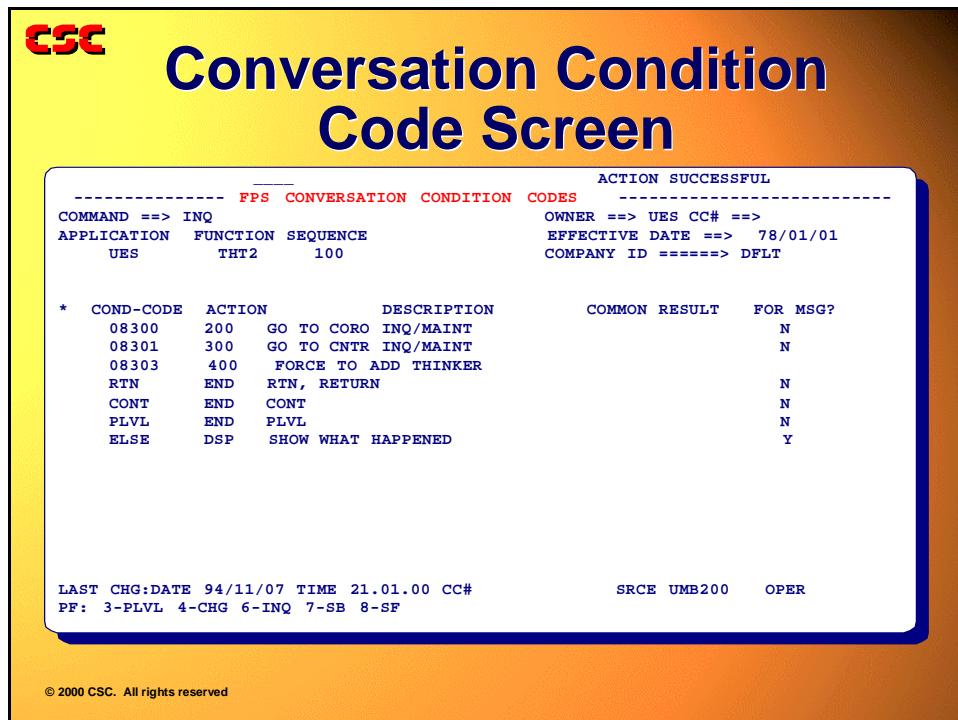
Umbrella Programming

FPS Maintenance Screens

Conversation Condition Code Screen

As can be seen on the previous logic flow chart, the condition codes contained on the Conversation Control entry are used by the FPS supervisor to determine the next step to be executed. FPS interrogates the list of condition codes contained on the Control entry against TCB-USER-CC to determine the ACTION to be taken. When the correct code is found the ACTION field is analyzed for the proper execution process.

The Conversation Condition Code screen looks like this:



The Condition Code screen may be displayed only from the basic conversation screen. This screen may be displayed using one of the following techniques:

1. From the Conversation Control Base Screen press PF8.
2. From the Conversation Control Base Screen, enter COND in the COMMAND field and press ENTER.

Notes:



FPS and CDMF

The values contained in the preceding screens are stored on CDMF as formats. The following table associates each screen with the format on which the screen data are stored.

SCREEN	FORMATID
Application Table Screen	48266
Additional Display Device Screen	1694
Menu Processor Control Data Screen	1686
Conversation Control Screen	1690
Conversation Condition Code Screen	1690
Alternate Command Screen	1693
Function Key Definition Screen	48323
Help Processor Screen	1688
Help Processor for CDMF Formats Screen	1688
Help Description Data Screen	1687
Operator Security Control Screen	1698

Notes:



Summary



Summary

- FPS Applications link to FPS Supervisor
- Processing of multiple screens with same set or transaction codes
- Application in control of processing of screens
- FPS gains information from Conversation Terminal File and FPS Control records
- Scenarios should be designed in detail through diagrams

© 2000 CSC. All rights reserved

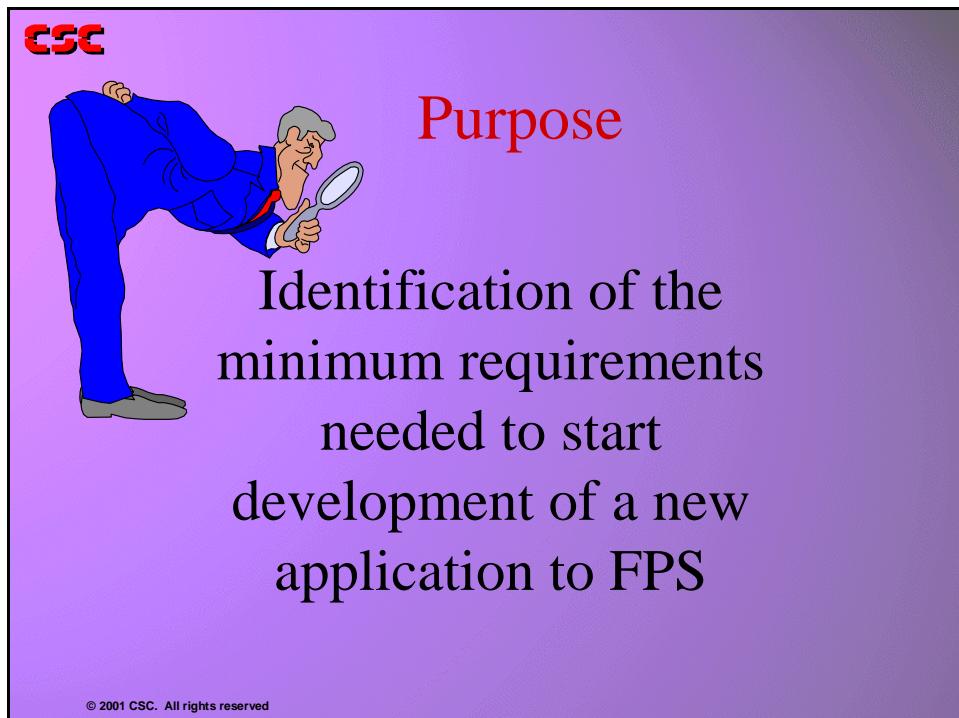
- All FPS Applications link to the FPS Supervisor, which performs the application's data communications needs and links to the application programs.
- The FPS System allows for the processing of more than one screen using the same set of transaction codes.
- FPS is designed for the application to be in control of the processing of screens. It provides a user-friendly system where an operator is guided through a business function, such as setup of a new customer.
- Online systems need to know the screen to display, the program to process the screen, the next screen, and the data to save before displaying the next screen. FPS gains this information from the Conversational Terminal File and FPS Control records.
- FPS scenarios should be designed in detail through diagrams.



FPS Application Requirements

21

Purpose



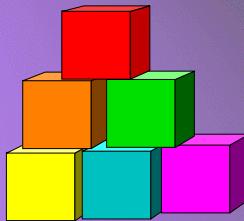
Notes:



Topics

CSC

Topics



- FPS Requirements
 - Transaction Definitions
 - Application Definition
 - FPS Control Entry for the Menu
 - Security Access
- Application Requirements
 - FPS Header Maps
 - Condition codes in Application Program

© 2000 CSC. All rights reserved

Objectives

CSC

Objectives



To become familiar with:

- Transactions needed for an FPS application
- Fields needed in an Application Definition
- Control entry needed by an application
- Application addition to a security control screen
- Map modification for an FPS application
- Application program modification for FPS communication

© 2000 CSC. All rights reserved



FPS Requirements

The four requirements for an application system to process under the FPS system are listed below:

CSC

FPS Requirements

- Three transaction definitions on the Process Dictionary
- An application definition entry for the application
- An FPS control entry defining the menu for the application
- An FPS security definition entry authorizing an operator/password combination to access the application

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

FPS Requirements

Transaction Definitions

Three transaction definitions must exist for an application which runs under the FPS system. The three transactions which are required must perform the following functions:

PROMPT	Display the menu screen or the last screen used.
WORK	Process the requested application function.
RESET	Re-initialize the CTF file for the current operator and display the menu.

Below are listed the three transaction codes for some applications running under FPS:

APPL ID	PROMPT RESUME	WORK	PROMPT RESET
CIS	HCIS	HCIT	HCIR
RPM	RPMP	RPMT	RPMR
TCS	TCS	TCST	TCSC
ILP	ILP	ILPT	ILPC
OTP	OTP	OTPL	OTPR
CDS	KD00	KD01	KDRS
IDS	IDSP	IDST	IDS R
UMB	UMB	UMBT	UMBR
DEM	DEMO	DEMT	DEMR
UES	UES	UEST	UESR

Notes:



Application Table Entry

Each application which runs under FPS must have an application table entry on CDMF. The application table defines the characteristics of the FPS application, including the Application ID of the application's transactions, a three-character application code which uniquely identifies the application, and the condition code set by the application programs to indicate the successful execution of the operator's request.

CSC **'Reset' Transaction Def.**

```
ACTION COMPLETE
----- UMBRELLA TRANSACTION DEFINITION INQUIRY/MAINTENANCE -----
COMMAND ==> BY TRAN CODE OWNER ==> TRD CC# ==>
APPLICATION ID ======> 99 ← Must match
FUNCTION ID ======> 103 FPS APP entry
SOURCE ID ======> 3
COMPANY ID LIST ======> ALL
EFFECTIVE DATE ======> 1978/01/01
TRANSACTION CODE ======> DBRR
TRANSACTION DESCRIPTION ==> TRAINING RESET TRANS
DL/I PSB NAME ======> PSSEDUA
DB2 PLAN NAME ======>
APPC: REMOTE PGM ==> CONVERSATION ==>
FORCE DG STORAGE AMODE 24 ==> NO (YES/NO)

TRANSACTION ACTIVITIES
* ACTIVITY * ACTIVITY * ACTIVITY
13587 13581

CHG:DATE 1989/11/20 TIME 17:06:09 CC# 1000 SRCE EDU9001 OPER GLT
PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-ACTS 11-TRAN -DEL -NEW
© 2001 CSC. All rights reserved
```

Notes:



Umbrella Programming

FPS Requirements

Control Definition for Main Menu

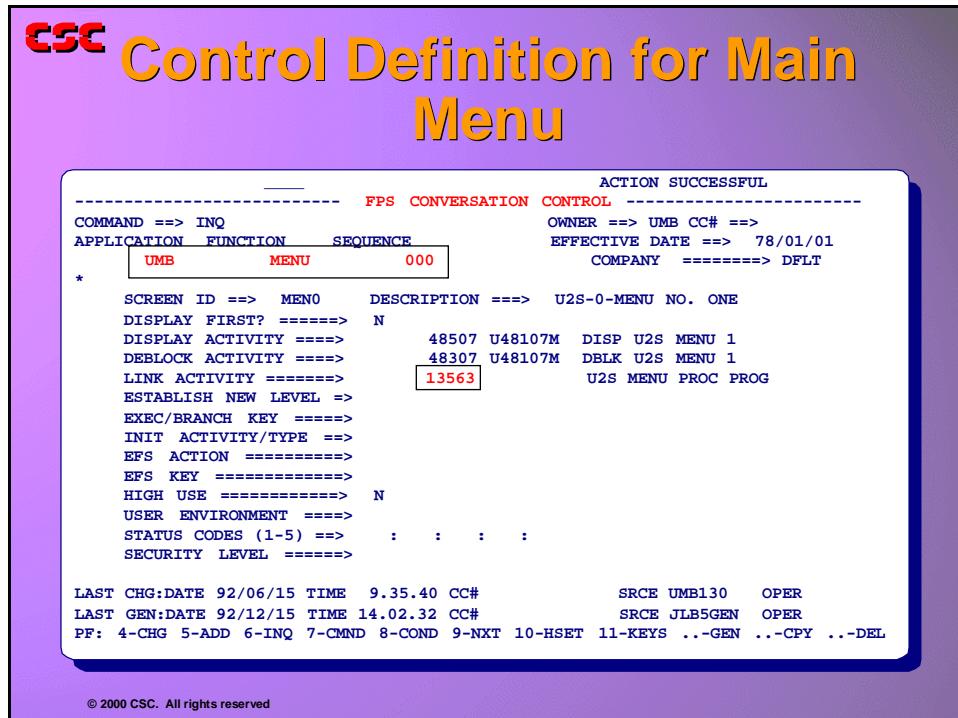
Each application executing under the FPS system must define a menu function (the first function) on an FPS Control entry. The item key for the MENU function must be in the following format:

AAAMENU000

where...

AAA = the alpha application code

The item key for the menu function must be in the above format because the FPS Supervisor program automatically builds the FPS control key for the menu function as seen above when an initial PROMPT or a RESET transaction is done.



Notes:



FPS Operator Security Control

Each application which runs under the FPS system must be referenced on at least one operator/password security record. The item key for the security record must be in the following format:

```
OOOOPPPP
where...
OOOO = Operator ID
PPPP = password
```

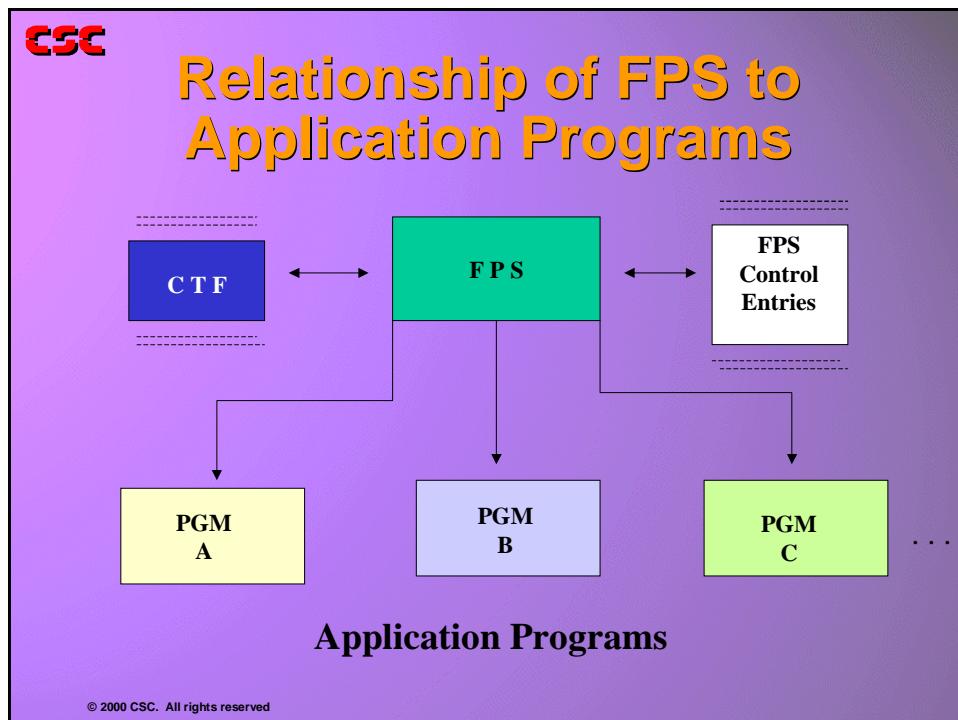
The security record lists every application which the operator/password combination is allowed to access and indicates the level of security associated with each application.

One delivered security record has a blank item key, indicating an Operator ID of blank and a password of blank. This record is the most logical one for adding your new application.

FPS Operator Security Control																																																																																				
ACTION SUCCESSFUL																																																																																				
----- FPS OPERATOR SECURITY CONTROL -----																																																																																				
COMMAND ==> NXT					CC# ==>																																																																															
OPERATOR ID ==>					EFFECTIVE DATE ==> 78/01/01																																																																															
PASSWORD =====>					COMPANY ID =====> DFLT																																																																															
E B C																																																																																				
<table border="0" style="width: 100%; border-collapse: collapse;"> <tr><td>* APP REC SENS FUNC SEC MAINT LVL M A M SEC</td><td>TERMINAL KEY ASSIGMNENT</td><td>OWNR</td></tr> <tr><td>123456789 123456789 0123456789 P L P LVL</td><td>E1...5...10...15...20...</td><td></td></tr> <tr><td>BOR YYYYYYYY YYYYYYYY</td><td>Y Y 3</td><td>PRCLDIHEAZZZZZZZZZZZZZZ BOR</td></tr> <tr><td>CDS YYYYYYYY YYYYYYYY</td><td>Y Y 3</td><td>PRCL E CDS</td></tr> <tr><td>CIS YYYYYYYY YYYYYYYY YYYYYYYY</td><td>Y Y Y</td><td>PRCLDIHEA RCLDIHEA CIS</td></tr> <tr><td>CLS YYYYYYYY YYYYYYYY</td><td>Y Y</td><td>PRCLDIHEAHZZZZZZZZZZZZZ CLS</td></tr> <tr><td>COL YYYYYYYY YYYYYYYY</td><td>Y Y 3</td><td>PRCLDIHEAZZZZZZZZZZZZZZ COL</td></tr> <tr><td>DEM NNNNNNNN YYYYYYYY</td><td>Y Y 3</td><td>PZLZBBBBBBBZAZLZZZZZZZZ INT</td></tr> <tr><td>FBS YYYYYYYY YYYYYYYY</td><td>Y Y 3</td><td>PACLZBBBBBBBZAZLZZZZZZZZ INT</td></tr> <tr><td>FIS YYYYYYYY YYYYYYYY YYYYYYYY</td><td>Y Y Y</td><td>PACLZBBBBBBBZACLDIHEZZZ INT</td></tr> <tr><td>FPS NNNNNNNN NNNNNNNN YYYYYYYY N Y N</td><td>3</td><td>PACLZBBBBBBBZAZLZZZZZZZZ FPS</td></tr> <tr><td>FSS YYYYYYYY YYYYYYYY</td><td>Y Y</td><td>PACLZBBBBBBBZACLZZZZZZZZ FSS</td></tr> <tr><td>IBA YYYYYYYY YYYYYYYY</td><td>Y Y 3</td><td>PACLZBBBBBBBZACLZZZZZZZZ FSS</td></tr> <tr><td>IDS NNNNNNNN NNNNNNNN</td><td>N N</td><td>PACLZBBBBBBBZACLZZZZZZZZ IDS</td></tr> <tr><td>LSM YYYYYYYY YYYYYYYY</td><td>Y Y</td><td>PRCLDIHEAX RCLDIHEAX LSM</td></tr> <tr><td colspan="10" style="text-align: center;">EMP=VIEW EMPLOYEE RECS BAL=VIEW BALANCES CMP=CROSS COMPANY BOUNDARIES</td></tr> <tr><td colspan="10" style="text-align: center;">LAST CHG:DATE 93/05/26 TIME 17.37.45 CC# 1 SRCE R002 OPER JLB</td></tr> <tr><td colspan="10" style="text-align: center;">PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT ..-DEL</td></tr> </table>										* APP REC SENS FUNC SEC MAINT LVL M A M SEC	TERMINAL KEY ASSIGMNENT	OWNR	123456789 123456789 0123456789 P L P LVL	E1...5...10...15...20...		BOR YYYYYYYY YYYYYYYY	Y Y 3	PRCLDIHEAZZZZZZZZZZZZZZ BOR	CDS YYYYYYYY YYYYYYYY	Y Y 3	PRCL E CDS	CIS YYYYYYYY YYYYYYYY YYYYYYYY	Y Y Y	PRCLDIHEA RCLDIHEA CIS	CLS YYYYYYYY YYYYYYYY	Y Y	PRCLDIHEAHZZZZZZZZZZZZZ CLS	COL YYYYYYYY YYYYYYYY	Y Y 3	PRCLDIHEAZZZZZZZZZZZZZZ COL	DEM NNNNNNNN YYYYYYYY	Y Y 3	PZLZBBBBBBBZAZLZZZZZZZZ INT	FBS YYYYYYYY YYYYYYYY	Y Y 3	PACLZBBBBBBBZAZLZZZZZZZZ INT	FIS YYYYYYYY YYYYYYYY YYYYYYYY	Y Y Y	PACLZBBBBBBBZACLDIHEZZZ INT	FPS NNNNNNNN NNNNNNNN YYYYYYYY N Y N	3	PACLZBBBBBBBZAZLZZZZZZZZ FPS	FSS YYYYYYYY YYYYYYYY	Y Y	PACLZBBBBBBBZACLZZZZZZZZ FSS	IBA YYYYYYYY YYYYYYYY	Y Y 3	PACLZBBBBBBBZACLZZZZZZZZ FSS	IDS NNNNNNNN NNNNNNNN	N N	PACLZBBBBBBBZACLZZZZZZZZ IDS	LSM YYYYYYYY YYYYYYYY	Y Y	PRCLDIHEAX RCLDIHEAX LSM	EMP=VIEW EMPLOYEE RECS BAL=VIEW BALANCES CMP=CROSS COMPANY BOUNDARIES										LAST CHG:DATE 93/05/26 TIME 17.37.45 CC# 1 SRCE R002 OPER JLB										PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT ..-DEL									
* APP REC SENS FUNC SEC MAINT LVL M A M SEC	TERMINAL KEY ASSIGMNENT	OWNR																																																																																		
123456789 123456789 0123456789 P L P LVL	E1...5...10...15...20...																																																																																			
BOR YYYYYYYY YYYYYYYY	Y Y 3	PRCLDIHEAZZZZZZZZZZZZZZ BOR																																																																																		
CDS YYYYYYYY YYYYYYYY	Y Y 3	PRCL E CDS																																																																																		
CIS YYYYYYYY YYYYYYYY YYYYYYYY	Y Y Y	PRCLDIHEA RCLDIHEA CIS																																																																																		
CLS YYYYYYYY YYYYYYYY	Y Y	PRCLDIHEAHZZZZZZZZZZZZZ CLS																																																																																		
COL YYYYYYYY YYYYYYYY	Y Y 3	PRCLDIHEAZZZZZZZZZZZZZZ COL																																																																																		
DEM NNNNNNNN YYYYYYYY	Y Y 3	PZLZBBBBBBBZAZLZZZZZZZZ INT																																																																																		
FBS YYYYYYYY YYYYYYYY	Y Y 3	PACLZBBBBBBBZAZLZZZZZZZZ INT																																																																																		
FIS YYYYYYYY YYYYYYYY YYYYYYYY	Y Y Y	PACLZBBBBBBBZACLDIHEZZZ INT																																																																																		
FPS NNNNNNNN NNNNNNNN YYYYYYYY N Y N	3	PACLZBBBBBBBZAZLZZZZZZZZ FPS																																																																																		
FSS YYYYYYYY YYYYYYYY	Y Y	PACLZBBBBBBBZACLZZZZZZZZ FSS																																																																																		
IBA YYYYYYYY YYYYYYYY	Y Y 3	PACLZBBBBBBBZACLZZZZZZZZ FSS																																																																																		
IDS NNNNNNNN NNNNNNNN	N N	PACLZBBBBBBBZACLZZZZZZZZ IDS																																																																																		
LSM YYYYYYYY YYYYYYYY	Y Y	PRCLDIHEAX RCLDIHEAX LSM																																																																																		
EMP=VIEW EMPLOYEE RECS BAL=VIEW BALANCES CMP=CROSS COMPANY BOUNDARIES																																																																																				
LAST CHG:DATE 93/05/26 TIME 17.37.45 CC# 1 SRCE R002 OPER JLB																																																																																				
PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT ..-DEL																																																																																				
© 2000 CSC. All rights reserved																																																																																				



Relationship of FPS to Application Programs



Notes:



Application Requirements

Introduction

There are four means by which communication can be established between the application program and FPS.

1. FPS Communication Data Group - DG 13589
2. Through extensive common usage of condition codes
3. Condition codes defined on PCD 1301
4. Conversational Terminal File, or CTF

Of these four means, the first two place requirements on the application system. CTF manipulation and maintenance is handled exclusively by FPS. The maintenance of condition codes on PCD 1301 is an application responsibility, but it is optional.

This section discusses the application's use of FPS header maps as a tool to house information in data group 13589. It also addresses the setting of condition codes within the application program to indicate the outcome of the program's processing.

Notes:



Umbrella Programming

Application Requirements

Communications Data Group 13589

```
DATA GROUP          COPYBOOK NAME I53589D

***** START OF I53589D ***** FPS COMMON COMMUNICATION DG      *****
*
*     DATA GROUP NUMBER    13589
*
*****
*
01  W-FPS-COMM-DATA-GROUP .
    05  W-ACTION-I53589D           PIC XX.
    05  W-RESULT-I53589D          PIC XX.
*
    05  W-FPS-COMM-REQUEST-TYPE   PIC X.
        88  W-FPS-COMM-REQUEST-PROMPT      VALUE 'P'.
        88  W-FPS-COMM-REQUEST-PROCESS     VALUE ''.
    05  W-FPS-COMM-APPL-FUNC .
        10  W-FPS-COMM-APPL-CD       PIC X(3).
        10  W-FPS-COMM-ORIG-FUNC      PIC X(4).
    05  W-FPS-COMM-NEXT-CMND      PIC X(4).
        88  W-FPS-COMM-NEXT-CMND-END    VALUE 'END '.
        88  W-FPS-COMM-NEXT-CMND-RTN    VALUE 'RTN '.
        88  W-FPS-COMM-NEXT-CMND-CONT   VALUE 'CONT' .
        88  W-FPS-COMM-NEXT-CMND-PREV-LVL  VALUE 'PLVL' .
        88  W-FPS-COMM-NEXT-CMND-ABND    VALUE 'ABND' .
        88  W-FPS-COMM-NEXT-CMND-RSET    VALUE 'RSET' .
        88  W-FPS-COMM-NEXT-CMND-HOLD-SCRN  VALUE 'HOLD' .
        88  W-FPS-COMM-NEXT-CMND-HELP     VALUE 'HELP' .
        88  W-FPS-COMM-NEXT-CMND-DFLT     VALUE 'DFLT' .
        88  W-FPS-COMM-NEXT-CMND-PRNT-IT   VALUE 'PRT '
                                         THRU 'PRT9'.
        88  W-FPS-COMM-NEXT-CMND-TRACE-ON  VALUE 'TRON' .
        88  W-FPS-COMM-NEXT-CMND-TRACE-OFF  VALUE 'TROF' .
        88  W-FPS-COMM-NEXT-CMND-TEST      VALUE 'TEST' .
        88  W-FPS-COMM-NEXT-CMND-PROD      VALUE 'PROD' .
        88  W-FPS-COMM-NEXT-CMND-INCR     VALUE 'INCR' 'I'.
        88  W-FPS-COMM-NEXT-CMND-DECR     VALUE 'DECR' 'D'.
        88  W-FPS-COMM-NEXT-CMND-EXIT     VALUE 'EXIT' .
        88  W-FPS-COMM-NEXT-CMND-STAT     VALUE 'STAT' .
        88  W-FPS-COMM-NEXT-CMND-TIME     VALUE 'TIME' .
        88  W-FPS-COMM-NEXT-CMND-UNFMT    VALUE 'UFMT' .
    05  FILLER REDEFINES          W-FPS-COMM-NEXT-CMND.
        10  FILLER                  PIC X(1).
            88  NEXT-CMND-MENU      VALUE '='.
        10  FILLER                  PIC X(3).
    05  W-FPS-COMM-CURR-SCRN-ID    PIC X(4).
    05  W-FPS-COMM-SYS-DATE      PIC 9(7) COMP-3.
    05  W-FPS-COMM-TIME         PIC 9(7) COMP-3.
    05  W-FPS-COMM-APPL-OP-ID    PIC X(4).
    05  W-FPS-COMM-PSWD         PIC X(4).
    05  W-FPS-COMM-CTF-KEY-INFO .
        10  W-FPS-COMM-LAST-OP-ID    PIC X(4).
        10  W-FPS-COMM-ORIG-APPL-ID  PIC 99.
    05  W-FPS-COMM-MODE          PIC X(3).
        88  W-FPS-COMM-MODE-PROD     VALUE 'PRD'
                                         ' '.
        88  W-FPS-COMM-MODE-TEST    VALUE 'TST' .
```



Umbrella Programming

Application Requirements

```

05 W-FPS-COMM-BASE-DG-FND-IND    PIC X.
     88 W-FPS-COMM-BASE-DG-FND          VALUE 'Y'.
05 W-FPS-COMM-MSG-COND-CODE      PIC XX.
05 W-FPS-COMM-MSG                  PIC X(30).
05 FILLER REDEFINES W-FPS-COMM-MSG.
     10 FILLER                      PIC X.
     10 W-FPS-COMM-MSG-FUNC-NO       PIC 9.
     10 FILLER                      PIC XX.
     10 W-FPS-COMM-MSG-EX-LVL       PIC 9.
     10 FILLER                      PIC X.
     10 W-FPS-COMM-MSG-APPL-CD      PIC X(3).
     10 FILLER                      PIC X(6).
     10 W-FPS-COMM-MSG-SCRN-ID      PIC X(4).
     10 FILLER                      PIC X(6).
     10 W-FPS-COMM-MSG-ORIG-FUNC    PIC X(4).
     10 FILLER                      PIC X.
05 FILLER REDEFINES W-FPS-COMM-MSG.
     10 FILLER                      PIC X(5).
     10 W-FPS-COMM-MSG-TIME         PIC 999999.
     10 FILLER                      PIC X(6).
     10 W-FPS-COMM-MSG-DATE         PIC 9999999.
     10 FILLER                      PIC X(6).
05 W-FPS-COMM-PER-MSG            PIC X(60).
     88 W-FPS-COMM-PER-MSG-CLEAR    VALUE '*'.
05 W-FPS-COMM-SYS-MSG            PIC X(60).
05 W-FPS-COMM-SYS-MSG-ACTN      PIC X.
     88 W-FPS-COMM-SYS-MSG-ADD      VALUE 'A'.
     88 W-FPS-COMM-SYS-MSG-CHG      VALUE 'C'.
     88 W-FPS-COMM-SYS-MSG-DEL      VALUE 'D'.
05 W-FPS-COMM-RTN-FROM-DISP-FLAG PIC X.
     88 W-FPS-COMM-RTN-FROM-DISP    VALUE 'Y'.
05 W-FPS-COMM-COND-CODE-APP     PIC XX.
*
*   05 W-FPS-COMM-PGM-FUNC-KEY      PIC X.
*   SUPV PROCESS CMND IF ENTERED, ELSE CURR SCREEN
*   88 W-FPS-COMM-PGM-PROCESS-CMND    VALUE 'P'.
*   SUPV--INCREASE FUNCTION NUMBER
*   88 W-FPS-COMM-PGM-INCR-FUNC-NO    VALUE 'I'.
*   SUPV--DECREASE FUNCTION NUMBER
*   88 W-FPS-COMM-PGM-DECRL-FUNC-NO   VALUE 'D'.
*   SUPV--RETURN TO PREV PROC LEVEL
*   88 W-FPS-COMM-PGM-RTN-PREV-LVL    VALUE 'L'.
*   BOTH--RETURN TO PREV PROCESSING POINT
*   88 W-FPS-COMM-PGM-RTN-BACK      VALUE 'R'.
*   PROG--HOLD CURR SCRNL AFTER PROCESSING
*   88 W-FPS-COMM-PGM-HOLD-SCRN      VALUE 'H'.
*   BOTH--CONTINUE ON IF SCREEN ALLOWS
*   88 W-FPS-COMM-PGM-CONTINUE-ON    VALUE 'C'.
*   SUPV--PROC CURR SCRNL, THEN EXIT TO CMND
*   88 W-FPS-COMM-PGM-EXIT-AFTER    VALUE 'X'.
*   SUPV--END CURRENT FUNCTION
*   88 W-FPS-COMM-PGM-END-FUNC      VALUE 'E'.
*   PROG--BATCH PROCESSING
*   88 W-FPS-COMM-PGM-FUNC-BATCH    VALUE 'B'.
*   SUPV--HELP REQUEST
*   88 W-FPS-COMM-PGM-FUNC-HELP      VALUE 'A'.
*   BOTH--FUNCTION EXIT TO MENU
*   88 W-FPS-COMM-PGM-FUNC-EXIT      VALUE 'F'.
*   FUNC KEYS WHICH DO NOT REQUIRE PSWD

```



Umbrella Programming

Application Requirements

```
      88 W-FPS-COMM-PGM-BYPASS-PSWD      VALUE 'B'.
*     FUNC KEYS DEFINED ON CDMF ID 48323
      88 W-FPS-COMM-PGM-FUNC-DEFER      VALUE 'Z'.
  05 W-FPS-COMM-STATUS .
    10 W-FPS-COMM-STATUS-1      PIC X.
    10 W-FPS-COMM-STATUS-2      PIC X.
    10 W-FPS-COMM-STATUS-3      PIC X.
    10 W-FPS-COMM-STATUS-4      PIC X.
    10 W-FPS-COMM-STATUS-5      PIC X.
      88 W-FPS-COMM-STATUS-5-PROD      VALUE 'P'.
      88 W-FPS-COMM-STATUS-5-TEST      VALUE 'T'.
  05 W-FPS-COMM-FUNC-EXIT-FLAG      PIC X.
    88 W-FPS-COMM-FUNC-EXIT      VALUE 'Y'.
  05 W-FPS-COMP-VALID-IND      PIC X(1).
  05 W-FPS-TEST-VALID-MODE      PIC X(1).
  05 W-FPS-REQ-APPL      PIC X(3).
  05 W-FPS-COMM-FUNC-BEGIN-SEQ      PIC X(3).
  05 W-FPS-COMM-FUNC-NO      PIC 9      COMP-3.
  05 W-FPS-COMM-RESULT      PIC XX.
  05 W-FPS-COMM-INIT-RESULT      PIC XX.
  05 FILLER      PIC X(6).
  05 W-FPS-COMM-APPL-GLBL-STATS .
    10 W-FPS-COMM-APPL-GLBL-STAT1 PIC X.
      88 W-FPS-COMM-APPL-GLBL-DG1-PRES      VALUE 'Y'.
      88 W-FPS-COMM-APPL-GLBL1-RESTORE      VALUE 'R'.
    10 W-FPS-COMM-APPL-GLBL-STAT2 PIC X.
      88 W-FPS-COMM-APPL-GLBL-DG2-PRES      VALUE 'Y'.
      88 W-FPS-COMM-APPL-GLBL2-RESTORE      VALUE 'R'.
    10 W-FPS-COMM-APPL-GLBL-STAT3 PIC X.
      88 W-FPS-COMM-APPL-GLBL-DG3-PRES      VALUE 'Y'.
      88 W-FPS-COMM-APPL-GLBL3-RESTORE      VALUE 'R'.
    10 W-FPS-COMM-APPL-GLBL-STAT4 PIC X.
      88 W-FPS-COMM-APPL-GLBL-DG4-PRES      VALUE 'Y'.
      88 W-FPS-COMM-APPL-GLBL4-RESTORE      VALUE 'R'.
  05 FILLER      PIC XX.
  05 W-FPS-COMM-MSG-CC-APP-PREV      PIC XX.
  05 W-FPS-COMM-CURR-EX-LVL      PIC S9(4) COMP.
  05 W-FPS-COMM-CO-ID      PIC XX.
  05 FILLER      PIC XX.
  05 FILLER      PIC XX.
  05 W-FPS-COMM-01690-READ-CNT      PIC S9(4) COMP.
  05 W-FPS-COMM-HALFWORD-WORK-X.
    10 W-FPS-COMM-HALFWORD-WORK PIC S9(4) COMP.
  05 W-FPS-COMM-FULLWORD-WORK      PIC S9(9) COMP.
  05 W-FPS-COMM-FULLWORD-WORK-X REDEFINES
      W-FPS-COMM-FULLWORD-WORK.
    10 W-FPS-COMM-FULLWORD-BYTE-1-2 PIC XX.
    10 W-FPS-COMM-FULLWORD-BYTE-3-4 PIC XX.
  05 W-FPS-COMM-MSG-COND-CODE-PREV      PIC XX.
  05 W-FPS-COMM-SCRN-ID-PREV      PIC X(4).
  05 W-FPS-COMM-RCD-SENS-INDS .
    10 W-FPS-COMM-RCD-SENS      PIC X      OCCURS 09 TIMES
      INDEXED BY W-FPS-COMM-SENS-INDX.
      88 W-FPS-COMM-RCD-SENS-OK      VALUE 'Y'.
  05 W-FPS-COMM-VIEW-BAL      PIC X.
    88 W-FPS-COMM-VIEW-BAL-OK      VALUE 'Y'.
  05 W-FPS-COMM-EMPL-RCDS      PIC X.
    88 W-FPS-COMM-EMPL-RCDS-OK      VALUE 'Y'.
  05 W-FPS-COMM-CROSS-COMP      PIC X.
```



Umbrella Programming

Application Requirements

```

      88 W-FPS-COMM-CROSS-COMP-OK          VALUE 'Y'.
05  W-FPS-COMM-MAINT-IND             PIC X.
      88 W-FPS-COMM-MAINT-OK          VALUE 'Y'.
05  W-FPS-COMM-FUNC-LVL            PIC 9   COMP-3.
05  W-FPS-COMM-OPER-LVL            PIC 9   COMP-3.
*
*          05 W-FPS-COMM-ACTION-CD          PIC X(40).
05  FILLER REDEFINES           W-FPS-COMM-ACTION-CD.
      10 W-FPS-COMM-ACTION-CD-1-1 PIC X(1).
          88 MENU-COMMAND          VALUE '='.
          88 DELIMETER            VALUE ',', '.', '/', ''
                                      SPACE.
      10 W-FPS-COMM-ACTION-CD-2-40 PIC X(39).
05  FILLER REDEFINES           W-FPS-COMM-ACTION-CD.
      10 W-FPS-COMM-ACTION-CD-1-4 PIC X(4).
          88 VALID-FPS-ACTION        VALUE 'END'
                                      'RTN' 'CONT' 'PLVL' 'ABND' 'EXIT' 'HOLD' 'HELP'
                                      'DFLT' 'TRON' 'TROF' 'TEST' 'PROD' 'INCR' 'DECR'
                                      'STAT' 'TIME' 'I' 'D' 'PRT' 'THRU' 'PRT9'.
      10 W-FPS-COMM-ACTION-CD-5-40 PIC X(36).
05  FILLER REDEFINES           W-FPS-COMM-ACTION-CD.
      10 W-FPS-COMM-ACTION-CD-1-6 PIC X(6).
          10 FILLER                PIC X(34).
05  W-FPS-COMM-ACTION-CD-WK40    PIC X(40).
05  FILLER REDEFINES           W-FPS-COMM-ACTION-CD-WK40.
          10 FILLER                PIC X(1).
              88 MENU-COMMAND-WK        VALUE '='.
      10 W-FPS-COMM-ACTION-CD-WK40-2-40 PIC X(39).
05  W-FPS-COMM-ACTION-CD-WK20    PIC X(20).
05  W-FPS-COMM-FORMAT-ID.
          10 FILLER                PIC XX.
          10 W-FPS-COMM-FORMAT-ID-H PIC XX.
*
*          05 W-FPS-COMM-PFKEY-DISPLAY.
      10 W-FPS-COMM-PFKEY-START      PIC X(4).
      10 W-FPS-COMM-PFKEY-DISPLAY-TABLE
                                      OCCURS 75 TIMES
                                      INDEXED BY W-FPS-COMM-PFKEY-INDEX
                                      PIC X.
*
*          05 W-FPS-COMM-PFKEY-WORK-AREA.
      10 W-FPS-COMM-PFKEY-VALUE.
          15 W-FPS-COMM-PFKEY-VALUE-N PIC 99.
      10 W-FPS-COMM-PFKEY-HYPHEN     PIC X.
      10 W-FPS-COMM-PFKEY-ACTION     PIC X(6).
          10 FILLER                PIC X.
05  W-FPS-COMM-PFKEY-WORK-TABLE
REDEFINES W-FPS-COMM-PFKEY-WORK-AREA.
      10 W-FPS-COMM-PFKEY-WORK-BYTE
                                      OCCURS 10 TIMES
                                      INDEXED BY W-FPS-COMM-WORK-INDEX
                                      PIC X.
*
*          FIELDS THAT USED TO BE HW ARE NOW FW HERE
05  W-FPS-COMM-FUNC-INIT-ACTY    PIC XXXX.
05  W-FPS-COMM-APPL-GLBL-DGS.
          10 W-FPS-COMM-APPL-GLBL-DG1 PIC XXXX.
          10 W-FPS-COMM-APPL-GLBL-DG2 PIC XXXX.
          10 W-FPS-COMM-APPL-GLBL-DG3 PIC XXXX.

```



Umbrella Programming

Application Requirements

```
10  W-FPS-COMM-APPL-GLBL-DG4 PIC XXXX.  
05  W-FPS-COMM-APPL-STAT-ACTY      PIC XXXX.  
05  W-FPS-COMM-APPL-SECUR-ACTY      PIC XXXX.  
05  W-FPS-COMM-PRE-FPS-EXIT        PIC XXXX.  
05  W-FPS-COMM-POST-FPS-EXIT       PIC XXXX.  
05  W-FPS-COMM-PREV-APPL-FUNC.  
   10  W-FPS-COMM-PREV-APPL-CD    PIC X(3).  
   10  W-FPS-COMM-PREV-FUNC       PIC X(4).  
  
*  
*          EVENT TRACKER NAVIGATION FIELDS  
05  W-FPS-COMM-ET-CAPT-FF         PIC X.  
05  W-FPS-COMM-ET-EV-TYP         PIC X(08).  
05  W-FPS-MONETARY-KEY          PIC X(3).  
05  W-FPS-PRES-CURRENCY-CD      PIC X(3).  
W589PCUR  
05  W-FPS-PRES-CURR-RND-IN      PIC X(1).  
W589PCRI  
* DO NOT ROUND PRESENTATION CURRENCY - ADDS AN EXTRA DECIMAL DIGIT  
  88  W-FPS-DO-NOT-ROUND-PRES-CURR           VALUE 'N'.  
* ROUND PRESENTATION CURRENCY  
  88  W-FPS-ROUND-PRES-CURR                 VALUE 'Y'.  
*  
05  FILLER                         PIC X(13).  
***** END OF I53589D *****
```

Notes:



Common Referenced Fields - DG13589



Most Frequently Read Fields by Application Programs

- W-FPS-COMM-NEXT-CMND
- W-FPS-COMM-PGM-FUNC-KEY
- W-FPS-COMM-RESULT
- W-FPS-COMM-CO-ID
- W-FPS-COMM-ACTION-CD

© 2000 CSC. All rights reserved

Common Modified Fields - DG13589



Most Frequently Updated Fields by Application Programs

- W-FPS-COMM-APP-CD
- W-FPS-COMM-NEXT-CMND
- W-FPS-COMM-FUNC-BEGIN-SEQ
- W-FPS-COMM-STATUS

© 2000 CSC. All rights reserved



Umbrella Programming

Application Requirements

FPS Header Maps

The FPS system communicates with the terminal operator through key fields of information such as requested function, processing level, Application ID, conversation number and company number. This key information is stored in core in the FPS Common Communication data group (data group 13589, copy book I53589D). This information is passed to the FPS system through the entry of the data in the FPS header lines on the application screens.

FPS uses the technique of "double deblocking" on all maps. The header lines for the application are deblocked by PEM as part of the transaction definition before the FPS supervisor is invoked. These header lines may appear different from application to application but must contain the same information.

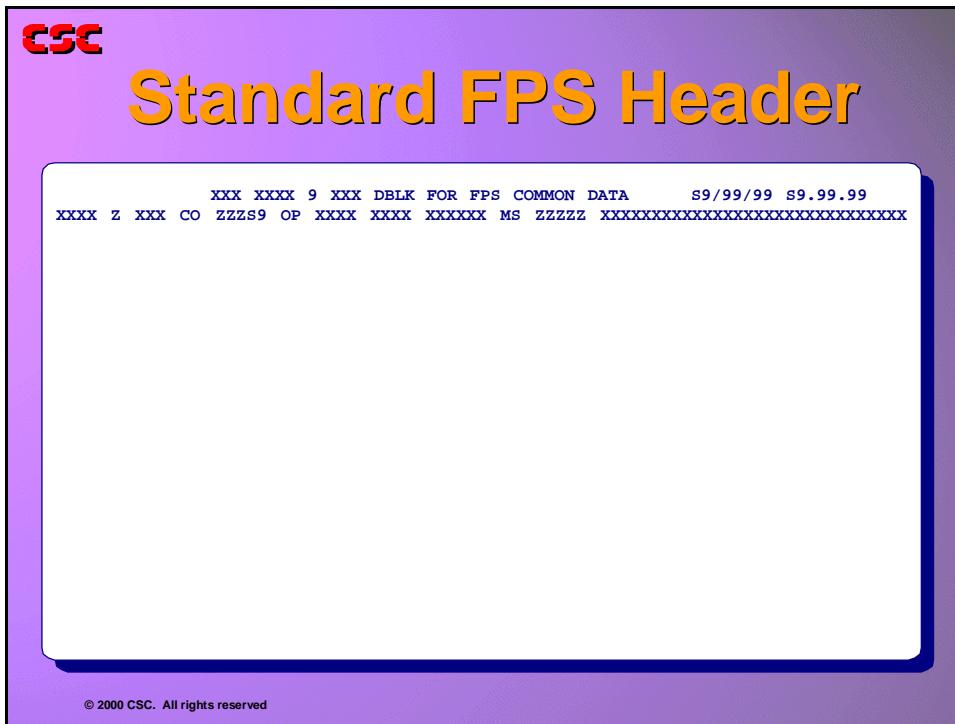
This key information is required for the FPS system to function and thus all applications screen maps must include one of the following:

1. The standard FPS header maps - I53589E and I53589F (APP# thru 99).
2. The standard FPS header maps - I53589E and I53589G (APP# over 99).
3. Maps which are user-designed and perform the same function as the standard FPS maps.

Notes:



I53589MA - Standard FPS Header



Used by several applications running under FPS.

Notes:



Umbrella Programming

Application Requirements

CSC

Map Definition Base

```
ACTION COMPLETE
----- MAP DEFINITION BASE SCREEN -----
COMMAND ==> INQ OWNER ==> UMB CC# ==>
LANGUAGE => ENU
MAP NAME => I53589M DEVICE => A CO GROUP => ALL EFF DATE => 78/01/01

DESCRIPTION =====> FPS STANDARD HEADER--TOP 2 LINES :
LINKNAME =====> I53589MM
TYPE (I,O,U,S) =====> U TRANCODE DYNAMIC? =====> N
TRANCODE =====> TRANCODE PROTECTED? ===> N
EXTENDED ATTRIBUTES? ==> N NON-PEM MAP? =====> N
MAP TYPE (DOC|SCR) =====> SCR MAXIMUM ROWS => 24 MAXIMUM CLMS => 80

EXTRA FPS DATA GROUPS
* DGID * DGID * DGID * DGID * DGID * DGID

NEW MAP KEY FOR COPY:
MAP NAME => DEVICE => CO GROUP => EFF DATE =>
LANGUAGE =>
LAST CHG:DATE 94/11/07 TIME 19.22.55 CC# SRCE UMB200 OPER
LAST GEN:DATE 95/02/12 TIME 13.20.49 CC# SRCE UMB200E6 OPER
PF: 4-CHG 5-ADD 6-INQ 7-BROWSE 8-EDIT 9-NXT 10-GEN 11-DSP ..-DOC ..-CPY ..-GRP
```

© 2000 CSC. All rights reserved

CSC

Map Definition Screen

```
ACTION COMPLETE
----- MAP DEFINITION SCREEN ----- MAX-ORIGIN=( 24 , 80 )
ORIGIN=( 1 , 1 ) MODE==> BRWS
COMMAND ROW CLM LEN (THRU ROW) TO/AT ROW CLM
-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6-----+-----7-----+
001 II53589E "DBLK FOR FPS COMMON DATA "II53589F
002
003
004
005
006
007
008
009
010
011
012
013
014
015
016
-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6-----+-----7-----+
PF: 3-PLVL 6-FDM2 7-SB 8-SF 9-FDM 10-SL 11-SR
```

© 2000 CSC. All rights reserved



I53589EA - Standard FPS 1st Include

CSC

I53589EA - Standard FPS 1st Include

ACTION COMPLETE							
MAP DEFINITION BASE SCREEN				OWNER ==> UMB CC# ==>			
COMMAND ==> INQ	DEVICE => A	CO GROUP => ALL	EFF DATE => 78/01/01	LINKNAME ==> INCLUDE	TRANCODE DYNAMIC? ==> N	TRANCODE PROTECTED? ==> N	NON-PEM MAP? ==> N
LANGUAGE => ENU	MAP NAME => I53589E	EXTENDED ATTRIBUTES? ==> N	MAP TYPE (DOC SCR) ==> SCR	MAXIMUM ROWS => 24	MAXIMUM CLMS => 80		
EXTRA FPS DATA GROUPS							
*	DGID	*	DGID	*	DGID	*	DGID
NEW MAP KEY FOR COPY:							
MAP NAME =>	DEVICE =>	CO GROUP =>	EFF DATE =>	LAST CHG:DATE 94/11/07 TIME 19.22.55 CC#	SRCE UMB200 OPER	LAST GEN:DATE TIME CC#	SRCE OPER
LANGUAGE =>	PF: 4-CHG 5-ADD 6-INQ 7-BROWSE 8-EDIT 9-NXT 10-GEN 11-DSP ...DOC ...CPY ...GRP						

© 2000 CSC. All rights reserved

CSC

Map Definition Screen

ACTION COMPLETE							
ORIGIN=(1 , 1) ---- MAP DEFINITION SCREEN ----- MAX-ORIGIN=(24 , 80)				MODE==> BRWS			
-----+---1---+---2---+---3---+---4---+---5---+---6---+---7---+							
001	XXXX	XXXX	9	XXX	011	013	015
002							
003							
004							
005							
006							
007							
008							
009							
010							
012							
014							
016							
-----+---1---+---2---+---3---+---4---+---5---+---6---+---7---+							
PF: 3-PLVL 6-FDM2 7-SB 8-SF 9-FDM 10-SL 11-SR							

© 2000 CSC. All rights reserved



Umbrella Programming

Application Requirements

I53589FA - Standard FPS 2nd Include(app<=99)

CSC 153589FA - Standard FPS 2nd Include (Applications 1 - 99)

```
ACTION COMPLETE
----- MAP DEFINITION BASE SCREEN -----
COMMAND ==> INQ OWNER ==> UMB CC# ==>
LANGUAGE ==> ENU
MAP NAME ==> I53589F DEVICE => A CO GROUP => ALL EFF DATE => 78/01/01
DESCRIPTION =====> FPS 2ND INCLUDE :
LINKNAME =====> INCLUDE
TYPE (I,O,U,S) =====> TRANCODE DYNAMIC? =====> N
TRANCODE =====> TRANCODE PROTECTED? ===> N
EXTENDED ATTRIBUTES? ==> N NON-PEM MAP? =====> N
MAP TYPE (DOC|SCR) ==> SCR MAXIMUM ROWS => 24 MAXIMUM CLMS => 80
EXTRA FPS DATA GROUPS
* DGID * DGID * DGID * DGID * DGID * DGID
```

NEW MAP KEY FOR COPY:

```
MAP NAME => DEVICE => CO GROUP => EFF DATE =>
LANGUAGE ==>
LAST CHG:DATE 94/11/07 TIME 19.22.55 CC# SRCE UMB200 OPER
LAST GEN:DATE TIME CC# SRCE OPER
PF: 4-CHG 5-ADD 6-INQ 7-BROWSE 8-EDIT 9-NXT 10-GEN 11-DSP ..-DOC ..-CPY ..-GRP
```

ORIGFUNC-Alphabetic, Length=4

© 2000 CSC. All rights reserved

CTFKEY-Alphabetic, Length=6

CSC

Map Definition Screen

```
ACTION COMPLETE
----- MAP DEFINITION SCREEN ----- MAX-ORIGIN=( 24 , 80 )
ORIGIN=( 1 , 1 ) ---- COMMAND ROW CLM LEN (THRU ROW) TO/AT ROW CLM MODE==> BRWS
-----+---1---+---2---+---3---+---4---+---5---+---6---+---7---+
001 .00 S9.
002 XXXX Z XXX"CO"ZZZS9"OP"XXXX XXXX XXXXXX"MS"ZZZZ XXXXXXXXXXXXXXXXXXXXXXX
003
004
005
006
007
008
009
010
011
012
013
014
015
016
-----+---1---+---2---+---3---+---4---+---5---+---6---+---7---+
PF: 3-PLVL 6-FDM2 7-SB 8-SF 9-FDM 10-SL 11-SR
```

© 2000 CSC. All rights reserved



I53589GA - Standard FPS 2nd Include(app>99)

CSC

153589GA - Standard FPS 2nd Include (Applications 100-65535)

```

----- ACTION COMPLETE -----
----- MAP DEFINITION BASE SCREEN -----
COMMAND ==> INQ OWNER ==> FPS CC# ==>
LANGUAGE => ENU
MAP NAME => I53589G DEVICE => A CO GROUP => ALL EFF DATE => 78/01/01
DESCRIPTION =====> FPS 2ND INCLUDE (APP ID BINARY) :
LINKNAME =====> INCLUDE
TYPE (I,O,U,S) =====> TRANCODE DYNAMIC? =====> N
TRANCODE =====> TRANCODE PROTECTED? ===> N
EXTENDED ATTRIBUTES? ==> N NON-PEM MAP? =====> N
MAP TYPE (DOC|SCR) ==> SCR MAXIMUM ROWS => 24 MAXIMUM CLMS => 80
EXTRA FPS DATA GROUPS
* DGID * DGID * DGID * DGID * DGID * DGID

NEW MAP KEY FOR COPY:
MAP NAME => DEVICE => CO GROUP => EFF DATE =>
LANGUAGE =>
LAST CHG:DATE 94/11/07 TIME 19.22.55 CC# SRCE UMB200 OPER
LAST GEN:DATE TIME CC# SRCE OPER
PF: 4-CHG 5-ADD 6-INQ 7-BROWSE 8-EDIT 9-NXT 10-GEN 11-DSP ..-DOC ..-CPY ..-GRP

```

BINAPPNM-Numeric, Length = 5

CTFKEY-Alphabetic, Length = 4

© 2000 CSC. All rights reserved

CSC

Map Definition Screen

```

----- ACTION COMPLETE -----
ORIGIN={ 1 , 1 } ---- MAP DEFINITION SCREEN ---- MAX-ORIGIN={ 24 , 80 }
COMMAND ROW CLM LEN (THRU ROW) TO/AT ROW CLM MODE==> BRWS
-----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+
001 .00 59.
002 ZZZZZZ Z XXX*CO*22259*0P*XXX XXXX XXXX* MS*ZZZZZ XXXXXXXXXXXXXXXXXX
003
004
005
006
007
008
009
010
011
012
013
014
015
016
-----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+
PF: 3-PLVL 6-FDM2 7-SB 8-SF 9-FDM 10-SL 11-SR

```

© 2000 CSC. All rights reserved



Umbrella Programming

Application Requirements

Setting of Condition Codes

It was discussed earlier that the application systems communicate with FPS through the use of condition codes. The application programs check to see if special conditions exist and then set the appropriate condition code. The applications set the condition code in the TCB-USER-CC variable in the Transaction Control Block (TCB). The TCB is data group number 1 and is defined by copybook P49000D.

Following is an example of an application program setting the condition code for FPS.

```
*****
*          I D E N T I F I C A T I O N      D I V I S I O N      *
*****
IDENTIFICATION DIVISION.
PROGRAM-ID.      Z47350.
AUTHOR.          HOGAN ASSOCIATES.
DATE-COMPILED.
REMARKS.
.
.
.

*****
*          D A T A      D I V I S I O N      *
*****
DATA DIVISION.
FILE SECTION.
WORKING-STORAGE SECTION.
SKIP1
01  FILLER          PIC X(8)
      VALUE 'Z47350' .
.
.

01  CF-CONSTANTS .
05  CF-CALL-CCP-NO-ACTS .
    10  FILLER          PIC S9(8) COMP
        VALUE +1398.
05  CF-PROCESS-EMP-RANDOM .
    10  FILLER          PIC S9(8) COMP
        VALUE +47360.
05  CF-PROCESS-EMP-SEQ .
    10  FILLER          PIC S9(8) COMP
        VALUE +47362.
05  CF-CALL-CDMF .
    10  FILLER          PIC S9(8) COMP
        VALUE +48000.
.
.
```



Umbrella Programming

Application Requirements

```
          01 COND-CODES.
05 FILLER                           PIC S9(8) COMP
          VALUE +47350.
05 FILLER                           PIC S9(8) COMP
          VALUE +47351.
05 FILLER                           PIC S9(8) COMP
          VALUE +47352.
05 FILLER                           PIC S9(8) COMP
          VALUE +47353.
05 FILLER                           PIC S9(8) COMP
          VALUE +47354.

.
.
.
01 FILLER REDEFINES COND-CODES.
* INVALID ACTION REQUESTED
05 FILLER                           PIC X(2).
05 COND-CODE-47350                  PIC X(2).
* EMPLOYEE DISPLAYED SUCCESSFULLY
05 FILLER                           PIC X(2).
05 COND-CODE-47351                  PIC X(2).
* EMPLOYEE NOT FOUND
05 FILLER                           PIC X(2).
05 COND-CODE-47352                  PIC X(2).
* UNDETERMINED ERROR
05 FILLER                           PIC X(2).
05 COND-CODE-47353                  PIC X(2).
* NO MORE EMPLOYEES
05 FILLER                           PIC X(2).
05 COND-CODE-47354                  PIC X(2).

.
.
.

*****
*          P R O C E D U R E      D I V I S I O N      *
*****
SKIP1
PROCEDURE DIVISION USING TRANSACTION-CONTROL-BLOCK
.

.

.

SKIP1
AA000-INITIALIZATION SECTION.
SKIP1
MOVE LOW-VALUES TO TCB-USER-CC.
SKIP1
AB000-MAINLINE SECTION.
SKIP1
IF W010-INQ-ACTION
    PERFORM BA000-PERFORM-INQUIRY
ELSE
    IF W010-ADD-ACTION
        PERFORM BB000-PERFORM-ADD
```



Umbrella Programming

Application Requirements

```
ELSE
    IF W010-CHG-ACTION
        PERFORM BC000-PERFORM-UPDATE
    ELSE
        IF W010-ACTION EQUAL CC-DEL
            PERFORM BD000-PERFORM-DELETE
        ELSE
            MOVE COND-CODE-47350 TO TCB-USER-CC.

SKIP1
AC000-FINALIZATION SECTION.
SKIP1
PERFORM ZZ000-END-OF-PROCESSING.
STOP RUN.
EJECT

.
.

*****
* THIS SECTION PERFORMS AN INQUIRY FOR EITHER A SPECIFIC      *
* EMPLOYEE OR NEXT EMPLOYEE IN THE DATA BASE DEPENDING ON      *
* THE ACTION REQUESTED BY THE OPERATOR ('INQ' OR 'NXT').       *

*****
SKIP1
BA000-PERFORM-INQUIRY SECTION.
SKIP1
PERFORM CD000-READ-EMP.
SKIP1
IF TCB-RESULT EQUAL TCB-OK
    MOVE COND-CODE-47351 TO TCB-USER-CC
    PERFORM CB000-FORMAT-SCREEN-DATA
    PERFORM CC000-SAVE-PREVIOUS-KEY-DATA
ELSE
    IF TCB-RESULT EQUAL TCB-ERR
        IF EMP-RESULT EQUAL DGR-NO-FIND
            MOVE COND-CODE-47352 TO TCB-USER-CC
        ELSE
            IF EMP-RESULT EQUAL DGR-END-DATA
                MOVE COND-CODE-47354 TO TCB-USER-CC
            ELSE
                MOVE COND-CODE-47353 TO TCB-USER-CC
            ELSE
                MOVE COND-CODE-47353 TO TCB-USER-CC.

SKIP1
BA999-EXIT.
SKIP1
EXIT.

.
```

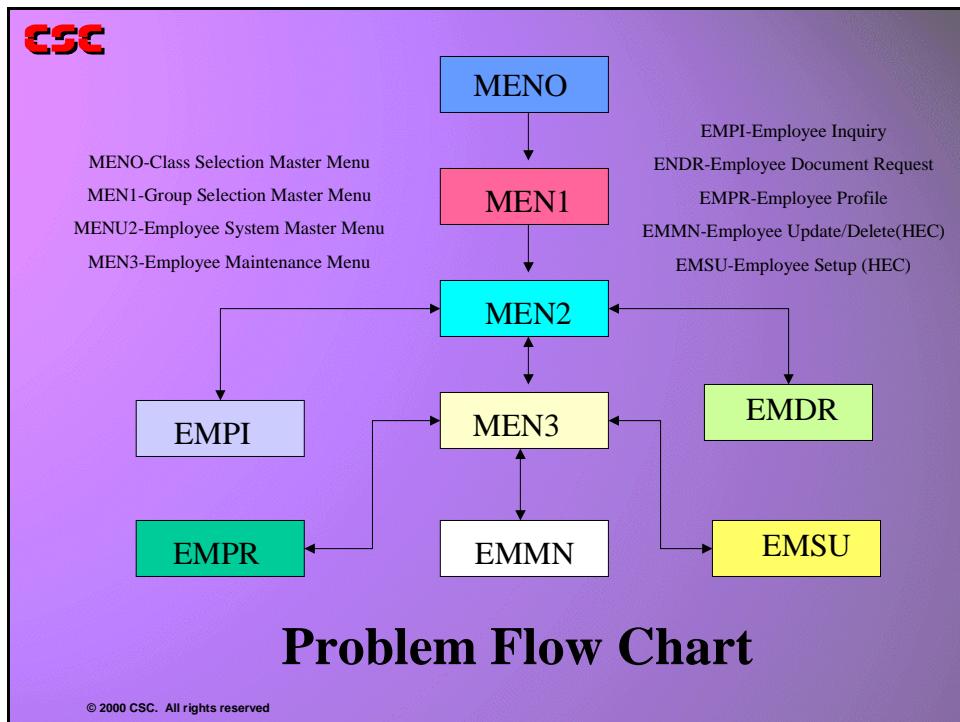


The Employee System - Scenario Flow

The exercises in this class are integrated around the development of a hypothetical Employee System. Through the development of the system, the student will be able to gain practice in maintaining many of the components of an FPS application.

The diagram shows the first five levels of the Employee System, beginning with the Demo System Class Selection Menu. This diagram provides students with a familiarity of the design of the system as well as introduces them to the overall flow through the screens.

The students will be providing additional pieces to the system through the completion of a series of exercises. The first exercise provides the function 'EMPI' shown on the fourth level, first box on the left.



MENO - Class Selection Master Menu
MEN1 - Group Selection Master Menu
MEN2 - Employee System Master Menu
MEN3 - Employee Maintenance Menu

EMPI - Employee Inquiry
EMDR - Employee Document Request
EMPR - Employee Profile
EMMN - Employee Update/Delete(HEC)
EMSU - Employee Setup (HEC)



Umbrella Programming

The Employee System - Scenario Flow

Problem Specifications—FPS Inquiry



Create an on-line function which performs an inquiry only on the Employee data base, given a specific Company Number and Employee ID number. The data to be subsequently displayed back to the operator is shown below.

COMPANY NUMBER

EMPLOYEE ID

EMPLOYEE FIRST NAME

EMPLOYEE LAST NAME

CURRENT EARNINGS

YEAR-TO-DATE EARNINGS

If the requested employee is located and the contents displayed properly, a message to that effect should also be displayed. If the requested employee cannot be located, a message to that effect should be displayed. All other conditions can be categorized together, and an appropriate message should be displayed.

Use PEM activities you create. Assume the terminal operator enters the transaction code 'DEMR' to reset the CTF and display the SYSTEM CLASS MENU. The operator then makes selection "1.xx.1" from this menu, where "1" is the Umbrella Programming class, "xx" equals your group number, and "1" is the employee inquiry option; e.g. "1.1.1" for group number 1 or "1.12.1" for group number 12. This causes your map to be displayed, and the operator can enter a valid COMPANY and EMPLOYEE ID.

The exercise is broken down into three (3) phases to emphasize specific topics within FPS system design and coding. The specific phases are described as follows:

I.

Creating a map definition including the required fields for data group 13589. Students are to use the same INCLUDE maps found in the standard header map, I53589M.

II.

Coding a COBOL program to read the Employee data base randomly. This program must satisfy FPS's requirement that all application programs must set the TCB-USER-CC field before normal termination.

III.

Coding the appropriate FPS Control entry to cause the Employee Inquiry function to be invoked from the EMPLOYEE SYSTEM MASTER MENU.



Map Definition - Phase I

In a previous section of this week's class, you built an online transaction (pair of transactions) to inquire on the EMP data base. In doing so, you had to create a map. You can copy the map you designed and modify it to work in the DEM System. The fields to be displayed are the same ones you displayed in the non-FPS application.

Base information for map definition:

- 1) Map Name:** Z999yyM, where "yy" is your group number + 20
- 2) Device:** A
- 3) Map Linkname:** Z999yyMs (Load module name), where "yy" is your group number + 20, and "s" is the map suffix (A=MFS,M=BMS)
- 4) Title:** Include your group number in the title information and as a literal on the map. (Inquire on map Z47350M to see where to place literal in map.)
- 5) Transaction:** DEMT
- 6) INCLUDE maps:** Use I53589E & I53589F. (Inquire on map Z47350M to see how to setup.)

Modify the map documentation and assembly JCL. Create ZUPCxxDP. You will need to replace the map name with your FPS map name. Submit the JCL and check its condition codes for a successful documentation, assembly, and link edit.

CICS only: Each time you assemble/linkedit, you will need to new copy the module.

Notes:



Umbrella Programming

The Employee System - Scenario Flow

Application Program - Phase II

In this phase of the Employee Inquiry Exercise you need to modify your COBOL program. The steps to successful completion of this phase are:

1. Verify that the following condition codes exist on format 1301:

Condition Code	Meaning
47351	Employee displayed
47352	Employee not found
47353	Undetermined error

Note: Use a value of 3 for "SOURCE" and blanks for "APPLICATION" and "ENVIRONMENT" in the key fields in format 1301.

2. Include the FPS Communication data group, copybook I53589D (01 W-FPS-COMM-DATA-GROUP).
3. In the EMP data base random access paragraph, set the condition code field to
47351—employee found
47352—employee not found
47353—undetermined error.

CICS only: Each time you compile/linkedit, you will need to new copy the module.

Notes:



Summary



Summary



- ✓ Application programs within FPS
 - ✓ Cannot issue data communication activities
 - ✓ Must set all needed condition codes
 - ✓ Must interrogate command field to determine last operator action
- ✓ Minimum requirements to set up FPS application
 - ✓ Prompt, work, reset must be defined
 - ✓ Application must be defined in FPS Application table
 - ✓ Must have an FPS Control record for a menu
 - ✓ Security entry for operator/password necessary
- ✓ Maps must have FPS header lines



© 2001 CSC. All rights reserved

22

- Application programs within FPS applications cannot issue data communication activities, must set all needed condition codes, and must interrogate the FPS command field to determine the last operator action.
- To set up a FPS application there are four minimum requirements:
 - At least three transactions (prompt, work, reset) must be defined
 - The application must be defined in the FPS Application Table
 - There must be a FPS Control record for a menu
 - There must be a security entry for operator/password.
- Maps used within FPS applications must have FPS header lines. These lines are generally provided through include maps.



Umbrella Programming

Summary



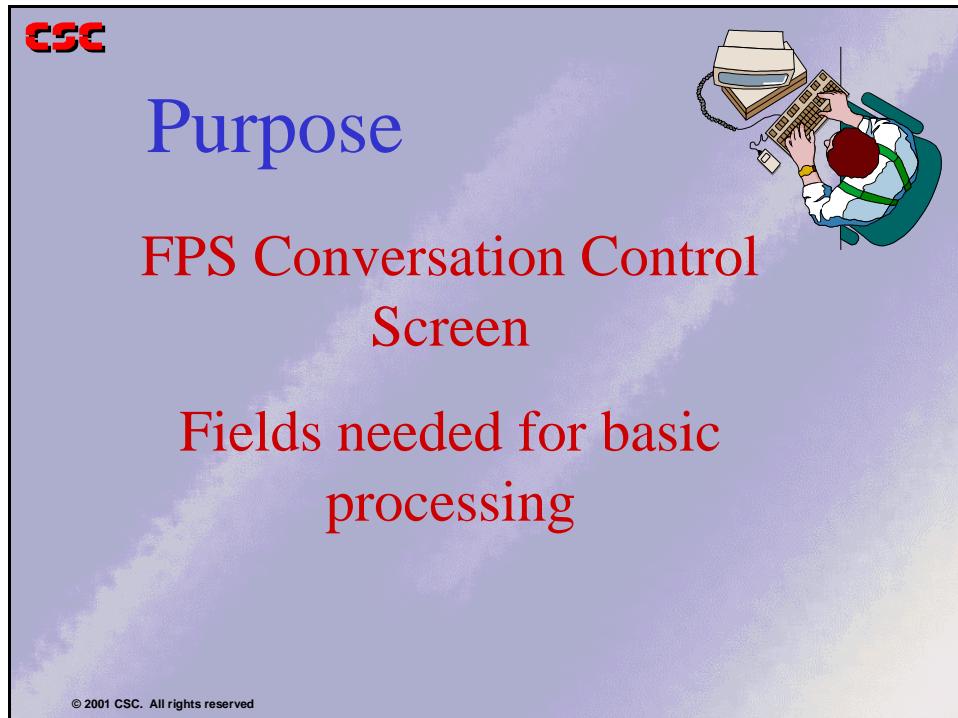
21-30

FPS Application Requirements

FPS Basic Logic

22

Purpose



Notes:

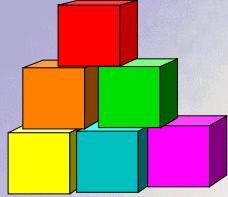


Topics

CSC

Topics

- Conversation Control Screen
- Conversation Condition Code Screen
- Communication methods between application programs and FPS
- Branching to a non-FPS application
- Basic Logic Fields
- Conversation Logic Flow



© 2000 CSC. All rights reserved 6/30/00

Objectives

CSC

Objectives



Explore:

- FPS Conversation Control and FPS Conversation Condition Code Screens
- Fields available for communication between application program and FPS
- Fields necessary for basic processing
- FPS Control and Condition Code entries that an application program can execute under FPS

© 2000 CSC. All rights reserved



Conversation Control Screen

The flow of logic executed by FPS is maintained on this screen. It is retrieved and interrogated for Displaying and Deblocking the correct maps, as well as executing the Link activity of the correct program for massaging the data properly. The entry also directs FPS to the next entry to be executed in the flow of the conversation.

Due to the amount of information contained on the Conversation Control record, the on-line maintenance is contained on two screens. The first contains the basic information and the second contains all the condition codes for that particular entry.

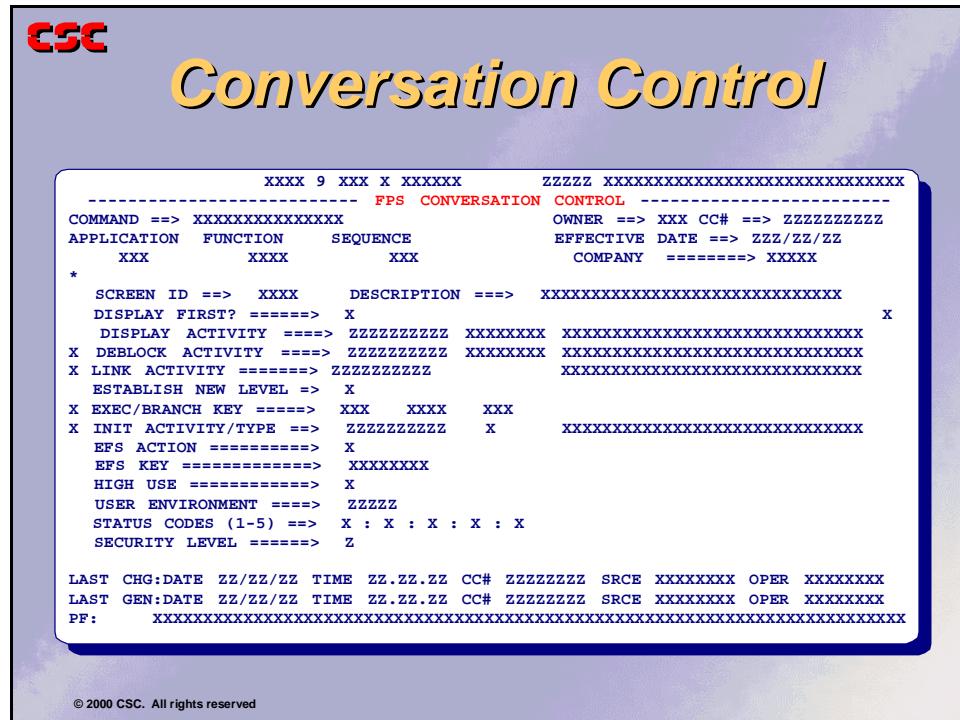
Notes:



Umbrella Programming

Conversation Control Screen

The FPS Basic Conversation Control screen looks like this:



The FPS Basic Conversation Control screen can be displayed using one of the following techniques:

1. From the Function Processing System Menu enter 3 in the SELECT OPTION field and press ENTER.
2. From the UMBRELLA System Master Menu, enter 6.3 in the SELECT OPTION field and press ENTER.
3. From any UMBRELLA System screen, enter =FPS.xyyyymmdd in the SELECT OPTION field and press ENTER.
4. From a cleared screen key in:

U FPS.xyyyymmdd

where xxx is the application mnemonic yyyy is the function (e.g., CONV) sss is the sequence number desired (e.g., 000).

If you do not enter a value for xxxyyys, the first Conversation record is presented.



Umbrella Programming

Conversation Control Screen

The fields on this screen are described as follows:

COMMAND

ADD (PF5)

Add a new conversation.

CHG (PF4)

Change the processing of a conversation.

CMND (PF7)

Invoke the screen to list the alternate commands that are valid in this conversation.

COND (PF8)

Invoke the screen to list the condition codes connected with this conversation.

CPY

Copy the basic information and all associated condition codes to another key.

DEL

Delete this conversation.

GEN

Create the generated version of this record that is to be used by the FPS Supervisor during execution of the conversation.

HSET (PF10)

Invoke the display of the Help processing control for the screen identified with this conversation.

INQ (PF6)

Display a requested conversation.

KEYS (PF11)

Invoke the display of the PF keys that are to be displayed when the DEBLOCK activity of this conversation is executed.

NXT (PF9)

Display the next conversation.

OWNER

The owner application identifier.

CC#

The change control number associated with any update function. The change control number must be compatible with the OWNER field.

EFFECTIVE DATE

The date this conversation is to effective to FPS.



Umbrella Programming

Conversation Control Screen

APPLICATION

The 3-byte mnemonic for the application.

FUNCTION

The user-defined function of this conversation.

SEQUENCE

The sequence number of the entry within the function.

COMPANY

The company number for which this conversation is valid. (The application record must have Y in the company default flag for this field to be a company number. Otherwise, it must be DFLT.)

* (action)

To view the ACTIVITY or go to the EXEC/BRANCH entry, enter:

S

With a command of INQ, displays the activity or goes directly to the conversation defined by the EXEC/BRANCH entry.

SCREEN ID

Enter a user-defined code to be placed in the FPS common header when a display action is entered from this conversation. This is used as part of the key when requesting Help text during execution of the conversation.

DESCRIPTION

The purpose of this screen or entry.

DISPLAY FIRST?

Enter Y if the DISPLAY activity is to be issued the first time the entry is entered. If it is not, the LINK activity is issued and the DISPLAY activity is not issued unless a DSP is requested on an accompanying condition code entry. This field is only used if the display activity is nonzero.

If the function/sequence of the Control entry is equal to MENU000, the DISPLAY FIRST is treated as display.

DISPLAY ACTIVITY

The activity to be issued if DISPLAY FIRST equals Y or if the action of the resulting condition code is equal to DSP. This activity must be a previously-defined activity when updating the conversation. (The map name and description of the activity are displayed by the maintenance program when ENTER is pressed.)

If the EXEC/BRANCH KEY is present for this conversation entry, the DISPLAY ACTIVITY is never executed.

DEBLOCK ACTIVITY

The activity to be issued when ENTER is pressed during the execution of this



Umbrella Programming

Conversation Control Screen

conversation. This activity must be a previously-defined activity when updating the Conversation entry (the map name and description of the activity are displayed by the maintenance program when ENTER is pressed). If the EXEC/BRANCH KEY is present for this Conversation entry, the DEBLOCK ACTIVITY is never executed.

LINK ACTIVITY

Enter a valid program link activity to be issued. This is a required field, unless the EXEC/BRANCH field is entered. These are mutually exclusive, with the exception of an EXEC/BRANCH to a non-FPS application. In this case the link activity must contain a valid program link, and the EXEC/BRANCH must contain FPSEXIT000, designating an exit from FPS to the appropriate LINK ACTIVITY (the description of the activity is displayed by the maintenance program when ENTER is pressed).

ESTABLISH NEW LEVEL

This field is used only in conjunction with the EXEC/BRANCH field. Y indicates a new level of processing is to be established. The default is N.

EXEC/BRANCH KEY

The key of the FPS Control entry to be processed.

If the EXEC/BRANCH KEY is present, the display, deblock, and link activities for this Conversation entry are ignored.

If the ESTABLISH NEW LEVEL flag is Y, this key is considered to be an EXEC and the current processing is saved in order to return at the completion of the execution. The condition codes on this Conversation entry are then checked for determining the next step in the conversation.

If the NEW LEVEL flag is N or space, this entry is considered a BRANCH and does not return, thereby ignoring all condition codes on this Conversation entry.

INIT ACTIVITY/TYPE

The activity to be used for initialization based on the value entered in the TYPE field. The type code determines when the INIT activity is to be issued. The following values are possible:

- L** Issue the INIT activity every time just before the link activity is issued.
- I** Issue the INIT activity the first time this entry is read, but not upon returning from a display of this entry. (A return from another entry causes the INIT activity to be issued again, however.)
- E** Issue the INIT activity at the end of this entry before proceeding to another entry.



Umbrella Programming

Conversation Control Screen

EFS Activity

This activity must be a previously-defined activity when updating the conversation. (The description of the activity is displayed by the maintenance program when ENTER is pressed.)

EFS ACTION

The TYPE of action being requested from the Edit and Format System (EFS). This is used in conjunction with the EFS KEY also being passed. The values are:

- 1 Edit only.
- 2 Edit and format.

EFS KEY

The key used by EFS when an EFS edit is requested by the application program. The program associated with the LINK activity must have a link to activity 13555 (which is program I53555) for execution of the EFS edit.

HIGH USE

This field is not currently used.

USER ENVIRONMENT

This code is placed into TCB-USER-ENVMT just before the link activity is issued. It can be checked by the application program as a parameter to allow variations in processing based on the entry being processed. If a value of 65000 is coded in this field, the value of TCB-USER-ENVMT during the processing of the previous entry is not overlaid.

STATUS CODES

This field can be used to qualify execution of the entry. Any values that are present are compared to the corresponding values present in W-FPS-COMM-STAT in the FPS communications data group I53589D. If the compared fields are unequal, the current entry is not executed. FPS then reads the next entry for processing.

(STATUS CODE 5 is reserved by FPS for the current terminal status. This field is set to T if the terminal is in a TEST status; otherwise, it is set to P for Production. The application program can use this status indicator to qualify execution of entries under test or production modes.)

SECURITY LEVEL

This field contains a value of zero to 9. If the value is zero, operators can always execute this entry regardless of the security information present on their password record. If the field is greater than zero, the security password record of the operator must contain Y in the corresponding FUNC SEC position in order to execute this entry.



Conversation Condition Code Screen

The condition codes contained on the Conversation Control entry are used by the FPS supervisor to determine the next step to be executed. FPS interrogates the list of condition codes contained on the FPS Control entry against TCB-USER-CC to determine the action to be taken. When the correct code is found, the ACTION field is analyzed for the proper execution process.

Certain condition codes are represented on this screen by an alphabetic mnemonic rather than their number. These condition codes have special meaning to FPS. The condition code 65000 falls into this category. It is represented by the alphabetic mnemonic ELSE and is the catch-all or default value. If the value in TCB-USER-CC cannot be found in the list of condition codes, the action listed with ELSE is executed. This must be the last condition code in the list, since the FPS supervisor quits interrogation upon finding this condition code.

Other alphabetic mnemonics for condition codes include CONT, RTN, PLVL, EXIT, and HELP. These codes and their actions will be discussed in a later section. On an ADD to the base screen the CONT, RTN, PLVL, and ELSE condition codes are automatically created. The ELSE condition code is mandatory although the others may be deleted.

There are five categories of actions which can be associated with a condition code. They are listed below.

DSP	Issue the Display Activity.
END	End function processing at current level.
DYF	Execute a dynamic function.
ABE	Request to abend.
SSS	SSS represents a three-digit sequence value. Pass control to the FPS Control entry with the current application and function ID's, and the specified sequence number.

Notes:



Umbrella Programming

Conversation Control Screen

The Conversation Condition Code screen looks like this:

The screenshot shows a terminal window titled "FPS Conversation Condition Codes". At the top, there is a header with fields for COMMAND, APPLICATION, and FUNCTION SEQUENCE, each containing "XXXXX". Below this is a table with columns for COND-CODE, ACTION, DESCRIPTION, COMMON RESULT, and FOR MSG?. The table contains 12 rows, each with an "X" in the first column and "XXXXX" in the other columns. At the bottom of the screen, there is a status bar with "LAST CHG:DATE ZZ/ZZ/ZZ TIME ZZ.ZZ.ZZ CC# BZZZZZZZZZ SRCE XXXXXXXX OPER XXXXXXXX" and "PF: XXXXXXXXX" followed by a series of "X"s.

*	COND-CODE	ACTION	DESCRIPTION	COMMON RESULT	FOR MSG?
X	XXXXX	XXX	XXXXXXXXXXXXXXXXXXXXXX	ZZZZZ	X
X	XXXXX	XXX	XXXXXXXXXXXXXXXXXXXXXX	ZZZZZ	X
X	XXXXX	XXX	XXXXXXXXXXXXXXXXXXXXXX	ZZZZZ	X
X	XXXXX	XXX	XXXXXXXXXXXXXXXXXXXXXX	ZZZZZ	X
X	XXXXX	XXX	XXXXXXXXXXXXXXXXXXXXXX	ZZZZZ	X
X	XXXXX	XXX	XXXXXXXXXXXXXXXXXXXXXX	ZZZZZ	X
X	XXXXX	XXX	XXXXXXXXXXXXXXXXXXXXXX	ZZZZZ	X
X	XXXXX	XXX	XXXXXXXXXXXXXXXXXXXXXX	ZZZZZ	X
X	XXXXX	XXX	XXXXXXXXXXXXXXXXXXXXXX	ZZZZZ	X
X	XXXXX	XXX	XXXXXXXXXXXXXXXXXXXXXX	ZZZZZ	X
X	XXXXX	XXX	XXXXXXXXXXXXXXXXXXXXXX	ZZZZZ	X

The Condition Code screen may be displayed only from the basic conversation screen. This screen may be displayed using one of the following techniques:

1. From the Conversation Control Base Screen press PF8.
2. From the Conversation Control Base Screen, enter COND in the COMMAND field and press ENTER.

The fields on this screen are described as follows:

COMMAND

CHG (PF4)

Change the condition codes of a conversation.

INQ (PF6)

Display the first 12 condition codes.

SB (PF7)

Display the previous 12 condition codes.

SF (PF8)

Display the next 12 condition codes.

OWNER

The owner application identifier.



CC#

The change control number associated with any update function. The change control number must be compatible with the OWNER field.

EFFECTIVE DATE

The date for which this conversation is to be effective to FPS.

APPLICATION

The 3-byte mnemonic for the application.

FUNCTION

The user-defined function of this conversation.

SEQUENCE

The sequence number of the entry within the function.

COMPANY ID

The company number for which this conversation is valid. (The application record must have Y in the company default flag for this field to be a company number. Otherwise, it must be DFLT.)

Note: OWNER, CC#, EFFECTIVE DATE, APPLICATION, FUNCTION, SEQUENCE, and COMPANY ID fields default to the corresponding fields on the "FPS Conversation Control" base screen.

*** (action)**

The action for this specific condition code. Valid values are:

A Add a condition code entry to the list.

C Change a condition code entry.

D Delete this condition code from the list.

COND-CODE

Enter a numeric condition code or one of the special FPS condition code mnemonics.

With an ADD to the base screen, the CONT, RTN, PLVL, and ELSE condition codes are automatically created. Any user-defined condition codes added by the operator are inserted just before the ELSE condition code.

RTN

Return to previous processing point.

CONT

Continue on.

PLVL

Return to previous processing level.



Umbrella Programming

Conversation Control Screen

HELP

Help screen processing requested. CC53553

EXIT

Go to Menu Processing. CC53545

ELSE

What to do if no other conditions are met. CC65000

If the value in TCB-USER-CC cannot be found in the list of condition codes, the action listed with 65000 is executed. This must be the last condition code in the list, because the FPS Supervisor quits interrogation upon finding this condition code.

ELSE cannot be deleted because it is required. However, if necessary, it can be changed to any other number and then that entry deleted.

ACTION

Enter one of the four FPS actions in the following list, or an alphanumeric sequence number of a Conversation Control entry.

DSP

Issue display activity.

The display activity coded on this entry is executed. If a display activity is not coded, the last screen is redisplayed and the current Control entry is overlaid with the entry belonging to the display activity.

END

End function processing at current level.

This action terminates the function at the current processing level. If the level is zero, the MENU function is initiated; otherwise, the level number is reduced by one and the processing status for the new level is restored. The condition code on the restored level then determines the action to be executed.

DYF

Execute a dynamic function.

This action builds the key for the Control entry from data in the FPS data group. APP is the value in W-FPS-COMM-APPL-CD, FUNCTION is the value in W-FPS-COMM-NEXT-CMND, and SEQ is the value in W-FPS-COMM-FUNC-BEGIN-SEQ. The application program must have placed the proper values in these fields.

ABE

Request to abend.

This action requests that FPS issue an abend. This is used for testing and for abnormal situations.



SSS

Sequence number of next entry to execute.

Using the APP and FUNC of the current entry, this sequence number is used to build the key of the next Conversation Control entry to execute.

DESCRIPTION

Enter a short description of what is executed when this condition code is the value of TCB-USER-CC. (This is for documentation only.)

COMMON RESULT

Enter a user-defined numeric value, or leave space. This field is moved into the W-FPS-COMM-RESULT field of data group 13589, and can be used by the application program for special processing. A value of 65000 in this field indicates the current value of TCB-USER-CC is to be placed in W-FPS-COMM-RESULT. A value of 65001 indicates that the current value of W-FPS-COMM-RESULT should be moved to TCB-USER-CC.

FOR MSG?

Enter N or Y. The default is Y. If the value is N, the condition and any message present are not saved for output on the displayed screen.

Notes:



Umbrella Programming

Methods of Communication Between Application Programs and FPS

Methods of Communication Between Application Programs and FPS

Three of the fields on the Conversation Control screen are used to communicate between FPS and application programs. A discussion of each of these fields and how it is used follows.

Status Qualifiers - Program to FPS

These fields will be set by an application program as it executes and then later be checked against values on the Conversation Control entry. The application programs set values in the corresponding fields in the variable W-FPS-COMM-STATUS in the FPS Common Data Group 13589. Those values are then checked against the values of the status qualifiers as set on the control entry. If the values in Data Group ID 13589 do not match all existing values on the entry, the entry is not executed and the next entry for the application is used for processing.

User Environment Code - FPS to Program

This field will be set by the control entry and then checked by the application program. This allows the application program to perform different actions based on which entry is being processed. The value of this field is moved into the TCB-USER-ENVMT variable before the link activity specified on the entry is issued.

As an example of usage, the CIS system uses the following values in TCB-USER-ENVMT to determine a) the type of access to be allowed, and b) if a function should allow the current customer (account) key to be:

NO SPEC PROCESSING	0
SETUP KEY CHG OK	1
" NO KEY CHG	2
MAINT KEY CHG OK	3
" NO KEY CHG	4
INQ KEY CHG OK	5
" NO KEY CHG	6

As another example of usage, the DPS system uses the following values in TCB-USER-ENVMT to determine the type of function to perform:

PRE/POST DISPLAY EXIT	10
REQUEST PROMPT	100
LINEUP PROMPT	200
REPRINT PROMPT	300



Common Result - Program to Program

This field allows programs to communicate with other programs and yet still have the processing flow defined by FPS control entries. The value in this field is moved into the W-FPS-COMM-RESULT field in the FPS Communication Data Group 13589 when a match is found on the condition code with which this value is associated. This field can be used for passing back result codes from lower processing levels or for checking by application programs which next obtain control.

If the value is 65000, FPS will move the current TCB-USER-CC to W-FPS-COMM-RESULT. A value of 65001 will cause the current value of W-FPS-COMM-RESULT to replace TCB-USER-CC before the action is taken.

Notes:



Umbrella Programming

Branching to a Non-FPS Application

Branching to a Non-FPS Application

If it is necessary to branch to a non-FPS application, both a LINK activity and EXEC/BRANCH are coded on the Control entry. The EXEC/BRANCH must be FPSEXIT000, which is a keyword to FPS and not an actual Control entry.

The ESTABLISH NEW LEVEL? field has a special meaning; if it is set or allowed to default to N, FPS will restart the current application at its MENU when prompted from non-FPS application. Coding Y in this field will cause FPS to restart at the current function and check the TCB-USER-CC set by the non-FPS application against the values coded on the function.

To re-enter an FPS application after the conversation has branched out of it, an interface program must change the field TCB-APPL-ID to the Application ID of the field to which a return is needed. Next, the program must allocate the FPS data groups, link to the FPS Supervisor Program, and end. When FPS retrieves the relevant CTF record for the conversation, it will recognize by a flag that the operator is returning from a prior conversation and will resume processing accordingly.

Notes:



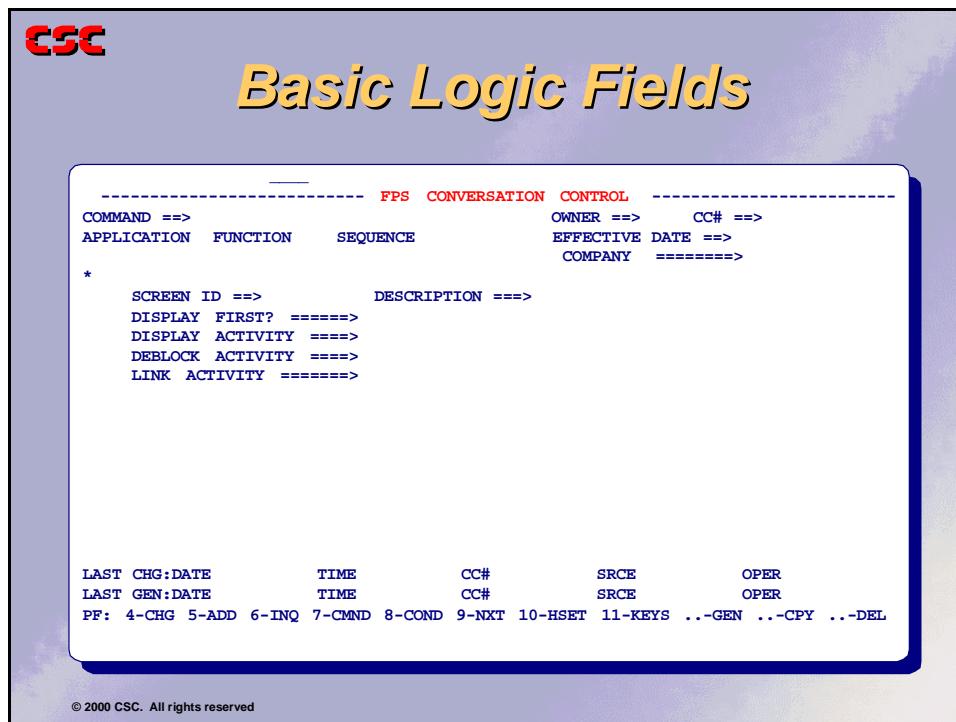
Basic Logic

Basic Logic Fields

The next two screens are extracts of the Conversation Control entry showing only the fields required for basic logic. The fields highlighted here perform two very specialized functions:

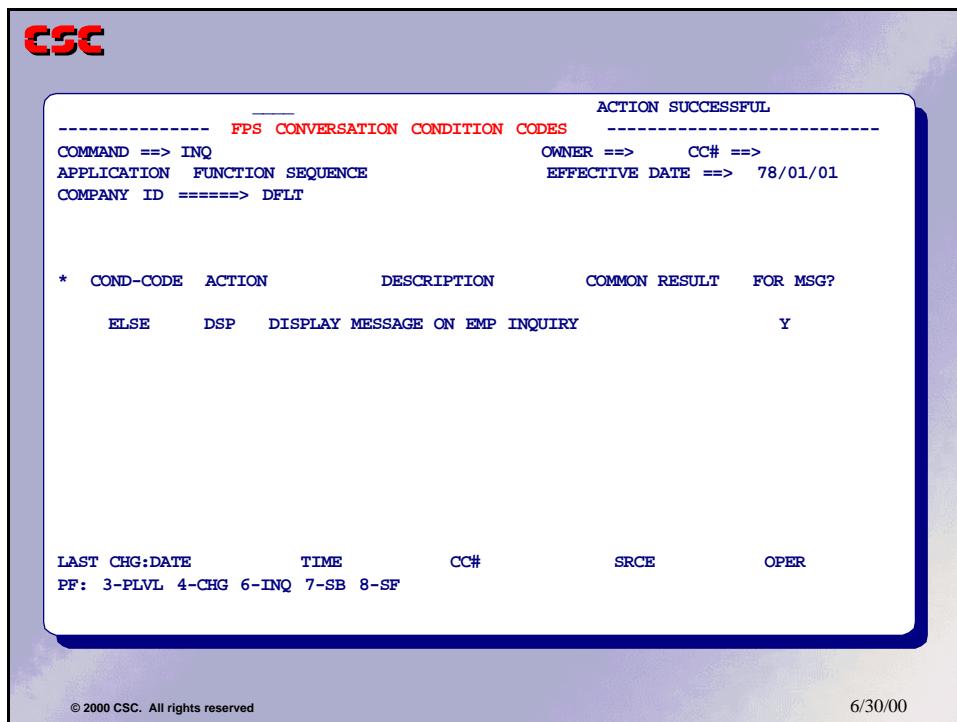
1. processing screen data and/or
 2. providing horizontal movement between Control entries.

In processing screen data the DISPLAY, DEBLOCK, and LINK activity fields play a major role. As we already know, the program to which FPS has linked has the requirement of setting a condition code. Once it has been done and the program terminates normally, the FPS Control module will gain control and compare the newly set value in TCB-USER-CC against the list of condition codes found on the FPS Conversation Condition Codes entry. Associated with each condition code is an action to be taken. This may entail branching to another FPS Control entry, displaying the screen, abending, or ending the current processing level. The branching action implies horizontal movement, i.e. the processing level is *not* increased when the branch takes place.



Umbrella Programming

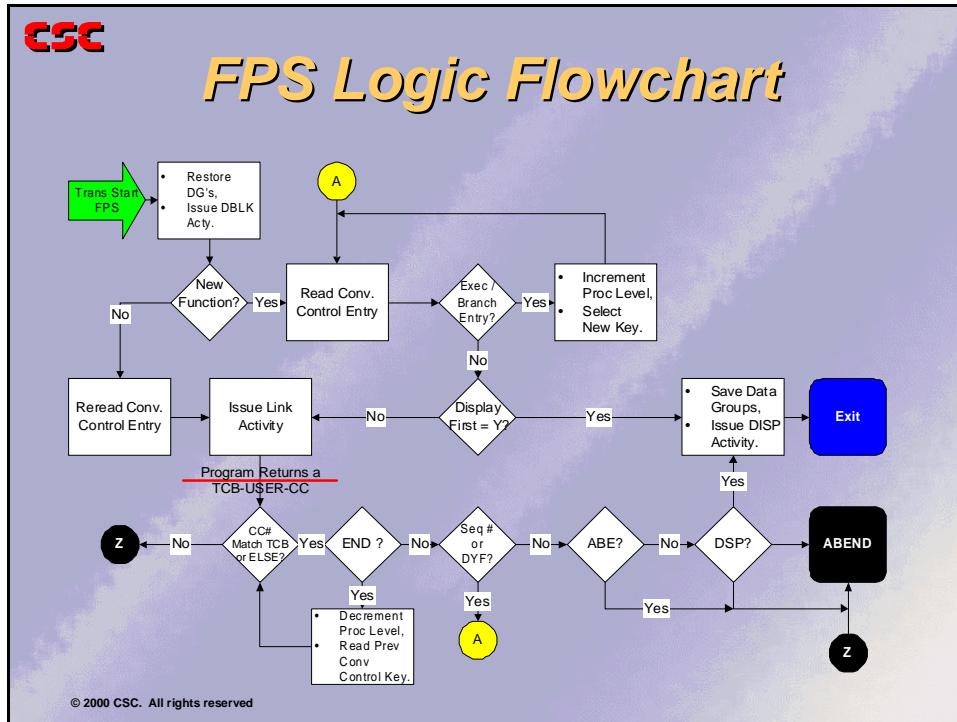
Basic Logic



Notes:



FPS Logic Flowchart



Notes:



Umbrella Programming

Basic Logic

FPS Basic Logic Flow UMBR Transaction

1. The transaction starts when the operator keys in UMBR from a cleared screen.
2. PEM looks up the Transaction Definition on the Process Dictionary. It has an Application ID of 48 and a Function ID of 53.

CSC FPS Basic Logic Flow 'UMBR' Transaction

ACTION COMPLETE	
UMBRELLA TRANSACTION DEFINITION INQUIRY/MAINTENANCE	
COMMAND ==>	BY TRAN CODE OWNER ==> UMB CC# ==>
APPLICATION ID ======>	48
FUNCTION ID ======>	53
SOURCE ID ======>	3
COMPANY ID LIST ======>	ALL
EFFECTIVE DATE ======>	78/01/01
TRANSACTION CODE ======>	UMBR
TRANSACTION DESCRIPTION ==>	UMBRELLA FPS INIT
DL/I PSB NAME ======>	PSBUMBA
DB2 PLAN NAME ======>	&&IBAU04
APPC: REMOTE PEM ==>	CONVERSATION ==>

*	ACTIVITY	TRANSACTION	ACTIVITIES	*	ACTIVITY	*	ACTIVITY	*	ACTIVITY
*	13587	ACTIVITY	13551	*	ACTIVITY	*	ACTIVITY	*	ACTIVITY

LAST CHG:DATE 94/11/07 TIME 19:22:55 CC# SRCE UMB200 OPER
PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-ACTS 11-TRAN ..-DEL ..-NEW

© 2000 CSC. All rights reserved

1. Start with operator keys in UMBR from a cleared screen.
2. PEM looks up Transaction Definition. It has an Application ID of 48 and a Function ID of 53.

Notes:



3. The first activity on the transaction allocates the work area needed by FPS.

CSC

```
48311 ACTION COMPLETE (WA ACTS)
----- ACTIVITY DEFINITION INQUIRY/MAINTENANCE -----
COMMAND ==> INQ OWNER ==> FPS CC# ==>

ACTIVITY ID ======> 13587     EFF DATE ==> 78/01/01
ACTIVITY MNEMONIC ==>          TYPE ======> WORK
DESCRIPTION ======> FPS-COMM DG INIT

WORKAREA

*** DATA GROUPS ***
*   DGID DISP *   DGID DISP *   DGID DISP *   DGID DISP
  48278 ANO    48266 ANO    13597 ANO    13591 ALC
  13590 ALC    13589 INT

LAST CHG:DATE 94/11/07 TIME 19.22.55 CC#
SRCE UMB200 OPER
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 11-CPY ..-NXTT ..-NEW ..-NXTA ..-DEL
```

3. 1st transaction activity work area needed by FPS.

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

Basic Logic

4. The next activity links to the FPS supervisor, program I53551.

CSC

```
48308 ACTION COMPLETE (LINK ACTS)
----- ACTIVITY DEFINITION INQUIRY/MAINTENANCE -----
COMMAND ==> INQ OWNER ==> FPS CC# ==>
ACTIVITY ID =====> 13551 EFF DATE ==> 78/01/01
ACTIVITY MNEMONIC ==> TYPE =====> LINK
DESCRIPTION =====> FPS CONTROL MODULE

LINK

PROGRAM ID ===> 13551

*** DATA GROUPS ***
* DGID * DGID * DGID * DGID * DGID

LAST CHG:DATE 94/11/07 TIME 19.22.55 CC# SRCE UMB200 OPER
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-PGMID 11-CPY ..-NXTT ..-NEW ..-NXTA
```

4. Next activity links to FPS supervisor, program I53551.

© 2000 CSC. All rights reserved

CSC

```
ACTION SUCCESSFUL
----- PROGRAM DEFINITION INQUIRY/MAINTENANCE -----
COMMAND ==> INQ OWNER ==> FPS CC# ==>
PROGRAM ID ===> 13551 EFF DATE ==> 78/01/01 PEM TECHNOLOGY => FULLWORD
LINKNAME ==> I53551 LANGUAGE ==> ALC SOURCE NAME ===> I53551
DESCRIPTION => FPS CONTROL MODULE STATUS ==> PROD
HIGH, MED, OR LOW USAGE? ==> HIGH USED ONLINE, BATCH OR BOTH? ==> BOTH

----- DATA GROUPS USED BY PROGRAM -----
* PP ---DGID--- * PP ---DGID--- * PP ---DGID---
1 13551 2 13589 3 13590
4 13591 5 48278 6 48266
7 13597 8 48009 9 48111
10 -DYNAMIC- 11 -DYNAMIC- 12 -DYNAMIC-

----- AUTHORIZED ACTIVITIES -----
* -ACTIVITY- * -ACTIVITY- * -ACTIVITY- * -ACTIVITY- * -ACTIVITY- * -ACTIVITY-
-- ALL --

LAST CHG:DATE 94/11/07 TIME 19.22.55 CC# SRCE UMB200 OPER
PF: 2-XREF 3-PLVL 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-BYLINK 11-BYID ..-ACTS
```

© 2000 CSC. All rights reserved



Umbrella Programming

Basic Logic

5. The FPS supervisor interrogates the TCB to determine the Application ID of the transaction which issued the link to the program and searches the Application Table sequentially until it finds a match.

This is the table entry for the "UMB" application. Each FPS application must have an entry in this table.

CSC

```
ACTION SUCCESSFUL
----- FPS APPLICATION TABLE -----
COMMAND ==> INQ           OWNER ==> SPS CC# ==>
APPLICATION ==> UMB          EFFECTIVE DATE ==> 78/01/01

APPLICATION ID ======>      48        TEST MODE ==> N
RESET TRANSACTION =====>    53
PRE-FPS EXIT ======>
POST-FPS EXIT ======>
FUNC-INIT ACTIVITY ==>
SUCCESSFUL COND CODE ==>
COMPANY DFLT VALID? ==> N

GLOBAL DATA GROUPS:
*      --DGID--      *      --DGID--      *      --DGID--      *      --DGID--
        48002

LAST CHG:DATE 94/11/07 TIME 19.22.55 CC#
PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-ALTD 9-NXT ..-DEL
SRCE UMB200   OPER
```

5. FPS supervisor interrogates TCB to determine Application ID which issued link to program . This is table entry for “UMB” application. Each FPS application must have an entry in table.

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

Basic Logic

6. The FPS supervisor checks the Function ID of the 'UMBR' transaction, 53, against the value in the field UMB-RESET-TXN. Since the two fields match, the CTF records associated with the UMB application are cleared out, and the FPS Control entry with a userkey of UMB/MENU/000 is read.



```
ACTION SUCCESSFUL
----- FPS CONVERSATION CONTROL -----
COMMAND ==> INQ          OWNER ==> UMB CC# ==>
APPLICATION   FUNCTION      SEQUENCE          EFFECTIVE DATE ==> 78/01/01
          UMB       MENU        000             COMPANY =====> DFLT
*
SCREEN ID ==> MENO      DESCRIPTION ==> U2S-0-MENU NO. ONE
DISPLAY FIRST? =====> N
DISPLAY ACTIVITY =====> 48507 U48107M DISP U2S MENU 1
DEBLOCK ACTIVITY =====> 48307 U48107M DBLK U2S MENU 1
LINK ACTIVITY =====> 13563 U2S MENU PROC PROG
ESTABLISH NEW LEVEL =>
EXEC/BRANCH KEY =====>
INIT ACTIVITY/TYPE ==>
EFS ACTION =====>
EFS KEY =====>
HIGH USE =====> N
USER ENVIRONMENT =====>
STATUS CODES (1-5) ==> : : : :
SECURITY LEVEL =====>

LAST CHG:DATE 94/11/07 TIME 21.01.00 CC#           SRCE UMB200    OPER
LAST GEN:DATE 95/01/10 TIME 15.52.49 CC#           SRCE UMB2006   OPER
PF: 4-CHG 5-ADD 6-INQ 7-CMND 8-COND 9-NXT 10-HSET 11-KEYS ..-GEN ..-CPY ..-DEL

6. FPS supervisor checks Function ID of 53 against value in UMB-RESET-TXN. As the 2 fields match, CTF records associated with UMB application are cleared and FPS Control entry with a userkey of UMB/MENU/000 is read.
```

© 2000 CSC. All rights reserved

Notes:



7. The FPS Conversation Condition Codes are part of the FPS Conversation Control, but only checked when control is returned to FPS from an application program.

CSC

ACTION SUCCESSFUL

----- FPS CONVERSATION CONDITION CODES -----

COMMAND ==> INQ	OWNER ==> UMB CC# ==>
APPLICATION FUNCTION SEQUENCE	EFFECTIVE DATE ==> 78/01/01
UMB MENU 000	COMPANY ID =====> DFLT

* COND-CODE ACTION DESCRIPTION COMMON RESULT FOR MSG?

00000	DYF	GO TO NEW FUNC BUILT BY MODULE		
EXIT	000	DUMMY XFER		
RTN	END	RETURN TO MAIN MENU IF PF1/RTN		
CONT	END	RETURN TO MAIN MENU IF PF2/RTN		
ELSE	DSP	ALL OTHERS, DISPLAY MESSAGE		Y

LAST CHG:DATE 94/11/07 TIME 21.01.00 CC# SRCE UMB200 OPER
PF: 3-PLVL 4-CHG 6-INQ 7-SB 8-SF

FPS Conversation Condition Codes are part of FPS Conversation Control, but only checked when control is returned to FPS from an application program.

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

Basic Logic

8. The Control entry with the userkey UMB/MENU/000 contains the data used by the FPS supervisor to control processing flow. Because the transaction is a reset transaction, the display activity will always be executed first, whether the "1ST?" flag is turned on or not. In this case, display activity 48507 will be executed. Since this is a display activity, the data referenced on the map (in this case, map U48107M) is written to the CTF before the activity is executed.



```
48304 ACTION COMPLETE (DC ACTS)
----- ACTIVITY DEFINITION INQUIRY/MAINTENANCE -----
COMMAND ==> INQ OWNER ==> UMB CC# ==>
ACTIVITY ID ======> 48507 EFF DATE ==> 78/01/01
ACTIVITY MNEMONIC ==> TYPE ======> DC
DESCRIPTION ======> DISP U2S MENU 1

DATA COMMUNICATIONS
MAPNAME ==> U48107M RETRY ==> Y
SERVICE ==> DISP ERASE ==> Y
DISP =====> WAIT PROMPT ==> N
*** DATA GROUPS ***
* DGID * DGID * DGID * DGID

LAST CHG:DATE 94/11/07 TIME 19.22.55 CC# SRCE UMB200 OPER
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-MAPID 11-CPY ..-NXTT ..-NEW ..-NXTA

© 2000 CSC. All rights reserved
```

7. Control entry with userkey UMB/MENU/000 contains data used to control processing flow. Display activity always be executed 1st. Display activity 48507 will be executed. Data referenced on map is written to CTF before activity is executed.

Notes:



9. With the display of the map, the transaction ends. Note that the map has the trancode UMBT.

CSC

```
ACTION COMPLETE
----- MAP DEFINITION BASE SCREEN -----
COMMAND ==> NXT          OWNER ==> UMB CC# ==>
LANGUAGE => ENU
MAP NAME => U48107M DEVICE => A      CO GROUP => ALL      EFF DATE => 78/01/01

DESCRIPTION =====> UMBRELLA SYSTEM MASTER MENU : 
LINKNAME =====> U48107MM
TYPE (I,O,U,S) =====> U           TRANCODE DYNAMIC? =====> N
TRANCODE =====> UMBT           TRANCODE PROTECTED? =====> N
EXTENDED ATTRIBUTES? ==> N           NON-PEM MAP? =====> N
MAP TYPE (DOC|SCR) =====> SCR      MAXIMUM ROWS => 24      MAXIMUM CLMS => 80

*     DGID      *     DGID      *     DGID      *     DGID      *     DGID

NEW MAP KEY FOR COPY:
MAP NAME =>           DEVICE =>           CO GROUP =>           EFF DATE =>
LANGUAGE =>
LAST CHG:DATE 94/11/07 TIME 19.22.55 CC#      59452 SRCE UMB200   OPER
LAST GEN:DATE 95/01/10 TIME 15.51.44 CC#      SRCE UMB2006   OPER
PF: 4-CHG 5-ADD 6-INQ 7-BROWSE 8-EDIT 9-NXT 10-GEN 11-DSP ..-DOC ..-CPY ..-GRP
```

© 2000 CSC. All rights reserved

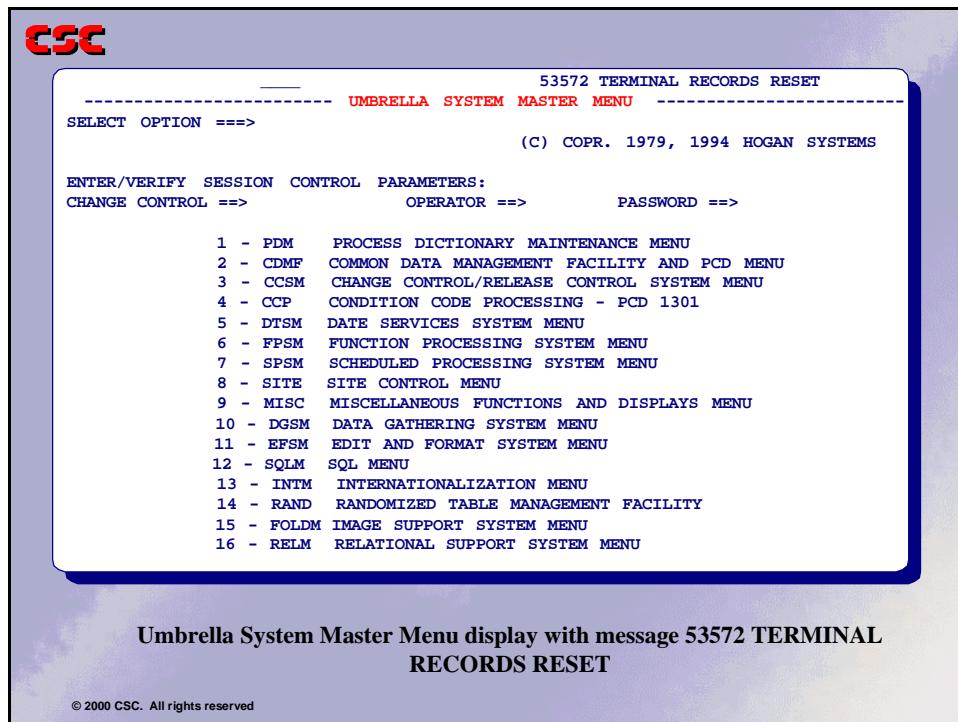
Notes:



Umbrella Programming

Basic Logic

The UMBRELLA SYSTEM MASTER MENU display with message 53572 TERMINAL RECORDS RESET.



Notes:



SCENARIO - CTF STATUS UPDATE

Processing Level: 00 CTL: UMB/MENU/000 DISP: 48507 DBLK: 48307

B/L

UMB/MENU/000

Notes:



Umbrella Programming

Basic Logic

Problem Specifications—FPS Inquiry



FPS Control Entry - Phase III

In this phase you will complete the exercise which was first introduced in a previous section. The first step of this phase is to define two PEM activities:

1. DEBLOCK activity
2. DISPLAY activity

The DC activities will use the map you defined in Phase I, Z999yyM, where "yy" is your group number + 20. You will also create an FPS Control entry to invoke the Employee Inquiry function in the Employee System. Steps to successful completion of this phase are:

1. Define the activities required for this function.

9993yy, where yy is your group number + 20, data communications (DC) to display map Z999yyM, where yy is your group number + 20.

9992yy, where yy is your group number + 20, data communications (DC) to deblock map Z999yyM where yy is your group number + 20.

2. Add userkey "DEMEMPx100", where "x" is the last digit of your two-digit group number, or "A", "B", or "C" for groups 10, 11, or 12, respectively.

FPS Control userkey "DEMEMPx100" will contain the display, deblock, and link activities as well as the condition code(s) you wish to 'trap'.

Assign a 'SCRN-ID' of "EMPx" in your FPS Control entry. "x" is defined as above.

When the "FPS Conversation Control" entry screen is added the "FPS Conversation Condition Codes" screen is automatically displayed with several predefined condition codes.

Delete all the default condition codes and condition code actions except for the ELSE condition code. The ACTION should be DSP, key your own description, there are no COMMON RESULTS being used and Y FOR MSG?.

3. In order for a conversation to be used in processing, it must be generated. On the base screen, enter GEN in the COMMAND field and press ENTER.
4. Enter transaction code 'DEMR' to reset the CTF and display the SYSTEM CLASS MENU. Make selection "1.xx.1" from this menu, where "1" is the Umbrella Programming class, "xx" equals your group number, and "1" is the employee inquiry option. This causes your map to be displayed, and then you can enter a valid COMPANY and EMPLOYEE ID.
5. Test your scenario with Company '1' and Employee Number '1111'. This employee is "TIBERIUS CAESAR".



6. Retest your scenario with Company '2' and Employee Number '12346'. This employee does not exist.

Note the messages returned to the screen in each case.

7. Use the FPS command PLVL. Note which action occurs. The FPS supervisor sets up this command to automatically return to the last function of the previous processing level.

Notes:

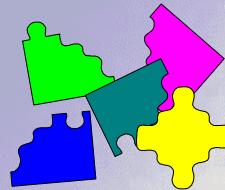


Summary



Summary

- Last operator action determined by FPS Conversation Control questioning command field
- Table determines where or what function is performed next
- Primary FPS Basic Logic fields
 - DISPLAY FIRST?
 - DISPLAY ACTIVITY
 - DEBLOCK ACTIVITY
 - LINK ACTIVITY



© 2000 CSC. All rights reserved

6/30/00



Summary

- Several communication methods possible between FPS and the application program
- Branching to a non-FPS application is possible



© 2000 CSC. All rights reserved

6/30/00



Umbrella Programming

Summary

- The FPS Conversation Control screen maintains the information to display and deblock a map, and link to an application program, so as to interrogate the FPS command field to determine the last operator action.
- The FPS Conversation Control Condition Code screen contains a table which can determine where or what function is performed next.
- The following are known as the primary FPS Basic Logic fields:
 - 1.DISPLAY FIRST?
 - 2.DISPLAY ACTIVITY
 - 3.DEBLOCK ACTIVITY
 - 4.LINK ACTIVITY
- Several methods of communication are possible between FPS and the application program.
- Branching to a non-FPS application is possible by specifying the special FPS interface key of FPS/EXIT/000.

Notes:



Umbrella Programming

Summary



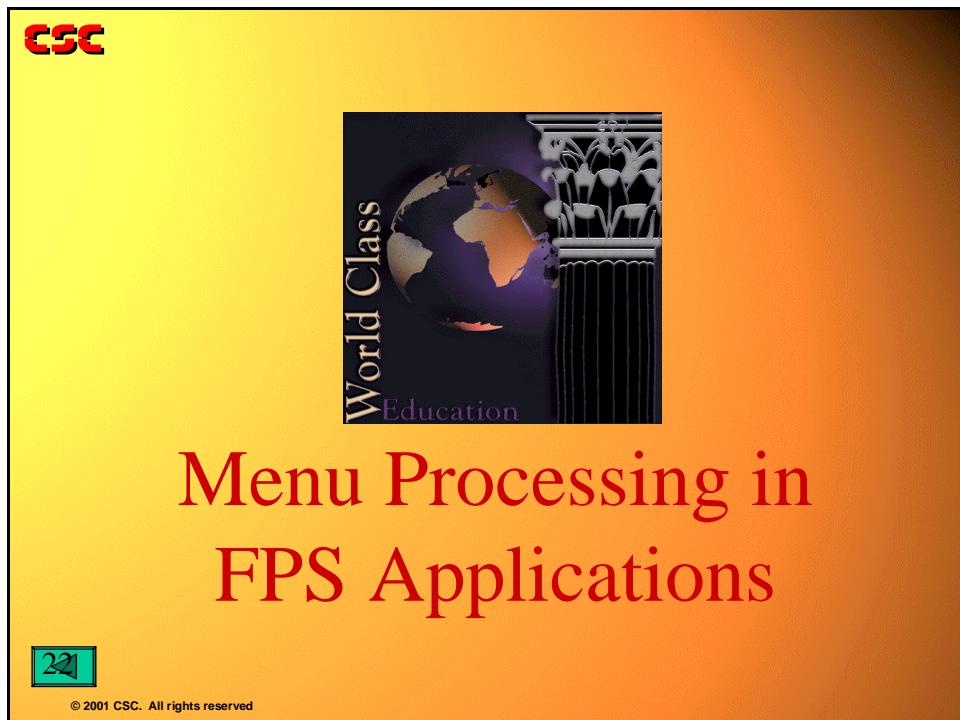
22-34

FPS Basic Logic

FPS Menu Processing

23

Purpose



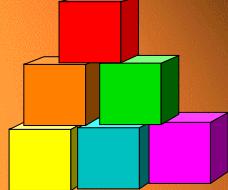
Notes:



Topic

CSC

Topic



- FPS Menu Processor

© 2000 CSC. All rights reserved

Objectives

CSC

Objectives



- List requirements for using the FPS menu processor
- Code an FPS Control entry to invoke the menu processor
- Add new prompt to existing Menu Processing entry

© 2000 CSC. All rights reserved



Overview

CSC

To Process the Menu



- * Use the FPS Menu Processor
 - * I53563
- * Code the functions of new FPS Control entry userkeys on map
- * Enter desired function directly

© 2000 CSC. All rights reserved

The starting point with all on-line systems is the main menu. With all FPS on-line systems the reset transaction will cause the application main menu to be displayed. However, there is more than one way to process the menu. Some of the more common methods include the following:

- Using the delivered FPS Menu Processor Activity, 13563
- Coding the functions of new FPS Control entry userkeys on the map where they can be accessed by an application program.
- Entering the desired function directly into the FPS NEXT COMMAND field.

This section will discuss the most common method, the FPS Menu Processor.

Notes:



Umbrella Programming

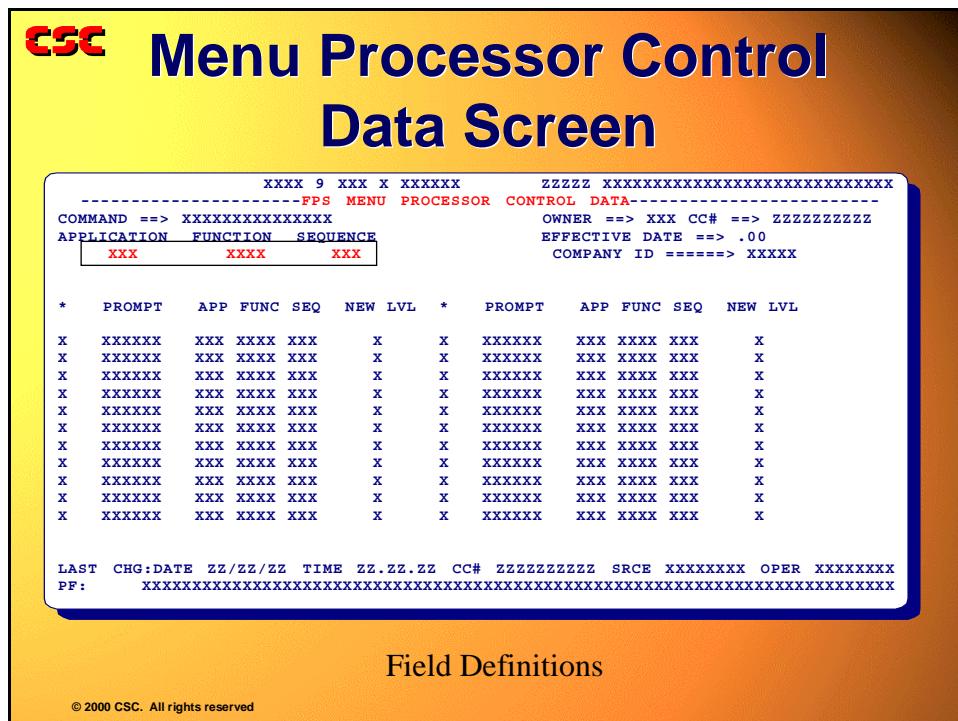
The FPS Menu Processor

The FPS Menu Processor

The most straightforward method of menu processing uses the FPS Menu Processing entries stored on CDMF.

Each Menu Processing entry contains a list of prompt values. Each prompt value is associated with a userkey for an FPS Control entry. When a terminal operator selects a menu prompt, FPS transfers control to the entry whose userkey corresponds to the menu prompt just selected.

The mapped image below displays the fields of the FPS Menu Processor Control Data screen.



Notes:



Umbrella Programming

The FPS Menu Processor

The fields on this screen are described as follows:

COMMAND

ADD (PF5)

Add a new menu control record.

CHG (PF4)

Change the prompt commands and/or the key of the conversation to be executed.

DEL

Delete this menu control record.

INQ (PF6)

Display a requested menu control record on file.

NXT (PF9)

Display the next menu control record on file.

SB (PF7)

Display the previous 22 prompt entries.

SF (PF8)

Display the next 22 prompt entries.

OWNER

The owner application identifier.

CC#

The change control number associated with any update function. The change control number must be compatible with the OWNER field.

EFFECTIVE DATE

The date that this menu control is to be effective.

APPLICATION

The 3-byte mnemonic for the application.

FUNCTION

The 4-character function of the menu entry.

SEQUENCE

The sequence number of the specific menu to be maintained.

COMPANY ID

The company number for which this menu control is valid. (The application record must have Y in the company default flag for this field to be a company number. Otherwise, it must be DFLT.)



Umbrella Programming

The FPS Menu Processor

* (action)

The action for this specific prompt. Valid values are:

A Add the prompt and key as specified by the PROMPT, APP, FUNC, and SEQ fields.

C Change the prompt and key as specified by the PROMPT, APP, FUNC, and SEQ fields.

D Delete the PROMPT command.

S With a command of INQ, display the Conversation Control entry with this key.

PROMPT

The value to start the conversation processing for the corresponding APP.FUNC.SEQ fields.

APP FUNC SEQ

The key of the Conversation Control entry that FPS is to execute for this prompt value.

NEW LVL

Enter Y if a new FPS level should be created (not currently supported).

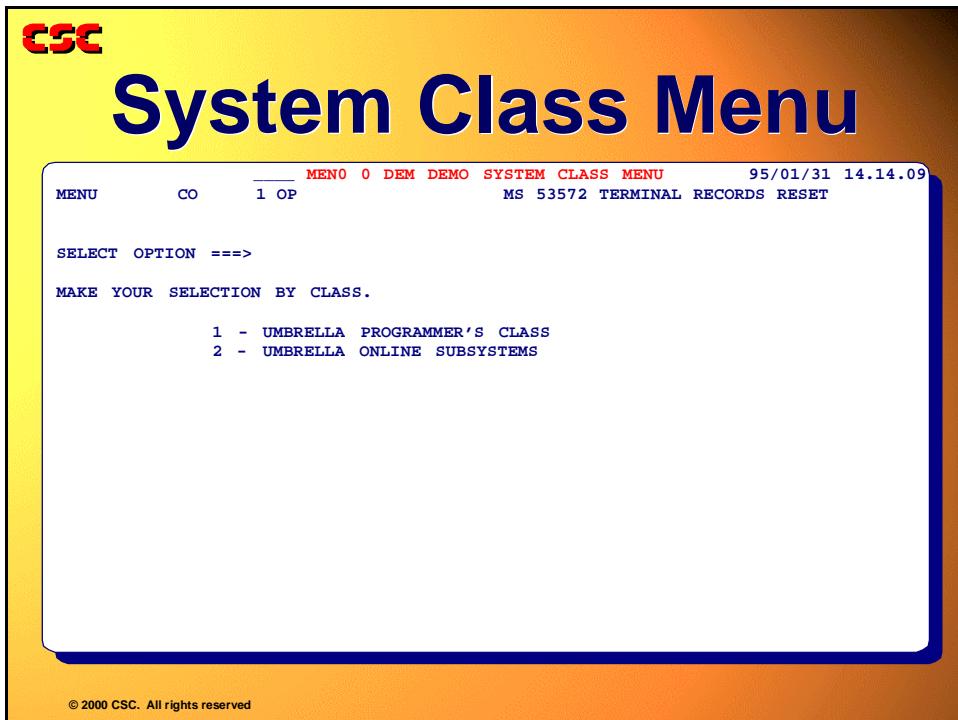
Notes:



Umbrella Programming

The FPS Menu Processor

Shown below is the SYSTEM CLASS MENU. This menu is used to drive the on-line exercises for the UMBRELLA classes. On the following page you will see the control and menu processing screens that direct its processing.



Notes:

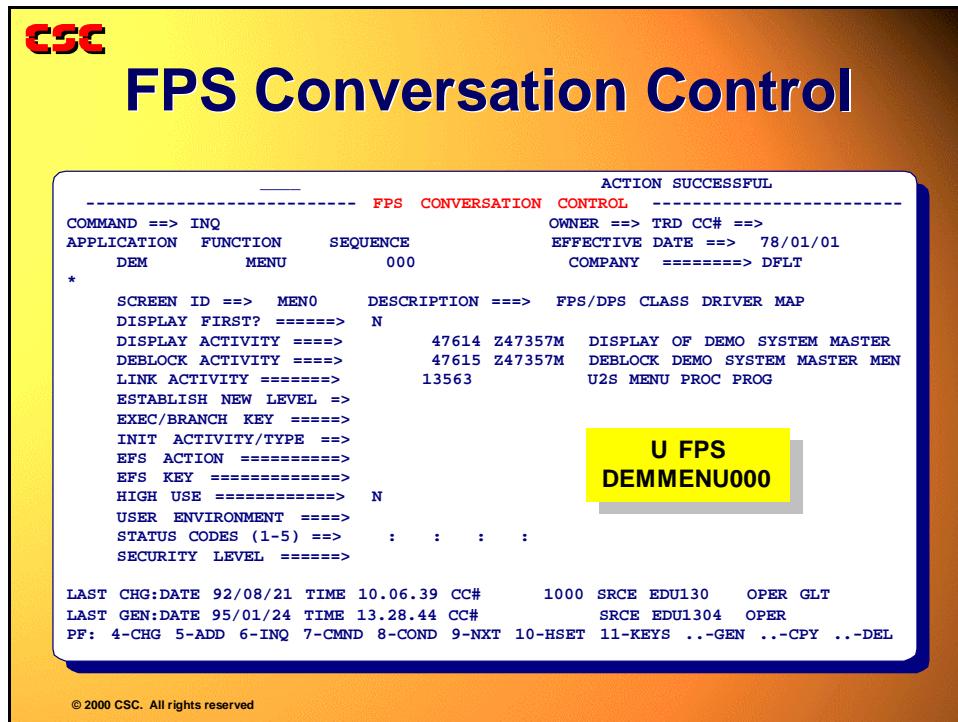


Umbrella Programming

The FPS Menu Processor

We already know that for application 'aaa' to run under FPS, a userkey of 'aaaMENU000' must be defined on the FPS Control entries. This entry is responsible for displaying and deblocking the master menu for the 'aaa' system. If this system is using FPS Menu Processing to process the menu, the following three requirements must also be met.

1. The Control entry must link to program I53563, the special Menu Processor program, using Activity 13563.
2. A Menu Processing entry must be defined with a key identical to the Control entry.
3. The Condition Code Control entry must include the condition code of 00000 with an action of DYF.



Notes:



CSC

Menu Processor Control Data

ACTION SUCCESSFUL

FPS MENU PROCESSOR CONTROL DATA			
COMMAND ==> INQ	OWNER ==> TRD CC# ==>		
APPLICATION FUNCTION SEQUENCE	EFFECTIVE DATE ==> 78/01/01		
DEM MENU 000	COMPANY ID =====> DFLT		
* PROMPT APP FUNC SEQ NEW LVL * PROMPT APP FUNC SEQ NEW LVL			
U UMB MENU 000	1 DEM MENU 020		
2 DEM MENU 010	UPC DEM MENU 020		
UOS DEM MENU 010			
U MENU.DEMMENU000			
LAST CHG:DATE 92/08/03 TIME 16.00.01 CC# 1000 SRCE EDU130 OPER GLT			
PF: 2-XREF 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT ...DEL			

© 2001 CSC. All rights reserved

CSC

COND-CODE 00000 ACTION DYF

ACTION SUCCESSFUL

FPS CONVERSATION CONDITION CODES			
COMMAND ==> INQ	OWNER ==> TRD CC# ==>		
APPLICATION FUNCTION SEQUENCE	EFFECTIVE DATE ==> 78/01/01		
DEM MENU 000	COMPANY ID =====> DFLT		
* COND-CODE ACTION DESCRIPTION COMMON RESULT FOR MSG?			
00000 DYF	GO TO NEW FUNC BUILT BY MODULE		
EXIT 000	DUMMY XFER		
RTN END	RETRN TO MAIN MENU IF PF1/RTN	Y	
CONT END	RETRN TO MAIN MENU IF PF2/RTN	Y	
HELP DSP	NO HELP YET	Y	
ELSE DSP	ALL OTHERS, DISPLAY MESSAGE	Y	

LAST CHG:DATE 92/08/21 TIME 10.06.39 CC# 1000 SRCE EDU130 OPER GLT

PF: 3-PLVL 4-CHG 6-INQ 7-SB 8-SF

© 2000 CSC. All rights reserved



Umbrella Programming

The FPS Menu Processor

- Program 153563 uses the terminal operator entered option, and searches the FPS Menu Processor table of the entry with the same key as the Conversation Control record. When it finds a match, the App Func and Seq values are inserted in the FPS Communications DG 13589. The fields are W-FPS-COMM-APPL-CD, W-FPS-COMM-NEXT-CMND, and W-FPS-COMM-FUNC-BEGIN-SEQ respectively.
- 153563 sets condition code 00000 in TCB-USER-CC to indicate that the entered option was found. If the option is not found, condition code 53542 is set instead.
- 153563 ends, and control is returned to FPS. Following a link execution, the Condition Code table attached to the Conversation Control entry is searched.
- CC 0 is associated with the Action code of DYF. This action signals the FPS processor program that it is to look in DG 13589 to determine which Conversation Control record key is to be processed next.
- The next Conversation Control record is typically a BE, causing the CD Display Activity to be issued.

Notes:



Problem Specifications—FPS Menu



The three options from the Employee System master menu are:

1. Employee Inquiry
2. Employee Maintenance
3. Employee Document Request

Both the Control and Menu Processor entries are already defined using the key of DEM/MENx/000, where "x" is the last digit of your two-digit group number, or "A", "B", or "C" for groups 10, 11, or 12, respectively. The Menu Processor entry DEM/MENx/000 designates a Control entry of DEM/EMPx/000 for Selection 1.

Selection 1 has been designed to take the operator to the Employee Inquiry Screen.

Your assignment is comprised of the following tasks:

1. Your user would like to be able to enter the function code EMPI directly into the "SELECT OPTION" field on the Employee System Master Menu to cause the Employee Inquiry option to be invoked. Modify the Menu Processor entry so that this request is satisfied.
2. This function should be coded as a "hidden option."
3. Test your work by executing the following System Flow diagram.

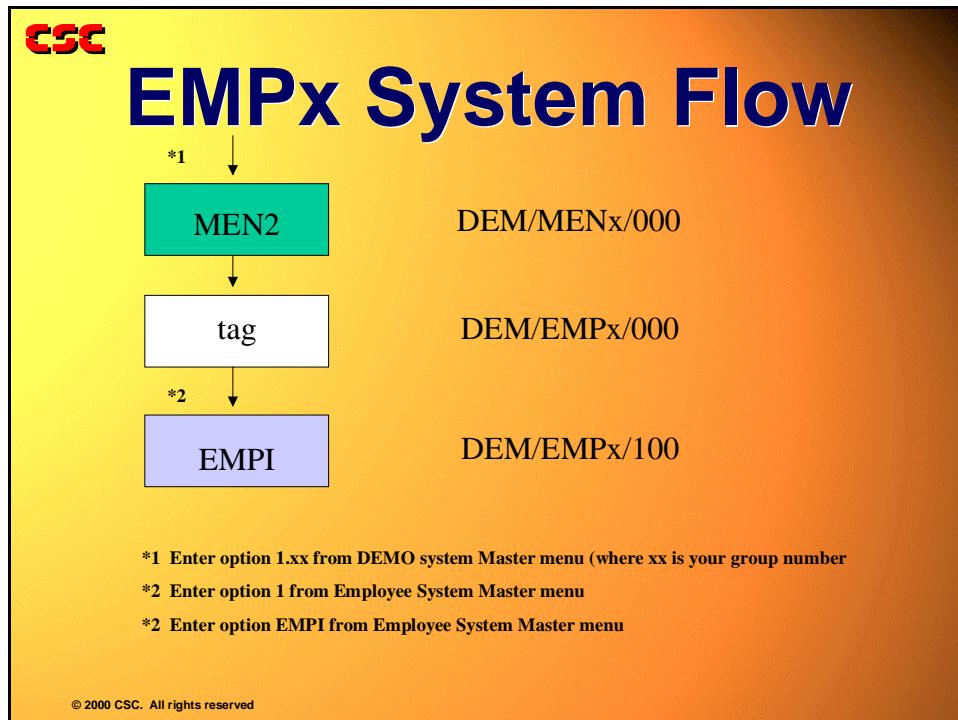
Notes:



Umbrella Programming

The FPS Menu Processor

EMPx System Flow



Notes:



FPS Branch/Exec Logic

24

Purpose



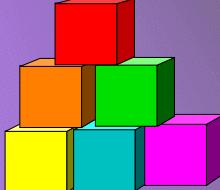
Notes:



Topics

CSC

Topics



- Branch logic
- Exec logic
- Walkthrough of exec logic
- End logic
- Walkthrough of end logic
- Combined exec and end logic

© 2000 CSC. All rights reserved

Objectives

CSC

Objectives



Explore:

- Fields needed to request branch or exec logic
- Difference between branch and exec logic
- Processing flow in conversation using exec logic
- Processing flow in conversation using end logic
- Control entries necessary to increase and decrease processing levels

© 2000 CSC. All rights reserved



Branch/Exec (Increase Level) Logic



Branch/Exec (increase Level) Logic

- Used to pass control from one application or function to another
- Transfer unconditional
- Contains no PEM activities
- Called tag points

© 2000 CSC. All rights reserved

Control is transferred from one FPS control entry to another by specifying a sequence number for the condition code action. Based on this request, FPS would build a new key with the same application ID, the same function ID, and the requested sequence number. While this technique is critical to being able to pass control from one entry to another, there are times in the construction of a scenario when we want to transfer control to a new function, or perhaps even to a completely new application. Branch and level control logic are used within an FPS conversation to pass control from one application and/or function to another. Below is an extract showing only the fields required for branch or exec processing.

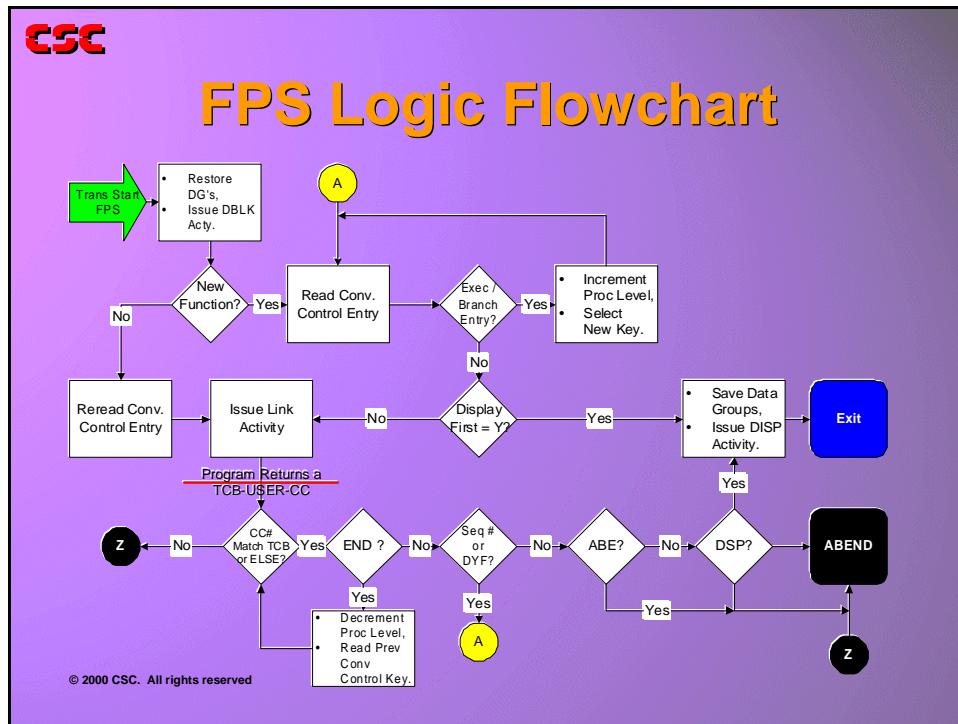
Notice that, unlike the transfer of control which occurs as the result of a condition code action, the transfer of control which occurs as the result of exec/branch logic is *unconditional* and occurs as soon as the entry is read. An entry with exec/branch logic is referred to as a 'Tag Point' entry. Notice that no PEM activities are coded on this type of entry. Additional fields that may be coded are: INIT ACTIVITY/TYPE and SECURITY LEVEL.



Umbrella Programming

Branch/Exec (Increase Level) Logic

FPS Logic Flowchart



Notes:



CSC

Necessary Fields

```

----- ACTION SUCCESSFUL -----
--FPS CONVERSATION CONTROL --
COMMAND ==> INQ          OWNER ==> CC# ==>
APPLICATION   FUNCTION      SEQUENCE           EFFECTIVE DATE ==> 78/01/01
                AAA        FFFF            SSS             COMPANY ======> DFLT
*
SCREEN ID ==> SSSS       DESCRIPTION ==> XXXXXXXXXXXXXXXXXXXXXXXXX
ESTABLISH NEW LEVEL ==>
EXEC/BRANCH KEY ======>

LAST CHG:DATE      TIME      CC#      SRCE      OPER
LAST GEN:DATE      TIME      CC#      SRCE      OPER
PF: 4-CHG 5-ADD 6-INQ 7-CMND 8-COND 9-NXT 10-HSET 11-KEYS ...-GEN ...-CPY ...-DEL

```

Additional fields that may be coded:

INIT ACTIVITY/TYPE
SECURITY LEVEL

© 2000 CSC. All rights reserved

If the NEW LEVEL field contains an 'N', then FPS will execute a branch, which involves the transfer of control to a new item without increasing the processing level of the conversation. In other words, we are causing horizontal (one-way) movement to occur. The following two control entries are examples which request branch processing. *No condition codes are required, since we will not be returning through this entry.*

Notes:



Umbrella Programming

Branch/Exec (Increase Level) Logic

CSC Establish New Level 'N'
Horizontal Branching

ACTION SUCCESSFUL		
FPS	CONVERSATION	CONTROL
COMMAND ==> INQ		OWNER ==> TRD CC# ==>
APPLICATION FUNCTION	SEQUENCE	EFFECTIVE DATE ==> 78/01/01
DEM	PP13	COMPANY ======> DFLT
*		
SCREEN ID ==> PP13	DESCRIPTION ==>	TEST EMP DB VAR DATA PGM
DISPLAY FIRST? ======>		
DISPLAY ACTIVITY ======>		
DEBLOCK ACTIVITY ======>		
LINK ACTIVITY ======>		
ESTABLISH NEW LEVEL =>	<input type="text"/>	
EXEC/BRANCH KEY ======>	UMB PRNT	010
INIT ACTIVITY/TYPE ==>		
EFS ACTION ======>		
EFS KEY ======>		
HIGH USE ======>	N	
USER ENVIRONMENT ======>		
STATUS CODES (1-5) ==>	:	:
SECURITY LEVEL ======>		
LAST CHG:DATE 92/08/21 TIME 9.54.39 CC#	1000	SRCE EDU130 OPER GLT
LAST GEN:DATE 95/01/24 TIME 13.28.44 CC#		SRCE EDU1304 OPER
PF: 4-CHG 5-ADD 6-INQ 7-CMND 8-COND 9-NXT 10-HSET 11-KEYS ..-GEN ..-CPY ..-DEL		

© 2000 CSC. All rights reserved

Notes:



Coding 'Y' for NEW LEVEL causes exec processing to be performed. In exec processing, not only is a new FPS control entry specified for processing, but the processing level is increased as well. In essence this entry is nothing more than a vehicle for continuing on to another entry, while increasing the processing level. Unlike the horizontal branch we saw earlier, condition codes must be present on this type of entry. The following screens illustrate a tag point coded for exec processing.

CSC

EXEC Logic Establish New Level 'Y'

			ACTION SUCCESSFUL
			FPS CONVERSATION CONTROL
COMMAND ==>	NXT		OWNER ==> RPM CC# ==>
APPLICATION	FUNCTION	SEQUENCE	EFFECTIVE DATE ==> 78/01/01
	RPM	ACCD	COMPANY =====> DFLT
*	SCREEN ID ==>	CULO	DESCRIPTION ==> RPM CUST LOCATE
	DISPLAY FIRST? =====>		
	DISPLAY ACTIVITY =====>		
	DEBLOCK ACTIVITY =====>		
	LINK ACTIVITY =====>		
	ESTABLISH NEW LEVEL ==>	Y	
	EXEC/BRANCH KEY =====>	CIS	CULO I00
	INIT ACTIVITY/TYPE ==>		
	EFS ACTION =====>		
	EFS KEY =====>		
	HIGH USE =====>	N	
	USER ENVIRONMENT =====>		
	STATUS CODES (1-5) ==>	:	:
	SECURITY LEVEL =====>		
LAST CHG:DATE	83/05/06 TIME	17.58.53 CC#	5761 SRCE ZLAZEAT OPER
LAST GEN:DATE	86/02/14 TIME	2.21.11 CC#	SRCE JMMFPS OPER
PF:	4-CHG 5-ADD 6-INQ 7-CMND 8-COND 9-NXT 10-HSET 11-KEYS ..-GEN ..-CPY ..-DEL		

Processing level increased

Must have condition codes specified

© 2000 CSC. All rights reserved

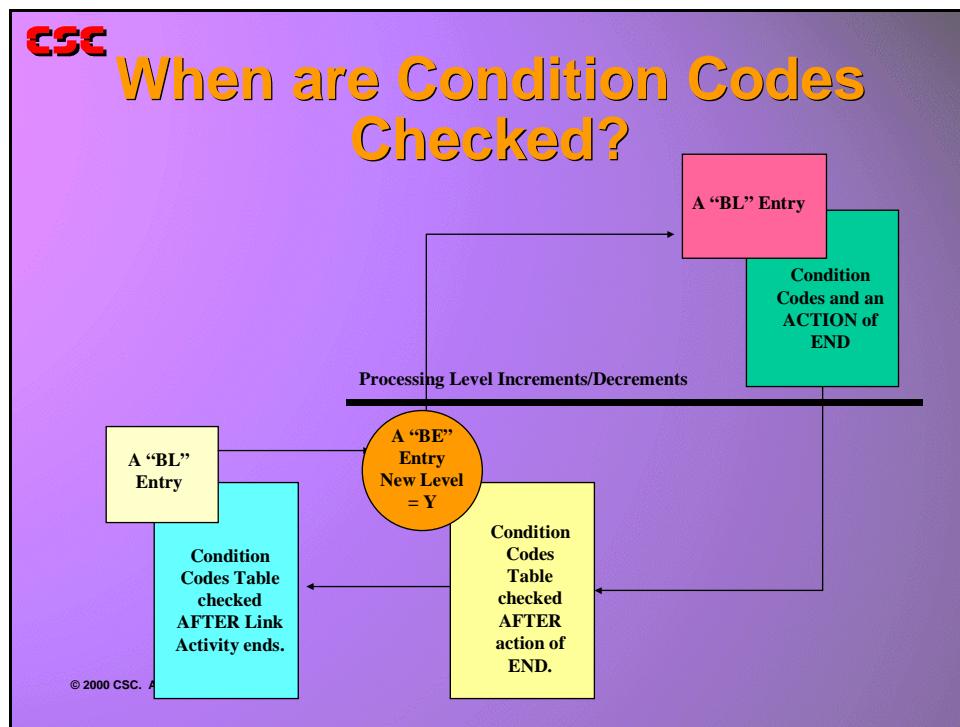
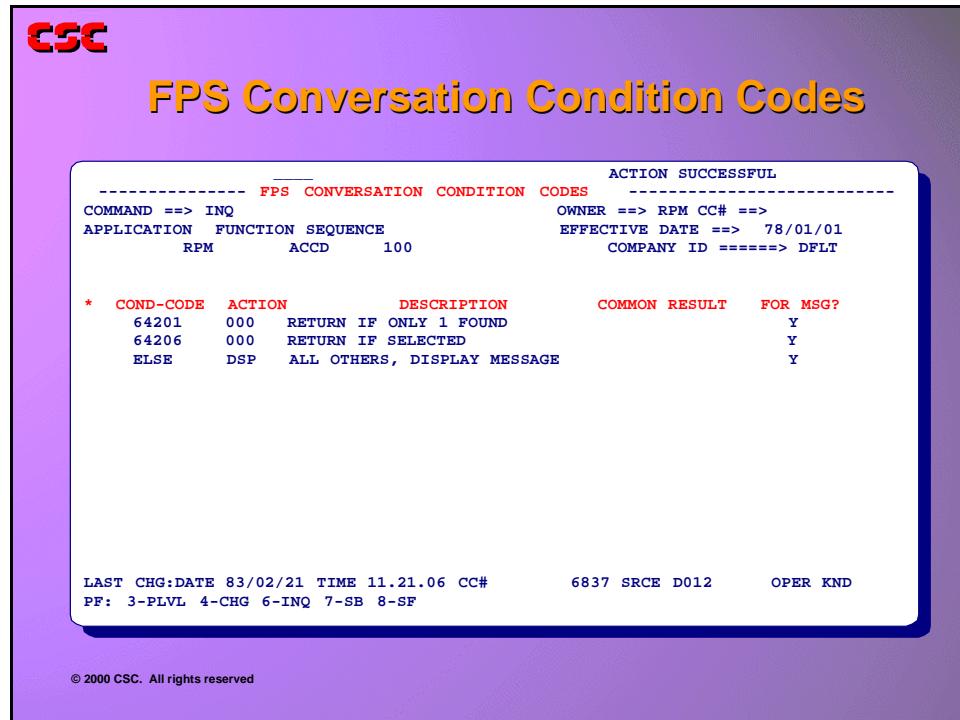
Notes:



Umbrella Programming

Branch/Exec (Increase Level) Logic

To display the FPS CONVERSATION CONDITION CODES press the PF8 key.



FPS Walkthrough--Request for a Data Group

Having previously accessed the UMBRELLA System Master Menu by keying in the 'UMBR' transaction, the operator now enters the prompt 'DG' in the COMMAND field, and presses the ENTER key.

Since the tran code for the UMBRELLA System Master Menu is 'UMBT,' PEM looks up this transaction on the Process Dictionary. The transaction contains three activities. The first, 13587, was also called from the 'UMBR' transaction and allocated FPS data groups. The second, 48300, is a Data Communications activity which deblocks map U48100M. This map contains the FPS header information for the UMBRELLA application. The third, 13551, was also issued by the 'UMBR' transaction and is a link to program I53551, the FPS supervisor.

CSC **FPS Walkthrough - Request for a Data Group**

ACTION COMPLETE			
UMBRELLA TRANSACTION DEFINITION		INQUIRY/MAINTENANCE	
COMMAND ==>	BY TRAN CODE	OWNER ==> UMB CC# ==>	
APPLICATION ID ======>	48		
FUNCTION ID ======>	100		
SOURCE ID ======>	3		
COMPANY ID LIST ======>	ALL		
EFFECTIVE DATE ======>	78/01/01		
TRANSACTION CODE ======>	UMBT	1 OF 2 CODES	
TRANSACTION DESCRIPTION ==>	UMBRELLA FPS TRANS.		
DL/I PSB NAME ======>	PSBUMBA		
DB2 PLAN NAME ======>	&IBAU04		
APPC: REMOTE PEM ==>	CONVERSATION ==>		
TRANSACTION ACTIVITIES			
*	ACTIVITY * ACTIVITY *	ACTIVITY * ACTIVITY *	ACTIVITY
	13587 48300	13551	
LAST CHG:DATE 94/11/07 TIME 19:22:55 CC# SRCE UMB200 OPER			
PF: 3-PLVL 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-ACTS 11-TRAN ..-DEL ..-NEW			

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

Branch/Exec (Increase Level) Logic

CSC

Activity Definition Inquiry/Maintenance

48304 ACTION COMPLETE (DC ACTS)					
COMMAND ==> INQ	ACTIVITY DEFINITION	INQUIRY/MAINTENANCE			
			OWNER ==> UMB CC# ==>		
	ACTIVITY ID ======>	48300	EFF DATE ==>	78/01/01	
	ACTIVITY MNEMONIC ==>		TYPE =====>	DC	
	DESCRIPTION ======>	DBLK U2S INC MAP--V1			
DATA COMMUNICATIONS					
	MAPNAME ==> U48100M		RETRY ==> Y		
	SERVICE ==> DBLK		ERASE ==> Y		
	DISP =====> WAIT		PROMPT ==> N		
* DGID	*** DATA	GROUPS	***		
* 13589	* DGID	* DGID	*	* DGID	
LAST CHG:DATE 94/11/07 TIME 19.22.55 CC# SRCE UMB200 OPER					
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-MAPID 11-CPY ..-NXTT ..-NEW ..-NXTA					

© 2000 CSC. All rights reserved

CSC

Map Definition Base Screen

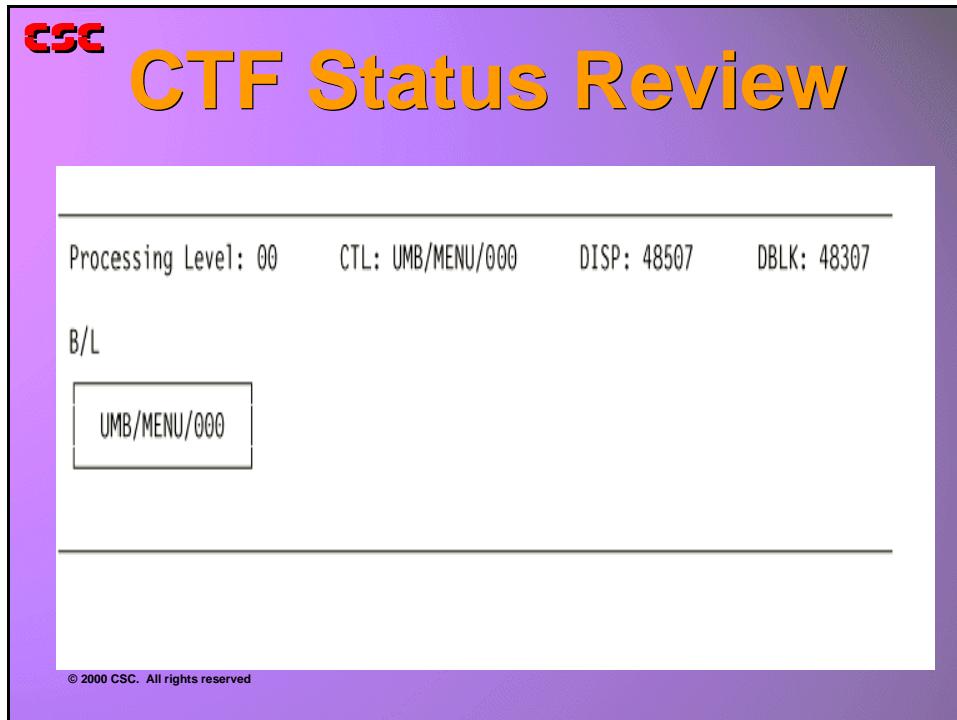
ACTION COMPLETE						
MAP DEFINITION BASE SCREEN						
COMMAND ==> NXT						OWNER ==> UMB CC# ==>
LANGUAGE ==> ENU						
MAP NAME => U48100M DEVICE => A	CO GROUP => ALL		EFF DATE => 78/01/01			
DESCRIPTION ======> UMB INITIAL FPS DEBLOCK MAP--V1						:
LINKNAME ======> U48100MM						
TYPE (I,O,U,S) ======> U						TRANCODE DYNAMIC? =====> N
TRANCODE ======> ----						TRANCODE PROTECTED? ==> N
EXTENDED ATTRIBUTES? ==> N						NON-PERM MAP? ======> N
MAP TYPE (DOC SCR) =====> SCR	MAXIMUM ROWS => 24		MAXIMUM CLMS => 80			
* DGID	* DGID	* DGID	* DGID	* DGID	* DGID	
NEW MAP KEY FOR COPY:						
MAP NAME =>	DEVICE =>	CO GROUP =>	EFF DATE =>			
LANGUAGE ==>						
LAST CHG:DATE 94/11/07 TIME 19.22.55 CC#			SRCE UMB200	OPER		
LAST GEN:DATE 95/01/10 TIME 15.51.44 CC#			SRCE UMB2006	OPER		
PF: 4-CHG 5-ADD 6-INQ 7-BROWSE 8-EDIT 9-NXT 10-GEN 11-DSP ..-DOC ..-CPY ..-GRP						

© 2000 CSC. All rights reserved



SCENARIO - CTF STATUS REVIEW

As the FPS supervisor receives control, review the previous transaction's CTF status.



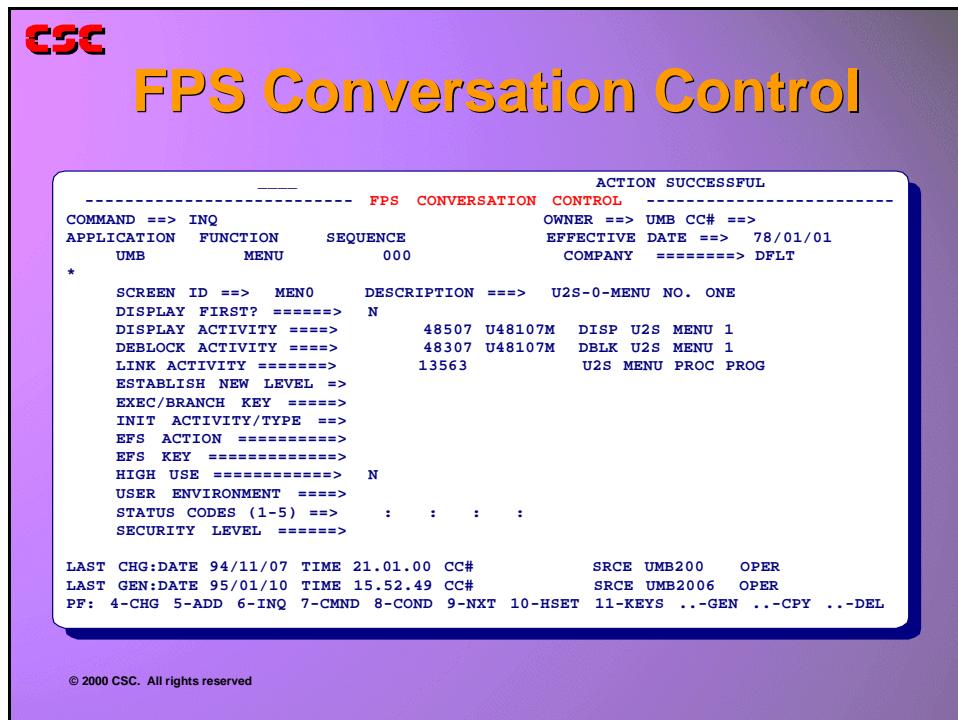
Notes:



Umbrella Programming

Branch/Exec (Increase Level) Logic

The FPS supervisor compares the Application ID of the 'UMBT' transaction with the Application Table and identifies the transaction as one associated with the 'UMB' application. The Function ID of this transaction does not match the field UMB-RESET-TXN, so the supervisor reads the information placed in Data Group 13589 from the header and builds the key to the CTF. The data groups saved on the CTF are restored, and the key to the FPS control entry, which in this case is UMB/MENU/000, is retrieved. The FPS supervisor then reads the item with a userkey of UMB/MENU/000. The deblock activity associated with this entry is executed, causing the 'DG' prompt to be placed into Data Group ID 13589.



Notes:



Umbrella Programming

Branch/Exec (Increase Level) Logic

Data Communications Deblock Activity ID 48307.

CSC

```
48304 ACTION COMPLETE (DC ACTS)
----- ACTIVITY DEFINITION INQUIRY/MAINTENANCE -----
COMMAND ==> INQ OWNER ==> UMB CC# ==>
ACTIVITY ID ======> 48307 EFF DATE ==> 78/01/01
ACTIVITY MNEMONIC ==> TYPE =====> DC
DESCRIPTION ======> DBLK U2S MENU 1

DATA COMMUNICATIONS

MAPNAME ==> U48107M RETRY ==> Y
SERVICE ==> DBLK ERASE ==> Y
DISP ==> WAIT PROMPT ==> N
*** DATA GROUPS ***
* DGID * DGID * DGID * DGID

LAST CHG:DATE 94/11/07 TIME 19.22.55 CC# SRCE UMB200 OPER
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-MAPID 11-CPY ..-NXTT ..-NEW ..-NXTA
```

Data Communications Deblock Activity ID 48307

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

Branch/Exec (Increase Level) Logic

The link activity on the control entry, in this case 13563, is executed next. This activity links to program I53563, which is the menu processor.

The screenshot shows a terminal window with the CSC logo at the top left. The title bar reads "Program Definition Inquiry/Maintenance". The main area displays the following information:

ACTION SUCCESSFUL

PROGRAM DEFINITION INQUIRY/MAINTENANCE

COMMAND ==> INQ OWNER ==> FPS CC# ==>

PROGRAM ID ==> 13563 EFF DATE ==> 78/01/01 PEM TECHNOLOGY => FULLWORD
LINKNAME ==> I53563 LANGUAGE ==> ALC SOURCE NAME ==> I53563
DESCRIPTION => U2S MENU PROC PROG STATUS ==> PROD
HIGH, MED, OR LOW USAGE? ==> LOW USED ONLINE, BATCH OR BOTH? ==> BOTH

DATA GROUPS USED BY PROGRAM

* PP ---DGID---	* PP ---DGID---	* PP ---DGID---
1 1686	2 1452	3 13589
4 13563	5 48278	6 48003

AUTHORIZED ACTIVITIES

* -ACTIVITY- * -ACTIVITY- * -ACTIVITY- * -ACTIVITY- * -ACTIVITY- * -ACTIVITY-
-- ALL --

LAST CHG:DATE 94/11/07 TIME 19.22.55 CC# SRCE UMB200 OPER
PF: 2-XREF 3-PLVL 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-BYLINK 11-BYID ..-ACTS

© 2000 CSC. All rights reserved

Notes:



The menu processor searches for a menu processing entry with the same userkey as the entry which is currently controlling FPS processing, UMB/MENU/000.



FPS Menu Processor Control Data

ACTION SUCCESSFUL							
-----FPS MENU PROCESSOR CONTROL DATA-----							
COMMAND ==> INQ				OWNER ==> UMB CC# ==>			
APPLICATION	FUNCTION	SEQUENCE		EFFECTIVE DATE ==>	78/01/01		
UMB	MENU	000		COMPANY ID =====>	DFLT		
*	PROMPT	APP FUNC SEQ	NEW LVL	*	PROMPT	APP FUNC SEQ	NEW LVL
0	UMB MENU 000			1	UMB MENU 100		
2	UMB MENU 200			3	UMB MENU 930		
4	130 1			5	UMB MENU 600		
6	UMB MENU 960			7	UMB MENU 970		
8	UMB MENU 980			9	UMB MENU 990		
10	UMB MENU 910			DPS	UMB U112 000		
DOC	UMB DOC1 000			ELE	UMB DG03 000		
DCT	UMB U116 000			RCS	UMB RCS1 000		
CC	UMB CCSM 000			X	UMB MENU 000		
DG	UMB DG01 000			TXN	UMB TX01 000		
FMT	UMB CD02 000			CDMF	UMB CDMF 000		

LAST CHG:DATE 94/11/07 TIME 21.01.00 CC# 58793 SRCE UMB200 OPER
PF: 2-XREF 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT ..-DEL

© 2000 CSC. All rights reserved

Having found the correct menu entry, the menu processor finds a match for the 'DG' prompt entered by the operator. The program then places the values 'UMB', 'DG01', and '000' into Data Group 13589, sets a condition code of 0, and ends.

Notes:



Umbrella Programming

Branch/Exec (Increase Level) Logic

The FPS supervisor resumes execution and checks the control entry with a userkey of UMB/MENU/000 for the condition code action associated with a condition code of 0. Since the action is DYF, the supervisor builds the new userkey UMB/DG01/000 from the values placed into Data Group 13589 by the menu processor, finds the Conversation Control entry with that userkey, and uses its instructions to continue processing.

CSC

FPS Conversation Condition Codes

ACTION SUCCESSFUL				
----- FPS CONVERSATION CONDITION CODES -----				
COMMAND ==> INQ	OWNER ==> UMB CC# ==>			
APPLICATION FUNCTION SEQUENCE	EFFECTIVE DATE ==> 78/01/01			
UMB MENU 000	COMPANY ID =====> DFLT			
*	COND-CODE	ACTION	DESCRIPTION	COMMON RESULT FOR MSG?
	00000	DYF	GO TO NEW FUNC BUILT BY MODULE	
	EXIT	000	DUMMY XFER	
	RTN	END	RETURN TO MAIN MENU IF PF1/RTN	
	CONT	END	RETURN TO MAIN MENU IF PF2/RTN	
	ELSE	DSP	ALL OTHERS, DISPLAY MESSAGE	Y

LAST CHG:DATE 94/11/07 TIME 21.01.00 CC#
PF: 3-PLVL 4-CHG 6-INQ 7-SB 8-SF

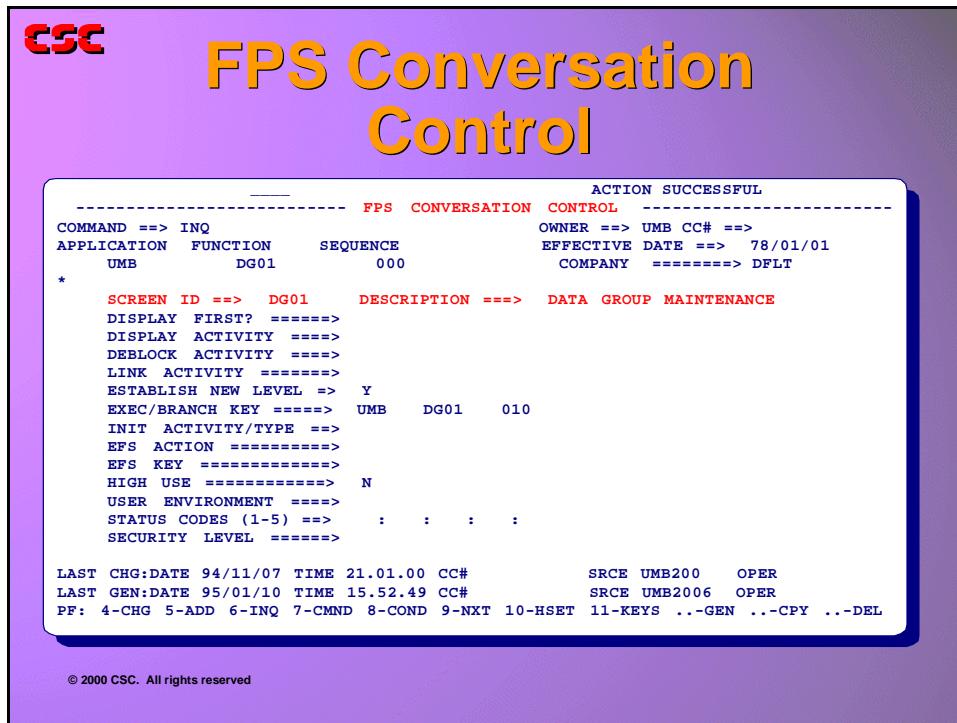
SRCE UMB200 OPER

© 2000 CSC. All rights reserved

Notes:



The control entry with a userkey of UMB/DG01/000 specifies that a new processing level be established and that the item with the userkey UMB/DG01/010 be used to control processing. Therefore, the FPS supervisor causes the data associated with the menu to be saved on the CTF and searches for the control item with the userkey UMB/DG01/010.



Notes:



Umbrella Programming

Branch/Exec (Increase Level) Logic

This entry causes link activity 13, which is a link to a NO-OP program, to be issued. Of course, when this program finishes execution, the value of TCB-USER-CC is still 0.

CSC **FPS Conversation Control**

FPS CONVERSATION CONTROL			ACTION SUCCESSFUL
COMMAND ==> NXT	OWNER ==> UMB CC# ==>	APPLICATION FUNCTION	SEQUENCE
UMB	EFFECTIVE DATE ==> 78/01/01	DG01	010
*	COMPANY =====> DFLT		
SCREEN ID ==> DG01	DESCRIPTION ==> DATA GROUP MAINTENANCE		
DISPLAY FIRST? =====> N			
DISPLAY ACTIVITY =====>	48530 U48130M DISP DG MAP 1		
DEBLOCK ACTIVITY =====>	48330 U48130M DBLK DG MAP 1		
LINK ACTIVITY =====>	13 PEM NO-OP ACTVTY EMU		
ESTABLISH NEW LEVEL =>			
EXEC/BRANCH KEY =====>			
INIT ACTIVITY/TYPE ==>			
EFS ACTION =====>			
EFS KEY =====>			
HIGH USE =====>	N		
USER ENVIRONMENT =====>			
STATUS CODES (1-5) ==>	:	:	:
SECURITY LEVEL =====>			
LAST CHG:DATE 94/11/07 TIME 21.01.00 CC#	SRCE UMB200 OPER		
LAST GEN:DATE 95/01/10 TIME 15.52.49 CC#	SRCE UMB2006 OPER		
PF: 4-CHG 5-ADD 6-INQ 7-CMND 8-COND 9-NXT 10-HSET 11-KEYS ..-GEN ..-CPY ..-DEL			

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

Branch/Exec (Increase Level) Logic

The FPS supervisor checks for the condition code action associated with a condition code of 0 and falls through to the ELSE condition. This line changes the sequence number of the userkey to 100 and sets the Common Result field to 48129.

CSC **FPS Conversation Condition Codes**

FPS CONVERSATION CONDITION CODES				ACTION SUCCESSFUL	
COMMAND ==> INQ	OWNER ==> UMB CC# ==>	APPLICATION FUNCTION SEQUENCE	EFFECTIVE DATE ==> 78/01/01	UMB	COMPANY ID =====> DFLT
DG01	010				
*	COND-CODE	ACTION	DESCRIPTION	COMMON RESULT	FOR MSG?
	EXIT	END	END IF REQUEST TO EXIT	48129	
	ELSE	100	ALL OTHER CONDITION CODES	48129	
LAST CHG:DATE 94/11/07 TIME 21.01.00 CC# PF: 3-PLVL 4-CHG 6-INQ 7-SB 8-SF				SRCE UMB200	OPER

© 2000 CSC. All rights reserved

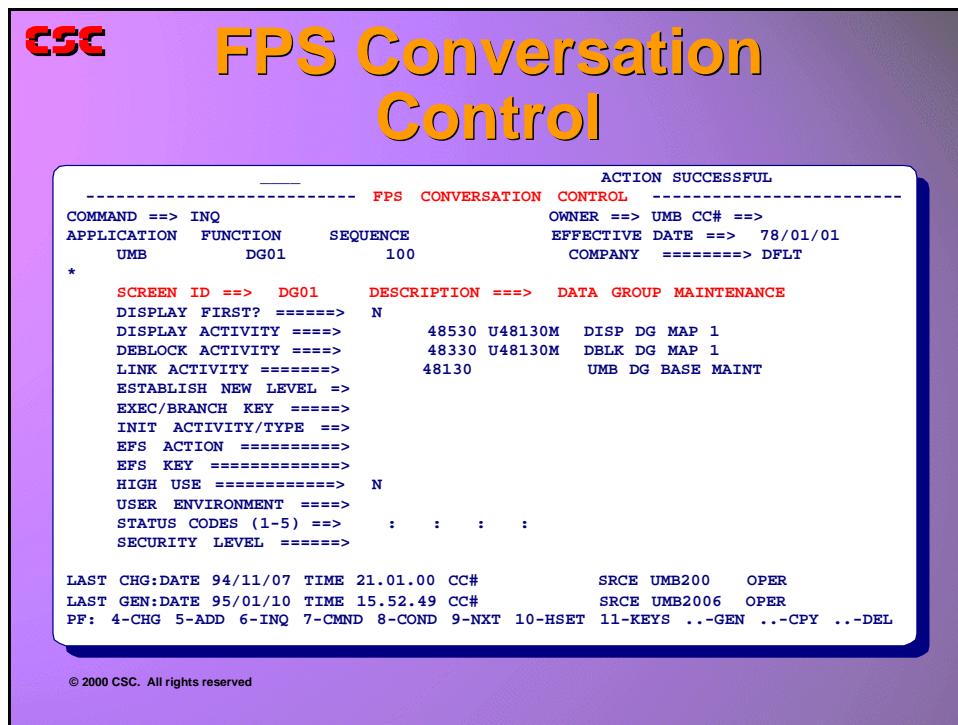
Notes:



Umbrella Programming

Branch/Exec (Increase Level) Logic

The control entries are again searched, this time for userkey UMB/DG01/100. The entry specifies that link activity 48130 be executed. Activity 48130 causes program U48130, the data group base maintenance program, to execute. The program executes and discovers that no Data Group ID has been specified. Therefore, it sets a condition code of 48355 and program U48130 ends.



Notes:



The FPS supervisor checks condition code 48355 against the condition code values listed on the entry with a userkey of UMB/DG01/100. Since there is no exact match, the action associated with the ELSE condition is used, and display activity 48530 is requested. As with all display activities, before the activity is actually executed, all data groups referenced on the map to be displayed, as well as all global data groups and data groups 13590 and 13591, are written to the CTF.

CSC

FPS Conversation Condition Codes

FPS CONVERSATION CONDITION CODES					ACTION SUCCESSFUL
COMMAND ==>	INQ	OWNER ==>	UMB CC# ==>		
APPLICATION	FUNCTION	SEQUENCE	EFFECTIVE DATE ==>	78/01/01	
UMB	DG01	100	COMPANY ID =====>	DFLT	
*	COND-CODE	ACTION	DESCRIPTION	COMMON RESULT	FOR MSG?
	RTN	END	END IF REQUEST TO RETURN---PF1		Y
	CONT	END	END IF REQUEST TO CONTINUE-PF2		Y
	EXIT	END	END IF REQST TO EXIT CONVERSTN		Y
	48149	150	DG VERIFY		
	48135	110	DG ELEMENT MAINT.		
	48033	120	OWNER APPL SCREEN		
	48139	130	DG GENERATION		
	48034	140	XREF SCREEN		
	ELSE	DSP	ALL OTHERS DISP		Y
LAST CHG:DATE 94/11/07 TIME 21.01.00 CC#			SRCE	UMB200	OPER
PF: 3-PLVL 4-CHG 6-INQ 7-SB 8-SF					
© 2000 CSC. All rights reserved					

Notes:



Umbrella Programming

Branch/Exec (Increase Level) Logic

Activity 48530 requests a display of map U48130M. Once this display has been accomplished, the transaction ends. As before, the tran code associated with this map is 'UMBT'.

The screenshot shows a terminal window titled "Activity Definition Inquiry/Maintenance". The title bar includes the CSC logo. The main display area shows the following information:

48304 ACTION COMPLETE (DC ACTS)

COMMAND ==>	INQ	OWNER ==>	UMB CC# ==>
ACTIVITY ID	48530	EFF DATE ==>	78/01/01
ACTIVITY MNEMONIC	DISP	TYPE ==>	DC
DESCRIPTION	DISP DG MAP 1		

DATA COMMUNICATIONS

MAPNAME ==>	U48130M	RETRY ==>	Y
SERVICE ==>	DISP	ERASE ==>	Y
DISP ==>	WAIT	PROMPT ==>	N
*** DATA GROUPS ***			
*	DGID	*	DGID
*	DGID	*	DGID
*	DGID	*	DGID

LAST CHG:DATE 94/11/07 TIME 19.22.55 CC# SRCE UMB200 OPER
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-MAPID 11-CPY ...-NXTT ..-NEW ..-NXTA

© 2000 CSC. All rights reserved

Notes:



The Map Definition for map U48130M.

CSC

Map Definition for Map U48130M

ACTION COMPLETE

----- MAP DEFINITION BASE SCREEN -----

COMMAND ==> NXT OWNER ==> UMB CC# ==>
 LANGUAGE => ENU
 MAP NAME => U48130M DEVICE => A CO GROUP => ALL EFF DATE => 78/01/01

DESCRIPTION =====> UMB DATA GROUP DEFINITION SETUP :
 LINKNAME =====> U48130MM
 TYPE (I,O,U,S) =====> U TRANCODE DYNAMIC? =====> N
 TRANCODE =====> UMBT TRANCODE PROTECTED? =====> N
 EXTENDED ATTRIBUTES? ==> N NON-PERM MAP? =====> N
 MAP TYPE (DOC|SCR) =====> SCR MAXIMUM ROWS => 24 MAXIMUM CLMS => 80

DGID	*	DGID	*	DGID	*	DGID	*
48003							

NEW MAP KEY FOR COPY:
 MAP NAME => DEVICE => CO GROUP => EFF DATE =>
 LANGUAGE =>
 LAST CHG:DATE 94/11/07 TIME 19.22.55 CC# SRCE UMB200 OPER
 LAST GEN:DATE 95/01/10 TIME 15.51.44 CC# SRCE UMB2006 OPER
 PF: 4-CHG 5-ADD 6-INQ 7-BROWSE 8-EDIT 9-NXT 10-GEN 11-DSP ...DOC ...CPY ...GRP

© 2000 CSC. All rights reserved

The Screen display for map U48130M.

CSC

Screen Display for Map U48130M

48355 ENTER REQUEST

----- PROCESS DICTIONARY DATA GROUP DEFINITION SETUP -----
 COMMAND ==> OWNER ==> CC# ==>

SPECIFY/VERIFY PARAMETERS BELOW:
 DATA GROUP ID NUMBER =====> DATA DICTIONARY ID =====>
 EFFECTIVE DATE =====>

DATA GROUP DESCRIPTION =====>
 LENGTH OF DATA GROUP =====>
 INITIALIZE (YES,NO,FLD) ==>
 INIT. CHARACTER IF 'YES' ==> X
 DATA GROUP TYPE =====> CSA ELIGIBLE ==>
 COPY BOOK NAME: COBOL =====> ALC ==>

NEW DATA GROUP KEY FOR COPY:
 DATA GROUP ID==> ALIAS==> EFFECTIVE DATE==>

LAST CHG:DATE TIME CC# SRCE OPER
 LAST GEN:DATE TIME CC# SRCE OPER
 PF: 2-XREF 4-CHG 5-ADD 6-INQ 7-BROWSE 8-EDIT 9-NXT 10-GEN 11-CPY ..-DEL

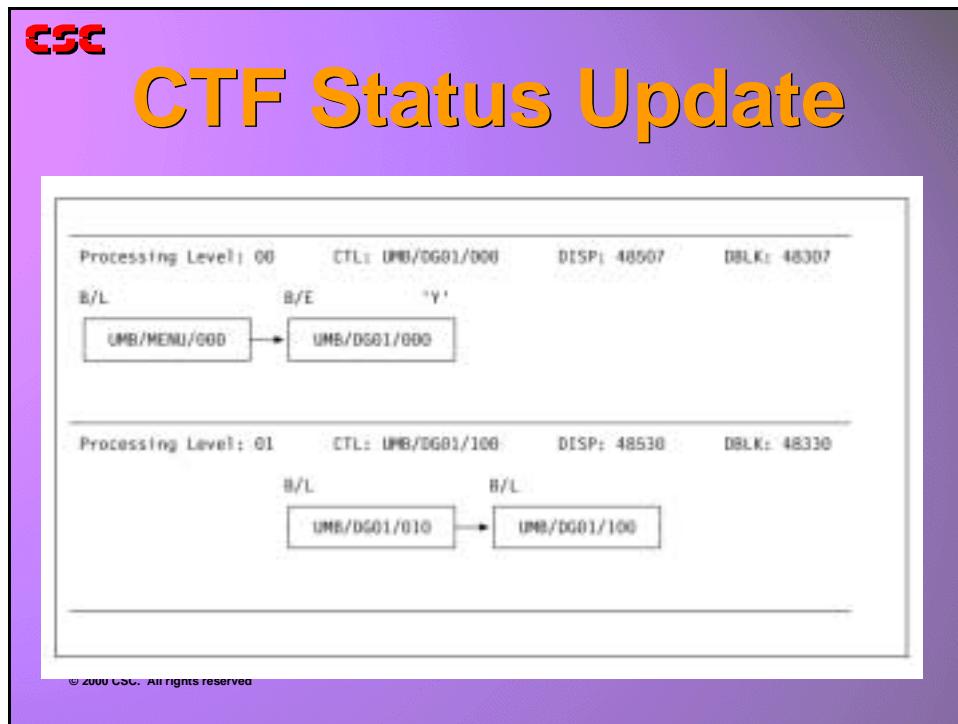
© 2000 CSC. All rights reserved



Umbrella Programming

Branch/Exec (Increase Level) Logic

SCENARIO - CTF STATUS UPDATE



Notes:



Action -END- (Decrease Level) Logic

When an FPS conversation condition code entry is associated with the 'END' action, the effects are as follows:

1. The current processing level number is decreased by one
2. The new current processing level status and datagroups are restored from CTF
3. The control entry for the new current processing level to be established is reread
4. The condition codes in the reread entry are checked to determine what action to take next.

Walkthrough of END Logic

Having completed an inquiry on a data group definition, the operator enters the 'PLVL' command. Since the 'UMBT' trancode was associated with the data group maintenance map, PEM again looks up the transaction definition associated with this trancode and again executes a work area activity to allocate the data groups used by FPS, a data communications activity to deblock the map's header, and a link to the FPS supervisor.

The FPS supervisor uses the header information to locate the appropriate record on the CTF and restores the data associated with the data group maintenance map. This record contains the ID of the deblock activity listed on the last control entry, activity 48330. This activity is executed, causing the remainder of the map to be deblocked.

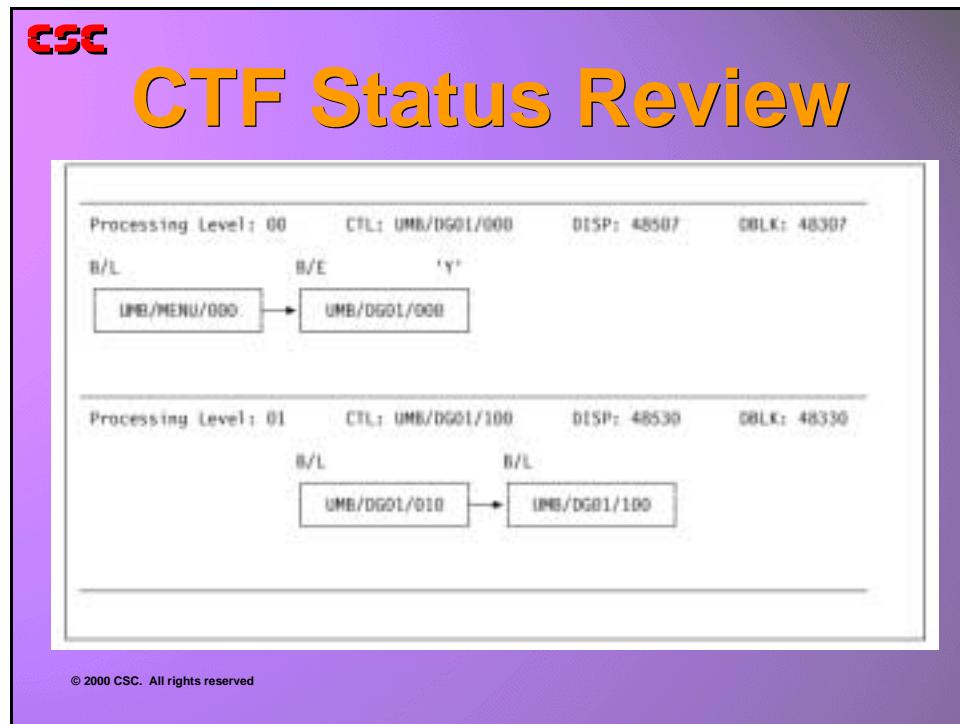
Notes:



Umbrella Programming

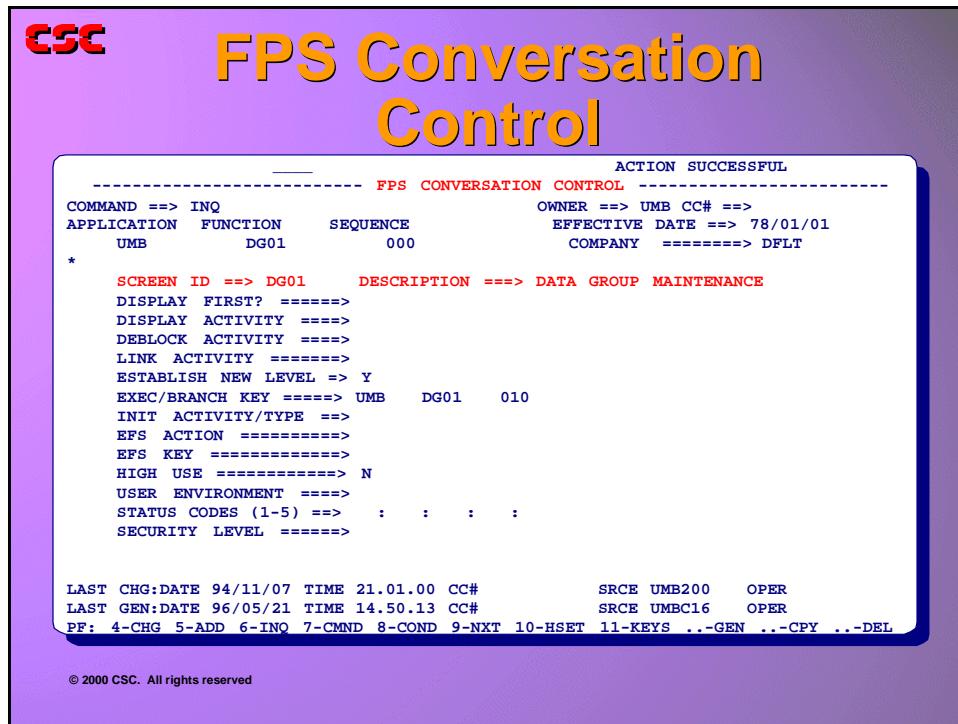
Action -END- (Decrease Level) Logic

SCENARIO - CTF STATUS REVIEW



Notes:





The FPS supervisor recognizes the 'PLVL' request as an FPS primary command. Therefore, the data on the CTF associated with the current processing level is deleted and the data associated with the previous level is restored. The entry last associated with the previous processing level, userkey UMB/DG01/000, is retrieved and is used to determine the action associated with the 'PLVL' command.

Notes:



Umbrella Programming

Action -END- (Decrease Level) Logic

The screenshot shows a terminal window titled "FPS Conversation Condition Codes". The title bar has the CSC logo on the left. The main area displays a command execution log:

ACTION SUCCESSFUL			
----- FPS CONVERSATION CONDITION CODES -----			
COMMAND ==> INQ	OWNER ==> UMB CC# ==>		
APPLICATION FUNCTION SEQUENCE	EFFECTIVE DATE ==> 78/01/01		
UMB DG01 000	COMPANY ID =====> DFLT		
* COND-CODE ACTION DESCRIPTION COMMON RESULT FOR MSG?			
EXIT END END IF REQUEST TO EXIT 48129			
ELSE DSP ALL OTHER CONDITION CODES 48129			

At the bottom of the window, there is a status message: "LAST CHG:DATE 94/11/07 TIME 21.01.00 CC# SRCE UMB200 OPER PF: 3-PLVL 4-CHG 6-INQ 7-SB 8-SF".

Small text at the bottom left reads: "© 2000 CSC. All rights reserved".

Since there is no specific action requested for PLVL, the default action associated with the ELSE condition, which is a display, is requested. Activity 48507 was the last display activity issued before the processing level was increased, so the information to be saved is again written to the CTF, and the map specified on this activity, Map ID U48107M, is displayed.

Notes:



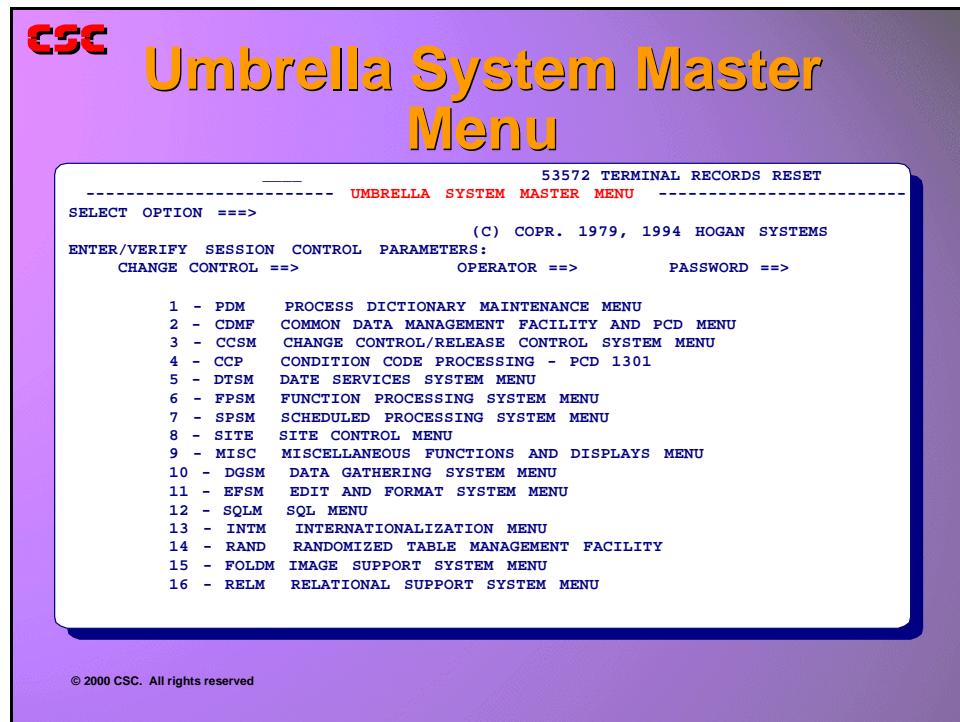
CSC

Map Definition Base Screen



Umbrella Programming

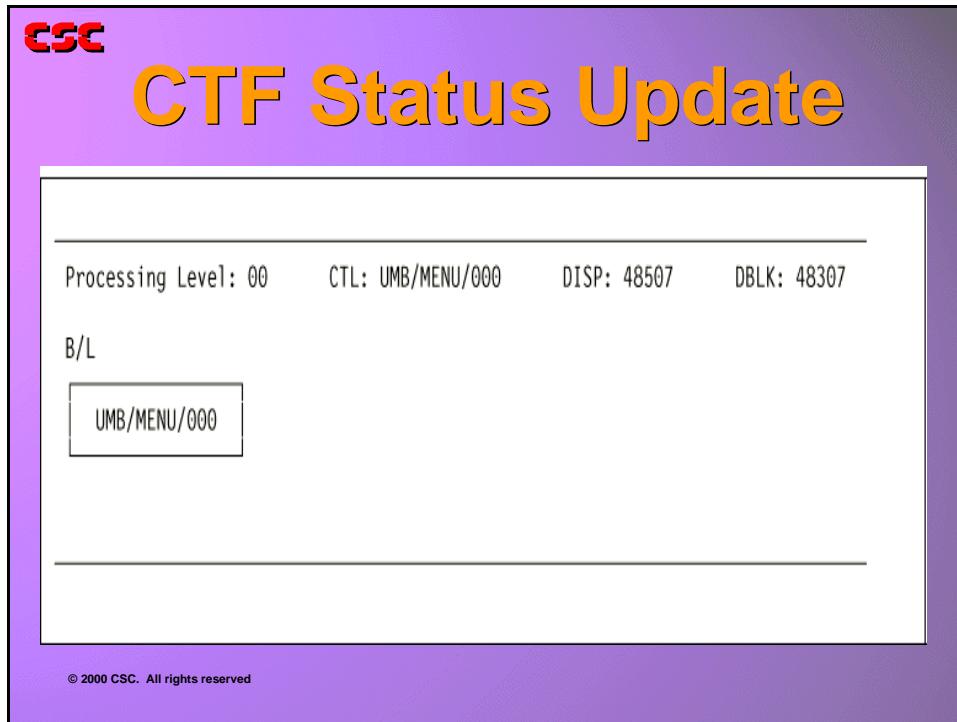
Action -END- (Decrease Level) Logic



Notes:



SCENARIO - CTF STATUS UPDATE



Notes:

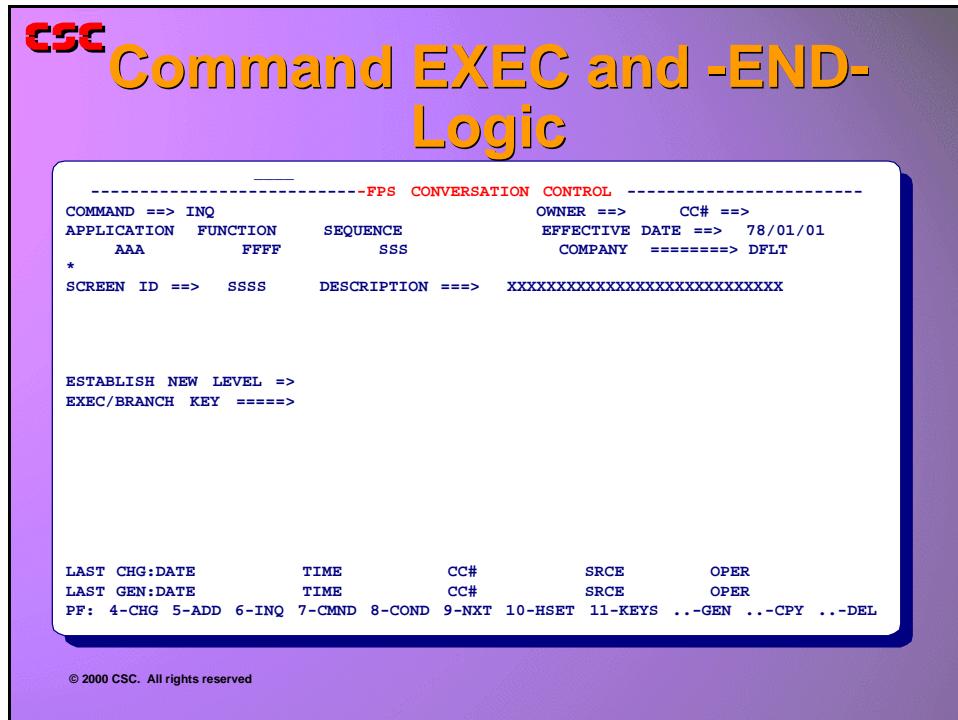


Umbrella Programming

Combined EXEC and END Logic

Combined EXEC and END Logic

Following are two extracts showing all fields required to increase the processing level, branch to a new control entry, and return to this entry to check condition codes. Remember, the condition codes are used only to determine the next action to take upon return. They are not used when increasing the processing level.



The slide features the CSC logo in red and blue at the top left. The main title "Command EXEC and -END- Logic" is displayed in large yellow text. Below the title is a command definition table:

COMMAND ==> INQ		OWNER ==>		CC# ==>
APPLICATION	FUNCTION	SEQUENCE		EFFECTIVE DATE ==> 78/01/01
AAA	FFFF	SSS		COMPANY =====> DFLT
*				
SCREEN ID ==>	SSSS	DESCRIPTION ==>	XXXXXXXXXXXXXXXXXXXXXX	

Below this is a section titled "ESTABLISH NEW LEVEL =>" followed by "EXEC/BRANCH KEY =====>". At the bottom of the slide is a footer: "© 2000 CSC. All rights reserved".

Notes:



CSC

Condition Codes for END LOGIC

FPS CONVERSATION CONDITION CODES				
COMMAND ==>	OWNER ==>	CC# ==>		
APPLICATION FUNCTION SEQUENCE	EFFECTIVE DATE ==>	78/01/01		
AAA FFFF SSS	COMPANY ID =====>	DFLT		
*	COND-CODE ACTION	DESCRIPTION	COMMON RESULT	FOR MSG?
	ELSE			

LAST CHG:DATE TIME CC# SRCE OPER
PF: 3-PLVL 4-CHG 6-INQ 7-SB 8-SF

© 2000 CSC. All rights reserved

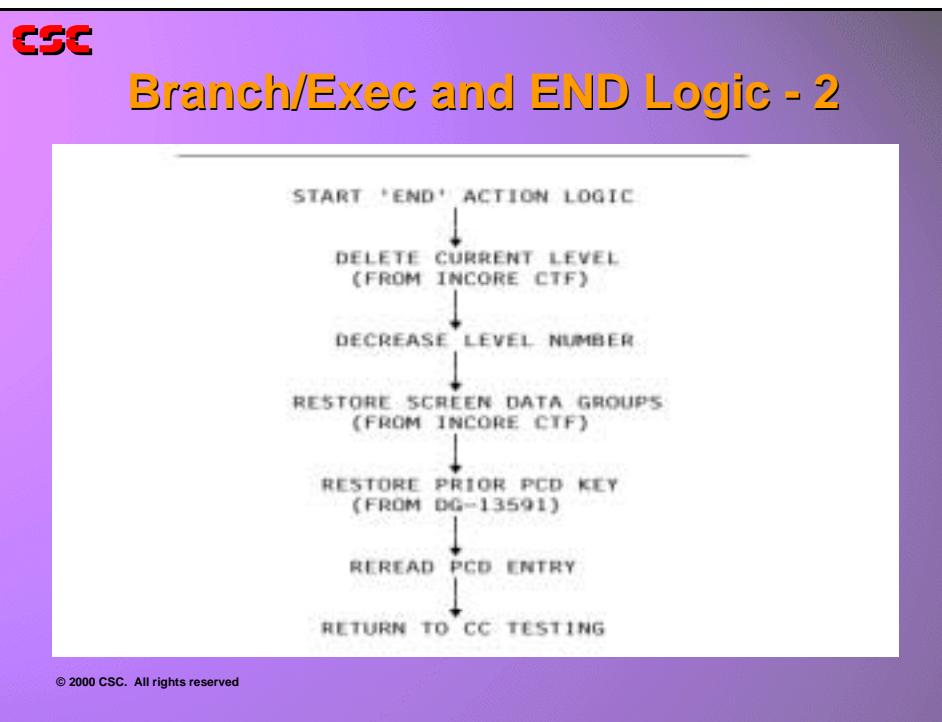
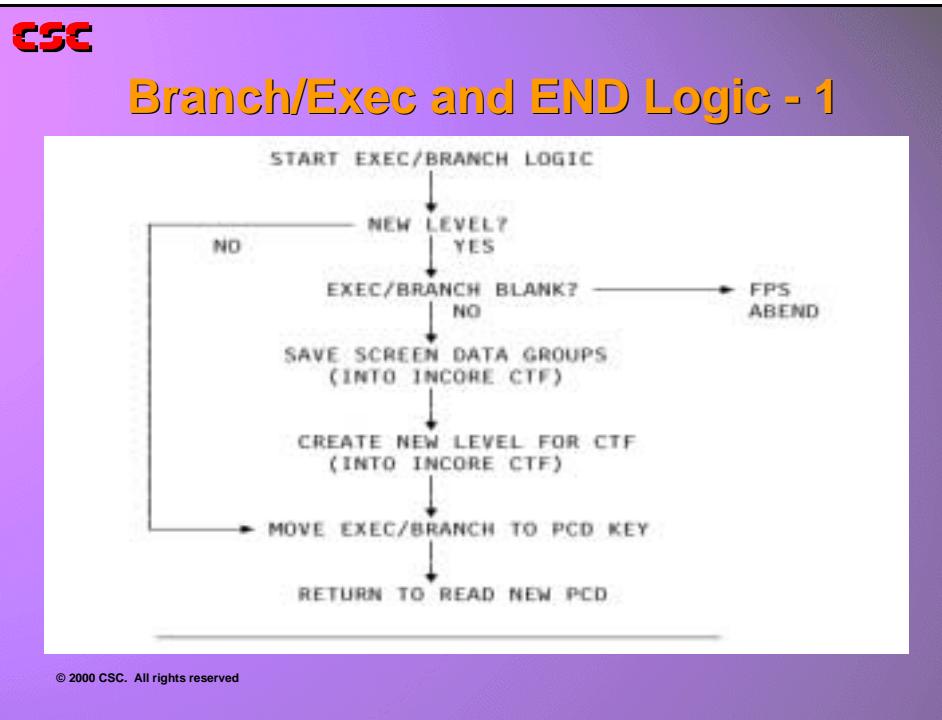
Notes:



Umbrella Programming

Combined EXEC and END Logic

Branch/Exec and END Logic

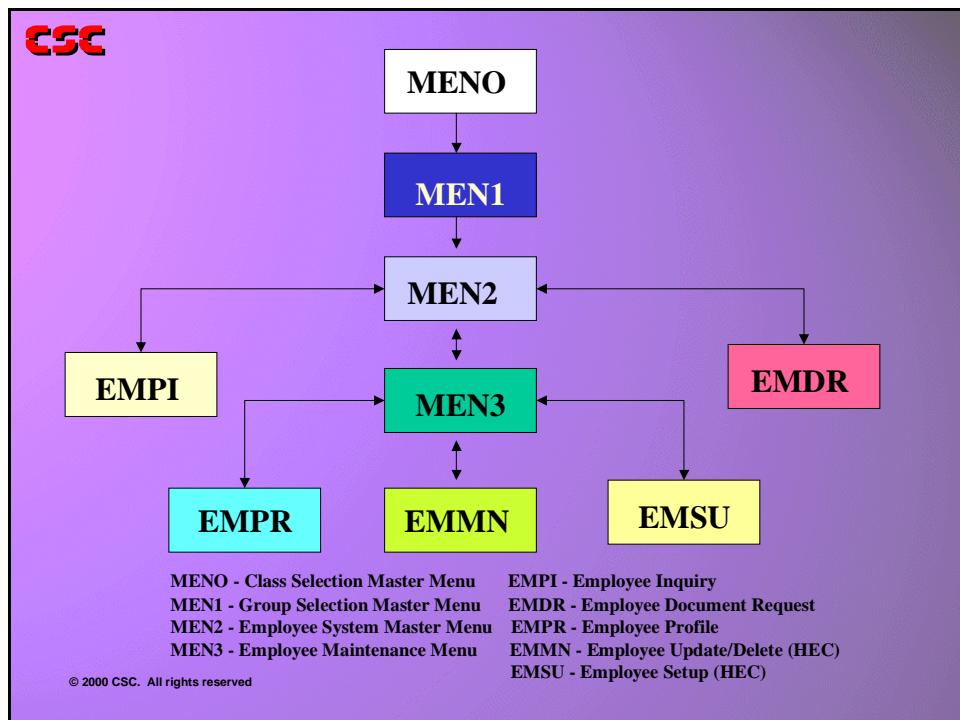


The Employee System - Scenario Flow

The exercises in this class are integrated around the development of a hypothetical Employee System. Through the development of the system, the student will be able to gain practice in maintaining many of the components of an FPS application.

The diagram shows the first five levels of the Employee System, beginning with the Demo System Class Selection Menu. This diagram provides students with a familiarity of the design of the system as well as introduces them to the overall flow through the screens.

The students will be providing additional pieces to the system through the completion of a series of exercises. The first exercise involves enhancing the Menu Scenario Flow, starting with function 'EMPR' shown on the fifth level, first box on the left.



Notes:



Umbrella Programming

The Employee System - Scenario Flow

Level Control Exercise I



In this exercise you will build a scenario which uses the following two functions, where "xx" is your group number:

EPxx - Employee Profile

ECxx - Employee Current Pay Data

Employee Profile is the function invoked when the operator makes selection "1" from the Employee Maintenance Menu. We have assigned an FPS Control userkey of "DEMEPxx000" to this selection, where "xx" is your group number.

1. Add "DEMEPxx000" to the FPS Control entries for company DFLT and effective date 78/01/01. This item should increase the processing level and branch to a new entry that represents the 'EPxx' function above. (Also see the diagram on the next page. The new entries are identified by path indicators.)
2. Create the necessary Control entries to build the scenario shown on the next page. Make sure to enter your screen names (e.g. EPxx) in the 'SCRN-ID' field on the Control entries. This will make it easy for you to recognize that you are executing your scenario when you are testing your work.
3. Required function information is given in the table below:

FUNCTION ID	DISPLAY ACTIVITY	DEBLOCK ACTIVITY	LINK ACTIVITY	TCB-USER ENVMT
EPxx	47370	47371	47378	0
ECxx	47372	47373	47379	0

4. Pertinent condition codes returned by the program are listed below:

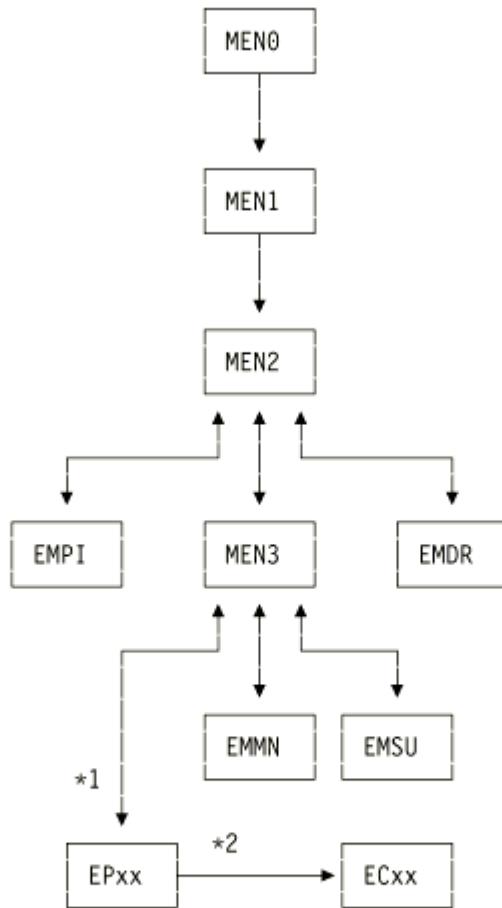
EPxx (PGM Z47350) - 47356 - Request for ECxx

When you first create your Control entries, delete all default condition codes and condition code actions except for the ELSE condition code. NOTE: This is not the only condition code that may be returned by program Z47350. However, at this time it is the only one we wish to 'trap'.

5. After coding the necessary information, test your scenario on the DEM system. From a cleared screen enter the reset transaction for the DEM system, 'DEMR.' From the SYSTEM CLASS MENU, make selection '1.xx.2', where "xx" is your group number, e.g. group 1 enters '1.1.2' and group 12 enters '1.12.2'. This takes you to the Employee Maintenance Menu. From this menu make selection '1'. Before pressing ENTER you may enter a value for COMPANY and EMPLOYEE ID. Use Company '1' and Employee ID '1111'. Alternatively, you may wait to enter these values in the 'EPxx' screen (along with an action of 'INQ'). From the 'EPxx' screen enter 'EMCP' in the ACTION field, provided the data for employee 1111 was successfully displayed previously.



Level Control Exercise I Flow



MEN0 – Class Selection Master Menu
 MEN1 – Group Selection Master Menu
 MEN2 – Employee System Master Menu
 MEN3 – Employee Maintenance Menu
 EMDR – Employee Document Request

ECxx – Employee Current Pay
 EPxx – Employee Profile
 EMMN – Employee Update/Delete
 EMSU – Employee Setup
 EMPI – Employee Inquiry

*1. By '1' or 'EMPR' in the Employee Maintenance menu (Processing level is increased)

*2. By 'EMCP' in the action field of 'EPxx' screen.



Umbrella Programming

The Employee System - Scenario Flow

Level Control Exercise I Scenario Worksheet



Optional Level Control Exercise II

In this exercise you will build the scenario shown on the next page. To simplify your tasks this exercise will be broken down into three (3) phases. Each phase will walk you through a task that will allow you to complete a basic unit of the overall structure. The next phase requires that you build on the preceding phase(s). Once all phases have been completed, the scenario on the next page will be complete and can then be tested. Please read the instructions very carefully for each phase.

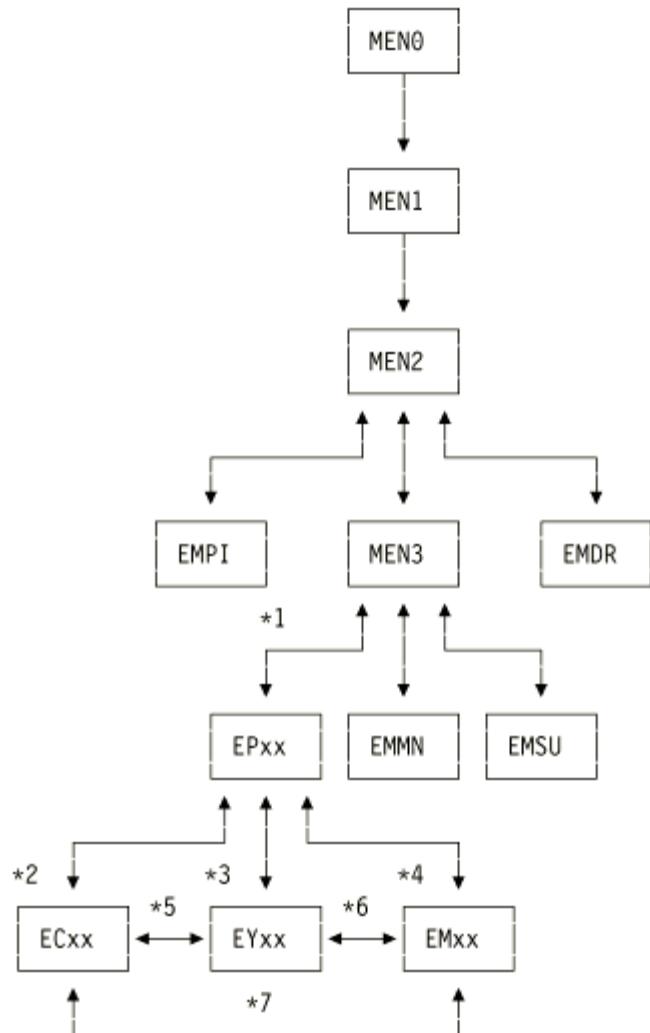
Notes:



Umbrella Programming

The Employee System - Scenario Flow

Level Control Exercise II Flow



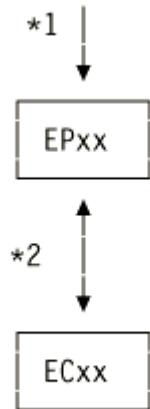
MEN0 - Class Selection Master Menu
 MEN1 - Group Selection Master Menu
 MEN2 - Employee System Master Menu
 MEN3 - Employee Maintenance Menu
 EMDR - Employee Document Request
 EYxx - Employee Y-T-D Pay

ECxx - Employee Current Pay
 EPxx - Employee Profile
 EMMN - Employee Update/Delete
 EMSU - Employee Setup
 EMPI - Employee Inquiry
 EMXX - Employee Misc. Information



Phase I

In this phase you will build the part of the scenario shown below. These pieces should look very familiar to you (review Level Control Exercise I).



- *1. By '1' or 'EMPR' in the Employee Maintenance menu (Processing level is increased)
- *2. By 'EMCP' in the action field of 'EPxx' screen.



Umbrella Programming

The Employee System - Scenario Flow

The information provided in all but the first step below is copied from the previous exercise. Verify that your data is consistent with these items.

1. Create the necessary FPS Control entries to build the scenario shown above. Make sure to enter your screen names (e.g. EPxx) in the 'SCRN-ID' field on the entries. This will make it easy for you to recognize that you are executing your scenario when you are testing your work.
2. Required function information is given in the table below:

FUNCTION ID	DISPLAY ACTIVITY	DEBLOCK ACTIVITY	LINK ACTIVITY	TCB-USER ENVMT
EPxx	47370	47371	47378	0
ECxx	47372	47373	47379	0

3. Pertinent condition codes returned by the program are listed below. Delete all default condition codes and condition code actions except for the ELSE condition code.

EPxx (PGM Z47350) - 47356 - Request for ECxx

NOTE: This is not the only condition code that may be returned by program Z47350. However, at this time it is the only one we wish to 'trap'.

4. Test your newly created scenario using the same test data from the previous exercise (COMPANY=1 and EMPLOYEE ID=1111).



Optional Exercise II - Phase I Scenario Worksheet

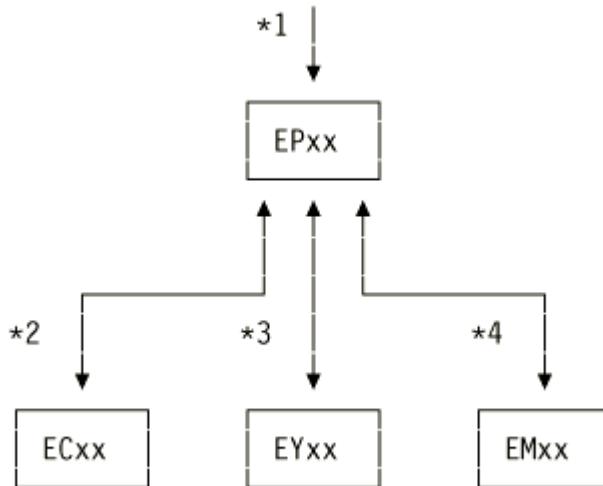


Umbrella Programming

The Employee System - Scenario Flow

Phase II

In this phase you will modify the scenario you built in the last phase according to the diagram below. Notice that paths 3 and 4 should be identical in nature to path 2 which you just created.



- *1. By '1' in the Employee Maintenance menu. (Processing level is increased)
- *2. By 'EMCP' in the ACTION field of the 'EPxx' screen.
Return by PF3 only.
- *3. By 'EMYP' in the ACTION field of the 'EPxx' screen.
Return by PF3 only.
- *4. By 'EMMI' in the ACTION field of the 'EPxx' screen.
Return by PF3 only.



Umbrella Programming

The Employee System - Scenario Flow

1. Create the necessary FPS Control entries to build the scenario shown above. Make sure to enter your screen names (e.g. EPxx) in the 'SCRN-ID' field. This will make it easy for you to recognize that you are executing your scenario when you are testing your work.
2. Required function information is given in the table below.

FUNCTION ID	DISPLAY ACTIVITY	DEBLOCK ACTIVITY	LINK ACTIVITY	TCB-USER ENVMT
EPxx	47370	47371	47378	0
ECxx	47372	47373	47379	0
EYxx	47374	47375	47380	0
EMxx	47376	47377	47381	0

3. Pertinent condition codes returned by the programs are listed below. Delete all default condition codes and condition code actions except for the ELSE condition code.

EPxx (PGM Z47350)

47356 - Request for ECxx

47357 - Request for EYxx

47358 - Request for EMxx

4. Test your newly created scenario using the same test data from the previous exercise (COMPANY=1 and EMPLOYEE ID=1111).



Umbrella Programming

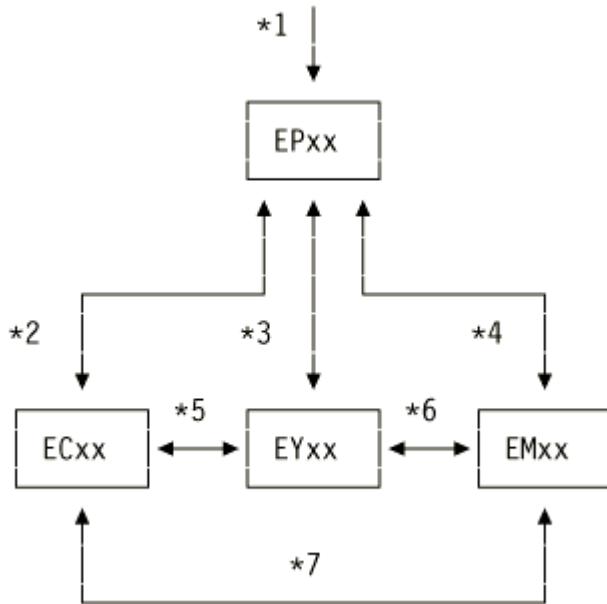
The Employee System - Scenario Flow

Exercise II - Phase II Scenario Worksheet



Optional Phase III

In this phase you will modify the scenario you built in the last phase according to the diagram below. After completing this phase, the exercise will be complete.



- *1. By '1' in the Employee Maintenance menu. (Processing level is increased)
- *2. By 'EMCP' in the ACTION field of the 'EPxx' screen. Return by PF3 only.
- *3. By 'EMYP' in the ACTION field of the 'EPxx' screen. Return by PF3 only.
- *4. By 'EMMI' in the ACTION field of the 'EPxx' screen. Return by PF3 only.
- *5. By 'EMYP' in the ACTION field of the 'ECxx' screen. Return by 'EMCP' in the ACTION field of the 'EYxx' screen.
- *6. By 'EMMI' in the ACTION field of the 'EYxx' screen. Return by 'EMYP' in the ACTION field of the 'EMxx' screen.
- *7. By 'EMCP' in the ACTION field of 'EMxx' screen. Return by 'EMMI' in the ACTION field of the 'ECxx' screen.



Umbrella Programming

The Employee System - Scenario Flow

1. Create the necessary FPS control entries to build the scenario shown above. Make sure to enter your screen names (e.g. EPxx) in the 'SCRN-ID' field. This will make it easy for you to recognize that you are executing your scenario when you are testing your work.

Use an FPS technique to complete this phase without adding a single new Control entry. Review END action logic flow.

HINT: Add a condition code on the FPS Control Entry with an END action to return to the previous level Control Entry. Then, add the same Condition Code to the Tag Point with a branch-to SEQ#.

2. Required function information is given in the table below:

FUNCTION ID	DISPLAY ACTIVITY	DEBLOCK ACTIVITY	LINK ACTIVITY	TCB-USER ENVMT
EPxx	47370	47371	47378	0
ECxx	47372	47373	47379	0
EYxx	47374	47375	47380	0
EMxx	47376	47377	47381	0

3. Pertinent condition codes returned by the programs are listed below. Delete all default condition codes and condition code actions except for the ELSE condition code.

EPxx (PGM Z47350)

47356 - Request for ECxx
47357 - Request for EYxx
47358 - Request for EMxx

ECxx (PGM Z47351)

47357 - Request for EYxx
47358 - Request for EMxx

EYxx (PGM Z47352)

47356 - Request for ECxx
47358 - Request for EMxx

EMxx (PGM Z47353)

47356 - Request for ECxx
47357 - Request for EYxx

4. Test your newly created scenario using the same test data from the previous exercise (COMPANY=1 and EMPLOYEE ID=1111).



Exercise II - Phase III Scenario Worksheet



Umbrella Programming

The Employee System - Scenario Flow



24-50

FPS Branch/Exec Logic

Sort Activities

25

Purpose

The slide has a purple background. In the top left corner is the red CSC logo. In the top right corner is a cartoon illustration of a person standing next to a large orange shelving unit filled with white boxes. One box on the shelf is falling out. In the center of the slide, the text 'Introduce and explain the Sort Activity Type' is written in a large, bold, black serif font. At the bottom left, there is small text that reads '© 2000 CSC. All rights reserved'. At the bottom right, there is small text that reads '7/5/00'.

© 2000 CSC. All rights reserved

7/5/00

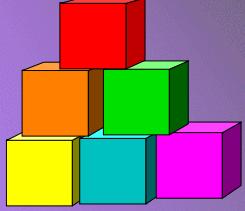
Notes:



Topic

CSC

Topic



Sort Activities

© 2000 CSC. All rights reserved

7/5/00

Objectives

CSC

Objectives



- Describe the EXIT types of Sort Activities
- Add Sort activity definitions to the Process Dictionary

© 2000 CSC. All rights reserved

7/5/00



Sort Activity

CSC

SORT Activity

- Like COBOL sort verb
- Application action opposite to exit type
- Input
 - Application program writes records to SORT
 - Application program closes sort file
 - Sort executes SORT parameters
 - PEM writes all sorted records to named SDB

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

Sort Activity



SORT Activity

- Output
 - PEM passes all records in named SDBs to SORT
 - Sort executes SORT parameters
 - Application program reads records from SORT
- Both
 - Application program writes records to SORT
 - Application program closes SORT file
 - Sort executes SORT parameters
 - Application program reads records from SORT

© 2000 CSC. All rights reserved

The sort activity enables the application program to dynamically sort a file without a separate job step. It is analogous to the COBOL verb SORT.

The sort activity interface is identical to sequential data base activities. That is, data bases are opened, closed, read, and written. The type of sort exit that is used depends on the activity definition.

If the input sort activity is used, the application program will be writing records to sort. When the application program requests a close action, PEM will then write all the sorted records to the sequential data base named in the sort activity.

If the output sort activity is used, the application program will be reading the output of sort. That is, PEM will pass all the records in the sequential data base named in the activity to sort and will pass the output records one at a time to the application program.

If the both sort activity is used, the application program will write records to sort, and after closing, read all the sorted records from sort. When the both option is used, no sequential data base is used.

The exit-type parameter refers to the activity's relationship to the sort program, while the default action represents the basic processing needs of your source code.

The default action will only work for one-half of the processing. Either the read or the write may be chosen. During the opposite half of processing, you must move the appropriate action to the DG ACTION FIELD.



Sort Activity Definition - IN(PUT)

CSC

SORT Activity Definition - IN(PUT)

```

----- ACTIVITY DEFINITION INQUIRY/MAINTENANCE -----
COMMAND ==> INQ OWNER ==> RPM CC# ==>

ACTIVITY ID ======> 8262 EFF DATE ==> 78/01/01
ACTIVITY MNEMONIC ==> TYPE =====> SORT
DESCRIPTION ======> RPM ANA TRG SRT

DISPOSITION ==> WAIT EXIT TYPE ======> IN FORMAT =====> F
DATA GROUP ==> 8250 DEFAULT ACTION ==> WRITE CORESIZE ==>
                SORT CARDS
1==> SORT FIELDS=(01,80,BI,A)
2==>

***** INPUT FILES *****
* FILENAME           * FILENAME           * FILENAME
RMAPSOUT

LAST CHG:DATE 83/03/31 TIME          CC#          SRCE RPS130   OPER KND
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-DGID 11-CPY ..-NXTT ..-NEW ..-NXTA

```

© 2000 CSC. All rights reserved

Sort Activity Definition - OUT(PUT)

CSC

SORT Activity Definition - OUT(PUT)

```

----- ACTIVITY DEFINITION INQUIRY/MAINTENANCE -----
COMMAND ==> INQ OWNER ==> SPS CC# ==>

ACTIVITY ID ======> 16400 EFF DATE ==> 78/01/01
ACTIVITY MNEMONIC ==> TYPE =====> SORT
DESCRIPTION ======> READ SORT REPORTS

DISPOSITION ==> WAIT EXIT TYPE ======> OUT FORMAT =====>
DATA GROUP ==> 3601 DEFAULT ACTION ==> READ CORESIZE ==>
                SORT CARDS
1==> SORT FIELDS=(018,001,BI,A,009,009,BI,A,019,030,BI,A)
2==>

***** INPUT FILES *****
* FILENAME           * FILENAME           * FILENAME
TID2T21

LAST CHG:DATE 94/10/05 TIME 11.58.38 CC# 15079 SRCE R001 OPER JXM
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-DGID 11-CPY ..-NXTT ..-NEW ..-NXTA

```

© 2000 CSC. All rights reserved



Umbrella Programming

Sort Activity

Sort Activity Definition - BOTH

CSC Sort Activity Definition - Both

```
48309 ACTION COMPLETE (SORT ACTS)
----- ACTIVITY DEFINITION INQUIRY/MAINTENANCE -----
COMMAND ==> INQ OWNER ==> PEM CC# ==>
ACTIVITY ID ======> 128 EFF DATE ==> 78/01/01
ACTIVITY MNEMONIC ==> TYPE ======> SORT
DESCRIPTION ======> AUDIT DRIVER INPUT

          SORT
DISPOSITION ==> WAIT      EXIT TYPE ======> BOTH      FORMAT =====> V
DATA GROUP ==>           101      DEFAULT ACTION ==> READ      CORESIZE ==>
                                SORT CARDS
1==>     SORT FIELDS=(5,2,A,15,8,A,7,8,A,23,97,A),FORMAT=BI
2==>
          **** INPUT FILES ****
* FILENAME           * FILENAME           * FILENAME

LAST CHG:DATE 94/11/07 TIME 19.22.55 CC#           SRCE UMB200    OPER
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-DGID 11-CPY ..-NXTT ..-NEW ..-NXTA
```

© 2000 CSC. All rights reserved 7/5/00

Notes:



Problem Specifications—SORT Activity



1. Establish a Sort activity, 9998xx, where xx is your group number.
2. The type of activity will be SORT.
3. There will be no mnemonic for this activity ID.
4. Please include your group number in the description of the activity ID.
5. The EXIT TYPE should be OUT.
6. Use datagroup 9994xx as the I/O area.
7. The DEFAULT ACTION should be READ.
8. Indicate SORT FIELDS that will sort LAST-NAME then FIRST-NAME, within your I/O area.
9. Enter values from the change control number you created.

Notes:



Problem Specifications—SORT Access



SORT Activity - Exercise I

1. Construct ALL of the necessary Process Dictionary components needed to support a batch JOB ZUPCXXEZ.
2. The first step of ZUPCXXEZ will execute program, ZUPCxxSY (where yy is your group number + 20), that will sequentially read the EMP hierarchical database and write an SDB record for each employee, with a record format mapped by datagroup 9994xx (where xx is your group number).
3. The second step of ZUPCXXEZ will execute program, ZUPCxxSZ (where zz is your group number + 40), that will incorporate SORT activity 9998xx, replacing the SDB input activity 9995yy, move each SDB field element back to its original HDB field element, and produce the same report output as program Z9994xx.
4. Enter values from the change control number you created in Chapter 5.
 - SDB definition, EMPxxOUT, where xx is your group number, was created in a prior exercise.
 - SDB activity, 9995xx, where xx is your group number, was created in a prior exercise.

Notes:



Batch Skeleton Program - ZUPCxxSY

```

MODULE NAME ZUPCxxSY

//ZUP{J}SC JOB (HOGN,{B},BEF),'PGM Z9994\\\\\\',MSGCLASS=9,
//                      TIME=(00,04),REGION=4M,NOTIFY=&SYSUID
//*
//P$ $$LIB JCLLIB ORDER=( {TL}.PROCLIB)
//*
//***** ****
//JS010    EXEC HOGNBC2B,
//          MASTER=' {L}.HEC.MASTER',
//          SYSMOD=' {L}.TESTLIB',
//          VSAM=' {V}'
//*-----
//* THIS PROC :
//*          EXECUTES LIBRARIAN
//*          VALIDATES PROGRAM VIA HOGAN PRECOMPILER
//*          COMPILES PROGRAM
//*          LINKS INTO TESTLIB (NAME CARD IS GENERATED)
//*-----
//SYSIN      DD *
-PID 9994\\\\\\
-OPT EXEC,TEMP
-SEL HECDUMMY <===== DO NOT CHANGE THIS STATEMENT
-REP ALL,NOAUDIT
***** ****
*           I D E N T I F I C A T I O N   D I V I S I O N
***** ****
      SKIP1
      IDENTIFICATION DIVISION.
      PROGRAM-ID.      Z9994\\\\\\.
      AUTHOR.          HOGAN SYSTEMS INC.
      DATE-COMPILED.
*REMARKS.      SKELETON PROGRAM FOR THE
*              UMBRELLA PROGRAMMERS CLASS
*              LAB EXERCISES.
      ENVIRONMENT DIVISION.
***** ****
*           E N V I R O N M E N T   D I V I S I O N
***** ****
      SKIP1
      CONFIGURATION SECTION.
      SOURCE-COMPUTER.   IBM-370.
      OBJECT-COMPUTER.   IBM-370.
      EJECT
***** ****
*           D A T A   D I V I S I O N
***** ****
      DATA DIVISION.
      SKIP3
      WORKING-STORAGE SECTION.
*-----
*   HOGAN LINKS ALL COBOL PROGRAMS AS REENTRANT. ONLY STATIC
*   VALUES ARE DEFINED IN WORKING STORAGE.
*-----
      77 CC-PHASE-NAME          PIC X(8)      VALUE 'Z9994\\\\\\'.
      77 CC-GROUP-ID            PIC XX       VALUE '\\\\\\'.
      SKIP3
*-----
*   THE FOLLOWING AREA CAN BE USED TO DEFINE PROGRAM CONSTANTS.
*   PEM USES BINARY FORMATED VALUES FOR ACTIVITY IDS, CONDITION
*   CODES, FORMAT IDS, ACTION CODES, ETC...
*   THERE ARE MANY DELIVERED COPYBOOK CONTAINING THESE BINARY

```



Umbrella Programming

Problem Specifications—SORT Access

```
* VALUES. THE ONES NEEDED FOR THIS CLASS HAVE BEEN INCLUDED.  
*-----  
01 BINARY-VALUES.  
05 FULLWORD-BINARY.  
    10 FILLER          PIC S9(8) COMP VALUE +0.  
    10 FILLER          PIC S9(8) COMP VALUE +0.  
    10 FILLER          PIC S9(8) COMP VALUE +0.  
05 FILLER REDEFINES FULLWORD-BINARY.  
    10 FIRST-FULLWORD  PIC XXXX.  
    10 SECOND-FULLWORD PIC XXXX.  
    10 THIRD-FULLWORD PIC XXXX.  
05 HALFWORD-BINARY.  
    10 FILLER          PIC S9(8) COMP VALUE +0.  
    10 FILLER          PIC S9(8) COMP VALUE +0.  
    10 FILLER          PIC S9(8) COMP VALUE +0.  
05 FILLER REDEFINES HALFWORD-BINARY.  
    10 FILLER          PIC XX.  
    10 FIRST-HALFWORD  PIC XX.  
    10 FILLER          PIC XX.  
    10 SECOND-HALFWORD PIC XX.  
    10 FILLER          PIC XX.  
    10 THIRD-HALFWORD PIC XX.  
EJECT  
*-----  
* THE FOLLOWING COPYBOOK P49002D CONTAINS THE VARIOUS RESULT  
* CODE VALUES THAT MAY BE PLACED IN THE TCB RESULT FIELD  
* (TCB-RESULT). THE TCB IS DEFINED BY COPYBOOK P49000D.  
*-----  
000100**** START OF P49002D ***** TCB RESULT CONSTANTS ***  
000200*  
000300 01 TCB-RESULT-CONSTANTS.  
000400      05 TCB-RESULTS.  
000500          10 TCB-OK.  
000600                  15 FILLER PIC S9(4) COMP VALUE +0000  
000700          10 TCB-ERR.  
000800                  15 FILLER PIC S9(4) COMP VALUE +0001  
000900          10 TCB-NO-ACT.  
001000                  15 FILLER PIC S9(4) COMP VALUE +0002  
001100          10 TCB-NOT-AUTH.  
001200                  15 FILLER PIC S9(4) COMP VALUE +0003  
001300          10 TCB-FAIL.  
001400                  15 FILLER PIC S9(4) COMP VALUE +0004  
001500          10 TCB-FULL.  
001600                  15 FILLER PIC S9(4) COMP VALUE +0005  
001700          10 TCB-ABEND-EXIT.  
001800                  15 FILLER PIC S9(4) COMP VALUE +0006  
001900          10 TCB-ROLLBACK.  
002000                  15 FILLER PIC S9(4) COMP VALUE +0007  
002100          10 TCB-DATA-BASE-FULL.  
002200                  15 FILLER PIC S9(4) COMP VALUE +0008  
002300          10 TCB-DB-NOT-AVAILABLE.  
002400                  15 FILLER PIC S9(4) COMP VALUE +0009  
002500          10 TCB-OK-CHECKPOINT.  
002600                  15 FILLER PIC S9(4) COMP VALUE +0010  
002700          10 TCB-FAIL-CHECKPOINT.  
002800                  15 FILLER PIC S9(4) COMP VALUE +0011  
002900          10 TCB-ABEND-TRANS.  
003000                  15 FILLER PIC S9(4) COMP VALUE +0012  
003100          10 TCB-DEFERRED.  
003200                  15 FILLER PIC S9(4) COMP VALUE +0020  
003300          10 TCB-RANDOMIZER-ERROR.  
003400                  15 FILLER PIC S9(4) COMP VALUE +0021  
003500          10 TCB-DATA-NOT-AVAILABLE.  
003600                  15 FILLER PIC S9(4) COMP VALUE +0022  
003700          10 TCB-DATA-FROM-GET-ONLY-DB.
```



Umbrella Programming

Problem Specifications—SORT Access

```

003800          15 FILLER PIC S9(4) COMP VALUE +0030
003900*
004000***** END OF P49002D ****
      EJECT
*-----
*   THE FOLLOWING COPYBOOK P49003D DEFINES SOME OF THE MORE
*   COMMONLY USED PEM ACTIVITIES. ACTIVITIES 1 THROUGH 100
*   ARE AUTOMATICALLY AUTHORIZED FOR USE BY ALL PROGRAMS AND
*   NEED NOT BE SPECIFIED IN THE PROGRAM DEFINITION IN THE
*   PROCESS DICTIONARY.
*-----
000100*** START OF P49003D *** PEM COMMON ACTIVITIES ***
000200*
000300 01  PEM-COMMON-ACTIVITIES.
000400    05 PEM-ACTIVITIES.
000500    10 CA-LONG-PEM-END-PROG.
000600          15 FILLER PIC S9(9) COMP VALUE +0001
000700    10 CA-LONG-PEM-END-TRANS.
000800          15 FILLER PIC S9(9) COMP VALUE +0002
000900    10 CA-LONG-PEM-DYN-DG-ALOC.
001000          15 FILLER PIC S9(9) COMP VALUE +0004
001100    10 CA-LONG-PEM-DYN-DG-REL.
001200          15 FILLER PIC S9(9) COMP VALUE +0005
001300    10 CA-LONG-PEM-DYN-DG-INIT.
001400          15 FILLER PIC S9(9) COMP VALUE +0006
001500    10 CA-LONG-PEM-SYSPRINT-WRITE.
001600          15 FILLER PIC S9(9) COMP VALUE +0008
001700    10 CA-LONG-PEM-NO-OP.
001800          15 FILLER PIC S9(9) COMP VALUE +0013
001900    10 CA-LONG-PEM-TRANS-DUMP-RETURN.
002000          15 FILLER PIC S9(9) COMP VALUE +0014
002100    10 CA-LONG-PEM-TRANS-DUMP-END.
002200          15 FILLER PIC S9(9) COMP VALUE +0015
002300    10 CA-LONG-PEM-DUMP-TCB-TRACE.
002400          15 FILLER PIC S9(9) COMP VALUE +0016
002500    10 CA-LONG-PEM-DUMP-DG-RETURN.
002600          15 FILLER PIC S9(9) COMP VALUE +0017
002700    10 CA-LONG-PEM-DUMP-DG-END.
002800          15 FILLER PIC S9(9) COMP VALUE +0018
002900    10 CA-LONG-PEM-ENABLE-ABEND-EXIT.
003000          15 FILLER PIC S9(9) COMP VALUE +0020
003100    10 CA-LONG-PEM-DISABLE-ABEND-EXIT.
003200          15 FILLER PIC S9(9) COMP VALUE +0021
003300    10 CA-LONG-PEM-CHECKPOINT.
003400          15 FILLER PIC S9(9) COMP VALUE +0028
003500    10 CA-LONG-PEM-DYN-DG-NO-INIT.
003600          15 FILLER PIC S9(9) COMP VALUE +0029
003700    10 CA-LONG-PEM-ROLLBACK.
003800          15 FILLER PIC S9(9) COMP VALUE +0030
003900    10 CA-LONG-PEM-DLI-SYNCPOINT.
004000          15 FILLER PIC S9(9) COMP VALUE +0032
004100    10 CA-LONG-PEM-USERCC-EXCEP.
004200          15 FILLER PIC S9(9) COMP VALUE +0035
004300    10 CA-LONG-PEM-MSG9-ABEND.
004400          15 FILLER PIC S9(9) COMP VALUE +0039
004500    10 CA-LONG-PEM-APPC-SYNCPOINT.
004600          15 FILLER PIC S9(9) COMP VALUE +0056
004700    10 CA-LONG-PEM-DYN-DG-LENGTH.
004800          15 FILLER PIC S9(9) COMP VALUE +0074
004900    10 CA-LONG-PEM-DYN-PTR-REL.
005000          15 FILLER PIC S9(9) COMP VALUE +0075
005100    10 CA-LONG-PEM-DYN-PTR-INIT.
005200          15 FILLER PIC S9(9) COMP VALUE +0076
005201    10 CA-LONG-PEM-DYN-PTR-ANO.
005202          15 FILLER PIC S9(9) COMP VALUE +0077

```



Umbrella Programming

Problem Specifications—SORT Access

```
005300      10 CA-LONG-PEM-PDG-HAS-CHANGED.
005400                               15 FILLER PIC S9(9) COMP VALUE +0088
005500*
005600      05 FILLER          REDEFINES PEM-ACTIVITIES.
005700                               10 FILLER PIC XX
005800      10 CA-PEM-END-PROG    PIC XX.
005900                               10 FILLER PIC XX
006000      10 CA-PEM-END-TRANS   PIC XX.
006100                               10 FILLER PIC XX
006200      10 CA-PEM-DYN-DG-ALOC  PIC XX.
006300                               10 FILLER PIC XX
006400      10 CA-PEM-DYN-DG-REL   PIC XX.
006500                               10 FILLER PIC XX
006600      10 CA-PEM-DYN-DG-INIT  PIC XX.
006700                               10 FILLER PIC XX
006800      10 CA-PEM-SYSPRINT-WRITE PIC XX.
006900                               10 FILLER PIC XX
007000      10 CA-PEM-NO-OP       PIC XX.
007100                               10 FILLER PIC XX
007200      10 CA-PEM-TRANS-DUMP-RETURN PIC XX.
007300                               10 FILLER PIC XX
007400      10 CA-PEM-TRANS-DUMP-END   PIC XX.
007500                               10 FILLER PIC XX
007600      10 CA-PEM-DUMP-TCB-TRACE  PIC XX.
007700                               10 FILLER PIC XX
007800      10 CA-PEM-DUMP-DG-RETURN  PIC XX.
007900                               10 FILLER PIC XX
008000      10 CA-PEM-DUMP-DG-END    PIC XX.
008100                               10 FILLER PIC XX
008200      10 CA-PEM-ENABLE-ABEND-EXIT PIC XX.
008300                               10 FILLER PIC XX
008400      10 CA-PEM-DISABLE-ABEND-EXIT PIC XX.
008500                               10 FILLER PIC XX
008600      10 CA-PEM-CHECKPOINT    PIC XX.
008700                               10 FILLER PIC XX
008800      10 CA-PEM-DYN-DG-NO-INIT  PIC XX.
008900                               10 FILLER PIC XX
009000      10 CA-PEM-ROLLBACK      PIC XX.
009100                               10 FILLER PIC XX
009200      10 CA-PEM-DLI-SYNCPOINT  PIC XX.
009300                               10 FILLER PIC XX
009400      10 CA-PEM-USERCC-EXCEP  PIC XX.
009500                               10 FILLER PIC XX
009600      10 CA-PEM-MSG9-ABEND    PIC XX.
009700                               10 FILLER PIC XX
009800      10 CA-PEM-APPC-SYNCPOINT PIC XX.
009900                               10 FILLER PIC XX
010000      10 CA-PEM-DYN-DG-LENGTH  PIC XX.
010100                               10 FILLER PIC XX
010200      10 CA-PEM-DYN-PTR-REL   PIC XX.
010300                               10 FILLER PIC XX
010400      10 CA-PEM-DYN-PTR-INIT   PIC XX.
010401                               10 FILLER PIC XX
010402      10 CA-PEM-DYN-PTR-ANO   PIC XX.
010500                               10 FILLER PIC XX
010600      10 CA-PEM-PDG-HAS-CHANGED PIC XX.
010700*
010800***** END OF P49003D ****
EJECT
*-----
*   THE FOLLOWING COPYBOOK P49022D DEFINES THE POSSIBLE VALUES
*   THAT MIGHT BE PLACED INTO A DATA GROUP ACTION FIELD PRIOR
*   TO ISSUING A PEM DATA BASE ACTIVITY.      DGA ACTION
*-----
```

000100*



Umbrella Programming

Problem Specifications—SORT Access

```

000200*--* START OF P49022D *----* PEM ACTION CODES *-----*
000300*
000400 01  PEM-DATA-GROUP-ACTION-CODES.
000500    05  DGA-CODE-VALUES.
000600    10  DGA-NO-OP.
000700                      15  FILLER PIC S9(4) COMP VALUE +0000
000800    10  DGA-READ.          15  FILLER PIC S9(4) COMP VALUE +0001
000900
001000    10  DGA-WRITE.        15  FILLER PIC S9(4) COMP VALUE +0002
001100    10  DGA-HOLD-P.       15  FILLER PIC S9(4) COMP VALUE +0003
001200
001300
001400    10  DGA-ERASE.        15  FILLER PIC S9(4) COMP VALUE +0004
001500
001600    10  DGA-READ-KEY-GE.   15  FILLER PIC S9(4) COMP VALUE +0005
001700
001800    10  DGA-FORCE-WRITE.  15  FILLER PIC S9(4) COMP VALUE +0006
001900
002000    10  DGA-READ-KEY-EQ.   15  FILLER PIC S9(4) COMP VALUE +0007
002100
002200    10  DGA-INSERT-FIRST.  15  FILLER PIC S9(4) COMP VALUE +0008
002300
002400    10  DGA-READ-LAST-REC. 15  FILLER PIC S9(4) COMP VALUE +0009
002500
002600    10  DGA-INSERT-LAST.  15  FILLER PIC S9(4) COMP VALUE +0010
002700
002800    10  DGA-READ-FIRST.   15  FILLER PIC S9(4) COMP VALUE +0011
002900
003000    10  DGA-INSERT-HERE.  15  FILLER PIC S9(4) COMP VALUE +0012
003100
003200    10  DGA-END-REQUEST.  15  FILLER PIC S9(4) COMP VALUE +0014
003300
003400    10  DGA-FORCE-NO-OP.  15  FILLER PIC S9(4) COMP VALUE +0016
003500
003600    10  DGA-CLOSE.        15  FILLER PIC S9(4) COMP VALUE +0036
003700
003800    10  DGA-OPEN-OUTPUT.  15  FILLER PIC S9(4) COMP VALUE +0037
003900
004000    10  DGA-OPEN-INPUT.   15  FILLER PIC S9(4) COMP VALUE +0038
004100
004200    10  DGA-OPEN-UPDATE.  15  FILLER PIC S9(4) COMP VALUE +0039
004300
004400    10  DGA-POINT.        15  FILLER PIC S9(4) COMP VALUE +0040
004500
004600    10  DGA-ENABLE-KEY-RANGE. 15  FILLER PIC S9(4) COMP VALUE +0041
004700
004800    10  DGA-DISABLE-KEY-RANGE. 15  FILLER PIC S9(4) COMP VALUE +0042
004900
005000    10  DGA-ENABLE-MULT-KRDB. 15  FILLER PIC S9(4) COMP VALUE +0043
005100
005200*
005300*--* END OF P49022D *-----*
EJECT
*-----
*   THE FOLLOWING COPYBOOK P49023D CONTAINS THE POSSIBLE VALUES
*   THAT MIGHT BE CONTAINED IN A DATA GROUP RESULTS FIELD AFTER
*   A PEM DATA BASE ACTIVITY HAS BEEN ISSUED.      DGR RESULT
*   # # #
*   WHEN CONTROL IS RETURNED TO AN APPLICATION PROGRAM ON
*   COMPLETION OF A DATA BASE ACTIVITY, EACH DATA GROUP INVOLVED
*   IN THE ACTIVITY WILL CONTAIN A RESULT CODE REFLECTING THE
*   RESULT OF THE REQUESTED ACTION.  IF EACH INDIVIDUAL DATA
*   GROUP'S RESULT FIELD IS ZERO, THE TCB RESULT FIELD
*   (TCB-RESULT) WILL ALSO CONTAIN ZERO.
*
```



Umbrella Programming

Problem Specifications—SORT Access

```
* TCB-RESULT WILL BE SET TO NON-ZERO IF ANY OF THE INDIVIDUAL
* DATA GROUPS IS NON-ZERO. IT IS THE RESPONSIBILITY OF THE
* APPLICATION PROGRAM TO DETERMINE THE CAUSE AND SEVERITY OF
* THE RESULT CONDITION. AN END-OF-DATA ON THE BASE DATA
* GROUP (EOF) WILL CAUSE A NON-ZERO VALUE TO BE RETURNED TO
* THE TCB-RESULT FIELD.
*-----
000100*** START OF P49023D *----* PEM RESULT CODES *-----
000200*
000300 01  PEM-DATA-GROUP-RESULT-CODES.
000400      05  DGR-CODE-VALUES.
000500      10  DGR-OK.
000600                      15  FILLER PIC S9(4) COMP VALUE +0000
000700      10  DGR-END-DATA.
000800                      15  FILLER PIC S9(4) COMP VALUE +0001
000900      10  DGR-I-O-ERR.
001000                      15  FILLER PIC S9(4) COMP VALUE +0002
001100      10  DGR-DUP-KEY.
001200                      15  FILLER PIC S9(4) COMP VALUE +0006
001300      10  DGR-NO-FIND.
001400                      15  FILLER PIC S9(4) COMP VALUE +0007
001500      10  DGR-SKEY-NO-FIND.
001600                      15  FILLER PIC S9(4) COMP VALUE +0008
001700      10  DGR-KRID-DISABLED.
001800                      15  FILLER PIC S9(4) COMP VALUE +0009
001900      10  DGR-DB-NOT-AVAIL.
002000                      15  FILLER PIC S9(4) COMP VALUE +0010
002100      10  DGR-SQLCA-ERROR.
002200                      15  FILLER PIC S9(4) COMP VALUE +0013
002300      10  DGR-LEN-ERR.
002400                      15  FILLER PIC S9(4) COMP VALUE +0014
002500*
002600*** END OF P49023D *-----
          EJECT
LINKAGE SECTION.
*****
*
*           L I N K A G E   S E C T I O N
*
*****
*
* THE FOLLOWING COPYBOOK P49000D DEFINES DATA GROUP 1
* TRANSACTION CONTROL BLOCK (TCB)
*
* THE APPLICATION CONTROL BLOCK IS CONTAINED WITHIN THE TCB.
* THE ACB IS USED FOR CDMF/PCD PROCESSING.
* COPYBOOK U48004D CONTAINS THE CDMF ACTION CODES
* COPYBOOK U48024D CONTAINS THE CDMF RESULT CODES
* ACTIVITY ID 48000
*
* THE TCB MUST BE THE FIRST DATA GROUP IN THE LINKAGE SECTION.
* THE TCB DOES NOT NEED TO BE LISTED IN THE UMB PROGRAM DEF.
* ALL OTHER DATA GROUPS MUST BE LISTED IN THE UMB PROGRAM DEF.
* IN THE SAME ORDER AS THEY ARE LISTED IN THE LINKAGE SECTION
* AND IN THE USING STATEMENT OF THIS APPLICATION PROGRAM.
*-----
000100*** START OF P49000D **** TCB ****
000200* THIS COPYBOOK NOW DEFINES FOUR DISTINCT DATA GROUPS:
000300* 1) D.G. 00001, THE USER TRANSACTION CONTROL BLOCK AND THE CDMF
000400*     APPLICATION CONTROL BLOCK;
000500* 2) D.G. 00012, THE SYSIN INPUT DATA GROUP;
000600* 3) D.G. 00013, THE SYSPRINT OUPUT DATA GROUP; AND
000700* 4) D.G. 00010, THE SECURITY CONTROL BLOCK, THE TCB USER AREA,
000800*     THE USER TCB EXTENSION AREA, AND THE INTERNAL TCB EXTENSION
000900*     AREA (ALC PROGRAMS ONLY).
```



Umbrella Programming

Problem Specifications—SORT Access

```

001000*****
001100      SKIP1
001200*****
001300* DATA GROUP 00001 (USER TRANSACTION CONTROL BLOCK)
001400*****
001500*
001600 01  TRANSACTION-CONTROL-BLOCK.
001700      05  FILLER          PIC XXXX.
001800      05  TCB-TRANS-NO    PIC XXXX.
001900      05  TCB-CO-ID      PIC XX.
002000      05  TCB-APPL-ID    PIC XX.
002100      05  TCB-FUNC-ID    PIC XX.
002200      05  TCB-SOURCE-TYPE.
002300          10  TCB-SOURCE-TYPE-N   PIC S9(4)  COMP.
002400          88  TCB-ONLINE        VALUE +3 +5 +6.
002500          88  TCB-BTCH         VALUE +4.
002600          88  TCB-APPL-SOURCE  VALUE +2.
002700          88  TCB-AUTHORIZATIONS  VALUE +5 +6.
002800          88  TCB-MANNED-TELLER  VALUE +5.
002900          88  TCB-UNMANNED-TELLER  VALUE +6.
003000      05  TCB-ACTIVITY.
003100          10  TCB-ACTIVITY-N    PIC S9(04) COMP.
003200      05  TCB-RESULT        PIC XX.
003300      05  TCB-USER-DATA.
003400          10  FILLER          PIC X(6).
003500          10  TCB-DATA-GROUP  PIC XX.
003600          10  TCB-PARM-POS.
003700          15  TCB-PARM-POS-N   PIC S9(4)  COMP.
003800      05  FILLER          REDEFINES TCB-USER-DATA.
003900          10  TCB-USER-INFO    PIC XXXX.
004000          10  TCB-USER-COND    PIC XX.
004100          10  FILLER          PIC X(4).
004200      05  FILLER          REDEFINES TCB-USER-DATA.
004300          10  TCB-USER-CC      PIC XX.
004400          10  TCB-USER-RESULT  PIC XX.
004500          10  TCB-USER-ENVMT  PIC XX.
004600          10  FILLER          PIC X(4).
004700      05  FILLER          REDEFINES TCB-USER-DATA.
004800          10  TCB-EOJ-CALL    PIC XXXX.
004900          10  FILLER          REDEFINES TCB-EOJ-CALL.
005000          15  TCB-SOT-CALL    PIC XXXX.
005100          10  FILLER          PIC X(6).
005200      05  TCB-TIME          PIC S9(7)  COMP-3.
005300      05  TCB-SYS-DATE    PIC S9(7)  COMP-3.
005400      05  TCB-SOURCE        PIC X(8).
005500      05  TCB-OPERATOR.
005600          10  TCB-UMBRELLA-OPERATOR  PIC X(8).
005700          10  TCB-OPERATOR-FILLER  PIC X(12).
005800          10  TCB-DXRF          REDEFINES TCB-OPERATOR-FILLER.
005900          15  TCB-DXRF-ID      PIC X(4).
006000          15  FILLER          PIC X(8).
006100          10  TCB-DYN-KEY-RANGE  REDEFINES TCB-OPERATOR-FILLER.
006200          15  TCB-DKR-ID      PIC X(4).
006300          15  FILLER          PIC X(8).
006400          10  TCB-ENQ           REDEFINES TCB-OPERATOR-FILLER.
006500          15  TCB-ENQ-ID      PIC X(8).
006600          15  FILLER          PIC X(4).
006700      05  TCB-DESTINATION  PIC X(8).
006800      05  TCB-TERM-DATA.
006900          10  TCB-PFKEY        PIC X.
007000          88  TCB-ENTER        VALUE QUOTE.
007100          88  TCB-PF01        VALUE '1'.
007200          88  TCB-PF02        VALUE '2'.
007300          88  TCB-PF03        VALUE '3'.
007400          88  TCB-PF04        VALUE '4'.

```



Umbrella Programming

Problem Specifications—SORT Access

```
007500      88 TCB-PF05          VALUE '5'.
007600      88 TCB-PF06          VALUE '6'.
007700      88 TCB-PF07          VALUE '7'.
007800      88 TCB-PF08          VALUE '8'.
007900      88 TCB-PF09          VALUE '9'.
008000      88 TCB-PF10          VALUE ':'.
008100      88 TCB-PF11          VALUE '#'.
008200      88 TCB-PF12          VALUE '@'.
008300      88 TCB-PF13          VALUE 'A'.
008400      88 TCB-PF14          VALUE 'B'.
008500      88 TCB-PF15          VALUE 'C'.
008600      88 TCB-PF16          VALUE 'D'.
008700      88 TCB-PF17          VALUE 'E'.
008800      88 TCB-PF18          VALUE 'F'.
008900      88 TCB-PF19          VALUE 'G'.
009000      88 TCB-PF20          VALUE 'H'.
009100      88 TCB-PF21          VALUE 'I'.
009200      88 TCB-PF22          VALUE '¢'.
009300      88 TCB-PF23          VALUE '.'.
009400      88 TCB-PF24          VALUE '<'.
009500      88 TCB-PFKEY-NOT-PRESENT  VALUE LOW-VALUE.
009600      10 FILLER           PIC XX.
009700      05 TCB-GENP-LOG        PIC X(1).
009800      88 TCB-GENP-NO-LOGGING  VALUE 'N'.
009900      88 TCB-GENP-LOGGING    VALUE 'Y' .
                                         LOW-VALUE.
010000
010100      05 TCB-EFFECTIVE-DATE  PIC S9(7) COMP-3.
010200      05 TCB-DEVICE-TYPE-2.
010300      10 TCB-DEVICE-TYPE    PIC X.
                                         VALUE 'A'.
010400      88 TCB-3270-2         VALUE 'B'.
010500      88 TCB-BATCH          VALUE 'C'.
010600      88 TCB-3270-1         VALUE 'D'.
010700      88 TCB-3604-DS1        VALUE 'E'.
010800      88 TCB-3604-DS3        VALUE 'F'.
010900      88 TCB-3604-DS4        VALUE 'G'.
011000      88 TCB-3600-JP         VALUE 'H'.
011100      88 TCB-3600-PB         VALUE 'I'.
011200      88 TCB-3600-LP         VALUE 'J'.
011300      88 TCB-3270-MOD1-PRINTER  VALUE 'K'.
011400      88 TCB-3270-MOD2-PRINTER  VALUE 'L'.
011500      88 TCB-TWX            VALUE 'M'.
011600      88 TCB-2470-MOD2       VALUE 'N'.
011700      88 TCB-2740-MOD1       VALUE 'O'.
011800      88 TCB-ALIEN-X         VALUE 'P'.
011900      88 TCB-ALIEN-Y         VALUE 'Q'.
012000      88 TCB-ALIEN-Z         VALUE 'R'.
012100*
012200***** RESERVED FOR TCB-DEVICE EXPANSION TO PIC X(2)
012300      10 FILLER           PIC X.
012400      05 TCB-LONG-ACTIVITY-N  PIC S9(09) COMP.
012500      05 TCB-LONG-ACTIVITY    REDEFINES TCB-LONG-ACTIVITY-N.
012600      10 TCB-LONG-ACT-HI     PIC XX.
012700      10 TCB-LONG-ACT-LO     PIC XX.
012800      05 TCB-LONG-DGID-N     PIC S9(09) COMP.
012900      05 TCB-LONG-DGID       REDEFINES TCB-LONG-DGID-N.
013000      10 TCB-LONG-DG-HI     PIC XX.
013100      10 TCB-LONG-DG-LO     PIC XX.
013200      05 TCB-USER-CC-APP     PIC XX.
013300      05 TCB-RESULT-2       PIC XX.
013400      EJECT
013500*****
013600* CDMF APPLICATION CONTROL BLOCK
013700*****
013800      05 CDMF-CONTROL-BLOCK.
013900      10 CDMF-ACTION        PIC XX.
```



Umbrella Programming

Problem Specifications—SORT Access

```

014000      10 CDMF-RESULT          PIC XX.
014100      10 CDMF-KEY-FIELDS.
014200      15 CDMF-FORMAT        PIC XXXX.
014300      15 CDMF-COID          PIC XX.
014400      15 CDMF-EFF-DATE      PIC S9(7) COMP-3.
014500      10 CDMF-EXP-DATE      PIC S9(7) COMP-3.
014600      10 CDMF-COID-FOUND    PIC XX.
014700      88 CDMF-DEFAULT-COID-FOUND  VALUE HIGH-VALUES.
014800      10 CDMF-EFF-DATE-FOUND  PIC S9(7) COMP-3.
014900      10 CDMF-HIGH-USE-FLAG   PIC X.
015000      88 CDMF-HIGH-USE-ITEM   VALUE 'Y'.
015100      88 CDMF-NON-PURGEABLE  VALUE 'P'.
015200*
015300***** ITEM OWNERSHIP IS ALWAYS RETURNED IN CDMF-OWNER-APPLICATION
015400***** OWNERSHIP MAY BE RETRIEVED AND UPDATED FROM DATA GROUP 480
015500***** IF THIS FLAG IS SET TO A 'Y'. OWNERSHIP MAY BE UPDATED FR
015600***** CDMF-OWNER-APPLICATION IF THIS FLAG IS SET TO A 'C'.
015700      10 CDMF-OWNER-APP-FLAG   PIC X.
015800      88 CDMF-OWNER-APP-REQUEST  VALUE 'Y'.
015900      88 CDMF-OWNER-APP-IN-CTL-BLK  VALUE 'C'.
016000      10 CDMF-ITEM-LOCATION    PIC X.
016100      88 CDMF-ITEM-FOUND-IN-TABLE  VALUE 'Y'.
016200      10 FILLER             PIC X.
016300      10 CDMF-CC-NO         PIC X(4).
016400      10 CDMF-LAST-CHANGE-DATA.
016500      15 CDMF-LAST-CHANGE-DATE  PIC S9(7) COMP-3.
016600      15 CDMF-LAST-CHANGE-TIME  PIC S9(7) COMP-3.
016700      15 CDMF-LAST-CHANGE-CC-NO  PIC X(4).
016800      15 CDMF-LAST-CHANGE-SOURCE  PIC X(8).
016900      15 CDMF-LAST-CHANGE-OPER   PIC X(8).
017000      10 CDMF-SECONDARY-KEY-ID  PIC X(4).
017100      10 CDMF-SUBSTITUTE-DGID  PIC X(4).
017200*
017300***** THIS FIELD IS FOR INTERNAL UMBRELLA USE ONLY.
017400      10 CDMF-RELEASE-CTL-DG-LEN  PIC XX.
017500*
017600***** THIS FIELD IS FOR INTERNAL UMBRELLA USE ONLY.
017700      10 CDMF-RELEASE-CTL-FLAGS  PIC X.
017800*
017900***** THE ITEM APPLICATION OWNERSHIP IS ALWAYS RETURNED IN THIS
018000***** FIELD. THIS FIELD MAY ONLY BE USED IN AN UPDATE WHEN
018100***** CDMF-OWNER-APP-FLAG IS SET TO A 'C'.
018200      10 CDMF-OWNER-APPLICATION  PIC X(3).
018300      10 FILLER             PIC X(2).
018400***** END OF DATA GROUP 00001 ****
018500      EJECT
018600*****
018700* DATA GROUP 12 (SYSIN INPUT DATA GROUP)
018800*-----
018900* NOTE: PROGRAMS THAT REFERENCE DATA GROUP 12 NEED NOT CODE THE
019000* DATA GROUP ON THE PROGRAM DEFINITION. INSTEAD, YOU MAY REFER
019100* TO THIS AREA DIRECTLY & ISSUE ACTIVITIES THAT REFERENCE DG 12.
019200*****
019300*
019400***** DATA GROUP CHAIN (DO NOT DESTROY)
019500      05 FILLER             PIC X(8).
019600      05 DATA-GROUP-12.
019700      10 DG12-ACTION        PIC XX.
019800      10 DG12-RESULT        PIC XX.
019900      10 DG12-CARD-IMAGE    PIC X(80).
020000***** END OF DATA GROUP 00012 ****
020100      SKIP1
020200*****
020300* DATA GROUP 13 (SYSPRINT OUTPUT DATA GROUP)
020400*-----

```



Umbrella Programming

Problem Specifications—SORT Access

```
020500* NOTE: PROGRAMS THAT REFERENCE DATA GROUP 13 NEED NOT CODE THE
020600* DATA GROUP ON THE PROGRAM DEFINITION. INSTEAD, YOU MAY REFER
020700* TO THIS AREA DIRECTLY & ISSUE ACTIVITIES THAT REFERENCE DG 13.
020800*****020900*
021000***** DATA GROUP CHAIN (DO NOT DESTROY)
021100    05 FILLER          PIC X(8).
021200    05 DATA-GROUP-13.
021300    10 DG13-ACTION    PIC XX.
021400    10 DG13-RESULT    PIC XX.
021500    10 DG13-CONTROL-CHAR PIC X.
021600    10 DG13-PRINT-DATA PIC X(132).
021700    10 FILLER          PIC X(7).
021800***** END OF DATA GROUP 00013 *****
021900    SKIP1
022000*****022100* DATA GROUP 10 (TCB EXTENSION AREAS)
022200*-----
022300* THIS DATA GROUP CONTAINS THE SECURITY CONTROL BLOCK, THE TCB
022400* USER AREA (FOR CLIENT USE), THE USER TCB EXTENSION AREA, AND
022500* THE INTERNAL TCB EXTENSION AREA.
022600* ***** NOTE: DO NOT CODE THIS DATA GROUP ON YOUR PROGRAM
022700* ***** DEFINITION. INSTEAD, YOU SHOULD REFER TO THIS
022800* ***** AREA DIRECTLY SINCE IT IS PART OF THE USER
022900* ***** TRANSACTION CONTROL BLOCK.
023000*****023100*
023200***** DATA GROUP CHAIN (DO NOT DESTROY)
023300    05 FILLER          PIC X(8).
023400    05 DATA-GROUP-10.
023500    EJECT
023600*****023700* SECURITY CONTROL BLOCK
023800*****023900    10 SECURITY-CONTROL-BLOCK.
024000    15 SCB-ACTION      PIC S9(4) COMP.
024100    15 SCB-ACTION-X REDEFINES SCB-ACTION
024200                                PIC XX.
024300    15 SCB-RESULT      PIC S9(4) COMP.
024400    88 SCB-AUTHORIZATION-VALID VALUE +0.
024500    88 SCB-AUTHORIZATION-FAILED VALUE +4.
024600    88 SCB-AUTHORIZATION-ERROR VALUE +8.
024700    88 SCB-EXT-SECURITY-INACTIVE VALUE +12.
024800    15 SCB-RESULT-X REDEFINES SCB-RESULT
024900                                PIC XX.
025000    15 SCB-VIOLATION-ACTION PIC S9(4) COMP.
025100    88 SCB-ABEND-TASK   VALUE +0.
025200    88 SCB-RETURN       VALUE +1.
025300    15 SCB-LOGGING-FLAG PIC X.
025400    88 SCB-LOG-EXCPTNS VALUE 'Y'.
025500    88 SCB-BYPASS-LOG  VALUE 'N'.
025600    15 SCB-PROCESSING-TYPE PIC X.
025700    15 SCB-PEM-FLAG1   PIC X.
025800    15 SCB-FUTURE-FLAGS PIC X(3).
025900    15 SCB-FORMAT-NUMBER PIC XXXX.
026000    15 SCB-FORMAT-NAME PIC X(10).
026100*
026200***** THIS FIELD IS FOR ALC PROGRAMS ONLY
026300    15 SCB-ADDR-FMT-TARGET-DG  PIC XXXX.
026400*
026500    15 SCB-TARGET-DG-ID   PIC XXXX.
026600    15 SCB-ITEM-OWNER   PIC X(3).
026700    15 SCB-PREV-OWNER   PIC X(3).
026800    15 SCB-MESSAGE-NO.
026900    20 SCB-MESSAGE-NO-N  PIC S9(4) COMP.
```



Umbrella Programming

Problem Specifications—SORT Access

```

027000      15  SCB-EXCEPTION-MESSAGE .
027100          20  SCB-RULE-NAME      PIC X(40) .
027200          20  FILLER           PIC X(4) .
027300      15  SCB-USER-DATA      PIC X(25) .
027400      15  SCB-RESERVED       PIC X(8) .
027500      15  FILLER           PIC X(1) .
027600      EJECT
027700*****
027800* TCB USER AREA
027900*-----
028000* THIS AREA IS RESERVED FOR CLIENTS AND WILL NEVER BE USED BY
028100* HOGAN SYSTEMS.
028200*****
028300      10  TCB-USER-AREA.
028400          15  FILLER           PIC X(104) .
028500      SKIP1
028600*****
028700* USER TCB EXTENSION AREA
028800*-----
028900* THIS AREA IS RESERVED FOR NEW TCB FIELDS TO BE ADDED AND
029000* UPDATED BY HOGAN SYSTEMS.
029100*****
029200      10  TCB-EXTENSION-AREA.
029300*          CURSOR POSITION AFTER A DEBLOCK; ROW AND COLUMN
029400      15  TCB-ROW            PIC S9(4) COMP.
029500      15  TCB-COLUMN          PIC S9(4) COMP.
029600      15  TCB-BATCH-DISP-OPTION PIC X.
029700          88  TCB-BATCH-DISP-DUMP    VALUE LOW-VALUES.
029800*          BATCH DISPLAY TO SYSPRINT IN DUMP FORMAT.
029900          88  TCB-BATCH-DISP-FORMAT  VALUE 'F'.
030000*          BATCH DISPLAY TO SYSPRINT IN SCREEN FORMAT
030100          88  TCB-BATCH-DISP-RETURN  VALUE 'R'.
030200*          BATCH DISPLAY DATA IN DG 47. NOT PRINTED.
030300      15  FILLER           PIC X.
030400      15  TCB-DYN-TXN-ID      PIC X(008) .
030500      15  TCB-OPTIONS-1       PIC X.
030600      15  TCB-OPTIONS-2       PIC X.
030700      15  FILLER           PIC X(040) .
030800      15  TCB-SQL-ACTION.
030900          20  TCB-SQL-ACTION-N   PIC S9(4) COMP.
031000      15  TCB-SQL-RESULT.
031100          20  TCB-SQL-RESULT-N   PIC S9(4) COMP.
031200      15  TCB-SQL-DYNPLAN     PIC X(008) .
031300      15  TCB-SQL-CURPLAN     PIC X(008) .
031400      15  TCB-SQL-DYN-SUBSID   PIC X(004) .
031500      15  TCB-SQL-SUBSID      PIC X(004) .
031600      15  TCB-CKPT-COUNT      PIC S9(9) COMP.
031700      15  FILLER           PIC X(005) .
031800      15  TCB-APPC-SERVICE-AREAS.
031900          20  TCB-APPC-DATA-GROUP  PIC X(4) .
032000          20  TCB-APPC-SYSTEM-KEY  PIC X(4) .
032100          20  TCB-APPC-APPL-KEY   PIC X(8) .
032200          20  TCB-APPC-XMIT-IMMED  PIC X(1) .
032300          88  TRANSMIT          VALUE 'Y'.
032400          88  TRAN-PREPARE-RECEIVE  VALUE 'R'.
032500          20  TCB-APPC-XMIT-ERROR  PIC X(1) .
032600*          **88 ISSUE-ERROR      VALUE +1.
032700*          **88 ISSUE-ERROR-W-DATA  VALUE +17.
032800*          **88 ISSUE-ABEND      VALUE +2.
032900          20  TCB-APPC-MORE-DATA   PIC X(1) .
033000          20  TCB-APPC-BYPSS-ERR   PIC X(1) .
033100          20  TCB-APPC-RETURN-CDE   PIC X(6) .
033200      15  FILLER           PIC X.
033300      15  TCB-VSAM-RELATIVE-NUMB PIC X(4) .
033400      15  TCB-MONETARY-KEY     PIC X(3) .

```



Umbrella Programming

Problem Specifications—SORT Access

```

033401      15 TCB-PRES-CURRENCY-CD    PIC X(3).
033500      15 TCB-LANGUAGE-KEY     PIC X(3).
033600      15 TCB-PACK-COLLECT-NAME PIC X(18).
033700      15 TCB-DYN-COLLECT-NAME.
033800      20 TCB-DYN-PLAN-PREF    PIC X(2).
033900      20 TCB-DYN-COMP-GRP    PIC X(8).
034000      20 TCB-DYN-PROC-GRP    PIC X(4).
034100      20 TCB-DYN-KEY-RANGE   PIC X(4).
034200      15 FILLER            PIC X(8).
034300      15 TCB-DB2-KRID        PIC X(4).
034400      15 TCB-LANG-ENABLED    PIC X(1).
034500      15 TCB-DEFAULT-LANG   PIC X(3).
034600      15 TCB-LANG-ENCODE-FLAG PIC X(1).
034700      15 TCB-PROC-GROUP     PIC X(4).
034800      15 TCB-PROC-GROUP-BRANCH PIC 9(5) COMP-3.
034900      15 TCB-PROCESSING-ID   REDEFINES
035000      TCB-PROC-GROUP-BRANCH PIC 9(5) COMP-3.
035100      15 TCB-UDFL-LANG      PIC X(3).
035200      15 TCB-SPS-IMPLODE-LANG PIC X(3).
035300      15 TCB-OPERATOR-REGION  PIC 9(5) COMP-3.
035400      15 TCB-OPERATOR-BRANCH PIC 9(5) COMP-3.
035500      15 TCB-DYN-LANG      PIC X(3).
035600      15 TCB-DMAP-ID       PIC X(7).
035700      15 TCB-DB2-PROC-GRP-ID PIC X(4).
035800      15 TCB-DB2-PROC-GRP-BRANCH PIC 9(5) COMP-3.
035900      15 TCB-DB2-PROCESSING-ID REDEFINES
036000      TCB-DB2-PROC-GRP-BRANCH PIC 9(5) COMP-3.
036100      15 TCB-USER-WORK-AREA.
036200      20 TCB-DYNAMIC-DG-ADDR POINTER.
036201      20 TCB-DYNAMIC-DG-ADDRESS REDEFINES
036202      TCB-DYNAMIC-DG-ADDR POINTER.
036300      20 FILLER           PIC X(8).
036400      15 TCB-HDP-RESERVED   PIC X(10).
036500      15 TCB-DFLT-LANG-ENCODE-BYTE
036600          PIC X(01).
036700      15 TCB-DB2-TEST-POOL-ID PIC X(02).
036800      15 TCB-PRES-CURR-RND-IN  PIC X(01).
036801* DO NOT ROUND PRESENTATION CURRENCY - ADDS AN EXTRA DECIMAL DIG
036802          88 TCB-DO-NOT-ROUND-PRES-CURR      VALUE 'N
036803* ROUND PRESENTATION CURRENCY
036804          88 TCB-ROUND-PRES-CURR      VALUE 'Y
036900      15 TCB-RAND-REC-ADDR-NR    PIC S9(9) COMP.
037000      15 TCB-JOB-ID          PIC X(8).
037100      15 TCB-OPERATOR-PROCESSING-ID PIC 9(5) COMP-3.
037200      15 TCB-DFLT-CENTURYWINDOW-CUTOFF PIC S999 COMP-3.
037300      15 TCB-CICS-STARTCODE   PIC XX.
037301          88 TCB-CICS-START-DPL      VALUE 'D '
037302          88 TCB-CICS-START-DPL-SYNC  VALUE 'DS'
037303          88 TCB-CICS-START-TD-TRIGGER  VALUE 'QD'
037304          88 TCB-CICS-START-CMD     VALUE 'S '
037305          88 TCB-CICS-START-CMD-DATA  VALUE 'SD'
037306          88 TCB-CICS-START-FEPI    VALUE 'SZ'
037307          88 TCB-CICS-START-TERMINAL-INPUT  VALUE 'TD'
037308          88 TCB-CICS-START-USER-ATTACH  VALUE 'U '
037309      15 FILLER           PIC X(661).
037400      15 FILLER           PIC X(200).
037500      SKIP1
037600***** END OF DATA GROUP 00010 *****
037700      SKIP1
037800*
037900***** END OF P49000D *****
EJECT
*-----
* THE FOLLOWING COPYBOOK Z47100D DEFINES DATA GROUP 47100
* BASE DATA GROUP EMP DATA BASE

```



Umbrella Programming

Problem Specifications—SORT Access

```

*   EMPLOYEE INFORMATION
*   EMP-KEY-GROUP
*-----
000010***** START OF Z47100D ***** EMP KEY ****
000020*
000030*   COPYBOOK Z47100D DEFINES DATA GROUP 47100, WHICH IS A KEY
000040*   DATA GROUP IN AN EMPLOYEE INFORMATION RECORD ON THE "EMP"
000050*   DATA BASE FOR EDUCATION CLASS USE.
000060*
000070*****
000080*
000090*   DATA GROUP NUMBER    47100
000100*
000110*****
000120 01  EMP-KEY-GROUP .
000130      10  EMP-ACTION          PIC XX.
000140      10  EMP-RESULT          PIC XX.
000150      05  EMP-KEY-GROUP-MOVE .
000160      10  EMP-CO-ID           PIC XX.
000170      10  EMP-KEY-ID           PIC 9(11) COMP-3.
000180      10  EMP-FILLER          PIC X(38).
000190***** END OF Z47100D ****
EJECT
*-----
*   THE FOLLOWING COPYBOOK Z47110D DEFINES DATA GROUP 47110
*   POSITIONAL DATA GROUP EMP DATA BASE
*   EMPLOYEE GENERAL INFORMATION
*   EMP-INFO-GROUP
*-----
000010***** START OF Z47110D ***** EMP INFORMATION ****
000020*
000030*   COPYBOOK Z47110D DEFINES DATA GROUP 47110, WHICH IS A
000040*   POSITIONAL DATA GROUP USED FOR "EMP" DATA BASE RECORDS
000050*   FOR EDUCATION CLASSES.
000060*
000070*****
000080*
000090*   DATA GROUP NUMBER    47110
000100*
000110*****
000120 01  EMP-INFO-GROUP .
000130      10  EMP-INFO-ACTION      PIC XX.
000140      10  EMP-INFO-RESULT      PIC XX.
000150      05  EMP-INFO-GROUP-MOVE .
000160      10  EMP-L-NAME          PIC X(15).
000170      10  EMP-F-NAME          PIC X(15).
000180      10  EMP-ADDRESS          PIC X(30).
000190      10  EMP-CITY            PIC X(20).
000200      10  EMP-STATE            PIC X(2).
000210      10  EMP-ZIP              PIC 9(9) COMP-3.
000220      10  EMP-RESERVED         PIC X(5).
000230      10  EMP-EEOC-CODE       PIC XX.
000240      10  EMP-SEX              PIC X.
000250      10  EMP-BIRTHDATE        PIC 9(7) COMP-3.
000260      10  EMP-EXEMP            PIC 999 COMP-3.
000270      10  EMP-INSUR-EXEMP      PIC 999 COMP-3.
000280      10  EMP-PHONE            PIC 9(11) COMP-3.
000290      10  EMP-INFO-FILLER      PIC X(87).
000300***** END OF Z47110D ****
EJECT
*-----
*   THE FOLLOWING COPYBOOK Z47120 DEFINES DATA GROUP 47120
*   POSITIONAL DATA GROUP EMP DATA BASE
*   EMPLOYEE JOB STATUS INFORMATION
*   EMP-JOB-STATUS

```



Umbrella Programming

Problem Specifications—SORT Access

```
*-----  
000010***** START OF Z47120D ***** EMP JOB STATUS *****  
000020*  
000030* COPYBOOK Z47120 DEFINES DATA GROUP 47120, WHICH IS A  
000040* POSITIONAL DATA GROUP USED FOR "EMP" DATA BASE RECORDS  
000050* FOR EDUCATION CLASSES.  
000060*  
000070*****  
000080*  
000090* DATA GROUP NUMBER 47120  
000100*  
000110*****  
000120 01 EMP-JOB-STATUS.  
000130      10 JOB-STAT-ACTION          PIC XX.  
000140      10 JOB-STAT-RESULT          PIC XX.  
000150      05 EMP-JOB-STATUS-MOVE.  
000160      10 JOB-STAT-DATE-HIRED    PIC 9(7)  COMP-3.  
000170      10 JOB-STAT-DATE-TERMD    PIC 9(7)  COMP-3.  
000180      10 JOB-STAT-MGR           PIC X(30).  
000190      10 JOB-STAT-DEPT          PIC XXX.  
000200      10 JOB-STAT-COST-CENTER    PIC XXX.  
000210      10 JOB-STAT-CLASS-POS.  
000211      15 JOB-STAT-CLASS          PIC XX.  
000212      15 JOB-STAT-POSITION        PIC XX.  
000230      10 JOB-STAT-FILLER         PIC X(98).  
000240***** END OF Z47120D *****  
      EJECT  
*-----  
* THE FOLLOWING COPYBOOK Z47130 DEFINES DATA GROUP 47130  
* POSITIONAL DATA GROUP EMP DATA BASE  
* EMPLOYEE CURRENT PAY INFORMATION  
* EMP-CURRENT-PAY  
*-----  
000010***** START OF Z47130D ***** EMP CURRENT PAY *****  
000020*  
000030* COPYBOOK Z47130D DEFINES DATA GROUP 47130, WHICH IS A  
000040* POSITIONAL DATA GROUP USED FOR "EMP" DATA BASE RECORDS  
000050* FOR EDUCATION CLASSES.  
000060*  
000070*****  
000080*  
000090* DATA GROUP NUMBER 47130  
000100*  
000110*****  
000120 01 EMP-CURRENT-PAY.  
000130      10 EMP-C-ACTION          PIC XX.  
000140      10 EMP-C-RESULT          PIC XX.  
000150      05 EMP-CURRENT-PAY-MOVE.  
000160      10 EMP-C-BEG-DATE        PIC 9(7)  COMP-3.  
000170      10 EMP-C-END-DATE        PIC 9(7)  COMP-3.  
000180      10 EMP-C-EARN-TOT        PIC S9(7)V99 COMP-3.  
000190      10 EMP-C-FED-INC         PIC S9(7)V99 COMP-3.  
000200      10 EMP-C-FICA            PIC S9(7)V99 COMP-3.  
000210      10 EMP-C-OTHER-TAX        PIC S9(7)V99 COMP-3.  
000220      10 EMP-C-DED-ADJ-TOT       PIC S9(7)V99 COMP-3.  
000230      10 EMP-C-NET-PAY         PIC S9(7)V99 COMP-3.  
000240      10 EMP-C-DED-ADJ OCCURS 10 TIMES.  
000250      15 EMP-C-DED-ADJ-CODE      PIC XX.  
000260      15 EMP-C-DED-ADJ-AMT        PIC S9(7)V99 COMP-3.  
000270      10 EMP-C-EARNINGS OCCURS 5 TIMES.  
000280      15 EMP-C-TYPE-CODE        PIC XX.  
000290      15 EMP-C-UNITS           PIC S9(5)   COMP-3.  
000300      15 EMP-C-AMTS            PIC S9(7)V99 COMP-3.  
000310      10 EMP-C-FILLER          PIC X(138).  
000320***** END OF Z47130D *****
```



Umbrella Programming

Problem Specifications—SORT Access

```

        EJECT
*-----
*   THE FOLLOWING COPYBOOK Z47140 DEFINES DATA GROUP 47140
*   POSITIONAL DATA GROUP EMP DATA BASE
*   EMPLOYEE YEAR TO DATE PAY INFORMATION
*   EMP-YEAR-TO-DATE-PAY
*-----
000010***** START OF Z47140D ***** EMP YEAR-TO-DATE PAY *****
000020*
000030*   COPYBOOK Z47140D DEFINES DATA GROUP 47140, WHICH IS A
000040*   POSITIONAL DATA GROUP USED FOR "EMP" DATA BASE RECORDS
000050*   FOR EDUCATION CLASSES.
000060*
000070*****
000080*
000090*   DATA GROUP NUMBER    47140
000100*
000110*****
000120 01  EMP-YEAR-TO-DATE-PAY.
000130      10  EMP-Y-ACTION          PIC XX.
000140      10  EMP-Y-RESULT          PIC XX.
000150      05  EMP-YEAR-TO-DATE-PAY-MOVE .
000160      10  EMP-Y-BEG-DATE         PIC 9(7)     COMP-3.
000170      10  EMP-Y-END-DATE         PIC 9(7)     COMP-3.
000180      10  EMP-Y-EARN-TOT        PIC S9(7)V99  COMP-3.
000190      10  EMP-Y-FED-INC         PIC S9(7)V99  COMP-3.
000200      10  EMP-Y-FICA           PIC S9(7)V99  COMP-3.
000210      10  EMP-Y-OTHER-TAX        PIC S9(7)V99  COMP-3.
000220      10  EMP-Y-DED-ADJ-TOT       PIC S9(7)V99  COMP-3.
000230      10  EMP-Y-NET-PAY         PIC S9(7)V99  COMP-3.
000240      10  EMP-Y-DED-ADJ OCCURS 10 TIMES.
000250      15  EMP-Y-DED-ADJ-CODE      PIC XX.
000260      15  EMP-Y-DED-ADJ-AMT       PIC S9(7)V99  COMP-3.
000270      10  EMP-Y-EARNINGS OCCURS 5 TIMES.
000280      15  EMP-Y-TYPE-CODE        PIC XX.
000290      15  EMP-Y-UNITS          PIC 9(5)     COMP-3.
000300      15  EMP-Y-AMTS           PIC S9(7)V99  COMP-3.
000310      10  EMP-Y-FILLER          PIC X(138).
000320***** END OF Z47140D *****
        EJECT
*****



*          P R O C E D U R E      D I V I S I O N
*
*-----*
*   FOR EASE IN DE-BUGGING, PLEASE MODIFY CODE ONLY IN CLASS
*   CODING SECTION AND MARKED AREAS.
*-----*
PROCEDURE DIVISION
    USING    TRANSACTION-CONTROL-BLOCK
            EMP-KEY-GROUP
            EMP-INFO-GROUP
            EMP-JOB-STATUS
            EMP-CURRENT-PAY
            EMP-YEAR-TO-DATE-PAY.

        EJECT
*-----
*   BA000-MAIN-LINE SECTION.
*-----


        BA100-READ-LOOP.

        PERFORM RD000-READ-EMP-DB-SEQUENTIAL THRU RD099-EXIT.

```



Umbrella Programming

Problem Specifications—SORT Access



Batch Skeleton Program - ZUPCxxSZ

```

MODULE NAME ZUPCxxSZ

//ZUP{J}SC JOB (HOGN,{B},BEF),'PGM Z9994\\\\\\',MSGCLASS=9,
//                      TIME=(00,04),REGION=4M,NOTIFY=&SYSUID
//*
//P$ $$LIB JCLLIB ORDER=( {TL}.PROCLIB)
//*
//***** ****
//JS010 EXEC HOGNBC2B,
//          MASTER=' {L}.HEC.MASTER',
//          SYSMOD=' {L}.TESTLIB',
//          VSAM=' {V}'
//*-----
//** THIS PROC :
//*          EXECUTES LIBRARIAN
//*          VALIDATES PROGRAM VIA HOGAN PRECOMPILER
//*          COMPILES PROGRAM
//*          LINKS INTO TESTLIB (NAME CARD IS GENERATED)
//*-----
//SYSIN DD *
-PID 9994\\\\\\
-OPT EXEC,TEMP
-SEL HECDUMMY <===== DO NOT CHANGE THIS STATEMENT
-REP ALL,NOAUDIT
***** ****
*          IDENTIFICATION DIVISION
*          I D E N T I F I C A T I O N   D I V I S I O N
***** ****
      SKIP1
      IDENTIFICATION DIVISION.
      PROGRAM-ID.    Z9994\\\\\\.
      AUTHOR.        HOGAN SYSTEMS INC.
      DATE-COMPILED.
*REMARKS.    SKELETON PROGRAM FOR THE
*          UMBRELLA PROGRAMMERS CLASS
*          LAB EXERCISES.
      ENVIRONMENT DIVISION.
***** ****
*          E N V I R O N M E N T   D I V I S I O N
***** ****
      SKIP1
      CONFIGURATION SECTION.
      SOURCE-COMPUTER.    IBM-370.
      OBJECT-COMPUTER.    IBM-370.
      EJECT
***** ****
*          D A T A   D I V I S I O N
***** ****
      DATA DIVISION.
      SKIP3
      WORKING-STORAGE SECTION.
*-----
*          HOGAN LINKS ALL COBOL PROGRAMS AS REENTRANT. ONLY STATIC
*          VALUES ARE DEFINED IN WORKING STORAGE.
*-----
      77 CC-PHASE-NAME          PIC X(8)    VALUE 'Z9994\\\\\\'.
      77 CC-GROUP-ID           PIC XX     VALUE '\\\\\\'.
      SKIP3
*-----
*          THE FOLLOWING AREA CAN BE USED TO DEFINE PROGRAM CONSTANTS.
*          PEM USES BINARY FORMATED VALUES FOR ACTIVITY IDS, CONDITION
*          CODES, FORMAT IDS, ACTION CODES, ETC...
*          THERE ARE MANY DELIVERED COPYBOOK CONTAINING THESE BINARY

```



Umbrella Programming

Problem Specifications—SORT Access

```
* VALUES. THE ONES NEEDED FOR THIS CLASS HAVE BEEN INCLUDED.  
*-----  
01 BINARY-VALUES.  
05 FULLWORD-BINARY.  
    10 FILLER          PIC S9(8) COMP VALUE +01013.  
    10 FILLER          PIC S9(8) COMP VALUE +01398.  
    10 FILLER          PIC S9(8) COMP VALUE +01900.  
    10 FILLER          PIC S9(8) COMP VALUE +47921.  
    10 FILLER          PIC S9(8) COMP VALUE +47922.  
    10 FILLER          PIC S9(8) COMP VALUE +48000.  
    10 FILLER          PIC S9(8) COMP VALUE +0.  
    10 FILLER          PIC S9(8) COMP VALUE +0.  
    10 FILLER          PIC S9(8) COMP VALUE +0.  
05 FILLER REDEFINES FULLWORD-BINARY.  
    10 PCD-LK-ACTIVITY-1013  PIC XXXX.  
    10 CCP-LK-ACTIVITY-1398  PIC XXXX.  
    10 DTS-LK-ACTIVITY-1900  PIC XXXX.  
    10 SPS-LK-ACTIVITY-47921 PIC XXXX.  
    10 SPS-LK-ACTIVITY-47922 PIC XXXX.  
    10 CDMF-LK-ACTIVITY-48000 PIC XXXX.  
    10 FIRST-FULLWORD      PIC XXXX.  
    10 SECOND-FULLWORD     PIC XXXX.  
    10 THIRD-FULLWORD      PIC XXXX.  
05 HALFWORD-BINARY.  
    10 FILLER          PIC S9(8) COMP VALUE +99\\\|.  
    10 FILLER          PIC S9(8) COMP VALUE +0.  
    10 FILLER          PIC S9(8) COMP VALUE +0.  
    10 FILLER          PIC S9(8) COMP VALUE +0.  
05 FILLER REDEFINES HALFWORD-BINARY.  
    10 FILLER          PIC XX.  
    10 COND-CODE-99\\\|  PIC XX.  
    10 FILLER          PIC XX.  
    10 FIRST-HALFWORD   PIC XX.  
    10 FILLER          PIC XX.  
    10 SECOND-HALFWORD  PIC XX.  
    10 FILLER          PIC XX.  
    10 THIRD-HALFWORD   PIC XX.  
EJECT  
*-----  
* THE FOLLOWING COPYBOOK P49002D CONTAINS THE VARIOUS RESULT  
* CODE VALUES THAT MAY BE PLACED IN THE TCB RESULT FIELD  
* (TCB-RESULT). THE TCB IS DEFINED BY COPYBOOK P49000D.  
*-----  
000100***** START OF P49002D ***** TCB RESULT CONSTANTS ***  
000200*  
000300 01 TCB-RESULT-CONSTANTS.  
000400 05 TCB-RESULTS.  
000500    10 TCB-OK.  
000600          15 FILLER PIC S9(4) COMP VALUE +0000  
000700    10 TCB-ERR.  
000800          15 FILLER PIC S9(4) COMP VALUE +0001  
000900    10 TCB-NO-ACT.  
001000          15 FILLER PIC S9(4) COMP VALUE +0002  
001100    10 TCB-NOT-AUTH.  
001200          15 FILLER PIC S9(4) COMP VALUE +0003  
001300    10 TCB-FAIL.  
001400          15 FILLER PIC S9(4) COMP VALUE +0004  
001500    10 TCB-FULL.  
001600          15 FILLER PIC S9(4) COMP VALUE +0005  
001700    10 TCB-ABEND-EXIT.  
001800          15 FILLER PIC S9(4) COMP VALUE +0006  
001900    10 TCB-ROLLBACK.  
002000          15 FILLER PIC S9(4) COMP VALUE +0007  
002100    10 TCB-DATA-BASE-FULL.  
002200          15 FILLER PIC S9(4) COMP VALUE +0008
```



Umbrella Programming

Problem Specifications—SORT Access

```

002300      10 TCB-DB-NOT-AVAILABLE.
002400          15 FILLER PIC S9(4) COMP VALUE +0009
002500      10 TCB-OK-CHECKPOINT.
002600          15 FILLER PIC S9(4) COMP VALUE +0010
002700      10 TCB-FAIL-CHECKPOINT.
002800          15 FILLER PIC S9(4) COMP VALUE +0011
002900      10 TCB-ABEND-TRANS.
003000          15 FILLER PIC S9(4) COMP VALUE +0012
003100      10 TCB-DEFERRED.
003200          15 FILLER PIC S9(4) COMP VALUE +0020
003300      10 TCB-RANDOMIZER-ERROR.
003400          15 FILLER PIC S9(4) COMP VALUE +0021
003500      10 TCB-DATA-NOT-AVAILABLE.
003600          15 FILLER PIC S9(4) COMP VALUE +0022
003700      10 TCB-DATA-FROM-GET-ONLY-DB.
003800          15 FILLER PIC S9(4) COMP VALUE +0030
003900*
004000***** END OF P49002D ****
EJECT
*-----
* THE FOLLOWING COPYBOOK U48004D DEFINES THE POSSIBLE ACTION
* CODES FOR CDMF/PCD PROCESSING.
* THE REQUESTED ACTION MUST BE PLACED INTO CDMF-ACTION FIELD
* IN THE APPPLIACTION CONTROL BLOCK PRIOR TO ISSUING THE
* LINK ACTIVITY 48000 FOR CDMF/PCD PROCESSING.
* THE ACB IS PART OF THE TCB. THE DEFINITION IS IN COPYBOOK
* P49000D INCLUDED IN THE LINKAGE SECTION OF THIS PROGRAM.
*-----
000100*** START OF U48004D *-----
000200* CDMF ACTION LIST FOR 'CDMF CONTROL BLOCK' AND MISCELLANEOUS
000300* CONSTANTS FOR THE 'CDMF CROSS REFERENCE CONTROL BLOCK'.
000400*-----
000500 01 CDMF-ACTION-LIST.
000600    05 CDMF-CODE-VALUES.
000700    10 CDMF-NO-OP.
000800          15 FILLER PIC S9(4) COMP VALUE +000
000900    10 CDMF-ADD.
001000          15 FILLER PIC S9(4) COMP VALUE +000
001100    10 CDMF-ADD-DUMMY.
001200          15 FILLER PIC S9(4) COMP VALUE +000
001300    10 CDMF-REPLACE.
001400          15 FILLER PIC S9(4) COMP VALUE +000
001500    10 CDMF-CHANGE.
001600          15 FILLER PIC S9(4) COMP VALUE +000
001700    10 CDMF-DELETE.
001800          15 FILLER PIC S9(4) COMP VALUE +000
001900    10 CDMF-INQ.
002000          15 FILLER PIC S9(4) COMP VALUE +000
002100    10 CDMF-NXT.
002200          15 FILLER PIC S9(4) COMP VALUE +000
002300    10 CDMF-NXTE.
002400          15 FILLER PIC S9(4) COMP VALUE +000
002500    10 CDMF-KGE.
002600          15 FILLER PIC S9(4) COMP VALUE +001
002700    SKIP1
002800*-----
002900*           ADVANCED CDMF RETRIEVAL ACTION CODES
003000*-----
003100    10 CDMF-INQ-FROM-TABLE.
003200          15 FILLER PIC S9(4) COMP VALUE +000
003300    10 CDMF-NXT-FROM-TABLE.
003400          15 FILLER PIC S9(4) COMP VALUE +000
003500    10 CDMF-NXTE-FROM-TABLE.
003600          15 FILLER PIC S9(4) COMP VALUE +001
003700    10 CDMF-KGE-FROM-TABLE.

```



Umbrella Programming

Problem Specifications—SORT Access

```
003800                      15 FILLER PIC S9(4) COMP VALUE +001
003900      10 CDMF-INQ-FROM-DATABASE.
004000                      15 FILLER PIC S9(4) COMP VALUE +001
004100      10 CDMF-NXT-FROM-DATABASE.
004200                      15 FILLER PIC S9(4) COMP VALUE +001
004300      10 CDMF-NXTE-FROM-DATABASE.
004400                      15 FILLER PIC S9(4) COMP VALUE +001
004500      10 CDMF-KGE-FROM-DATABASE.
004600                      15 FILLER PIC S9(4) COMP VALUE +001
004700      SKIP1
004800*-----*
004900*      ADVANCED CDMF LOGGING ACTION CODES
005000*-----*
005100      10 CDMF-LOG-BEFORE-CHANGE.
005200                      15 FILLER PIC S9(4) COMP VALUE +001
005300      10 CDMF-LOG-BEFORE-DELETE.
005400                      15 FILLER PIC S9(4) COMP VALUE +001
005500      10 CDMF-LOG-AFTER-ADD.
005600                      15 FILLER PIC S9(4) COMP VALUE +001
005700      10 CDMF-LOG-AFTER-CHANGE.
005800                      15 FILLER PIC S9(4) COMP VALUE +002
005900      SKIP1
006000*-----*
006100* CONSTANTS FOR THE 'CDMF CROSS REFERENCE CONTROL BLOCK', DATA
006200* GROUP 48008, COPYBOOK U48008D
006300* NOTE: THE '66' LEVEL IS USED TO KEEP WORKING STORAGE TO A
006400*     MINIMUM. THE 'RENAMES' CLAUSE HERE MEANS THAT THE ENTRY HAS
006500*     THE SAME EXACT VALUES AS THE ENTRY IT REDEFINES (RENAMES).
006600*-----*
006700*
006800**** VALUES FOR U008-SOURCE-TYPE
006900      10 XREF-SOURCE-TYPE-HDB          PIC X    VALUE HIGH-VALUES.
007000*
007100**** VALUES FOR U008-ACTION (ACTION ON SOURCE ENTRY W/IN HOST)
007200      66 XREF-INQ-SOURCE-ENTRY        RENAMES CDMF-ADD.
007300      66 XREF-1ST-SOURCE-ENTRY        RENAMES CDMF-ADD-DUMMY.
007400      66 XREF-NXT-SOURCE-ENTRY        RENAMES CDMF-CHANGE.
007500      66 XREF-1ST-ENTRY-NXT-HOST      RENAMES CDMF-DELETE.
007600      66 XREF-KEY-GE-SOURCE-ENTRY      RENAMES CDMF-INQ.
007700      66 XREF-ADD-SOURCE-ENTRY        RENAMES CDMF-NXT.
007800      66 XREF-DEL-SOURCE-ENTRY        RENAMES CDMF-NXTE.
007900      66 XREF-LAST-CALL-DR-PHS2      RENAMES CDMF-INQ-FROM-TABLE
008000*
008100**** VALUES FOR U008-RESULT
008200      66 XREF-ACTION-SUCCESSFUL      RENAMES CDMF-NO-OP.
008300      66 XREF-SOURCE-ENTRY-NOT-FOUND  RENAMES CDMF-ADD.
008400      66 XREF-HOST-KEY-NOT-FOUND      RENAMES CDMF-ADD-DUMMY.
008500      66 XREF-END-OF-ENTRIES-FOR-HOST
008600                      RENAMES CDMF-REPLACE.
008700      66 XREF-NO-SUCH-SOURCE-FMT      RENAMES CDMF-DELETE.
008800      66 XREF-NO-SUCH-HOST-FMT        RENAMES CDMF-INQ.
008900      66 XREF-DB-NOT-AVAILABLE       RENAMES CDMF-NXT-FROM-TABLE
009000*
009100**** VALUES FOR U008-FEEDBACK (FEEDBACK FROM SUCCESSFUL UPDATES)
009200      66 XREF-DEL-HOST-NOT-FOUND      RENAMES CDMF-ADD.
009300      66 XREF-DEL-NO-XRFS-ON-HOST      RENAMES CDMF-ADD-DUMMY.
009400      66 XREF-DEL-NO-MATCHING-ENTRY    RENAMES CDMF-REPLACE.
009500      66 XREF-ADD-DUPE-ENTRY        RENAMES CDMF-DELETE.
009600      66 XREF-ADD-NO-EXP-DATA        RENAMES CDMF-INQ.
009700      66 XREF-ADD-DUMMY-ADDED       RENAMES CDMF-NXT.
009800*
009900*---* END OF U48004D *-----*
          EJECT
*-----*
*   THE FOLLOWING COPYBOOK U48024D DEFINES THE POSSIBLE RESULT
```



Umbrella Programming

Problem Specifications—SORT Access

```

*   CODES RETURNED BY CDMF/PCD PROCESSING.
*   THE RESULT FOR THE ACTION IS PLACED INTO CDMF-RESULT FIELD
*   IN THE APPPLIACTION CONTROL BLOCK PRIOR TO RETURNING TO THIS
*   APPLICATION PROGRAM.
*   THE ACB IS PART OF THE TCB.  THE DEFINITION IS IN COPYBOOK
*   P49000D INCLUDED IN THE LINKAGE SECTION OF THIS PROGRAM.
*-----
000100*---* START OF U48024D *----* CDMF RESULT LIST *-----
000200*
000300 01 CDMF-RESULT-LIST.
000400    05 CDMF-RESULT-CODES.
000500*   CDMF-NO-ERRORS
000600           10 FILLER PIC S9(9) COMP VALUE +0000
000700*   CDMF-DB-NOT-AVAILABLE
000800           10 FILLER PIC S9(9) COMP VALUE +0000
000900*   CDMF-INVALID-ACTION
001000           10 FILLER PIC S9(9) COMP VALUE +4800
001100*   CDMF-ITEM-NOT-FOUND
001200           10 FILLER PIC S9(9) COMP VALUE +4800
001300*   CDMF-FORMAT-NOT-FOUND
001400           10 FILLER PIC S9(9) COMP VALUE +4800
001500*   CDMF-UNABLE-TO-ALLOC-DG
001600           10 FILLER PIC S9(9) COMP VALUE +4800
001700*   CDMF-END-OF-FORMAT
001800           10 FILLER PIC S9(9) COMP VALUE +4800
001900*   CDMF-DUPE-KEY-ON-ADD
002000           10 FILLER PIC S9(9) COMP VALUE +4800
002100*   CDMF-INVALID-EFF-DATE
002200           10 FILLER PIC S9(9) COMP VALUE +4800
002300*   CDMF-INVALID-APPL
002400           10 FILLER PIC S9(9) COMP VALUE +4800
002500*   CDMF-SKEY-NOT-FOUND
002600           10 FILLER PIC S9(9) COMP VALUE +4800
002700*   CDMF-SKEY-READ-ERR
002800           10 FILLER PIC S9(9) COMP VALUE +4801
002900*   CDMF-INVALID-CC-NO
003000           10 FILLER PIC S9(9) COMP VALUE +4801
003100*   CDMF-INVALID-FAMILY
003200           10 FILLER PIC S9(9) COMP VALUE +4801
003300*   CDMF-NO-OWNER-CHANGE
003400           10 FILLER PIC S9(9) COMP VALUE +4801
003500*   CDMF-SECURITY-VIOLATION
003600           10 FILLER PIC S9(9) COMP VALUE +4803
003700*   CDMF-SECURITY-INACTIVE
003800           10 FILLER PIC S9(9) COMP VALUE +4803
003900*   CDMF-DUMMY-REC-FOUND
004000           10 FILLER PIC S9(9) COMP VALUE +4809
004100*   CDMF-CCNR-MISSING
004200           10 FILLER PIC S9(9) COMP VALUE +4837
004300*   CDMF-CCNR-CLOSED
004400           10 FILLER PIC S9(9) COMP VALUE +4837
004500    05 FILLER           REDEFINES CDMF-RESULT-CODES.
004600           10 FILLER PIC XX.
004700    10 CDMF-NO-ERRORS      PIC XX.
004800           10 FILLER PIC XX.
004900    10 CDMF-DB-NOT-AVAILABLE  PIC XX.
005000           10 FILLER PIC XX.
005100    10 CDMF-INVALID-ACTION    PIC XX.
005200           10 FILLER PIC XX.
005300    10 CDMF-ITEM-NOT-FOUND      PIC XX.
005400           10 FILLER PIC XX.
005500    10 CDMF-FORMAT-NOT-FOUND    PIC XX.
005600           10 FILLER PIC XX.
005700    10 CDMF-UNABLE-TO-ALLOC-DG    PIC XX.
005800           10 FILLER PIC XX.

```



Umbrella Programming

Problem Specifications—SORT Access

```
005900      10 CDMF-END-OF-FORMAT      PIC XX.
006000
006100      10 CDMF-DUPE-KEY-ON-ADD    PIC XX.
006200
006300      10 CDMF-INVALID-EFF-DATE   PIC XX.
006400
006500      10 CDMF-INVALID-APPL       PIC XX.
006600
006700      10 CDMF-SKEY-NOT-FOUND     PIC XX.
006800
006900      10 CDMF-SKEY-READ-ERR      PIC XX.
007000
007100      10 CDMF-INVALID-CC-NO      PIC XX.
007200
007300      10 CDMF-INVALID-FAMILY     PIC XX.
007400
007500      10 CDMF-NO-OWNER-CHANGE    PIC XX.
007600
007700      10 CDMF-SECURITY-VIOLATION PIC XX.
007800
007900      10 CDMF-SECURITY-INACTIVE  PIC XX.
008000
008100      10 CDMF-DUMMY-REC-FOUND   PIC XX.
008200
008300      10 CDMF-CCNR-MISSING     PIC XX.
008400
008500      10 CDMF-CCNR-CLOSED      PIC XX.
008600*
008700*----* END OF U48024D *-----  
EJECT  
*-----  
* THE FOLLOWING COPYBOOK P49003D DEFINES SOME OF THE MORE  
* COMMONLY USED PEM ACTIVITIES. ACTIVITIES 1 THROUGH 100  
* ARE AUTOMATICALLY AUTHORIZED FOR USE BY ALL PROGRAMS AND  
* NEED NOT BE SPECIFIED IN THE PROGRAM DEFINITION IN THE  
* PROCESS DICTIONARY.  
*-----  
000100**** START OF P49003D ***** PEM COMMON ACTIVITIES ***  
000200*
000300 01  PEM-COMMON-ACTIVITIES.  
000400 05  PEM-ACTIVITIES.  
000500 10  CA-LONG-PEM-END-PROG.  
000600
000700 10  CA-LONG-PEM-END-TRANS.  
000800
000900 10  CA-LONG-PEM-DYN-DG-ALOC.  
001000
001100 10  CA-LONG-PEM-DYN-DG-REL.  
001200
001300 10  CA-LONG-PEM-DYN-DG-INIT.  
001400
001500 10  CA-LONG-PEM-SYSPRINT-WRITE.  
001600
001700 10  CA-LONG-PEM-NO-OP.  
001800
001900 10  CA-LONG-PEM-TRANS-DUMP-RETURN.  
002000
002100 10  CA-LONG-PEM-TRANS-DUMP-END.  
002200
002300 10  CA-LONG-PEM-DUMP-TCB-TRACE.  
002400
002500 10  CA-LONG-PEM-DUMP-DG-RETURN.  
002600
002700 10  CA-LONG-PEM-DUMP-DG-END.  
002800
```



Umbrella Programming

Problem Specifications—SORT Access

```

002900    10 CA-LONG-PEM-ENABLE-ABEND-EXIT.
003000                                15 FILLER PIC S9(9) COMP VALUE +0020
003100    10 CA-LONG-PEM-DISABLE-ABEND-EXIT.
003200                                15 FILLER PIC S9(9) COMP VALUE +0021
003300    10 CA-LONG-PEM-CHECKPOINT.
003400                                15 FILLER PIC S9(9) COMP VALUE +0028
003500    10 CA-LONG-PEM-DYN-DG-NO-INIT.
003600                                15 FILLER PIC S9(9) COMP VALUE +0029
003700    10 CA-LONG-PEM-ROLLBACK.
003800                                15 FILLER PIC S9(9) COMP VALUE +0030
003900    10 CA-LONG-PEM-DLI-SYNCPOINT.
004000                                15 FILLER PIC S9(9) COMP VALUE +0032
004100    10 CA-LONG-PEM-USERCC-EXCEP.
004200                                15 FILLER PIC S9(9) COMP VALUE +0035
004300    10 CA-LONG-PEM-MSG9-ABEND.
004400                                15 FILLER PIC S9(9) COMP VALUE +0039
004500    10 CA-LONG-PEM-APPC-SYNCPOINT.
004600                                15 FILLER PIC S9(9) COMP VALUE +0056
004700    10 CA-LONG-PEM-DYN-DG-LENGTH.
004800                                15 FILLER PIC S9(9) COMP VALUE +0074
004900    10 CA-LONG-PEM-DYN-PTR-REL.
005000                                15 FILLER PIC S9(9) COMP VALUE +0075
005100    10 CA-LONG-PEM-DYN-PTR-INIT.
005200                                15 FILLER PIC S9(9) COMP VALUE +0076
005201    10 CA-LONG-PEM-DYN-PTR-ANO.
005202                                15 FILLER PIC S9(9) COMP VALUE +0077
005300    10 CA-LONG-PEM-PDG-HAS-CHANGED.
005400                                15 FILLER PIC S9(9) COMP VALUE +0088
005500*
005600    05 FILLER          REDEFINES PEM-ACTIVITIES.
005700                                10 FILLER PIC XX
005800    10 CA-PEM-END-PROG      PIC XX.
005900                                10 FILLER PIC XX
006000    10 CA-PEM-END-TRANS     PIC XX.
006100                                10 FILLER PIC XX
006200    10 CA-PEM-DYN-DG-ALOC     PIC XX.
006300                                10 FILLER PIC XX
006400    10 CA-PEM-DYN-DG-REL      PIC XX.
006500                                10 FILLER PIC XX
006600    10 CA-PEM-DYN-DG-INIT      PIC XX.
006700                                10 FILLER PIC XX
006800    10 CA-PEM-SYSPRINT-WRITE    PIC XX.
006900                                10 FILLER PIC XX
007000    10 CA-PEM-NO-OP          PIC XX.
007100                                10 FILLER PIC XX
007200    10 CA-PEM-TRANS-DUMP-RETURN    PIC XX.
007300                                10 FILLER PIC XX
007400    10 CA-PEM-TRANS-DUMP-END      PIC XX.
007500                                10 FILLER PIC XX
007600    10 CA-PEM-DUMP-TCB-TRACE     PIC XX.
007700                                10 FILLER PIC XX
007800    10 CA-PEM-DUMP-DG-RETURN     PIC XX.
007900                                10 FILLER PIC XX
008000    10 CA-PEM-DUMP-DG-END        PIC XX.
008100                                10 FILLER PIC XX
008200    10 CA-PEM-ENABLE-ABEND-EXIT    PIC XX.
008300                                10 FILLER PIC XX
008400    10 CA-PEM-DISABLE-ABEND-EXIT    PIC XX.
008500                                10 FILLER PIC XX
008600    10 CA-PEM-CHECKPOINT        PIC XX.
008700                                10 FILLER PIC XX
008800    10 CA-PEM-DYN-DG-NO-INIT        PIC XX.
008900                                10 FILLER PIC XX
009000    10 CA-PEM-ROLLBACK         PIC XX.
009100                                10 FILLER PIC XX

```



Umbrella Programming

Problem Specifications—SORT Access

```
009200      10 CA-PEM-DLI-SYNCPOINT          PIC XX.           10 FILLER PIC XX
009300
009400      10 CA-PEM-USERCC-EXCEP          PIC XX.           10 FILLER PIC XX
009500
009600      10 CA-PEM-MSG9-ABEND          PIC XX.           10 FILLER PIC XX
009700
009800      10 CA-PEM-APPC-SYNCPOINT          PIC XX.           10 FILLER PIC XX
009900
010000      10 CA-PEM-DYN-DG-LENGTH          PIC XX.           10 FILLER PIC XX
010100
010200      10 CA-PEM-DYN-PTR-REL          PIC XX.           10 FILLER PIC XX
010300
010400      10 CA-PEM-DYN-PTR-INIT          PIC XX.           10 FILLER PIC XX
010401
010402      10 CA-PEM-DYN-PTR-ANO          PIC XX.           10 FILLER PIC XX
010500
010600      10 CA-PEM-PDG-HAS-CHANGED        PIC XX.           10 FILLER PIC XX
010700*
010800***** END OF P49003D *****  
EJECT  
*-----  
* THE FOLLOWING COPYBOOK P49022D DEFINES THE POSSIBLE VALUES  
* THAT MIGHT BE PLACED INTO A DATA GROUP ACTION FIELD PRIOR  
* TO ISSUING A PEM DATA BASE ACTIVITY.    DGA ACTION  
*-----  
000100*  
000200*---* START OF P49022D *---* PEM ACTION CODES *-----  
000300*  
000400 01  PEM-DATA-GROUP-ACTION-CODES.  
000500 05  DGA-CODE-VALUES.  
000600 10  DGA-NO-OP.  
000700
000800 10  DGA-READ.                  15  FILLER PIC S9(4) COMP VALUE +0000
000900
001000 10  DGA-WRITE.                 15  FILLER PIC S9(4) COMP VALUE +0001
001100
001200 10  DGA-HOLD-P.                15  FILLER PIC S9(4) COMP VALUE +0002
001300
001400 10  DGA-ERASE.                 15  FILLER PIC S9(4) COMP VALUE +0003
001500
001600 10  DGA-READ-KEY-GE.          15  FILLER PIC S9(4) COMP VALUE +0004
001700
001800 10  DGA-FORCE-WRITE.         15  FILLER PIC S9(4) COMP VALUE +0005
001900
002000 10  DGA-READ-KEY-EQ.          15  FILLER PIC S9(4) COMP VALUE +0006
002100
002200 10  DGA-INSERT-FIRST.        15  FILLER PIC S9(4) COMP VALUE +0007
002300
002400 10  DGA-READ-LAST-REC.       15  FILLER PIC S9(4) COMP VALUE +0008
002500
002600 10  DGA-INSERT-LAST.        15  FILLER PIC S9(4) COMP VALUE +0009
002700
002800 10  DGA-READ-FIRST.          15  FILLER PIC S9(4) COMP VALUE +0010
002900
003000 10  DGA-INSERT-HERE.        15  FILLER PIC S9(4) COMP VALUE +0011
003100
003200 10  DGA-END-REQUEST.        15  FILLER PIC S9(4) COMP VALUE +0012
003300
003400 10  DGA-FORCE-NO-OP.        15  FILLER PIC S9(4) COMP VALUE +0014
003500
003600 10  DGA-CLOSE.              15  FILLER PIC S9(4) COMP VALUE +0016
003700
003800 10  DGA-OPEN-OUTPUT.        15  FILLER PIC S9(4) COMP VALUE +0036
003900
004000 10  DGA-OPEN-INPUT.         15  FILLER PIC S9(4) COMP VALUE +0037
```



Umbrella Programming

Problem Specifications—SORT Access

```

004100                               15 FILLER PIC S9(4) COMP VALUE +0038
004200      10 DGA-OPEN-UPDATE.          15 FILLER PIC S9(4) COMP VALUE +0039
004300                               15 FILLER PIC S9(4) COMP VALUE +0040
004400      10 DGA-POINT.              15 FILLER PIC S9(4) COMP VALUE +0041
004500      10 DGA-ENABLE-KEY-RANGE.    15 FILLER PIC S9(4) COMP VALUE +0042
004700      10 DGA-DISABLE-KEY-RANGE.   15 FILLER PIC S9(4) COMP VALUE +0043
004800      10 DGA-ENABLE-MULT-KRDB.    15 FILLER PIC S9(4) COMP VALUE +0044
005000      10 DGA-ENABLE-MULT-KRDB.    15 FILLER PIC S9(4) COMP VALUE +0045
005100                               15 FILLER PIC S9(4) COMP VALUE +0046
005200*
005300*----* END OF P49022D *-----*
                                EJECT
*-----
*   THE FOLLOWING COPYBOOK P49023D CONTAINS THE POSSIBLE VALUES
*   THAT MIGHT BE CONTAINED IN A DATA GROUP RESULTS FIELD AFTER
*   A PEM DATA BASE ACTIVITY HAS BEEN ISSUED.      DGR RESULT
*           # # #
*   WHEN CONTROL IS RETURNED TO AN APPLICATION PROGRAM ON
*   COMPLETION OF A DATA BASE ACTIVITY, EACH DATA GROUP INVOLVED
*   IN THE ACTIVITY WILL CONTAIN A RESULT CODE REFLECTING THE
*   RESULT OF THE REQUESTED ACTION.  IF EACH INDIVIDUAL DATA
*   GROUP'S RESULT FIELD IS ZERO, THE TCB RESULT FIELD
*   (TCB-RESULT) WILL ALSO CONTAIN ZERO.
*
*   TCB-RESULT WILL BE SET TO NON-ZERO IF ANY OF THE INDIVIDUAL
*   DATA GROUPS IS NON-ZERO.  IT IS THE RESPONSIBILITY OF THE
*   APPLICATION PROGRAM TO DETERMINE THE CAUSE AND SEVERITY OF
*   THE RESULT CONDITION.  AN END-OF-DATA ON THE BASE DATA
*   GROUP (EOF) WILL CAUSE A NON-ZERO VALUE TO BE RETURNED TO
*   THE TCB-RESULT FIELD.
*-----
000100*---* START OF P49023D *---* PEM RESULT CODES *-----*
000200*
000300 01  PEM-DATA-GROUP-RESULT-CODES.
000400      05 DGR-CODE-VALUES.
000500      10 DGR-OK.
000600                               15 FILLER PIC S9(4) COMP VALUE +0000
000700      10 DGR-END-DATA.          15 FILLER PIC S9(4) COMP VALUE +0001
000800                               15 FILLER PIC S9(4) COMP VALUE +0002
000900      10 DGR-I-O-ERR.          15 FILLER PIC S9(4) COMP VALUE +0003
001000                               15 FILLER PIC S9(4) COMP VALUE +0004
001100      10 DGR-DUP-KEY.          15 FILLER PIC S9(4) COMP VALUE +0005
001200                               15 FILLER PIC S9(4) COMP VALUE +0006
001300      10 DGR-NO-FIND.          15 FILLER PIC S9(4) COMP VALUE +0007
001400                               15 FILLER PIC S9(4) COMP VALUE +0008
001500      10 DGR-SKEY-NO-FIND.     15 FILLER PIC S9(4) COMP VALUE +0009
001600                               15 FILLER PIC S9(4) COMP VALUE +000A
001700      10 DGR-KRID-DISABLED.    15 FILLER PIC S9(4) COMP VALUE +000B
001800                               15 FILLER PIC S9(4) COMP VALUE +000C
001900      10 DGR-DB-NOT-AVAIL.     15 FILLER PIC S9(4) COMP VALUE +000D
002000                               15 FILLER PIC S9(4) COMP VALUE +000E
002100      10 DGR-SQLCA-ERROR.     15 FILLER PIC S9(4) COMP VALUE +000F
002200                               15 FILLER PIC S9(4) COMP VALUE +0010
002300      10 DGR-LEN-ERR.         15 FILLER PIC S9(4) COMP VALUE +0011
002400                               15 FILLER PIC S9(4) COMP VALUE +0012
002500*
002600*----* END OF P49023D *-----*
                                EJECT
*-----
*   THE FOLLOWING COPYBOOK T58007D CONTAINS ALL THE POSSIBLE
*   ACTIONS THAT MIGHT BE REQUESTED OF THE DATE SERVICES SYSTEM.
*
*   THE REQUESTED ACTION CODE MUST BE LOADED INTO THE ACTION

```



Umbrella Programming

Problem Specifications—SORT Access

```
* FIELD DCB-ACTION IN THE DATE CONTROL BLOCK DATA GROUP 2000
* PRIOR TO THE LINK TO DATE SERVICES ACTIVITY 1900.
* THE DATE CONTROL BLOCK IS DEFINED BY COPYBOOK T58001D
* INCLUDED IN THE LINKAGE SECTION OF THIS PROGRAM.
*-----
000100*--- START OF T58007D *----* DATE ROUTINE ACTIONS *-----
000200*
000300*
000400*
000500*****{*}
000600*
000700 01 DCB-ACTIONS.
000800    05 DCB-PROG-LINK.
000900        10 DCB-LINK-DSV-LONG.
001000        15 FILLER      PIC S9(9) COMP VALUE +1900.
001100        10 FILLER      REDEFINES DCB-LINK-DSV-LONG.
001200        15 FILLER      PIC XX.
001300        15 DCB-LINK-DSV  PIC XX.
001400*
001500    05 DCB-PROG-ACTIONS.
001600* GREGORIAN TO JULIAN
001700        10 DCB-AC-GREG-TO-JUL.
001800        15 FILLER      PIC S9(4) COMP VALUE +0001.
001900* GREGORIAN TO CALCULATION
002000        10 DCB-AC-GREG-TO-CALC.
002100        15 FILLER      PIC S9(4) COMP VALUE +0002.
002200* GREGORIAN TO NUMERIC DISPLAY
002300        10 DCB-AC-GREG-TO-NUM-DSP.
002400        15 FILLER      PIC S9(4) COMP VALUE +0003.
002500* GREGORIAN TO ALPHANUMERIC DISPLAY
002600        10 DCB-AC-GREG-TO-ALPHA-DSP.
002700        15 FILLER      PIC S9(4) COMP VALUE +0004.
002800* JULIAN TO GREGORIAN
002900        10 DCB-AC-JUL-TO-GREG.
003000        15 FILLER      PIC S9(4) COMP VALUE +0010.
003100* JULIAN TO CALCULATION
003200        10 DCB-AC-JUL-TO-CALC.
003300        15 FILLER      PIC S9(4) COMP VALUE +0011.
003400* JULIAN TO NUMERIC DISPLAY
003500        10 DCB-AC-JUL-TO-NUM-DSP.
003600        15 FILLER      PIC S9(4) COMP VALUE +0012.
003700* JULIAN TO ALPHANUMERIC DISPLAY
003800        10 DCB-AC-JUL-TO-ALPHA-DSP.
003900        15 FILLER      PIC S9(4) COMP VALUE +0013.
004000* CALCULATION TO GREGORIAN
004100        10 DCB-AC-CALC-TO-GREG.
004200        15 FILLER      PIC S9(4) COMP VALUE +0020.
004300* CALCULATION TO JULIAN
004400        10 DCB-AC-CALC-TO-JUL.
004500        15 FILLER      PIC S9(4) COMP VALUE +0021.
004600* CALCULATION TO NUMERIC DISPLAY
004700        10 DCB-AC-CALC-TO-NUM-DSP.
004800        15 FILLER      PIC S9(4) COMP VALUE +0022.
004900* CALCULATION TO ALPHANUMERIC DISPLAY
005000        10 DCB-AC-CALC-TO-ALPHA-DSP.
005100        15 FILLER      PIC S9(4) COMP VALUE +0023.
005200* BUMP TO NEXT DATE GREGORIAN
005300        10 DCB-AC-BND-GREG.
005400        15 FILLER      PIC S9(4) COMP VALUE +0030.
005500* BUMP TO NEXT DATE JULIAN
005600        10 DCB-AC-BND-JUL.
005700        15 FILLER      PIC S9(4) COMP VALUE +0031.
005800* BUMP TO NEXT DATE CALCULATION
005900        10 DCB-AC-BND-CALC.
006000        15 FILLER      PIC S9(4) COMP VALUE +0032.
```



Umbrella Programming

Problem Specifications—SORT Access

```

006100* BUMP TO NEXT DATE GREGORIAN 360
006200      10 DCB-AC-BND-G360.
006300      15 FILLER      PIC S9(4) COMP VALUE +0033.
006400* BUMP TO NEXT BUSINESS DATE GREGORIAN
006500      10 DCB-AC-BNBD-GREG.
006600      15 FILLER      PIC S9(4) COMP VALUE +0040.
006700* BUMP TO NEXT BUSINESS DATE JULIAN
006800      10 DCB-AC-BNBD-JUL.
006900      15 FILLER      PIC S9(4) COMP VALUE +0041.
007000* BUMP TO NEXT BUSINESS DATE CALCULATION
007100      10 DCB-AC-BNBD-CALC.
007200      15 FILLER      PIC S9(4) COMP VALUE +0042.
007300* GREGORIAN DAY DIFFERENCE
007400      10 DCB-AC-DIFF-GREG.
007500      15 FILLER      PIC S9(4) COMP VALUE +0050.
007600* JULIAN DAY DIFFERENCE
007700      10 DCB-AC-DIFF-JUL.
007800      15 FILLER      PIC S9(4) COMP VALUE +0051.
007900* CALCULATION DAY DIFFERENCE
008000      10 DCB-AC-DIFF-CALC.
008100      15 FILLER      PIC S9(4) COMP VALUE +0052.
008200* GREGORIAN 360 DAY DIFFERENCE
008300      10 DCB-AC-DIFF-G360.
008400      15 FILLER      PIC S9(4) COMP VALUE +0053.
008500* BUSINESS DAY DIFFERENCE GREGORIAN
008600      10 DCB-AC-BUS-DIFF-GREG.
008700      15 FILLER      PIC S9(4) COMP VALUE +0060.
008800* BUSINESS DAY DIFFERENCE JULIAN
008900      10 DCB-AC-BUS-DIFF-JUL.
009000      15 FILLER      PIC S9(4) COMP VALUE +0061.
009100* BUSINESS DAY DIFFERENCE CALCULATION
009200      10 DCB-AC-BUS-DIFF-CALC.
009300      15 FILLER      PIC S9(4) COMP VALUE +0062.
009400* GREGORIAN MONTHS DIFFERENCE
009500      10 DCB-AC-MO-DIFF-GREG.
009600      15 FILLER      PIC S9(4) COMP VALUE +0070.
009700* HOLIDAY TABLE REQUEST
009800      10 DCB-AC-HOLIDAYS.
009900      15 FILLER      PIC S9(4) COMP VALUE +0080.
010000* HOLIDAY TABLE RENEW REQUEST
010100      10 DCB-AC-RENEW-HOLIDAYS.
010200      15 FILLER      PIC S9(4) COMP VALUE +0081.
010300* FORMATED DATE (CYYMMDD) TO GREGORIAN
010400      10 DCB-AC-FMT00-GREG.
010500      15 FILLER      PIC S9(4) COMP VALUE +0100.
010600* FORMATED DATE (YY-MM-DD) TO GREGORIAN
010700      10 DCB-AC-FMT01-GREG.
010800      15 FILLER      PIC S9(4) COMP VALUE +0101.
010900* FORMATED DATE (MMDDYY) TO GREGORIAN
011000      10 DCB-AC-FMT02-GREG.
011100      15 FILLER      PIC S9(4) COMP VALUE +0102.
011200* FORMATED DATE (MM-DD-YY) TO GREGORIAN
011300      10 DCB-AC-FMT03-GREG.
011400      15 FILLER      PIC S9(4) COMP VALUE +0103.
011500* FORMATED DATE GREGORIAN TO (MMDDYYC)
011600      10 DCB-AC-FMT04-GREG.
011700      15 FILLER      PIC S9(4) COMP VALUE +0104.
011800* FORMATED DATE GREGORIAN TO (MMDDYY)
011900      10 DCB-AC-FMT05-GREG.
012000      15 FILLER      PIC S9(4) COMP VALUE +0105.
012100* DATE STATUS CHECK ALL
012200      10 DCB-AC-DSC-ALL.
012300      15 FILLER      PIC S9(4) COMP VALUE +0130.
012400* DATE STATUS CHECK PHYSICAL FIRST OF MONTH
012500      10 DCB-AC-DSC-PHY-FOM.

```



Umbrella Programming

Problem Specifications—SORT Access

```
012600           15 FILLER          PIC S9(4) COMP VALUE +0131.
012700* DATE STATUS CHECK PHYSICAL FIRST OF QUARTER
012800           10 DCB-AC-DSC-PHY-FOQ.
012900           15 FILLER          PIC S9(4) COMP VALUE +0132.
013000* DATE STATUS CHECK PHYSICAL FIRST OF YEAR
013100           10 DCB-AC-DSC-PHY-FOY.
013200           15 FILLER          PIC S9(4) COMP VALUE +0133.
013300* DATE STATUS CHECK PHYSICAL END OF MONTH
013400           10 DCB-AC-DSC-PHY-EOM.
013500           15 FILLER          PIC S9(4) COMP VALUE +0134.
013600* DATE STATUS CHECK PHYSICAL END OF QUARTER
013700           10 DCB-AC-DSC-PHY-EOQ.
013800           15 FILLER          PIC S9(4) COMP VALUE +0135.
013900* DATE STATUS CHECK PHYSICAL END OF YEAR
014000           10 DCB-AC-DSC-PHY-EOY.
014100           15 FILLER          PIC S9(4) COMP VALUE +0136.
014200* DATE STATUS CHECK BUSINESS FIRST OF MONTH
014300           10 DCB-AC-DSC-BUS-FOM.
014400           15 FILLER          PIC S9(4) COMP VALUE +0137.
014500* DATE STATUS CHECK BUSINESS FIRST OF QUARTER
014600           10 DCB-AC-DSC-BUS-FOQ.
014700           15 FILLER          PIC S9(4) COMP VALUE +0138.
014800* DATE STATUS CHECK BUSINESS FIRST OF YEAR
014900           10 DCB-AC-DSC-BUS-FOY.
015000           15 FILLER          PIC S9(4) COMP VALUE +0139.
015100* DATE STATUS CHECK BUSINESS END OF MONTH
015200           10 DCB-AC-DSC-BUS-EOM.
015300           15 FILLER          PIC S9(4) COMP VALUE +0140.
015400* DATE STATUS CHECK BUSINESS END OF QUARTER
015500           10 DCB-AC-DSC-BUS-EOQ.
015600           15 FILLER          PIC S9(4) COMP VALUE +0141.
015700* DATE STATUS CHECK BUSINESS END OF YEAR
015800           10 DCB-AC-DSC-BUS-EOY.
015900           15 FILLER          PIC S9(4) COMP VALUE +0142.
016000* DATE STATUS CHECK BUSINESS DAY OR NON BUSINESS DAY
016100           10 DCB-AC-DSC-BUS-DAY.
016200           15 FILLER          PIC S9(4) COMP VALUE +0143.
016300* DATE STATUS CHECK DAY OF WEEK
016400           10 DCB-AC-DSC-WEEK-DAY.
016500           15 FILLER          PIC S9(4) COMP VALUE +0144.
016600* DATE STATUS CHECK LEAP YEAR
016700           10 DCB-AC-DSC-LEAP-YEAR.
016800           15 FILLER          PIC S9(4) COMP VALUE +0145.
016900* DATE STATUS CHECK PHYSICAL FIRST OF WEEK
017000           10 DCB-AC-DSC-PHY-FOW.
017100           15 FILLER          PIC S9(4) COMP VALUE +0146.
017200* DATE STATUS CHECK PHYSICAL END OF WEEK
017300           10 DCB-AC-DSC-PHY-EOW.
017400           15 FILLER          PIC S9(4) COMP VALUE +0147.
017500* DATE STATUS CHECK BUSINESS FIRST OF WEEK
017600           10 DCB-AC-DSC-BUS-FOW.
017700           15 FILLER          PIC S9(4) COMP VALUE +0148.
017800* DATE STATUS CHECK BUSINESS END OF WEEK
017900           10 DCB-AC-DSC-BUS-EOW.
018000           15 FILLER          PIC S9(4) COMP VALUE +0149.
018100* CALCULATE CURRENT BUSINESS DAY
018200           10 DCB-AC-CALC-CURR-BUS.
018300           15 FILLER          PIC S9(4) COMP VALUE +0160.
018400* CALCULATE NEXT WEEK DAY
018500           10 DCB-AC-CALC-NXT-WKDAY.
018600           15 FILLER          PIC S9(4) COMP VALUE +0161.
018700* CONVERT GREGORIAN TO SQL FORMATTED DATE
018800           10 DCB-HOGAN-TO-SQL-DATE.
018900           15 FILLER          PIC S9(4) COMP VALUE +0200.
019000* CONVERT SQL FORMATTED DATE TO GREGORIAN
```



Umbrella Programming

Problem Specifications—SORT Access

```

019100          10 DCB-SQL-TO-HOGAN-DATE.
019200            15 FILLER      PIC S9(4) COMP VALUE +0201.
019300*   CONVERT HOGAN TO SQL TIME
019400            10 DCB-HOGAN-TO-SQL-TIME.
019500            15 FILLER      PIC S9(4) COMP VALUE +0202.
019600*   CONVERT SQL TO HOGAN TIME
019700            10 DCB-SQL-TO-HOGAN-TIME.
019800            15 FILLER      PIC S9(4) COMP VALUE +0203.
019900*
020000*
020100*----* END OF T58007D *-----*
      EJECT
*-----
*   THE FOLLOWING COPYBOOK T58008D IS USED TO DEFINE THE POSSIBLE
*   RESULT VALUES RETURNED FROM A DATE SERVICES REQUEST IN THE
*   DCB-RESULT FIELD OF THE DATE CONTROL BLOCK DATA GROUP 2000.
*-----
000100*---* START OF T58008D *---* DATE ROUTINE RESULTS *-----
000200*
000300*
000400*
000500*****000600*
000700 01 DCB-RESULTS.
000800    05 DCB-RESULT-VALUES.
000900*   SUCCESSFUL COMPLETION
001000      10 FILLER      PIC S9(8) COMP VALUE +00000.
001100*   INVALID ACTION
001200      10 FILLER      PIC S9(8) COMP VALUE +58001.
001300*   INVALID GREGORIAN MONTH
001400      10 FILLER      PIC S9(8) COMP VALUE +58002.
001500*   INVALID GREGORIAN DAY
001600      10 FILLER      PIC S9(8) COMP VALUE +58003.
001700*   INVALID JULIAN DAY
001800      10 FILLER      PIC S9(8) COMP VALUE +58004.
001900*   INVALID CALCULATION DATE
002000      10 FILLER      PIC S9(8) COMP VALUE +58005.
002100*   INVALID DIFFERENCE FACTOR
002200      10 FILLER      PIC S9(8) COMP VALUE +58006.
002300*   INVALID CYCLE PERIOD
002400      10 FILLER      PIC S9(8) COMP VALUE +58050.
002500*   INVALID FREQ CODE
002600      10 FILLER      PIC S9(8) COMP VALUE +58051.
002700*   INVALID FREQ WEEK
002800      10 FILLER      PIC S9(8) COMP VALUE +58052.
002900*   INVALID FREQ DATE
003000      10 FILLER      PIC S9(8) COMP VALUE +58053.
003100*   INVALID FREQ BUMP
003200      10 FILLER      PIC S9(8) COMP VALUE +58054.
003300*   INVALID FREQ CYCLE
003400      10 FILLER      PIC S9(8) COMP VALUE +58055.
003500*   INVALID CYCLE EXCEPT
003600      10 FILLER      PIC S9(8) COMP VALUE +58056.
003700*   INVALID FREQ AMT
003800      10 FILLER      PIC S9(8) COMP VALUE +58057.
003900*   INVALID GREGORIAN MONTH IN DATE OUT
004000      10 FILLER      PIC S9(8) COMP VALUE +58102.
004100*   INVALID GREGORIAN DAY IN DATE OUT
004200      10 FILLER      PIC S9(8) COMP VALUE +58103.
004300*   INVALID JULIAN DAY IN DATE OUT
004400      10 FILLER      PIC S9(8) COMP VALUE +58104.
004500*   INVALID CALCULATION DATE IN DATE OUT
004600      10 FILLER      PIC S9(8) COMP VALUE +58105.
004700*   INVALID ACTION FOR DATE
004800      10 FILLER      PIC S9(8) COMP VALUE +58106.

```



Umbrella Programming

Problem Specifications—SORT Access

```
004900* NOT EFFECTIVE
005000    10 FILLER      PIC S9(8) COMP     VALUE +58150.
005100* MORE THAN 15 CYCLES
005200    10 FILLER      PIC S9(8) COMP     VALUE +58151.
005300* INVALID FROM-THRU RANGE
005400    10 FILLER      PIC S9(8) COMP     VALUE +58152.
005500* INVALID FIRST-LAST RANGE
005600    10 FILLER      PIC S9(8) COMP     VALUE +58153.
005700* WARNING CALCULATED DATES HAVE BEEN ADJUSTED.
005800    10 FILLER      PIC S9(8) COMP     VALUE +58154.
005900* SQL DATE IS INVALID
006000    10 FILLER      PIC S9(8) COMP     VALUE +58200.
006100* SQL FORMAT ON SITE CONTROL RECORD NOT SUPPORTED
006200    10 FILLER      PIC S9(8) COMP     VALUE +58201.
006300*
006400*-----*
006500*
006600*
006700    05 DCB-RESULT-NAMES    REDEFINES DCB-RESULT-VALUES.
006800    10 FILLER.
006900    15 FILLER          PIC XX.
007000    15 DCB-RS-OK        PIC XX.
007100    10 FILLER.
007200    15 FILLER          PIC XX.
007300    15 DCB-RS-INVALID-ACTION PIC XX.
007400    10 FILLER.
007500    15 FILLER          PIC XX.
007600    15 DCB-RS-INVALID-GREG-MONTH PIC XX.
007700    10 FILLER.
007800    15 FILLER          PIC XX.
007900    15 DCB-RS-INVALID-GREG-DAY  PIC XX.
008000    10 FILLER.
008100    15 FILLER          PIC XX.
008200    15 DCB-RS-INVALID-JUL-DAY  PIC XX.
008300    10 FILLER.
008400    15 FILLER          PIC XX.
008500    15 DCB-RS-INVALID-CALC-DATE PIC XX.
008600    10 FILLER.
008700    15 FILLER          PIC XX.
008800    15 DCB-RS-INVALID-DIFF-FACTOR PIC XX.
008900    10 FILLER.
009000    15 FILLER          PIC XX.
009100    15 DCB-RS-INVALID-CYCLE-PERIOD PIC XX.
009200    10 FILLER.
009300    15 FILLER          PIC XX.
009400    15 DCB-RS-INVALID-FREQ-CODE  PIC XX.
009500    10 FILLER.
009600    15 FILLER          PIC XX.
009700    15 DCB-RS-INVALID-FREQ-WEEK  PIC XX.
009800    10 FILLER.
009900    15 FILLER          PIC XX.
010000    15 DCB-RS-INVALID-FREQ-DATE  PIC XX.
010100    10 FILLER.
010200    15 FILLER          PIC XX.
010300    15 DCB-RS-INVALID-FREQ-BUMP  PIC XX.
010400    10 FILLER.
010500    15 FILLER          PIC XX.
010600    15 DCB-RS-INVALID-FREQ-CYCLE PIC XX.
010700    10 FILLER.
010800    15 FILLER          PIC XX.
010900    15 DCB-RS-INVALID-CYCLE-EXCEPT PIC XX.
011000    10 FILLER.
011100    15 FILLER          PIC XX.
011200    15 DCB-RS-INVALID-FREQ-AMT  PIC XX.
011300    10 FILLER.
```



Umbrella Programming

Problem Specifications—SORT Access

```

011400      15 FILLER          PIC XX.
011500      15 DCB-RS-INVALID-GREG-MO-DO  PIC XX.
011600      10 FILLER.
011700      15 FILLER          PIC XX.
011800      15 DCB-RS-INVALID-GREG-DAY-DO  PIC XX.
011900      10 FILLER.
012000      15 FILLER          PIC XX.
012100      15 DCB-RS-INVALID-JUL-DAY-DO  PIC XX.
012200      10 FILLER.
012300      15 FILLER          PIC XX.
012400      15 DCB-RS-INVALID-CALC-DATE-DO PIC XX.
012500      10 FILLER.
012600      15 FILLER          PIC XX.
012700      15 DCB-RS-INVALID-DATE-ACT   PIC XX.
012800      10 FILLER.
012900      15 FILLER          PIC XX.
013000      15 DCB-RS-NOT-EFFECTIVE   PIC XX.
013100      10 FILLER.
013200      15 FILLER          PIC XX.
013300      15 DCB-RS-MORE-CYCLES   PIC XX.
013400      10 FILLER.
013500      15 FILLER          PIC XX.
013600      15 DCB-RS-INVALID-FT-RANGE PIC XX.
013700      10 FILLER.
013800      15 FILLER          PIC XX.
013900      15 DCB-RS-INVALID-FL-RANGE PIC XX.
014000      10 FILLER.
014100      15 FILLER          PIC XX.
014200      15 DCB-RS-DATE-ADJUST   PIC XX.
014300      10 FILLER.
014400      15 FILLER          PIC XX.
014500      15 DCB-RS-SQL-DATE-INVALID PIC XX.
014600      10 FILLER.
014700      15 FILLER          PIC XX.
014800      15 DCB-RS-SQL-FORMAT-NOT-SUP PIC XX.
014900*
015000*----* END OF T58008D *-----*
      EJECT
      LINKAGE SECTION.
*****
*
*           L I N K A G E   S E C T I O N
*
*****
*-----*
*   THE FOLLOWING COPYBOOK P49000D DEFINES DATA GROUP 1
*   TRANSACTION CONTROL BLOCK (TCB)
*
*   THE APPLICATION CONTROL BLOCK IS CONTAINED WITHIN THE TCB.
*   THE ACB IS USED FOR CDMF/PCD PROCESSING.
*   COPYBOOK U48004D CONTAINS THE CDMF ACTION CODES
*   COPYBOOK U48024D CONTAINS THE CDMF RESULT CODES
*   ACTIVITY ID 48000
*
*   THE TCB MUST BE THE FIRST DATA GROUP IN THE LINKAGE SECTION.
*   THE TCB DOES NOT NEED TO BE LISTED IN THE UMB PROGRAM DEF.
*   ALL OTHER DATA GROUPS MUST BE LISTED IN THE UMB PROGRAM DEF.
*   IN THE SAME ORDER AS THEY ARE LISTED IN THE LINKAGE SECTION
*   AND IN THE USING STATEMENT OF THIS APPLICATION PROGRAM.
*
*-----*
000100**** START OF P49000D ***** TCB *****
000200* THIS COPYBOOK NOW DEFINES FOUR DISTINCT DATA GROUPS:
000300* 1) D.G. 00001, THE USER TRANSACTION CONTROL BLOCK AND THE CDMF
000400*     APPLICATION CONTROL BLOCK;
000500* 2) D.G. 00012, THE SYSIN INPUT DATA GROUP;

```



Umbrella Programming

Problem Specifications—SORT Access

```
000600* 3) D.G. 00013, THE SYSPRINT OUPUT DATA GROUP; AND
000700* 4) D.G. 00010, THE SECURITY CONTROL BLOCK, THE TCB USER AREA,
000800*     THE USER TCB EXTENSION AREA, AND THE INTERNAL TCB EXTENSION
000900*     AREA (ALC PROGRAMS ONLY).
001000*****SKIP1*****
001100     SKIP1
001200*****
001300* DATA GROUP 00001 (USER TRANSACTION CONTROL BLOCK)
001400*****
001500*
001600 01 TRANSACTION-CONTROL-BLOCK.
001700     05 FILLER             PIC XXXX.
001800     05 TCB-TRANS-NO      PIC XXXX.
001900     05 TCB-CO-ID        PIC XX.
002000     05 TCB-APPL-ID      PIC XX.
002100     05 TCB-FUNC-ID      PIC XX.
002200     05 TCB-SOURCE-TYPE.
002300         10 TCB-SOURCE-TYPE-N    PIC S9(4) COMP.
002400         88 TCB-ONLINE          VALUE +3 +5 +6.
002500         88 TCB-BTCH            VALUE +4.
002600         88 TCB-APPL-SOURCE    VALUE +2.
002700         88 TCB-AUTHORIZATIONS  VALUE +5 +6.
002800         88 TCB-MANNED-TELLER   VALUE +5.
002900         88 TCB-UNMANNED-TELLER  VALUE +6.
003000     05 TCB-ACTIVITY.
003100         10 TCB-ACTIVITY-N    PIC S9(04) COMP.
003200     05 TCB-RESULT          PIC XX.
003300     05 TCB-USER-DATA.
003400         10 FILLER            PIC X(6).
003500         10 TCB-DATA-GROUP    PIC XX.
003600         10 TCB-PARM-POS.
003700             15 TCB-PARM-POS-N  PIC S9(4) COMP.
003800     05 FILLER              REDEFINES TCB-USER-DATA.
003900         10 TCB-USER-INFO    PIC XXXX.
004000         10 TCB-USER-COND    PIC XX.
004100         10 FILLER            PIC X(4).
004200     05 FILLER              REDEFINES TCB-USER-DATA.
004300         10 TCB-USER-CC      PIC XX.
004400         10 TCB-USER-RESULT   PIC XX.
004500         10 TCB-USER-ENVMT   PIC XX.
004600         10 FILLER            PIC X(4).
004700     05 FILLER              REDEFINES TCB-USER-DATA.
004800         10 TCB-EOJ-CALL    PIC XXXX.
004900         10 FILLER            REDEFINES TCB-EOJ-CALL.
005000             15 TCB-SOT-CALL  PIC XXXX.
005100         10 FILLER            PIC X(6).
005200     05 TCB-TIME            PIC S9(7) COMP-3.
005300     05 TCB-SYS-DATE       PIC S9(7) COMP-3.
005400     05 TCB-SOURCE          PIC X(8).
005500     05 TCB-OPERATOR.
005600         10 TCB-UMBRELLA-OPERATOR  PIC X(8).
005700         10 TCB-OPERATOR-FILLER   PIC X(12).
005800         10 TCB-DXRF            REDEFINES TCB-OPERATOR-FILLER.
005900             15 TCB-DXRF-ID    PIC X(4).
006000             15 FILLER          PIC X(8).
006100         10 TCB-DYN-KEY-RANGE  REDEFINES TCB-OPERATOR-FILLER.
006200             15 TCB-DKR-ID    PIC X(4).
006300             15 FILLER          PIC X(8).
006400         10 TCB-ENQ              REDEFINES TCB-OPERATOR-FILLER.
006500             15 TCB-ENQ-ID    PIC X(8).
006600             15 FILLER          PIC X(4).
006700     05 TCB-DESTINATION    PIC X(8).
006800     05 TCB-TERM-DATA.
006900         10 TCB-PFKKEY        PIC X.
007000             88 TCB-ENTER      VALUE QUOTE.
```



Umbrella Programming

Problem Specifications—SORT Access

```

007100      88 TCB-PF01          VALUE '1'.
007200      88 TCB-PF02          VALUE '2'.
007300      88 TCB-PF03          VALUE '3'.
007400      88 TCB-PF04          VALUE '4'.
007500      88 TCB-PF05          VALUE '5'.
007600      88 TCB-PF06          VALUE '6'.
007700      88 TCB-PF07          VALUE '7'.
007800      88 TCB-PF08          VALUE '8'.
007900      88 TCB-PF09          VALUE '9'.
008000      88 TCB-PF10          VALUE ':'.
008100      88 TCB-PF11          VALUE '#'.
008200      88 TCB-PF12          VALUE '@'.
008300      88 TCB-PF13          VALUE 'A'.
008400      88 TCB-PF14          VALUE 'B'.
008500      88 TCB-PF15          VALUE 'C'.
008600      88 TCB-PF16          VALUE 'D'.
008700      88 TCB-PF17          VALUE 'E'.
008800      88 TCB-PF18          VALUE 'F'.
008900      88 TCB-PF19          VALUE 'G'.
009000      88 TCB-PF20          VALUE 'H'.
009100      88 TCB-PF21          VALUE 'I'.
009200      88 TCB-PF22          VALUE '¢'.
009300      88 TCB-PF23          VALUE '.'.
009400      88 TCB-PF24          VALUE '<'.
009500      88 TCB-PFKEY-NOT-PRESENT  VALUE LOW-VALUE.
009600      10 FILLER           PIC XX.
009700      05 TCB-GENP-LOG       PIC X(1).
009800      88 TCB-GENP-NO-LOGGING  VALUE 'N'.
009900      88 TCB-GENP-LOGGING    VALUE 'Y' ''.
010000          LOW-VALUE.
010100      05 TCB-EFFECTIVE-DATE   PIC S9(7) COMP-3.
010200      05 TCB-DEVICE-TYPE-2.
010300      10 TCB-DEVICE-TYPE     PIC X.
010400          88 TCB-3270-2          VALUE 'A'.
010500          88 TCB-BATCH           VALUE 'B'.
010600          88 TCB-3270-1           VALUE 'C'.
010700          88 TCB-3604-DS1          VALUE 'D'.
010800          88 TCB-3604-DS3           VALUE 'E'.
010900          88 TCB-3604-DS4           VALUE 'F'.
011000          88 TCB-3600-JP            VALUE 'G'.
011100          88 TCB-3600-PB            VALUE 'H'.
011200          88 TCB-3600-LP            VALUE 'I'.
011300          88 TCB-3270-MOD1-PRINTER  VALUE 'J'.
011400          88 TCB-3270-MOD2-PRINTER  VALUE 'K'.
011500          88 TCB-TWX              VALUE 'L'.
011600          88 TCB-2470-MOD2          VALUE 'M'.
011700          88 TCB-2740-MOD1          VALUE 'N'.
011800          88 TCB-ALIEN-X           VALUE 'X'.
011900          88 TCB-ALIEN-Y           VALUE 'Y'.
012000          88 TCB-ALIEN-Z           VALUE 'Z'.
012100*
012200***** RESERVED FOR TCB-DEVICE EXPANSION TO PIC X(2)
012300      10 FILLER           PIC X.
012400      05 TCB-LONG-ACTIVITY-N   PIC S9(09) COMP.
012500      05 TCB-LONG-ACTIVITY     REDEFINES TCB-LONG-ACTIVITY-N.
012600      10 TCB-LONG-ACT-HI        PIC XX.
012700      10 TCB-LONG-ACT-LO        PIC XX.
012800      05 TCB-LONG-DGID-N         PIC S9(09) COMP.
012900      05 TCB-LONG-DGID           REDEFINES TCB-LONG-DGID-N.
013000      10 TCB-LONG-DG-HI          PIC XX.
013100      10 TCB-LONG-DG-LO          PIC XX.
013200      05 TCB-USER-CC-APP        PIC XX.
013300      05 TCB-RESULT-2          PIC XX.
013400      EJECT
013500*****

```



Umbrella Programming

Problem Specifications—SORT Access

```
013600* CDMF APPLICATION CONTROL BLOCK
013700*****
013800    05  CDMF-CONTROL-BLOCK.
013900    10  CDMF-ACTION          PIC XX.
014000    10  CDMF-RESULT         PIC XX.
014100    10  CDMF-KEY-FIELDS.
014200        15  CDMF-FORMAT      PIC XXXX.
014300        15  CDMF-COID        PIC XX.
014400        15  CDMF-EFF-DATE    PIC S9(7)  COMP-3.
014500    10  CDMF-EXP-DATE      PIC S9(7)  COMP-3.
014600    10  CDMF-COID-FOUND    PIC XX.
014700        88  CDMF-DEFAULT-COID-FOUND  VALUE HIGH-VALUES.
014800    10  CDMF-EFF-DATE-FOUND  PIC S9(7)  COMP-3.
014900    10  CDMF-HIGH-USE-FLAG   PIC X.
015000        88  CDMF-HIGH-USE-ITEM  VALUE 'Y'.
015100        88  CDMF-NON-PURGEABLE  VALUE 'P'.
015200*
015300***** ITEM OWNERSHIP IS ALWAYS RETURNED IN CDMF-OWNER-APPLICATION
015400***** OWNERSHIP MAY BE RETRIEVED AND UPDATED FROM DATA GROUP 480
015500***** IF THIS FLAG IS SET TO A 'Y'. OWNERSHIP MAY BE UPDATED FR
015600***** CDMF-OWNER-APPLICATION IF THIS FLAG IS SET TO A 'C'.
015700    10  CDMF-OWNER-APP-FLAG   PIC X.
015800        88  CDMF-OWNER-APP-REQUEST  VALUE 'Y'.
015900        88  CDMF-OWNER-APP-IN-CTL-BLK  VALUE 'C'.
016000    10  CDMF-ITEM-LOCATION    PIC X.
016100        88  CDMF-ITEM-FOUND-IN-TABLE  VALUE 'Y'.
016200    10  FILLER             PIC X.
016300    10  CDMF-CC-NO         PIC X(4).
016400    10  CDMF-LAST-CHANGE-DATA.
016500        15  CDMF-LAST-CHANGE-DATE  PIC S9(7)  COMP-3.
016600        15  CDMF-LAST-CHANGE-TIME  PIC S9(7)  COMP-3.
016700        15  CDMF-LAST-CHANGE-CC-NO  PIC X(4).
016800        15  CDMF-LAST-CHANGE-SOURCE  PIC X(8).
016900        15  CDMF-LAST-CHANGE-OPER   PIC X(8).
017000    10  CDMF-SECONDARY-KEY-ID  PIC X(4).
017100    10  CDMF-SUBSTITUTE-DGID  PIC X(4).
017200*
017300***** THIS FIELD IS FOR INTERNAL UMBRELLA USE ONLY.
017400    10  CDMF-RELEASE-CTL-DG-LEN  PIC XX.
017500*
017600***** THIS FIELD IS FOR INTERNAL UMBRELLA USE ONLY.
017700    10  CDMF-RELEASE-CTL-FLAGS  PIC X.
017800*
017900***** THE ITEM APPLICATION OWNERSHIP IS ALWAYS RETURNED IN THIS
018000***** FIELD. THIS FIELD MAY ONLY BE USED IN AN UPDATE WHEN
018100***** CDMF-OWNER-APP-FLAG IS SET TO A 'C'.
018200    10  CDMF-OWNER-APPLICATION  PIC X(3).
018300    10  FILLER             PIC X(2).
018400***** END OF DATA GROUP 00001 ****
018500    EJECT
018600*****
018700* DATA GROUP 12 (SYSIN INPUT DATA GROUP)
018800*-----
018900* NOTE: PROGRAMS THAT REFERENCE DATA GROUP 12 NEED NOT CODE THE
019000* DATA GROUP ON THE PROGRAM DEFINITION. INSTEAD, YOU MAY REFER
019100* TO THIS AREA DIRECTLY & ISSUE ACTIVITIES THAT REFERENCE DG 12.
019200*****
019300*
019400***** DATA GROUP CHAIN (DO NOT DESTROY)
019500    05  FILLER             PIC X(8).
019600    05  DATA-GROUP-12.
019700    10  DG12-ACTION       PIC XX.
019800    10  DG12-RESULT       PIC XX.
019900    10  DG12-CARD-IMAGE   PIC X(80).
020000***** END OF DATA GROUP 00012 ****
```



Umbrella Programming

Problem Specifications—SORT Access

```

020100      SKIP1
020200*****
020300* DATA GROUP 13 (SYSPRINT OUTPUT DATA GROUP)
020400*-----
020500* NOTE: PROGRAMS THAT REFERENCE DATA GROUP 13 NEED NOT CODE THE
020600* DATA GROUP ON THE PROGRAM DEFINITION. INSTEAD, YOU MAY REFER
020700* TO THIS AREA DIRECTLY & ISSUE ACTIVITIES THAT REFERENCE DG 13.
020800*****
020900*
021000***** DATA GROUP CHAIN (DO NOT DESTROY)
021100      05 FILLER          PIC X(8).
021200      05 DATA-GROUP-13.
021300      10 DG13-ACTION    PIC XX.
021400      10 DG13-RESULT    PIC XX.
021500      10 DG13-CONTROL-CHAR PIC X.
021600      10 DG13-PRINT-DATA PIC X(132).
021700      10 FILLER          PIC X(7).
021800***** END OF DATA GROUP 00013 *****
021900      SKIP1
022000*****
022100* DATA GROUP 10 (TCB EXTENSION AREAS)
022200*-----
022300* THIS DATA GROUP CONTAINS THE SECURITY CONTROL BLOCK, THE TCB
022400* USER AREA (FOR CLIENT USE), THE USER TCB EXTENSION AREA, AND
022500* THE INTERNAL TCB EXTENSION AREA.
022600* ***** NOTE: DO NOT CODE THIS DATA GROUP ON YOUR PROGRAM
022700* ***** DEFINITION. INSTEAD, YOU SHOULD REFER TO THIS
022800* ***** AREA DIRECTLY SINCE IT IS PART OF THE USER
022900* ***** TRANSACTION CONTROL BLOCK.
023000*****
023100*
023200***** DATA GROUP CHAIN (DO NOT DESTROY)
023300      05 FILLER          PIC X(8).
023400      05 DATA-GROUP-10.
023500      EJECT
023600*****
023700* SECURITY CONTROL BLOCK
023800*****
023900      10 SECURITY-CONTROL-BLOCK.
024000      15 SCB-ACTION        PIC S9(4) COMP.
024100      15 SCB-ACTION-X REDEFINES SCB-ACTION
024200                  PIC XX.
024300      15 SCB-RESULT        PIC S9(4) COMP.
024400      88 SCB-AUTHORIZATION-VALID VALUE +0.
024500      88 SCB-AUTHORIZATION-FAILED VALUE +4.
024600      88 SCB-AUTHORIZATION-ERROR VALUE +8.
024700      88 SCB-EXT-SECURITY-INACTIVE VALUE +12.
024800      15 SCB-RESULT-X REDEFINES SCB-RESULT
024900                  PIC XX.
025000      15 SCB-VIOLATION-ACTION PIC S9(4) COMP.
025100      88 SCB-ABEND-TASK   VALUE +0.
025200      88 SCB-RETURN       VALUE +1.
025300      15 SCB-LOGGING-FLAG PIC X.
025400      88 SCB-LOG-EXCPTNS VALUE 'Y'.
025500      88 SCB-BYPASS-LOG  VALUE 'N'.
025600      15 SCB-PROCESSING-TYPE PIC X.
025700      15 SCB-PEM-FLAG1   PIC X.
025800      15 SCB-FUTURE-FLAGS PIC X(3).
025900      15 SCB-FORMAT-NUMBER PIC XXXX.
026000      15 SCB-FORMAT-NAME  PIC X(10).
026100*
026200***** THIS FIELD IS FOR ALC PROGRAMS ONLY
026300      15 SCB-ADDR-FMT-TARGET-DG  PIC XXXX.
026400*
026500      15 SCB-TARGET-DG-ID   PIC XXXX.

```



Umbrella Programming

Problem Specifications—SORT Access

```
026600      15 SCB-ITEM-OWNER      PIC X(3).
026700      15 SCB-PREV-OWNER     PIC X(3).
026800      15 SCB-MESSAGE-NO.
026900      20 SCB-MESSAGE-NO-N   PIC S9(4)  COMP.
027000      15 SCB-EXCEPTION-MESSAGE.
027100      20 SCB-RULE-NAME      PIC X(40).
027200      20 FILLER            PIC X(4).
027300      15 SCB-USER-DATA      PIC X(25).
027400      15 SCB-RESERVED       PIC X(8).
027500      15 FILLER            PIC X(1).
027600      EJECT
027700*****
027800* TCB USER AREA
027900*-----
028000* THIS AREA IS RESERVED FOR CLIENTS AND WILL NEVER BE USED BY
028100* HOGAN SYSTEMS.
028200*****
028300      10 TCB-USER-AREA.
028400      15 FILLER            PIC X(104).
028500      SKIP1
028600*****
028700* USER TCB EXTENSION AREA
028800*-----
028900* THIS AREA IS RESERVED FOR NEW TCB FIELDS TO BE ADDED AND
029000* UPDATED BY HOGAN SYSTEMS.
029100*****
029200      10 TCB-EXTENSION-AREA.
029300*      CURSOR POSITION AFTER A DEBLOCK; ROW AND COLUMN
029400      15 TCB-ROW             PIC S9(4)  COMP.
029500      15 TCB-COLUMN          PIC S9(4)  COMP.
029600      15 TCB-BATCH-DISP-OPTION PIC X.
029700      88 TCB-BATCH-DISP-DUMP  VALUE LOW-VALUES.
029800*      BATCH DISPLAY TO SYSPRINT IN DUMP FORMAT.
029900      88 TCB-BATCH-DISP-FORMAT VALUE 'F'.
030000*      BATCH DISPLAY TO SYSPRINT IN SCREEN FORMAT
030100      88 TCB-BATCH-DISP-RETURN VALUE 'R'.
030200*      BATCH DISPLAY DATA IN DG 47. NOT PRINTED.
030300      15 FILLER            PIC X.
030400      15 TCB-DYN-TXN-ID     PIC X(008).
030500      15 TCB-OPTIONS-1      PIC X.
030600      15 TCB-OPTIONS-2      PIC X.
030700      15 FILLER            PIC X(040).
030800      15 TCB-SQL-ACTION.
030900      20 TCB-SQL-ACTION-N   PIC S9(4)  COMP.
031000      15 TCB-SQL-RESULT.
031100      20 TCB-SQL-RESULT-N   PIC S9(4)  COMP.
031200      15 TCB-SQL-DYNPLAN    PIC X(008).
031300      15 TCB-SQL-CURPLAN    PIC X(008).
031400      15 TCB-SQL-DYN-SUBSID  PIC X(004).
031500      15 TCB-SQL-SUBSID     PIC X(004).
031600      15 TCB-CKPT-COUNT     PIC S9(9)  COMP.
031700      15 FILLER            PIC X(005).
031800      15 TCB-APPC-SERVICE-AREAS.
031900      20 TCB-APPC-DATA-GROUP  PIC X(4).
032000      20 TCB-APPC-SYSTEM-KEY  PIC X(4).
032100      20 TCB-APPC-APPL-KEY    PIC X(8).
032200      20 TCB-APPC-XMIT-IMMED  PIC X(1).
032300      88 TRANSMIT           VALUE 'Y'.
032400      88 TRAN-PREPARE-RECEIVE VALUE 'R'.
032500      20 TCB-APPC-XMIT-ERROR  PIC X(1).
032600*      **88 ISSUE-ERROR      VALUE +1.
032700*      **88 ISSUE-ERROR-W-DATA  VALUE +17.
032800*      **88 ISSUE-ABEND       VALUE +2.
032900      20 TCB-APPC-MORE-DATA   PIC X(1).
033000      20 TCB-APPC-BYPSS-ERR   PIC X(1).
```



Umbrella Programming

Problem Specifications—SORT Access

```

033100          20 TCB-APPC-RETURN-CDE  PIC X(6).
033200          15 FILLER             PIC X.
033300          15 TCB-VSAM-RELATIVE-NUMB PIC X(4).
033400          15 TCB-MONETARY-KEY    PIC X(3).
033401          15 TCB-PRES-CURRENCY-CD PIC X(3).
033500          15 TCB-LANGUAGE-KEY   PIC X(3).
033600          15 TCB-PACK-COLLECT-NAME PIC X(18).
033700          15 TCB-DYN-COLLECT-NAME.
033800          20 TCB-DYN-PLAN-PREF  PIC X(2).
033900          20 TCB-DYN-COMP-GRP   PIC X(8).
034000          20 TCB-DYN-PROC-GRP   PIC X(4).
034100          20 TCB-DYN-KEY-RANGE  PIC X(4).
034200          15 FILLER             PIC X(8).
034300          15 TCB-DB2-KRID      PIC X(4).
034400          15 TCB-LANG-ENABLED   PIC X(1).
034500          15 TCB-DEFAULT-LANG  PIC X(3).
034600          15 TCB-LANG-ENCODE-FLAG PIC X(1).
034700          15 TCB-PROC-GROUP    PIC X(4).
034800          15 TCB-PROC-GROUP-BRANCH PIC 9(5) COMP-3.
034900          15 TCB-PROCESSING-ID  REDEFINES
035000          TCB-PROC-GROUP-BRANCH PIC 9(5) COMP-3.
035100          15 TCB-UDFL-LANG     PIC X(3).
035200          15 TCB-SPS-IMPLODE-LANG PIC X(3).
035300          15 TCB-OPERATOR-REGION  PIC 9(5) COMP-3.
035400          15 TCB-OPERATOR-BRANCH  PIC 9(5) COMP-3.
035500          15 TCB-DYN-LANG      PIC X(3).
035600          15 TCB-DMAP-ID       PIC X(7).
035700          15 TCB-DB2-PROC-GRP-ID PIC X(4).
035800          15 TCB-DB2-PROC-GRP-BRANCH PIC 9(5) COMP-3.
035900          15 TCB-DB2-PROCESSING-ID REDEFINES
036000          TCB-DB2-PROC-GRP-BRANCH PIC 9(5) COMP-3.
036100          15 TCB-USER-WORK-AREA.
036200          20 TCB-DYNAMIC-DG-ADDR POINTER.
036201          20 TCB-DYNAMIC-DG-ADDRESS  REDEFINES
036202              TCB-DYNAMIC-DG-ADDR POINTER.
036300          20 FILLER             PIC X(8).
036400          15 TCB-HDP-RESERVED   PIC X(10).
036500          15 TCB-DFLT-LANG-ENCODE-BYTE
036600              PIC X(01).
036700          15 TCB-DB2-TEST-POOL-ID PIC X(02).
036800          15 TCB-PRES-CURR-RND-IN PIC X(01).
036801* DO NOT ROUND PRESENTATION CURRENCY - ADDS AN EXTRA DECIMAL DIG
036802          88 TCB-DO-NOT-ROUND-PRES-CURR           VALUE 'N
036803* ROUND PRESENTATION CURRENCY
036804          88 TCB-ROUND-PRES-CURR           VALUE 'Y
036900          15 TCB-RAND-REC-ADDR-NR  PIC S9(9) COMP.
037000          15 TCB-JOB-ID        PIC X(8).
037100          15 TCB-OPERATOR-PROCESSING-ID PIC 9(5) COMP-3.
037200          15 TCB-DFLT-CENTURYWINDOW-CUTOFF PIC S999 COMP-3.
037300          15 TCB-CICS-STARTCODE  PIC XX.
037301          88 TCB-CICS-START-DPL           VALUE 'D '
037302          88 TCB-CICS-START-DPL-SYNC    VALUE 'DS'
037303          88 TCB-CICS-START-TD-TRIGGER  VALUE 'QD'
037304          88 TCB-CICS-START-CMD      VALUE 'S '
037305          88 TCB-CICS-START-CMD-DATA  VALUE 'SD'
037306          88 TCB-CICS-START-FEPI    VALUE 'SZ'
037307          88 TCB-CICS-START-TERMINAL-INPUT  VALUE 'TD'
037308          88 TCB-CICS-START-USER-ATTACH  VALUE 'U '
037309          15 FILLER             PIC X(661).
037400          15 FILLER             PIC X(200).
037500          SKIP1
037600***** END OF DATA GROUP 00010 ****
037700          SKIP1
037800*
037900***** END OF P49000D ****

```



Umbrella Programming

Problem Specifications—SORT Access

```
EJECT
*-----
* THE FOLLOWING COPYBOOK Z47100D DEFINES DATA GROUP 47100
* BASE DATA GROUP EMP DATA BASE
* EMPLOYEE INFORMATION
* EMP-KEY-GROUP
*-----
000010***** START OF Z47100D ***** EMP KEY *****
000020*
000030* COPYBOOK Z47100D DEFINES DATA GROUP 47100, WHICH IS A KEY
000040* DATA GROUP IN AN EMPLOYEE INFORMATION RECORD ON THE "EMP"
000050* DATA BASE FOR EDUCATION CLASS USE.
000060*
000070*****
000080*
000090* DATA GROUP NUMBER 47100
000100*
000110*****
000120 01 EMP-KEY-GROUP.
000130      10 EMP-ACTION          PIC XX.
000140      10 EMP-RESULT          PIC XX.
000150      05 EMP-KEY-GROUP-MOVE.
000160      10 EMP-CO-ID           PIC XX.
000170      10 EMP-KEY-ID           PIC 9(11) COMP-3.
000180      10 EMP-FILLER          PIC X(38).
000190***** END OF Z47100D *****
EJECT
*-----
* THE FOLLOWING COPYBOOK Z47110D DEFINES DATA GROUP 47110
* POSITIONAL DATA GROUP EMP DATA BASE
* EMPLOYEE GENERAL INFORMATION
* EMP-INFO-GROUP
*-----
000010***** START OF Z47110D ***** EMP INFORMATION *****
000020*
000030* COPYBOOK Z47110D DEFINES DATA GROUP 47110, WHICH IS A
000040* POSITIONAL DATA GROUP USED FOR "EMP" DATA BASE RECORDS
000050* FOR EDUCATION CLASSES.
000060*
000070*****
000080*
000090* DATA GROUP NUMBER 47110
000100*
000110*****
000120 01 EMP-INFO-GROUP.
000130      10 EMP-INFO-ACTION    PIC XX.
000140      10 EMP-INFO-RESULT    PIC XX.
000150      05 EMP-INFO-GROUP-MOVE.
000160      10 EMP-L-NAME         PIC X(15).
000170      10 EMP-F-NAME         PIC X(15).
000180      10 EMP-ADDRESS         PIC X(30).
000190      10 EMP-CITY          PIC X(20).
000200      10 EMP-STATE          PIC X(2).
000210      10 EMP-ZIP            PIC 9(9) COMP-3.
000220      10 EMP-RESERVED        PIC X(5).
000230      10 EMP-EOC-CODE       PIC XX.
000240      10 EMP-SEX             PIC X.
000250      10 EMP-BIRTHDATE       PIC 9(7) COMP-3.
000260      10 EMP-EXEMP           PIC 999 COMP-3.
000270      10 EMP-INSUR-EXEMP     PIC 999 COMP-3.
000280      10 EMP-PHONE           PIC 9(11) COMP-3.
000290      10 EMP-INFO-FILLER     PIC X(87).
000300***** END OF Z47110D *****
EJECT
```



Umbrella Programming

Problem Specifications—SORT Access

```

* THE FOLLOWING COPYBOOK Z47120 DEFINES DATA GROUP 47120
* POSITIONAL DATA GROUP EMP DATA BASE
* EMPLOYEE JOB STATUS INFORMATION
* EMP-JOB-STATUS
*-----
000010***** START OF Z47120D ***** EMP JOB STATUS *****
000020*
000030* COPYBOOK Z47120 DEFINES DATA GROUP 47120, WHICH IS A
000040* POSITIONAL DATA GROUP USED FOR "EMP" DATA BASE RECORDS
000050* FOR EDUCATION CLASSES.
000060*
000070*****
000080*
000090* DATA GROUP NUMBER 47120
000100*
000110*****
000120 01 EMP-JOB-STATUS.
000130      10 JOB-STAT-ACTION          PIC XX.
000140      10 JOB-STAT-RESULT         PIC XX.
000150      05 EMP-JOB-STATUS-MOVE.
000160      10 JOB-STAT-DATE-HIRED    PIC 9(7) COMP-3.
000170      10 JOB-STAT-DATE-TERMD   PIC 9(7) COMP-3.
000180      10 JOB-STAT-MGR           PIC X(30).
000190      10 JOB-STAT-DEPT          PIC XXX.
000200      10 JOB-STAT-COST-CENTER  PIC XXX.
000210      10 JOB-STAT-CLASS-POS.    PIC XX.
000211      15 JOB-STAT-CLASS        PIC XX.
000212      15 JOB-STAT-POSITION     PIC XX.
000230      10 JOB-STAT-FILLER       PIC X(98).
000240***** END OF Z47120D *****
EJECT
*-----
* THE FOLLOWING COPYBOOK Z47130 DEFINES DATA GROUP 47130
* POSITIONAL DATA GROUP EMP DATA BASE
* EMPLOYEE CURRENT PAY INFORMATION
* EMP-CURRENT-PAY
*-----
000010***** START OF Z47130D ***** EMP CURRENT PAY *****
000020*
000030* COPYBOOK Z47130D DEFINES DATA GROUP 47130, WHICH IS A
000040* POSITIONAL DATA GROUP USED FOR "EMP" DATA BASE RECORDS
000050* FOR EDUCATION CLASSES.
000060*
000070*****
000080*
000090* DATA GROUP NUMBER 47130
000100*
000110*****
000120 01 EMP-CURRENT-PAY.
000130      10 EMP-C-ACTION          PIC XX.
000140      10 EMP-C-RESULT          PIC XX.
000150      05 EMP-CURRENT-PAY-MOVE.
000160      10 EMP-C-BEG-DATE        PIC 9(7) COMP-3.
000170      10 EMP-C-END-DATE        PIC 9(7) COMP-3.
000180      10 EMP-C-EARN-TOT        PIC S9(7)V99 COMP-3.
000190      10 EMP-C-FED-INC         PIC S9(7)V99 COMP-3.
000200      10 EMP-C-FICA            PIC S9(7)V99 COMP-3.
000210      10 EMP-C-OTHER-TAX        PIC S9(7)V99 COMP-3.
000220      10 EMP-C-DED-ADJ-TOT      PIC S9(7)V99 COMP-3.
000230      10 EMP-C-NET-PAY         PIC S9(7)V99 COMP-3.
000240      10 EMP-C-DED-ADJ OCCURS 10 TIMES.
000250      15 EMP-C-DED-ADJ-CODE    PIC XX.
000260      15 EMP-C-DED-ADJ-AMT      PIC S9(7)V99 COMP-3.
000270      10 EMP-C-EARNINGS OCCURS 5 TIMES.
000280      15 EMP-C-TYPE-CODE       PIC XX.

```



Umbrella Programming

Problem Specifications—SORT Access

```
000290           15 EMP-C-UNITS          PIC S9(5) COMP-3.
000300           15 EMP-C-AMTS          PIC S9(7)V99 COMP-3.
000310           10 EMP-C-FILLER        PIC X(138).
000320***** END OF Z47130D *****
EJECT
*-----
*   THE FOLLOWING COPYBOOK Z47140 DEFINES DATA GROUP 47140
*   POSITIONAL DATA GROUP EMP DATA BASE
*   EMPLOYEE YEAR TO DATE PAY INFORMATION
*   EMP-YEAR-TO-DATE-PAY
*-----
000010***** START OF Z47140D ***** EMP YEAR-TO-DATE PAY *****
000020*
000030*   COPYBOOK Z47140D DEFINES DATA GROUP 47140, WHICH IS A
000040*   POSITIONAL DATA GROUP USED FOR "EMP" DATA BASE RECORDS
000050*   FOR EDUCATION CLASSES.
000060*
000070*****
000080*
000090*   DATA GROUP NUMBER    47140
000100*
000110*****
000120 01  EMP-YEAR-TO-DATE-PAY.
000130           10 EMP-Y-ACTION          PIC XX.
000140           10 EMP-Y-RESULT          PIC XX.
000150           05 EMP-YEAR-TO-DATE-PAY-MOVE.
000160           10 EMP-Y-BEG-DATE         PIC 9(7) COMP-3.
000170           10 EMP-Y-END-DATE         PIC 9(7) COMP-3.
000180           10 EMP-Y-EARN-TOT        PIC S9(7)V99 COMP-3.
000190           10 EMP-Y-FED-INC         PIC S9(7)V99 COMP-3.
000200           10 EMP-Y-FICA           PIC S9(7)V99 COMP-3.
000210           10 EMP-Y-OTHER-TAX        PIC S9(7)V99 COMP-3.
000220           10 EMP-Y-DED-ADJ-TOT        PIC S9(7)V99 COMP-3.
000230           10 EMP-Y-NET-PAY         PIC S9(7)V99 COMP-3.
000240           10 EMP-Y-DED-ADJ OCCURS 10 TIMES.
000250           15 EMP-Y-DED-ADJ-CODE       PIC XX.
000260           15 EMP-Y-DED-ADJ-AMT        PIC S9(7)V99 COMP-3.
000270           10 EMP-Y-EARNINGS OCCURS 5 TIMES.
000280           15 EMP-Y-TYPE-CODE        PIC XX.
000290           15 EMP-Y-UNITS          PIC 9(5) COMP-3.
000300           15 EMP-Y-AMTS          PIC S9(7)V99 COMP-3.
000310           10 EMP-Y-FILLER         PIC X(138).
000320***** END OF Z47140D *****
EJECT
*-----
*   THE FOLLOWING COPYBOOK T58001D DEFINES DATA GROUP 2000
*   DATE CONTROL BLOCK.
*
*   THE DATE CONTROL BLOCK IS USED FOR DATE SERVICES
*   COMMUNICATION. ACTIVITY ID 1900
*
*   COPYBOOK T58007D CONTAINS DCB ACTION FIELDS
*   COPYBOOK T58008D CONTAINS DCB RESULT FIELDS
*-----
000100*--- START OF T58001D *--- DATE CONTROL BLOCK *-----
000200*
000300*   DATA GROUP - 2000
000400*   LINK ACTIVITY - 1900
000500*
000600*****
000700*
000800 01  DATE-CONTROL-BLOCK.
000900           05 DCB-ACTION          PIC XX.
001000           05 DCB-RESULT          PIC XX.
001100           05 DCB-DATE-IN         PIC S9(7) COMP-3.
```



Umbrella Programming

Problem Specifications—SORT Access

```

001200      05 DCB-DATE-OUT          PIC S9(7) COMP-3.
001300      05 DCB-DIFF-FACTOR    PIC X(1).
001400          88 DCB-DIFF-DAYS          VALUE 'D'.
001500          88 DCB-DIFF-MONTHS        VALUE 'M'.
001600          88 DCB-DIFF-YEARS        VALUE 'Y'.
001700      05 DCB-DIFF-AMT          PIC S9(7) COMP-3.
001800      05 DCB-EDIT-CHAR        PIC X(1).
001900      05 DCB-RAW-DATE-AREA    PIC X(25).
002000*
002100      05 DCB-DISPLAY         REDEFINES DCB-RAW-DATE-AREA.
002200          10 DCB-NUM-WC.
002300          15 DCB-NUM-C          PIC X.
002400          15 DCB-NUM-WOC.
002500          20 FILLER          PIC X(8).
002600          15 FILLER          PIC X(16).
002700*
002800      05 DCB-HOLIDAYS OCCURS 15 TIMES.
002900          10 DCB-HOLIDAY        PIC S9(7) COMP-3.
003000      05 DCB-DAY-OF-WEEK    PIC 9(1).
003100          88 DCB-DOW-SUN          VALUE 1.
003200          88 DCB-DOW-MON          VALUE 2.
003300          88 DCB-DOW-TUE          VALUE 3.
003400          88 DCB-DOW-WED          VALUE 4.
003500          88 DCB-DOW-THR          VALUE 5.
003600          88 DCB-DOW-FRI          VALUE 6.
003700          88 DCB-DOW-SAT          VALUE 7.
003800      05 DCB-DATE-STATUS.
003900          10 DCB-FOM          PIC X(1).
004000          88 DCB-FOM-BUS         VALUE 'B' 'T'.
004100          88 DCB-FOM-CLNDR        VALUE 'P' 'T'.
004200          10 DCB-FOQ          PIC X(1).
004300          88 DCB-FOQ-BUS         VALUE 'B' 'T'.
004400          88 DCB-FOQ-CLNDR        VALUE 'P' 'T'.
004500          10 DCB-FOY          PIC X(1).
004600          88 DCB-FOY-BUS         VALUE 'B' 'T'.
004700          88 DCB-FOY-CLNDR        VALUE 'P' 'T'.
004800          10 DCB-EOM          PIC X(1).
004900          88 DCB-EOM-BUS         VALUE 'B' 'T'.
005000          88 DCB-EOM-CLNDR        VALUE 'P' 'T'.
005100          10 DCB-EOQ          PIC X(1).
005200          88 DCB-EOQ-BUS         VALUE 'B' 'T'.
005300          88 DCB-EOQ-CLNDR        VALUE 'P' 'T'.
005400          10 DCB-EOY          PIC X(1).
005500          88 DCB-EOY-BUS         VALUE 'B' 'T'.
005600          88 DCB-EOY-CLNDR        VALUE 'P' 'T'.
005700          10 DCB-BUS-DAY        PIC X(1).
005800          88 DCB-BDAY          VALUE 'B'.
005900          10 DCB-WEEK-DAY       PIC X(1).
006000          88 DCB-WDAY          VALUE 'W'.
006100          10 DCB-LEAP-YEAR      PIC X(1).
006200          88 DCB-LYEAR          VALUE 'L'.
006300          10 DCB-FOW           PIC X.
006400          88 DCB-FOW-BUS         VALUE 'B' 'T'.
006500          88 DCB-FOW-CLNDR        VALUE 'P' 'T'.
006600          10 DCB-EOW           PIC X.
006700          88 DCB-EOW-BUS         VALUE 'B' 'T'.
006800          88 DCB-EOW-CLNDR        VALUE 'P' 'T'.
006900      05 DCB-CURR-BUSINESS-DAY  PIC 99.
007000      05 DCB-FISCAL-STATUS.
007100          10 DCB-FISCAL-FOQ      PIC X(1).
007200          88 DCB-FISCAL-FOO-BUS    VALUE 'B' 'T'.
007300          88 DCB-FISCAL-FOQ-CLNDR  VALUE 'P' 'T'.
007400          10 DCB-FISCAL-FOY      PIC X(1).
007500          88 DCB-FISCAL-FOY-BUS    VALUE 'B' 'T'.
007600          88 DCB-FISCAL-FOY-CLNDR  VALUE 'P' 'T'.

```



Umbrella Programming

Problem Specifications—SORT Access

```
007700      10 DCB-FISCAL-EOQ  PIC X(1).  
007800          88 DCB-FISCAL-EOQ-BUS           VALUE 'B' 'T'.  
007900          88 DCB-FISCAL-EOQ-CLNDR         VALUE 'P' 'T'.  
008000      10 DCB-FISCAL-EOY  PIC X(1).  
008100          88 DCB-FISCAL-EOY-BUS           VALUE 'B' 'T'.  
008200          88 DCB-FISCAL-EOY-CLNDR         VALUE 'P' 'T'.  
008300*  
008400      05 DCB-SQL-FIELDS.  
008500          10 DCB-SQL-DATE-IN    PIC X(10).  
008600          10 DCB-SQL-DATE-OUT   PIC X(10).  
008700          10 DCB-SQL-TIME     PIC X(08).  
008800          10 DCB-PACK-TIME   PIC S9(07)  COMP-3.  
008900          10 DCB-SQL-OVRD-DATE PIC X(03).  
          88 DCB-SQL-EUR-DATE           VALUE 'EUR'.  
009100          88 DCB-SQL-ISO-DATE           VALUE 'ISO'.  
009200          88 DCB-SQL-JIS-DATE           VALUE 'JIS'.  
009300          88 DCB-SQL-USA-DATE           VALUE 'USA'.  
009400          10 DCB-SQL-OVRD-TIME  PIC X(03).  
009500          88 DCB-SQL-EUR-TIME           VALUE 'EUR'.  
009600          88 DCB-SQL-ISO-TIME           VALUE 'ISO'.  
009700          88 DCB-SQL-JIS-TIME           VALUE 'JIS'.  
009800          88 DCB-SQL-USA-TIME           VALUE 'USA'.  
009900*  
010000      05 DCB-BRANCH-ID-X.  
010100          10 DCB-BRANCH-ID      PIC 9(05)  COMP-3.  
010200*  
010300 01 FILLER REDEFINES DATE-CONTROL-BLOCK.  
010400      05 DCB-DATA          PIC X(17).  
010500*  
010600*-----* END OF T58001D *-----  
          EJECT  
*-----  
* THE FOLLOWING COPYBOOK I57101D DEFINES DATA GROUP 1452  
* PCD CONTROL BLOCK  
*  
* THE PCD CONTROL BLOCK IS USED FOR PCD PROCESSING  
* ACTIVITIES 1013 AND 1014  
*  
* COPYBOOK I57104D CONTAINS PCD ACTION FIELDS  
* COPYBOOK I57105D CONTAINS PCD RESULT FIELDS  
*-----  
000100*****  
000200*          I57101D - PCD CONTROL BLOCK  
000300*****  
000400*  
000500*          PEM - DG #1452, ALLOCATE ACTIVITY #1530  
000600*****  
000700 01 PCD-CONTROL-BLOCK.  
000800      05 PCD-ACTION          PIC XX.  
000900      05 PCD-RESULT          PIC XX.  
001000      05 PCD-INPUT-SECTION.  
001100          10 PCD-COID          PIC XX.  
001200          10 PCD-ID            PIC XX.  
001300          10 PCD-USER-KEY        PIC X(11).  
001400          10 PCD-EFFECT-DATE    PIC S9(7)  COMP-3.  
001500          10 PCD-EXPIRE-DATE    PIC S9(7)  COMP-3.  
001600          10 FILLER            PIC X.  
001700      05 PCD-OUTPUT-SECTION.  
001800          10 PCD-OUT-COID        PIC XX.  
001900          10 PCD-OUT-ID          PIC XX.  
002000          10 PCD-OUT-USER-KEY    PIC X(11).  
002100          10 PCD-OUT-EFFECT-DATE PIC S9(7)  COMP-3.  
002200          10 PCD-OUT-EXPIRE-DATE PIC S9(7)  COMP-3.  
002300          10 FILLER            PIC X.  
002400      05 PCD-EFFECT-PERIOD.
```



Umbrella Programming

Problem Specifications—SORT Access

```

002500      10 PCD-FIRST-EFFECT      PIC S9(7) COMP-3.
002600      10 PCD-LAST-EFFECT      PIC S9(7) COMP-3.
002700      05 PCD-SWTCH          PIC XX.
002800      05 PCD-AMSG           PIC X(8).
002900      05 PCD-RMSG           PIC X(17).
003000      05 PCD-CCNO           PIC S9(8) COMP.
003100      05 FILLER            PIC X.
003200      05 PCD-COID-USED      PIC XX.
003300      05 PCD-USER-FIELD-1    PIC X(40).
003400      05 PCD-OWNER          PIC X(03).
003500      05 FILLER            PIC X(01).
003600*
003700*****EJECT*****
*-----*
*   THE FOLLOWING COPYBOOK Z47190D DEFINES DATA GROUP 47190
*   TARGET DATA GROUP FOR PCD 47190
*   THIS IS A CONVERTED PCD
*   CONTAINS POSITION CODE DESCRIPTIONS WITHIN JOB CLASS.
*-----*
*****START OF Z47190D - POSITION CODE DESCRIPTIONS
*   DATA GROUP NUMBER 47190
*   USER KEY = JOB CLASS (TQE00000 FROM DATA GROUP 47120)
*   RECURSIVE KEY = POSITION CODE
*****END OF Z47190D
01  EMP-POSITION-CODE-DESCRIPTION.
    05  EMP-POS-ACT-TQE00000  PIC XX.
    05  EMP-POS-RES-TQE00000  PIC XX.
    05  EMP-POS-KEY-TQE00000  PIC XX.
    05  EMP-POSITION-CODE      REDEFINES EMP-POS-KEY-TQE00000
                                PIC XX.
        05  EMP-POS-DESCRIPTION  PIC X(40).
*
*   END OF Z47190D
EJECT
*-----*
*   THE FOLLOWING COPYBOOK Z47192D DEFINES DATA GROUP 47192
*   TARGET DATA GROUP FOR PCD 47192
*   THIS IS A CDMF/PCD
*   CONTAINS POSITION CODE DESCRIPTIONS WITHIN JOB CLASS.
*-----*
Z 4 7 1 9 2 D
*
*   THIS DG DEFINES THE POSITION DESCRIPTION RECORD WHICH IS
*   THE TARGET DATA GROUP FOR CDMF FORMAT 47192 USED IN THE
*   APPLICATIONS PROGRAMMING CLASS.  THE KEY IS COMPRISED OF
*   THE JOB CLASS AND THE POSITION CODE (BOTH FOUND IN
*   DG 47120 OF THE EMP DATA BASE).  THE DESCRIPTION IS USED
*   IN THE REPORT GENERATED BY PROGRAM 478XX.
*   *** DGID = 47192 ***
*****END OF Z47192D
01  POSITION-DESCRIPTION-RECORD.
    05  POSITION-CODE-ACTION      PIC XX.
    05  POSITION-CODE-RESULT      PIC XX.
    05  POSITION-JOB-CLASS       PIC XX.
    05  POSITION-CODE           PIC XX.
        05  POSITION-DESCRIPTION  PIC X(40).
*
*   END OF Z47192D
EJECT
*****END OF Z47192D
*-----*
*   P R O C E D U R E      D I V I S I O N

```



Umbrella Programming

Problem Specifications—SORT Access



Umbrella Programming

Problem Specifications—SORT Access

```

DA999-EXIT.
  EXIT.
  EJECT
*
PC000-READ-CDMF-FORMAT.
*
PC099-EXIT.
  EXIT.
  EJECT
*
CC000-LINK-TO-CCP.
*
CC999-EXIT.
  EXIT.
  SKIP3
*
PR000-PRINT-REPORT.
*
MOVE CC-GROUP-ID          TO EMP-EEOC-CODE.
MOVE SPS-LK-ACTIVITY-47922 TO TCB-LONG-ACTIVITY.
PERFORM CA000-CALL-PEM.

PR999-EXIT.
  EXIT.
  EJECT
*
SS000-READ-INPUT-SDB.
*
SS099-EXIT.
  EXIT.
*
*           END OF CLASS CODING SECTION
*
  EJECT
*
XX000-HOUSE-KEEPING SECTION.
*
IF TCB-BTCH
  MOVE SPS-LK-ACTIVITY-47921 TO TCB-LONG-ACTIVITY
  PERFORM CA000-CALL-PEM.

XX999-EXIT.
  EXIT.
  SKIP3
*
YY000-ERROR-PROCESSING SECTION.
*
MOVE CA-LONG-PEM-TRANS-DUMP-END TO TCB-LONG-ACTIVITY.
PERFORM CA000-CALL-PEM.

YY999-EXIT.
  EXIT.
  SKIP3
*
ZZ000-END-OF-PROCESSING SECTION.
*
```



Umbrella Programming

Problem Specifications—SORT Access

```
MOVE CA-LONG-PEM-END-PROG TO TCB-LONG-ACTIVITY.  
PERFORM CA000-CALL-PEM.
```

```
STOP RUN.
```

```
/* **** */
```

Notes:



Batch Execute JCL - ZUPCXXEZ

```
MODULE NAME ZUPCXXEZ

//ZUP{J}EZ JOB (HOGN,{B},BEF), 'EXEC SORT II',MSGCLASS=9,
//                      TIME=(00,04),REGION=4M,NOTIFY=&SYSUID
///*
//P$$LIB  JCLLIB ORDER=( {TL}.PROCLIB)
//*
//JS010    EXEC HOGNBPEM,
//          TEMPLIB=' {L}.TESTLIB',
//          VSAM=' {V}'
//*-----*
///*  EXEC YOUR PROGRAM IN BATCH           *
//*-----*
//EMP      DD DSN=&VSAM..EMP,DISP=SHR,DCB=BUFNO=10
//EMP\\OUT  DD DSN=&&EMP\\SDB,DISP=(NEW,PASS,DELETE),
//          UNIT=SYSDA,SPACE=(TRK,(1,1),RLSE),
//          DCB=(RECFM=FB,LRECL=96,BLKSIZE=0)
//SYSIN    DD *                                <===== PLACE YOUR TRANSACTIONS INFO HERE
//*
//JS020    EXEC HOGNBPEM,
//          TEMPLIB=' {L}.TESTLIB',
//          VSAM=' {V}'
//*-----*
///*  EXEC YOUR PROGRAM IN BATCH AND SORT YOUR SEQUENTIAL DATABASE   *
//*-----*
//SORTWK01  DD UNIT=SYSDA,SPACE=(CYL,5)
//SORTWK02  DD UNIT=SYSDA,SPACE=(CYL,5)
//SORTWK03  DD UNIT=SYSDA,SPACE=(CYL,5)
//EMP\\IN   DD DSN=&&EMP\\SDB,DISP=(OLD,DELETE)
//SYSIN    DD *                                <===== PLACE YOUR TRANSACTIONS INFO HERE
//*****
```

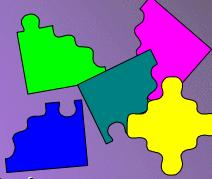
Notes:



Summary

CSC

Summary



- Unit of work under Hogan is an activity
- EXIT types of the Sort activity are
 - IN(PUT)
 - OUT(PUT)
 - BOTH
- IN(PUT) and OUT(PUT) EXIT types require SDB to be specified

© 2000 CSC. All rights reserved

7/5/00

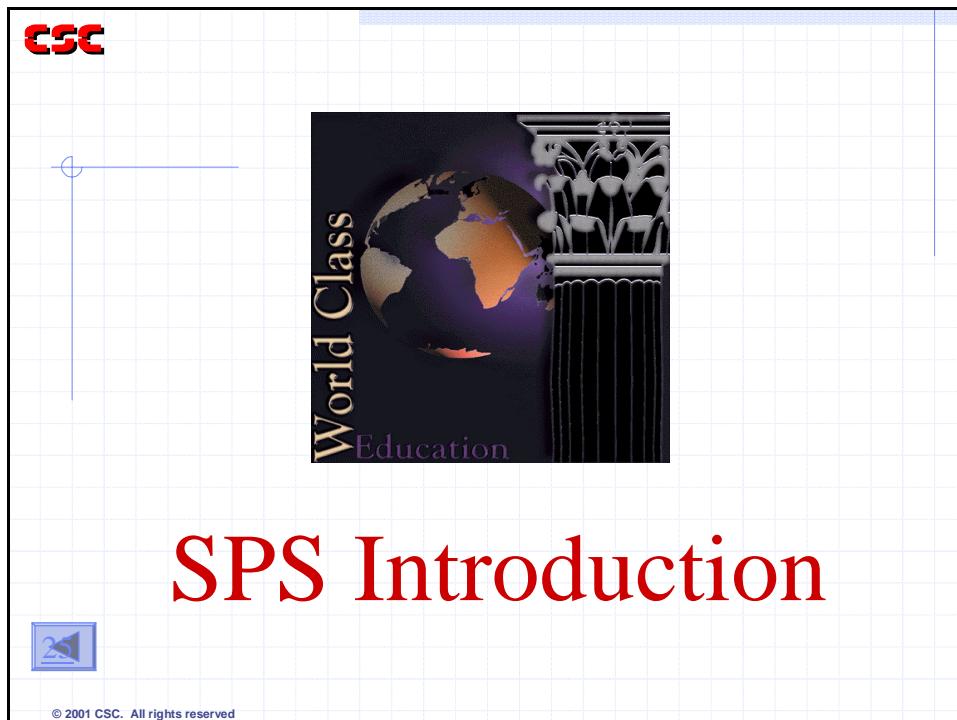
Notes:



Scheduled Processing System - SPS

26

Purpose



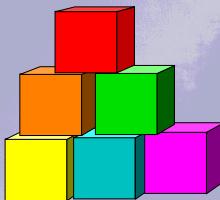
Notes:



Topics



Topics



- SPS Screens
- Line Definitions (SPS Programs)
- Format Definitions (SPS Subroutines)
- Maintenance of SPS Programs
- SPS Features and Processing Flow
- SPS Debugging Aids

© 2000 CSC. All rights reserved

Notes:



Objectives

CSC

Objectives



- Identify design objects of SPS
- Name four processing phases involved in report production
- Use SPS screens
- Identify function performed by a Line Definition
- Name components of key to a Line Definition

© 2000 CSC. All rights reserved

CSC

Objectives



- List four Line Definition commands
- Identify function performed by a Format Definition
- Name components of key to a Format Definition
- List seven Format Definition commands
- Describe several SPS Debugging Aids

© 2000 CSC. All rights reserved



Scheduled Processing System

The Scheduled Processing System (SPS) is one of the Umbrella Application Development Support Systems. As a low-level language, SPS enables the programmer to code logic online, and using the GEN command, create an executable module, ready to use either in the online or batch environment. The structure of the SPS language consists of two types of processing parameters, or instructions, Line Definitions used to define Mainline Logic and Format Definitions used to define subroutines. Data is presented to SPS from many different sources. Both batch and online application programs can link to SPS based on a reportable incident. This data is then processed by SPS according to user-defined parameters to sort and format the data. SPS is a utility language, designed to utilize the features and constructs of the Umbrella System. Its most frequent use is the capture and subsequent printing of reportable data.

The processing parameters are stored on PCDs so that they can be defined and maintained external to the user programs. They allow the user to specify rules for many program/report processing tasks, including:

- Identifying elements for sort keys
- Formatting data
- Defining criteria for special processing due to changes in data (break processing)
- Clearing, rolling and accumulating totals
- Defining page overflow and page heading criteria

SPS Execution

An application program or batch job stream can invoke the SPS subsystem by issuing a link to program ID 1850. The exact form of processing performed by the subsystem is determined by the value of the link activity and the Line Definition ID. For example:

- If the link activity has a value greater than 1255, then SPS executes either the implosion phase or a unit program. The Report ID takes the same value as the ID of the activity that linked to SPS.
- If the link activity is 1250, then SPS performs the explosion or output phase.
- A link activity of 1251 will invoke HEADING processing
- A link activity of 1255 invoke BREAK processing.

This class will focus on SPS as a data reporter.



SPS Report Production Phases

The creation of an SPS report involves four phases. They are:

1. Implosion
2. Sort
3. Explosion
4. Output

The purpose of the implosion phase is to build a data string record based on a reportable incident from an online or batch application. Data string records are written to a sequential file for later batch processing. Data string records, data group 3601, contain three areas:

- The Hogan Key contains the data string ID, the report id and several other fields.
- The User Key contains the data field(s) needed to control the sort sequence of the report.
- The User Data area contains the data fields to be printed on the report. Any field from the data string may be formatted to the print line.

The remaining three phases of sort, explosion, and report production are executed in a batch job or job steps.

During the sort phase, the data strings records are sorted. The user can supply a sort activity, or default to the delivered sort activity 16400.

During the explosion phase, the data groups are rebuilt. The data fields loaded into the data string record during the implosion phase are moved into the application data group(s) so that the data fields can be accessed for printing.

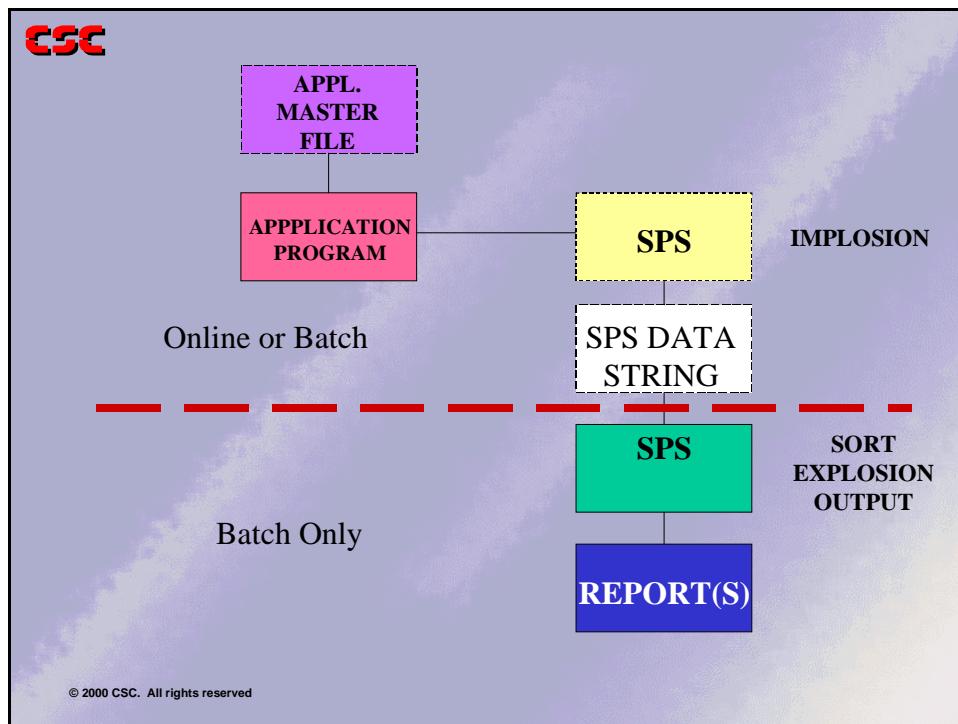
During the output phase, the data fields are moved or edited into the print area, data group 1031, and printed. Multiple print lines may be printed from one data string record. Simple arithmetic and logical operations can be performed. SPS allows activities to be issued from its process. If a code was formatted into the data string and the report is to print the description of the code, a CDMF or PCD read activity could be coded. Break and new page logic are executed where necessary.

On the next page you will see a diagram of how SPS report production interfaces with application processing.



Umbrella Programming

Scheduled Processing System

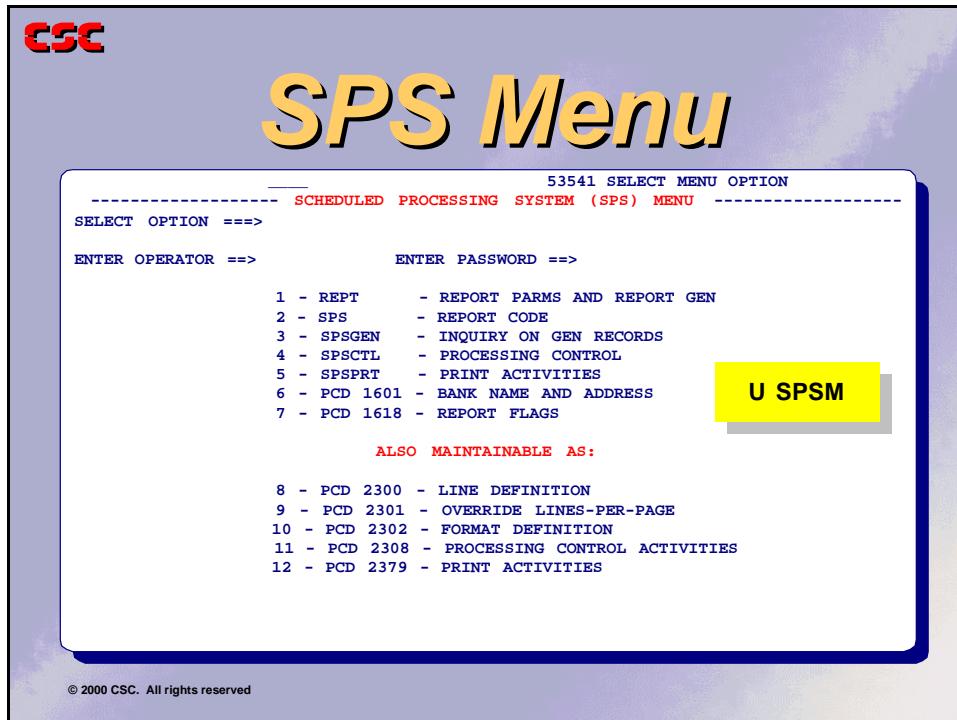


Notes:



SPS Menu Screen

SPS is a subsystem of the Umbrella that generates reports in a batch environment. Processing parameters that define a report are stored on CDMF as SPS programs, called Line Definitions, and SPS subroutines, called Format Definitions. SPS has an online language that is composed of Line Definition commands and Format Definition commands. The parameters needed to control the line and element processing of a report can be accessed and maintained online. From a cleared screen, enter U SPSM and press ENTER to display the "Scheduled Processing System (SPS) Menu".



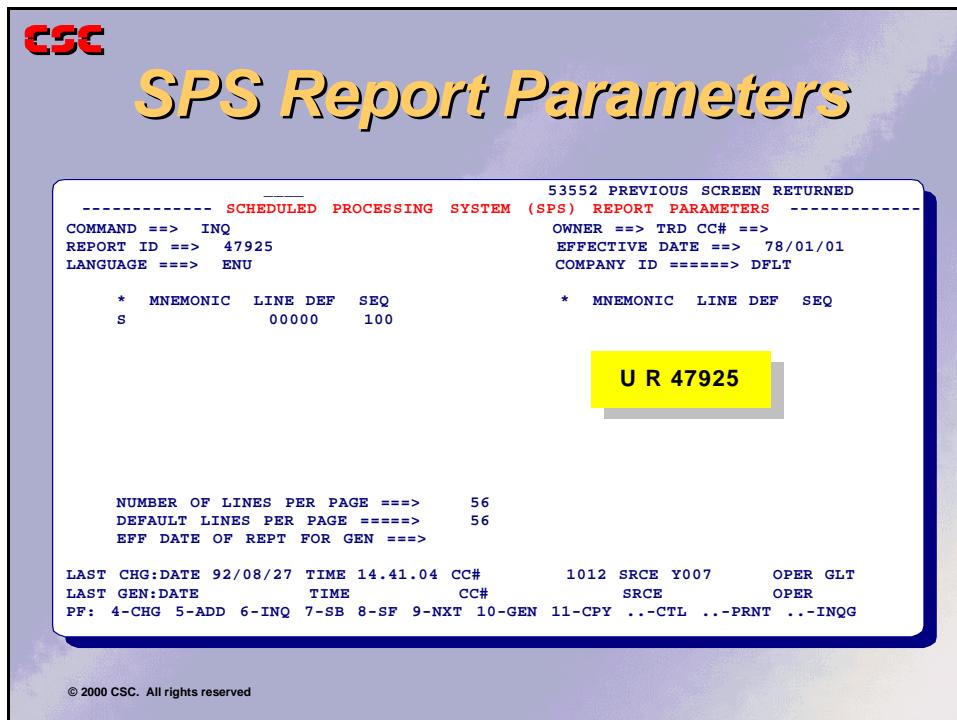
Notes:



Umbrella Programming

SPS Menu Screen

Entering Option 1 from the menu displays the "Scheduled Processing System (SPS) Report Parameters" screen.



This screen functions in a similar way to a PDS Directory containing multiple programs. Each line definition represents one program in the Directory. From this screen you can inquire on a program by entering S below the asterisk.

Notes:



CSC

SPS Report (Program) Code

SCHEDULED PROCESSING SYSTEM					ACTION SUCCESSFUL
					(SPS) REPORT CODE
COMMAND ==> INQ					OWNER ==> TRD CC# ==>
LANG RPT ID MNEMONIC LINE DEF SEQ					EFFECTIVE DATE ==> 78/01/01
ENU 47925 00000 100					COMPANY ID =====> DFLT
* I	VALUE	SEQ	INSTRUCTION		COMMENT
P	1101				SET PRINT CARRIAGE CONTROL
A	1279				PEM ACTIVITY
S	100				SHARED FORMAT DEFINITION
		100	MOVE 'GROUP' TO -034/005		
		200	MOVE TQE47409 TO -040		EMP-Y-DED-ADJ-CODE
		300	MOVE 'EMPLOYEE LISTING FROM DB' TO -050/024		SHARED FORMAT DEFINITION
		S 200			

LAST CHG:DATE 86/03/27 TIME 14.41.04 CC# 1012 SRCE Y007 OPER MTD
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL

© 2000 CSC. All rights reserved

This screen combines the line and format definition commands for the program and displays the start of one Line Definition for Report ID 47925. Line Definitions are keyed by Language Code, Report ID, Report Mnemonic, Line Definition, and Sequence Number. This particular Line Definition has a Language Code of ENU, a Report ID of 47925, a Mnemonic of blanks, a Line Definition of 00000, and a Sequence Number of 100. There could be other Line Definitions for this program; each would perform an independent function and would be executed independently of any other Line Definition.

The Line Definition commands are placed on the far left of the screen under the I (for instruction). A numeric value accompanies most commands. On the above screen, notice that there are three different types of command--P, A, and S. The comments on the far right indicate the function of each command. As you can see, a command of P sets the print carriage control and a command of A causes an activity to be issued.

A command of S executes a shared format definition. Shared Format Definitions contain element processing commands. They can be shared. Like Line Definitions, they are keyed by Language Code, Report ID and Mnemonic. They are also keyed by format definition. The first Shared Format Definition, S 100, has three statements in it. Each statement has its own sequence number. The comments for the statements within a format definition identify the element being manipulated or the operation being performed.

The entire program is displayed using multiple screens. To view the rest of the program definition, press PF8. The screen will scroll forward. PF7 scrolls backward.



Umbrella Programming

SPS Menu Screen

When you reach the end of the current Line Definition, the definition will wrap around to the beginning again.

Notes:



Line Definitions (Programs)



Line Definitions/SPS Programs

- One function
- Several within SPS program
- Determines
 - Program logic flow
 - Vertical spacing
- Calling programs
 - Invoke subroutines

© 2000 CSC. All rights reserved

Line Definition is the term used in SPS processing to represent logical executable entities. They are analogous to programs in traditional data processing. Line Definitions generally are written to execute one particular function, such as printing a line or moving data from an SPS Data Group to an Application Data Group.

Usually several Line Definitions are required to generate an SPS Report.

The instructions allowed in Line Definitions do little more than control report flow and vertical spacing. One command does invoke routines to manipulate data. Line Definitions, thus, are calling programs to invoke Format Definitions to manipulate data on the reports. Format definitions are analogous to performed subroutines

Notes:



Umbrella Programming

Line Definition Commands

Line Definition Commands

Line Definition processing is primarily used to control the lines of a report. In a real sense, it is a command processor. There are four primary Line Definition Commands used in this class. They are:



Line Definition Commands

- A..... Issue a PEM activity
- P.....Set printer carriage control byte
- S..... Execute a shared format definition
for the default company
- E..... End execution of a line definition

OPERAND VALUE

© 2000 CSC. All rights reserved

The four commands A, P, S, and E are discussed in this section. Additional commands are discussed as they become relevant within the course.

The general format of a Line Definition Command is:

INSTRUCTION VALUE

Where the INSTRUCTION is a letter representing the command and the VALUE is a numeric identifier associated with the command. They are coded in the Report Code Screen under the column heading “I” for instruction and value.

Notes:



A—Issue a PEM Activity

The A Command allows the user to issue PEM Process Dictionary Activities. The number of the activity to be issued is placed in the value field next to the command.

ScreenText: A 1279

Issue PEM Activity 1279. Activity 1279 is used to print an SPS Report output line.

Note: An SPS program has the authority to issue ANY activity on the Process Dictionary.

Notes:



Umbrella Programming

Line Definition Commands

P—Set Print Carriage Control

This command is used to set the printer carriage control.

P 0ABCC

VALUE SYNTAX: 0ABCC

Where: 0=ZERO

A=0 - DO AFTER

1 - DO IMMEDIATELY

B=0 - SPACE

1 - SKIP

C=XX - NUMBER TO SPACE (1-3) OR CHANNEL TO SKIP TO (1-12)

EXAMPLES: Skip to channel 1 immediately - P 01101

Space 1 after printing - P 00001

Notes:



S—Execute Shared Format Definition

The S Line Definition Command is used to execute a Shared Format Definition. (Subroutine) Shared Format Definitions contain Shared Format Definition Commands. Shared Format Definitions are also called Format Definitions. Shared Format Commands are discussed later in the class.

ScreenText: S 15010

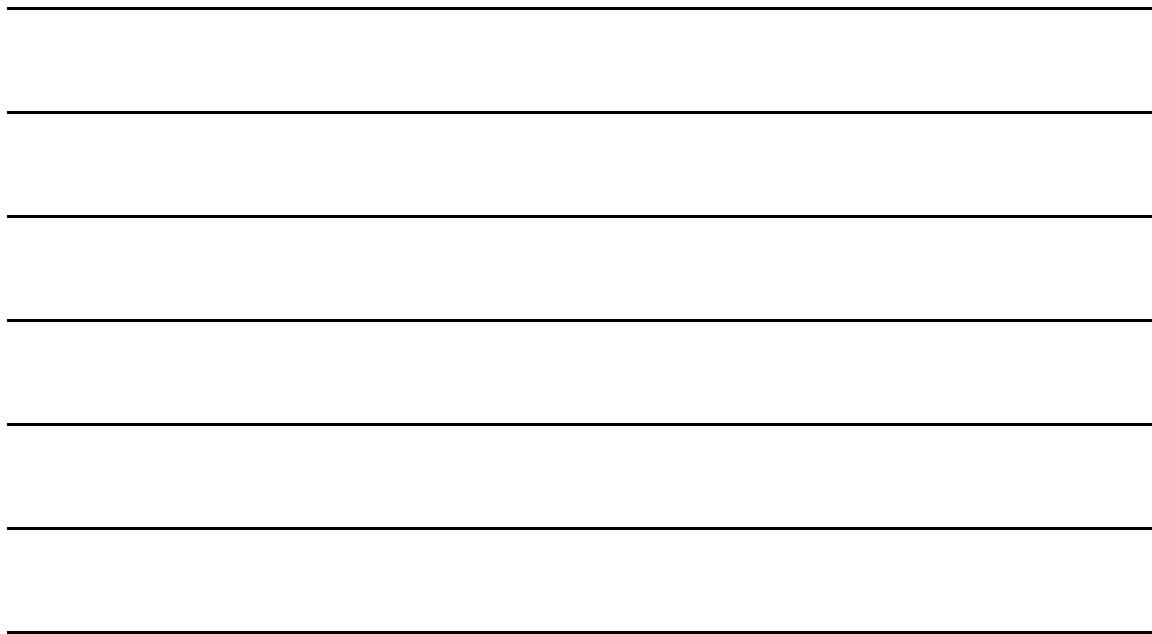
This Line Definition Command is used for tasks performed within a report. These tasks may be repeated for a given report. The command is similar to a perform of a subroutine in a COBOL program.

The S 15010 above represents the invocation of the Shared Format Definition identified by the number 15010. Shared Formats are defined within SPS Report ID/Mnemonic combinations.

Multiple invocations of the same Format Definition are possible. Just as a subroutine may be called from many different places in a COBOL program, a Shared Format Definition is invoked as many times as needed from Line Definitions under the Report ID/Mnemonic combination.

The Shared Format Definition Command is for usage only with reports for the default company (DFLT).

Notes:



Umbrella Programming

Line Definition Commands

E—End Line Definition Processing

This command ends execution of a Line Definition. Control is returned to the logic that initially invoked the Line Definition. The command is

ScreenText :

E

Each Line Definition must end with an E command explicitly coded as above.

Notes:



Format Definitions (Subroutines)

CSC

Format Definitions/SPS Subroutines

- **5 digit numbers**
- **Performed repeatedly**
- **Shared among programs within
same report-ID/mnemonic
combination**
- **Single task with few instructions**

© 2000 CSC. All rights reserved

Shared Format Definitions are logical executable entities that are analogous to subroutines in traditional data processing. A Shared Format Definition is also referred to as a Format Definition in this Class Manual and the Umbrella System Support Systems Programming Guide. Format Definitions are identified by 5-digit numbers.

They can be performed repeatedly so common tasks only need to be coded once and can be reused. These subroutines can be used by Line Definitions under the same Report ID/Mnemonic combination. It is recommended that a Format Definition perform a single task and, thus, contain few instructions.

The instructions within Format Definitions are called Format Definition Commands. These commands operate on the Data Element level and are used to manipulate fields and control the horizontal spacing of a report.

There are Format Definition Commands that perform arithmetic functions, move and edit data, compare data elements, and perform GOTO functions.



Umbrella Programming

Shared Format Definition Commands

Shared Format Definition Commands

Shared Format Definition Commands:

ACTIVITY.....	ISSUE PEM ACTIVITY
ADD.....	ADD INPUT ELEMENT TO OUTPUT ELEMENT
CLEAR.....	INITIALIZE TO LOW VALUES BY FIELD TYPE
EDIT.....	NUMERIC DATA TO PRINT LINE WITH EDIT PATTERN
END.....	END EXECUTION OF A LINE DEFINITION
GOTO.....	TRANSFER CONTROL TO ANOTHER COMMAND
IF (COMPARE).....	COMPARE OUTPUT-ELEMENT TO INPUT-ELEMENT
MOVE.....	MOVE INPUT-ELEMENT TO OUTPUT-ELEMENT
PRINT.....	WRITE A LINE
SKIP.....	ESTABLISH CARRIAGE CONTROL CHARACTER
SPACE.....	ESTABLISH CARRIAGE CONTROL CHARACTER
SUBTRACT.....	SUBTRACT INPUT-ELEMENT FROM OUTPUT-ELEMENT

SPS instructions are similar to COBOL instructions, in syntax. The command must be the first word in the instruction. Words must be separated by a space. Extra spaces are ignored.

Notes:



Optional Words

The following words have no meaning to the SPS compiler and may be inserted or left out when coding instructions. SPS will insert them automatically where appropriate.

BY

FROM

GO (IF NOT USED AS A COMMAND)

GOTO (IF NOT USED AS A COMMAND)

INDEXING

INTO

LINECOUNT

ON

PRINTING

THEN

TIMES

TO

USING.

Format Definition Commands

The order of presentation of commands in this manual is determined by function. The commands performing arithmetic functions are grouped together. Commands that set the carriage control character and print are in sequence.

Notes:



Umbrella Programming

Format Definition Commands

CLEAR Instruction

```
CLEAR OUTPUT-ELEMENT (( INDEXING) N TIMES)
OR
CLEAR OUTPUT-ELEMENT (TO 'X')
```

The OUTPUT-ELEMENT is set to zeros for packed and binary fields, character fields are set to spaces. If a value is specified, the entire field is set to the single character value.

Multiple adjacent elements with the same length and data type may be cleared for packed or binary data types. The number of these adjacent elements is specified by **N TIMES**.

ScreenTexts:

1. CLEAR TLE50109
2. CLEAR TLE50102
3. CLEAR TLE50100 07 TIMES
4. CLEAR \$01 INDEXING 20 TIMES
5. CLEAR PRTLINE
6. CLEAR PRTLINE TO '-'

EXAMPLE 1: The packed field TLE50109 is set to zero.

EXAMPLE 2: The binary field TLE50102 is set to zero.

EXAMPLE 3: All Data Elements from TLE50100 through TLE50106 are set equal to zero. These seven fields are defined contiguously.

EXAMPLE 4: The accumulators \$01 through \$20 are set equal to zero. These fields are defined contiguously.

EXAMPLE 5: The print line is set to spaces.

EXAMPLE 6: The print line is set to all dashes.



ADD Instruction

ADD INPUT-ELEMENT TO OUTPUT-ELEMENT (INDEXING N TIMES)

The INPUT-ELEMENT is ADDED to the OUTPUT-ELEMENT. The total is in the OUTPUT-ELEMENT. The INPUT-ELEMENT may be constant or packed field. The OUTPUT-ELEMENT must be packed.

INPUT-ELEMENTS can be a numeric literal or constant identified by a *. The largest value is 9999999.

If the INPUT-ELEMENT is character (not numeric) a condition code is set and processing abends.

The ADD instruction supports adding a low level set of totals to a higher level set of totals where N equals the number of adds to be performed. The accumulators within each level must be the same length. The INPUT-ELEMENTS must be packed to use this type of adding.

ScreenTexts:

1. ADD TRANAMT TO CURRBAL
2. ADD /01 TO /02
3. ADD *1 TO ITEMCNT
4. ADD \$01 TO \$0102 INDEXING 5 TIMES

EXAMPLE 1: The transaction amount field is added to the current balance field.

EXAMPLE 2: The value stored in the /01 accumulator is added to the /02 accumulator. (The SET Command establishes these accumulators)

EXAMPLE 3: 1 is added to ITEMCNT.

EXAMPLE 4: The \$01, \$02, \$03, \$04, and \$05 accumulators are added to the \$0102, \$0202, \$0302, \$0402, AND \$0502 accumulators, respectively. (The SET Command establishes these accumulators.)



Umbrella Programming

Format Definition Commands

SUBTRACT Instruction

```
SUB(TRACT) INPUT-ELEMENT FROM OUTPUT-ELEMENT ( INDEXING N TIMES )
```

The INPUT-ELEMENT is SUBTRACTED from the OUTPUT-ELEMENT. The result is placed in the OUTPUT-ELEMENT. The INPUT-ELEMENT may be character or packed data type. The OUTPUT-ELEMENT must be packed.

If the INPUT-ELEMENT is character (non-numeric) a condition code is set and processing abends.

The SUBTRACT instruction supports subtracting a low level set of totals from a higher level set of totals where N equals the number of subtracts to perform. The accumulators within each level must be the same length. The input elements must be a packed data type in this instruction.

ScreenTexts:

1. SUBTRACT TRANAMT FROM CURRBAL
2. SUBTRACT /01 FROM /02
3. SUBTRACT *1 FROM ITEMCNT
4. SUBTRACT \$01 FROM \$0102 INDEXING 5 TIMES

EXAMPLE 1: The value stored in the field TRANAMT is SUBTRACTED from the value stored in the field CURRBAL. The DIFFERENCE is stores in the field CURRBAL.

EXAMPLE 2: The value stored in the whole number accumulator /01 is SUBTRACTED from the value in the whole number accumulator /02.

EXAMPLE 3: 1 is SUBTRACTED from the field ITEMCNT.

EXAMPLE 4: The \$01, \$02, \$03, \$04, and \$05 accumulators are SUBTRACTED from the \$0102, \$0202, \$0302, \$0402, and \$0502 accumulators, respectively.



MOVE Instruction

```
MOVE INPUT-ELEMENT TO OUTPUT-ELEMENT
OR
MOVE 'XXXXXXXX' TO OUTPUT-ELEMENT
```

The INPUT-ELEMENT is moved to the OUTPUT-ELEMENT. Data conversion is performed if necessary. If the OUTPUT-ELEMENT is packed or binary and the INPUT-ELEMENT is character, but not numeric, a condition code is set and processing abends.

The maximum length of a constant is 30 characters excluding the beginning and ending quotes.

A MOVE totally replaces the contents of a field.

ScreenTexts:

1. MOVE TCE50017 TO TLE50107
2. MOVE 'PAGE' TO -125/004
3. MOVE TQE47303 TO -015/010

EXAMPLE 1: The value stored in the field TCE50017 is moved to field TLE50107.

EXAMPLE 2: The constant value PAGE is moved to position 125 of data group 1031 (the print line).

EXAMPLE 3: Illustrates the move of a numeric field to the print line or data string. The packed field is unpacked and 10 positions are moved to position 15. There is no edit pattern or sign.

If SPS must determine the length of a move, it bases the length on the sending field. If this length is not obtainable, it takes the length of the receiving field.

Notes:



Umbrella Programming

Format Definition Commands

Data String and Print Line Position Notation

The SPS PRINT LINE is formatted by moving fields to positions within DATA GROUP 1031. DATA STRINGS are created by moves to positions within DATA GROUP 3601. The position is indicated by:

-NNN/LLL

Where NNN is the starting position and LLL is the length. The position notation syntax consists of four parts:

- [.] NNN [.] / [.] [.] LLL [.]

All parameters are optional, with the exception of the "-". The LLL parameter is required ONLY for the EDIT Format Definition and data names referencing Matrix variables.

The number range of the Shared Format Definition is used to determine which Data Group is being referenced.

Range 15000 - 16999	Data Group 3601
All other ranges	Data Group 1031

ScreenTexts:

1. MOVE TCE50017 TO -
2. MOVE TQE47204 TO -015
3. MOVE 'PAGE NUMBER:' TO -125/004
4. MOVE TQE47102 TO - /010

EXAMPLE 1: The value stored in the field TCE50017 is moved to the next available unused byte of DG3601 or DG1031, for the defined length of TCE50017.

EXAMPLE 2: The value stored in the field TQE47204 is moved to position 15 of DG3601 or DG1031, for the defined field length of TQE47204.

EXAMPLE 3: The first 4 bytes of the constant value 'PAGE NUMBER' are moved to position 125 of DG3601 or DG1031.

EXAMPLE 4: The first 10 bytes of the value stored in the field TQE47102 are moved to the next available unused byte of DG3601 or DG1031.



EDIT Instruction

EDIT INPUT-ELEMENT (IN)TO OUTPUT-ELEMENT USING EDIT-PATTERN.

The INPUT-ELEMENT is edited through a user-defined or delivered edit pattern and the result is placed into the OUTPUT-ELEMENT.

Editing is always done on packed data fields, so conversion of character or binary data types is performed as necessary. For numeric fields the edit command is usually preferred to the move.

There are two types of edit patterns:

- DEFAULT
- USER-CODED

Default Edit Patterns

There are three default edit patterns delivered.

\$ AN EDIT PATTERN FOR A MONEY ELEMENT.

/ AN EDIT PATTERN FOR A NUMBER ELEMENT.

.NN AN EDIT PATTERN FOR A DATE

The money element edit pattern (\$) characteristics are listed below.

- Z,ZZZ,ZZZ,ZZZ,ZZZ.99-
- GENERATED FOR THE SPECIFIC INPUT-ELEMENT SIZE.
- SIGNIFICANCE FORCED AT THE DECIMAL POINT.
- HAS TWO DECIMAL PLACES.
- HAS A TRAILING DASH IF THE ELEMENT IS NEGATIVE.
- INPUT-ELEMENT PACKED DATA TYPE.

The number element edit pattern (/) characteristics are listed below.

- ZZZ,ZZZ,ZZZ,ZZZ,ZZZ
- GENERATED FOR THE SPECIFIC INPUT ELEMENT SIZE.
- SIGNIFICANCE FORCED BEFORE THE LAST DIGIT.
- HAS NO DECIMAL PLACES.
- NEGATIVE NUMBERS NOT INDICATED.
- INPUT-ELEMENT PACKED DATA TYPE.



Umbrella Programming

Format Definition Commands

The date edit pattern is described below. The NN is used to select the type of date format as follows:

.00 RESULTS THE SITE DEFAULT PATTERN.

.01 RESULTS IN A MM-DD-YY FORMAT.

.02 RESULTS IN A MM/DD/YY FORMAT.

.03 RESULTS IN A MM DD YY FORMAT.

.04 RESULTS IN A MMDDYY FORMAT.

.11 RESULTS IN A CYY-MM-DD FORMAT.

.12 RESULTS IN A CYY/MM/DD FORMAT.

.13 RESULTS IN A CYY MM DD FORMAT.

.14 RESULTS IN A CYYMMDD FORMAT.

.21 RESULTS IN A DD-MM-YY FORMAT.

.22 RESULTS IN A DD/MM/YY FORMAT.

.23 RESULTS IN A DD MM YY FORMAT.

.24 RESULTS IN A DDMMYY FORMAT.

The editing logic edits the standard packed date format of CYYMMDD into the specified format.

Notes:



User-Coded Edit Pattern

The user-coded edit pattern is like a COBOL edit mask where the following characters can be used:

- 9
- Z
- S
- \$
- -

The editing characters are replaced with digits from the INPUT-ELEMENT. Significance is forced after the first 9 or S. The Z and \$ float.

```
ScreenTexts:   ZZZ,ZZZ,ZZ9.99-
                  ZZZ,ZZZ,S99.99-
                  $$$,$$9,999.99-
```

The user may choose the fill character. If the first digit of the edit pattern is not a \$, /, ., 9, or Z, then the character is used as the fill character, such as, *9999 results in the presentation of 123 as **123.

If the user-coded edit pattern has more characters than the INPUT-ELEMENT has digits, the edit is rejected.

If the edit is accepted, the INPUT-ELEMENT is edited into a work area and then moved to the OUTPUT-ELEMENT under the following rules:

- If the edited data is larger than the OUTPUT-ELEMENT, then the OUTPUT-ELEMENT is filled with the right most digits of the edited data.
- If the OUTPUT-ELEMENT is larger than the edited data, then the edited data is moved to the left side of the OUTPUT-ELEMENT padding the right side with spaces.
- If an edit pattern of matching number of 9's to edited data occurs, SPS truncates the high order position if it is not significant.



Umbrella Programming

Format Definition Commands

ScreenTexts:

1. EDIT CURRBAL INTO -029/015 USING \$
2. EDIT ITEMCNT INTO -031/006 USING /
3. EDIT TCB\$DATE INTO -010/008 USING .02
4. EDIT DOL\$AMT INTO -081/010 USING ZZ,ZZ9.99-
5. EDIT DOL\$AMT INTO -081/011 USING \$ \$\$,\$\$9.99-

EXAMPLE 1: The field CURRBAL is edited using the delivered dollar edit pattern. The last 15 digits of the edited data are moved to print position 29.

EXAMPLE 2: The field ITEMCNT is edited using the delivered number edit pattern. The last 6 digits of the edited data are moved to print position 31.

EXAMPLE 3: The IPL DATE stored in TCB\$DATE is edited using one of the delivered date edit patterns and the edited data prints as MM/DD/YY beginning in print position 10.

EXAMPLE 4: A user-coded edit pattern is used to edit the value in DOL\$AMT.

EXAMPLE 5: A user-coded edit pattern is used to edit the value in DOL\$AMT. The \$ floats.

Notes:



SPACE Instruction

SPACE (OUTPUT-ELEMENT) N (TIMES) BEFORE/AFTER

This command is used to set the printer carriage control. N is the number of times to space either before or after printing. The maximum value of N is 3.

Once the SPACE command is executed, it stays in effect for all PRINT commands until the carriage control character is reset.

ScreenTexts:

1. SPACE 2 AFTER
2. SPACE 1 TIMES BEFORE

EXAMPLE 1: The carriage control character is set to double space after printing.

EXAMPLE 2: The printer is spaced one line prior to printing.

SKIP Instruction

SKIP (OUTPUT-ELEMENT) (TO) N BEFORE/AFTER

This instruction sets the printer carriage control for a channel skip. The maximum value of N is 12. Skip either BEFORE or AFTER printing.

Once a SPACE or SKIP command is given, it is operative for all print commands until the carriage control character is changed.

ScreenTexts:

1. SKIP TO 1 BEFORE
2. SKIP TO 12 AFTER

EXAMPLE 1: An immediate skip to channel one occurs on the next print activity.

EXAMPLE 2: At the next request for a print, the printer skips to channel 12 after printing.



Umbrella Programming

Format Definition Commands

PRINT Instruction

PRINT

This instruction generates PEM activity ID 1279. Activity 1279 is a link activity to program 1879, the SPS PRINT CONTROL PROGRAM. The program controls all involvement with the printer. Usually, the link is used to cause a detail line to be printed from data group 1031. The link is also issued along with a skip to channel 1 (top of page).

Notes:



IF (Compare) Instruction

```
IF OUTPUT-ELEMENT XX INPUT-ELEMENT (THEN)
OR
IF OUTPUT-ELEMENT XX INPUT-ELEMENT GOTO #NNNNN
```

WHERE XX IS:

EQ	- EQUAL
NE	- NOT EQUAL
LT	- LESS THAN
NLT	- NOT LESS THAN
GT	- GREATER THAN
NGT	- NOT GREATER THAN

#NNNNN is the Sequence ID of a Format Definition Command within the same Format Definition.

#EXITF can be substituted for #NNNNN. Control is transferred out of the Format Definition to the next Line Definition Command.

With the IF/THEN comparison, the OUTPUT-ELEMENT is compared to the INPUT-ELEMENT. If the comparison condition is true, the Format Definition processing continues with the next instruction. If the comparison condition is false, then the next Line Definition Command is executed.

If a GOTO is specified, the GOTO is processed on a true condition. The sequence ID (#NNNNN) specified must be within the same Format Definition. If the condition is false, processing continues with the next Format Definition Command.

DATA TYPES SUPPORTED ARE:

OUTPUT-ELEMENT	INPUT-ELEMENT
BINARY	BINARY
BINARY	LITERAL
CHARACTER	CHARACTER
CHARACTER	LITERAL/CONSTANT
PACKED	PACKED
PACKED	LITERAL

The OUTPUT-ELEMENT cannot be a constant or numeric literal.

If the OUTPUT-ELEMENT is character, the compare is set for the length of the OUTPUT-ELEMENT.



Umbrella Programming

Format Definition Commands

ScreenTexts:

1. IF TRANAMT GT CURRBAL THEN
2. IF TRANAMT LT CURRBAL GOTO #1400
3. IF TRANAMT NE CURRBAL GOTO #EXITF

EXAMPLE 1:

If the value stored in the field TRANAMT is greater than the value in CURRBAL, then processing continues with the next Format Definition Command. Otherwise, control passes to the next Line Definition Command.

EXAMPLE 2:

If the value stored in TRANAMT is less than the value in CURRBAL, control transfers to sequence number 1400 within the same Format Definition. Otherwise, processing continues with the next Format Definition Command.

EXAMPLE 3:

If the value stored in TRANAMT is not equal to the value in CURRBAL, control transfers to the next Line Definition Command. Otherwise, processing continues with the next Format Definition instruction.

GOTO Instruction

GOTO #NNNNNN

OR

GOTO #EXITF

This instruction transfers processing to another Format Definition Command. The specified sequence ID (#NNNNNN) must be within the same Format Definition.

If #EXITF is specified, control transfers to the next Line Definition Command.

Notes:



END Instruction

END

When the END command is issued by a Format Definition, the Line Definition that invoked the Format Definition is terminated.

Assume Line Definition 10 is invoked and it in turn invokes Format Definitions 100, 200, AND 300. The function of this Line Definition is to process transaction types 1, 2, AND 3. As transactions can only be of one type, processing should terminate when there is a transaction type match.

Extending this example, if the function of Format Definition 100 is to process transaction type 1, there is no need to process the two remaining Format Definitions. SPS code to accomplish this processing could use the END Format Definition command.

ScreenText :

```
LINE DEFINITION 10
    S 100
        IF TRANTYPE EQ *1 THEN
            MOVE
            MOVE
        END
    S 200
        IF TRANTYPE EQ *2 THEN
            MOVE
            MOVE
        END
    S 300
        MOVE
        MOVE
    E
```

If the EQ condition is satisfied, control falls through to the instructions that process the specific transaction type. When an END statement is executed, the Line Definition is immediately terminated.

Notes:



Umbrella Programming

Format Definition Commands

ACTIVITY Instruction

(ACTIVITY) NNNNNNNNNN

Where NNNNNNNNNN IS A PEM ACTIVITY.

This instruction allows the user to issue a PEM Process Dictionary Activity.

ScreenTexts:

1. ACTIVITY 16735
2. 16735
3. ACTIVITY 1900

EXAMPLES 1 AND 2:

The PEM ACTIVITY 16735 is executed during SPS processing. ACTIVITY 16735 writes data group 3601 to the SPS String File.

EXAMPLE 3:

Issue a link activity to Date Services.

Notes:



SPS Program Walk Through

Now that we have discussed some of the commands used by SPS, let's review Report ID 47925.

CSC

SPS Report Walk Through

```

----- SCHEDULED PROCESSING SYSTEM ----- ACTION SUCCESSFUL
COMMAND ==> INQ (SPS) REPORT CODE -----
LANG RPT ID MNEMONIC LINE DEF SEQ OWNER ==> TRD CC# ==>
ENU 47925 00000 100 EFFECTIVE DATE ==> 78/01/01
* I VALUE SEQ ----- INSTRUCTION ----- COMPANY ID =====> DFLT
    P 1101 COMMENT
          SET PRINT CARRIAGE CONTROL

    A 1279 PEM ACTIVITY

    S 100 SHARED FORMAT DEFINITION

    100 MOVE 'GROUP' TO -034/005
    200 MOVE TQE47409 TO -040 EMP-Y-DED-ADJ-CODE
    300 MOVE 'EMPLOYEE LISTING FROM DB' TO -050/024
    S 200 SHARED FORMAT DEFINITION

LAST CHG:DATE 86/03/27 TIME 14.41.04 CC# 1012 SRCE Y007 OPER MTD
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL

```

© 2000 CSC. All rights reserved

After positioning the printer at the top of a page by setting the carriage control to channel 1 and issuing activity 1279, this Line Definition executes format definition 100. Format definition 100 contains three move statements. Sequence number 100 is the move of the literal GROUP to the print line data group (data group 1031). The expression -034/005 indicates that the literal GROUP is to be moved to print position 34 for a length of 5.

Sequence number 200 moves a data element to print position 40. Notice that in this entry, the length of the field is not specified. Data element IDs are defined on the data group definition in columns 73 through 80. These data element IDs are the only way SPS can access a modifiable field. The COBOL copybook name of the element is provided in the comments.



Umbrella Programming

SPS Program Walk Through

To display the data element, enter U ELE TQE47409 from a cleared screen.

The screenshot shows the CSC Element Definition screen. At the top, it says "Element Definition". Below that is a large text area containing the following data:

ACTION COMPLETE	
PROCESS	DICTIONARY ELEMENT DEFINITION INQ/MAINTENANCE
COMMAND ==> INQ <=BYELE	OWNER: TRD CC# ==>
ELEMENT NAME==> TQE47409 :	
COBOL NAME==> EMP-Y-DED-ADJ-CODE	:
ELEMENT ID==> DATA GROUP==> 47140	EFFECTIVE DATE==> 77/01/01
DESCRIPTION==> EMP-Y-DED-ADJ-CODE	:
EDIT PATTERN==>	:
SHORT DESC==> EMP-Y-DED-ADJ-CODE	:
DISPLACEMENT: 42 LENGTH: 2 TYPE==> C	
IS ELE RECURSIVE?	ON AUDIT REPT?==> USED IN INITIALIZATION?====>
SPS :USER EXIT?==>	EXIT ACT==> FILE MAINTENANCE?=====> N
FIELDS: EFS EXIT?==>	EFS TABLE=====> SENSITIVE DATA?=====>
:HIGH VOL.?==>	SPS DESCRIPTION=====> EMP-Y-DED-ADJ-CODE
DOCUMENTATION KEY?==>	
-DOC- COMMAND=>	MODE: FROM LINE=> THRU LINE=> TO LINE=>

In the center of the screen, there is a yellow box containing two bullet points:

- ! or INCR in Command
- = ELE.TQE47409 in Command

At the bottom of the screen, there is some footer text:

LAST CHG:DATE 92/07/20 TIME 11.15.35 CC# SRCE F10F OPER GLT
PF: 3-PLVL 4-CHG 6-INQ 7-DSB 8-DSF 9-NXT 10-DCHG

© 2000 CSC. All rights reserved

From this screen you can see that element TQE47409 is defined in data group 47140. The field itself is a two byte character field. The length of the element defines the length of the move.

Notes:



CSC

Scheduled Processing System

```

----- SCHEDULED PROCESSING SYSTEM ----- ACTION SUCCESSFUL
COMMAND ==> SF (SPS) REPORT CODE -----
LANG RPT ID MNEMONIC LINE DEF SEQ OWNER ==> TRD CC# ==>
ENU 47925 00000 100 EFFECTIVE DATE ==> 78/01/01
                                         COMPANY ID =====> DFLT

* I     VALUE   SEQ ----- INSTRUCTION ----- COMMENT
      50                   TCB-EFFECTIVE-DATE
      EDIT TPE49022 INTO -089/008 USING .01
      100                  MOVE 'PAGE #' TO -111/006
      200                  ADD *1 TO PAGENO BUMP PAGE NUMBER
      300                  EDIT PAGENO INTO -117/007 USING ZZZZZZ
      400                  MOVE '0' TO W04 RE-SET LINE COUNTER
P          3           SET PRINT CARRIAGE CONTROL
A          1279          PEM ACTIVITY

LAST CHG:DATE 86/03/27 TIME 14.41.04 CC# 1012 SRCE Y007 OPER MTD
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL

```

D or DECR in Command PF8

© 2000 CSC. All rights reserved

Format Definition 200 introduces the ADD command in sequence number 200. The *1 indicates a numeric value of 1. This amount is to be added to PAGENO, which is a field provided by SPS to record page numbers. The result of the operation will remain in the PAGENO field.

The EDIT command performs two functions. It moves the data and edits the value according to the edit pattern following the USING clause.

W04 is an SPS work counter that is being used to count lines.

Following the last command for the format definition, the screen continues with Line Definition commands. The next command sets the print carriage control to space three lines after the detail line is printed. Because this command is followed by activity 1279, the first line of the report is printed and the printer is positioned on the fourth line of the report. Issuing activity 1279 not only prints the line, but it also causes the detail line to be cleared. The P command following activity 1279 requests that the report be single spaced after the next line is printed.

Notice that spacing commands work differently under SPS than they do when you are using ASCII carriage control. With ASCII, a request for spacing occurs {before} the line is printed. With SPS, the spacing normally occurs {after} the line is printed. If you instruct SPS to space before the line is printed, the detail line will be cleared before it is printed.



Umbrella Programming

SPS Program Walk Through

The screenshot shows the CSC Scheduled Processing System interface. At the top, it displays "SCHEDULED PROCESSING SYSTEM" and "ACTION SUCCESSFUL". Below this, it shows command details: "COMMAND ==> SF", "OWNER ==> TRD CC# ==> DF01", "EFFECTIVE DATE ==> 78/01/01", and "COMPANY ID =====> DF01". The main area contains SPS code:

```
----- SCHEDULED PROCESSING SYSTEM -----  
----- (SPS) REPORT CODE -----  
COMMAND ==> SF  
OWNER ==> TRD CC# ==>  
EFFECTIVE DATE ==> 78/01/01  
COMPANY ID =====> DF01  
  
----- INSTRUCTION -----  
* I VALUE SEQ COMMENT  
P 1 SET PRINT CARRIAGE CONTROL  
  
S 300 PF8 SHARED FORMAT DEFINITION  
  
100 MOVE 'JOB' TO -055/003  
200 PRINT  
S 400 SHARED FORMAT DEFINITION  
  
100 MOVE 'DEPT' TO -001/004  
150 MOVE 'EMPLOYEE ID' TO -009/011  
  
LAST CHG:DATE 86/03/27 TIME 14.41.04 CC# 1012 SRCE Y007 OPER MTD  
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL  
  
© 2000 CSC. All rights reserved
```

A yellow box highlights the "PF8" key, which is labeled "SHARED FORMAT DEFINITION".

Format Definition 300 contains an example of a print occurring from within a format definition. After this print is executed, SPS will be building the fifth line of the report. On the next page is a copy of what the report will look like through the first five lines. Spend a few moments comparing it with the SPS code that generated it.

Notes:



Umbrella Programming

SPS Program Walk Through

-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6-----+-----7-----+-----8-----+-----9-----+-----0-----+-----1-----+-----2-----+-----3-----
 GROUP XX EMPLOYEE LISTING FROM DB MM-DD-YY PAGE #ZZZZZZ
 DEPT EMPLOYEE ID EMPLOYEE NAME JOB CL/POS EEOC SEX DATE HIRE CUR EARNINGS Y-T-D-EARNINGS

-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6-----+-----7-----+-----8-----+-----9-----+-----0-----+-----1-----+-----2-----+-----3-----

Notes:



SPS Code Batch Print

On the following pages is a listing of SPS code printed in batch. Below is example JCL for printing SPS Code in Batch.

```
//JS010      EXEC HOGNBPEM
//-----*
//*
//*      PRINT THE SPS SOURCE ON CDMF - PCD 2300, 2302, 2308
//*
//*-----*
//SYSIN    DD    *      INTRODUCE CONTROL CARDS TO PRINT SPS PARAMETERS
1 54 21
#1031@@ @47923@47925%
//-----*
```

Notes:



Umbrella Programming
SPS Code Batch Print

```

REPORT=47925      LINE=00000      SEQ=00100      COMPANY=08888      HEADING ID=00000      EFF=0780101
P 00000001101
A 00000001279
S 00000000100

FORMAT=00100      COMPANY=08888      EFF=0780101
SEQ      INSTRUCTION
00100 MOVE 'GROUP' TO -034/005
00200 MOVE TQE47409 TO -040
00300 MOVE 'EMPLOYEE LISTING FROM DB' TO -050/024
                                         COMMENT
                                         EMP-Y-DED-ADJ-CODE

S 0000000200

FORMAT=00200      COMPANY=08888      EFF=0780101
SEQ      INSTRUCTION
00050 EDIT TPE49022 INTO -089/008 USING .01
00100 MOVE 'PAGE #' TO -111/006
00200 ADD *1      TO PAGENO
00300 EDIT PAGENO  INTO -117/007 USING ZZZZZZ9
00400 MOVE '0' TO W04
                                         COMMENT
                                         TCB-EFFECTIVE-DATE
                                         BUMP PAGE NUMBER
                                         RE-SET LINE COUNTER

P 00000000003
A 00000001279
P 00000000001
S 00000000300

FORMAT=00300      COMPANY=08888      EFF=0780101
SEQ      INSTRUCTION
00100 MOVE 'JOB' TO -055/003
00200 PRINT
                                         COMMENT

S 00000000400

FORMAT=00400      COMPANY=08888      EFF=0780101
SEQ      INSTRUCTION
00100 MOVE 'DEPT' TO -001/004
00150 MOVE 'EMPLOYEE ID' TO -009/011
00200 MOVE 'EMPLOYEE NAME' TO -031/013
00300 MOVE 'CL/POS' TO -054/006
00325 MOVE 'EOC' TO -065/004
00350 MOVE 'SEX' TO -077/003
00375 MOVE 'DATE HIRE' TO -088/009
00400 MOVE 'CUR EARNINGS' TO -103/012
00450 MOVE 'Y-T-D-EARNINGS' TO -117/014
                                         COMMENT

P 00000000002
A 00000001279
E 00000000000

REPORT=47923      LINE=00000      SEQ=00100      COMPANY=08888      HEADING ID=00000      EFF=0780101
S 00000000100

FORMAT=00100      COMPANY=08888      EFF=0780101
SEQ      INSTRUCTION
00100 IF W03      EQ  *0      THEN
00200 ACTIVITY 47925
00300 MOVE '1' TO W03
                                         COMMENT
                                         FIRST TIME THRU?
                                         SET FLAG SO DONE ONLY ONCE

S 00000000200

FORMAT=00200      COMPANY=08888      EFF=0780101
SEQ      INSTRUCTION
00100 MOVE TQE47204 TO -002/003
00200 MOVE TQE47002 TO -008/012
00300 MOVE TQE47101 TO -023/015
00400 MOVE TQE47102 TO -038/015
00450 MOVE TQE47206 TO -054/002
00500 MOVE TQE47207 TO -058/002
00600 MOVE TQE47108 TO -066
                                         COMMENT
                                         JOB-STAT-DEPT
                                         EMP-KEY-ID
                                         EMP-L-NAME
                                         EMP-F-NAME
                                         JOB-STAT-CLASS
                                         JOB-STAT-POSITION
                                         EMP-EOC-CODE

```



Umbrella Programming

SPS Code Batch Print

```
00640 MOVE TQE47109 TO -078          EMP-SEX
00670 EDIT TQE47201 INTO -089/008 USING .01   JOB-STAT-POSITION
00700 EDIT TQE47303 INTO -102/012 USING $    EMP-C-EARN-TOT
00800 EDIT TQE47403 INTO -118/012 USING $    EMP-Y-EARN-TOT

P 0000000001

A 0000001279

S 0000000300

FORMAT=00300 COMPANY=08888 EFF=0780101
SEQ INSTRUCTION
00100 ADD *1      TO W04           COMMENT
00200 IF W04      EQ  *45        SIMPLE LINE COUNTER
00300 ACTIVITY 47925      THEN
                                HAVE WE A FULL PAGE?

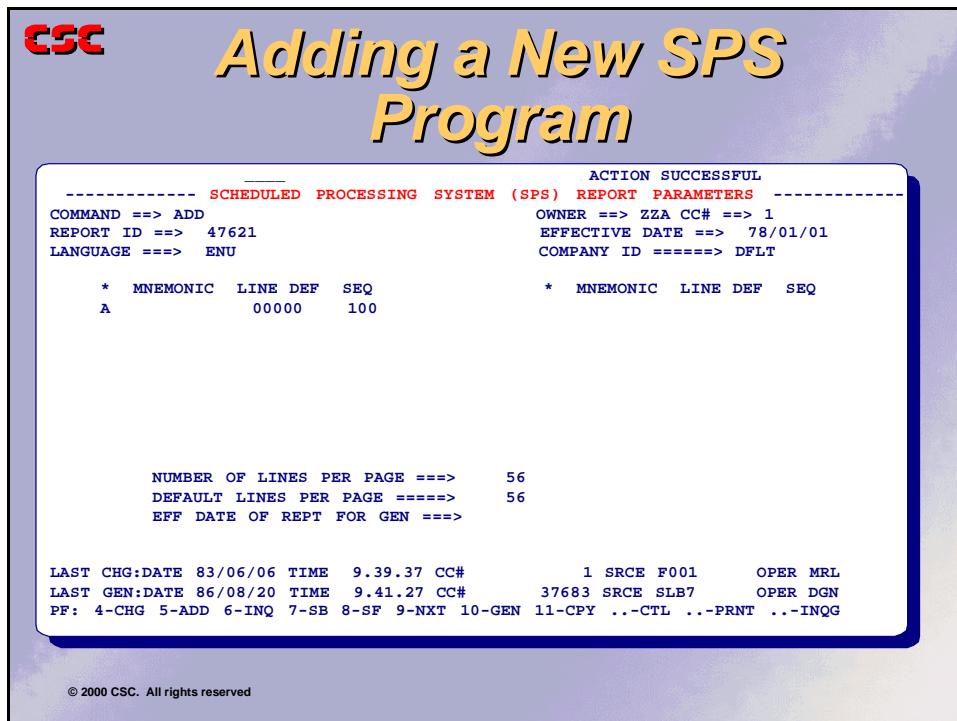
E 0000000000
```

Notes:



Online Coding of an SPS Program

All SPS programs are coded directly online. In this section of the class manual there are several examples for coding SPS. As in most Umbrella applications, a command and action are both required when adding, changing, and deleting code. The example below illustrates adding a Report and Line Definition to the "Scheduled Processing System (SPS) Report Parameters" screen



On the next page, the example uses the S select action to select and display the "Scheduled Processing System (SPS) Report Code" screen.

Notes:



Umbrella Programming

SPS Code Batch Print

CSC

Select Line Definition

```
ACTION SUCCESSFUL
----- SCHEDULED PROCESSING SYSTEM (SPS) REPORT PARAMETERS -----
COMMAND ==> INQ OWNER ==> INT CC# ==> 1
REPORT ID ==> 47621 EFFECTIVE DATE ==> 78/01/01
LANGUAGE ===> ENU COMPANY ID =====> DFLT

* MNEMONIC LINE DEF SEQ * MNEMONIC LINE DEF SEQ
S 00000 100

NUMBER OF LINES PER PAGE ==> 56
DEFAULT LINES PER PAGE =====> 56
EFF DATE OF REPT FOR GEN ==>

LAST CHG:DATE 91/09/23 TIME 9.38.15 CC# SRCE M014 OPER GLT
LAST GEN:DATE TIME CC# SRCE M014 OPER GLT
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-GEN 11-CPY ..-CTL ..-PRNT ..-INQG

© 2000 CSC. All rights reserved
```

CSC

E Line Definition Generated

```
ACTION SUCCESSFUL
----- SCHEDULED PROCESSING SYSTEM (SPS) REPORT CODE -----
COMMAND ==> INQ OWNER ==> INT CC# ==> 1
LANG RPT ID MNEMONIC LINE DEF SEQ EFFECTIVE DATE ==> 78/01/01
ENU 47621 00000 100 COMPANY ID =====> DFLT

* I VALUE SEQ ----- INSTRUCTION ----- COMMENT
E END PROCESSING

LAST CHG:DATE 91/09/23 TIME 9.38.15 CC# SRCE M014 OPER GLT
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL

© 2000 CSC. All rights reserved
```



To add a Line Definition Command, you overkey an existing Line Definition Command using the CHG command and the A action.

CSC

Adding Line Definition Commands

```
----- SCHEDULED PROCESSING SYSTEM ----- ACTION SUCCESSFUL
----- (SPS) REPORT CODE -----
COMMAND ==> CHG OWNER ==> INT CC# ==> 1
LANG RPT ID MNEMONIC LINE DEF SEQ EFFECTIVE DATE ==> 78/01/01
ENU 47621 00000 100 COMPANY ID =====> DFLT

* I VALUE SEQ ----- INSTRUCTION ----- COMMENT
A P 1101
```

Overtyping =
Insert Before

```
LAST CHG:DATE 91/09/23 TIME 9.38.59 CC# SRCE M014 OPER GLT
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL
```

© 2000 CSC. All rights reserved

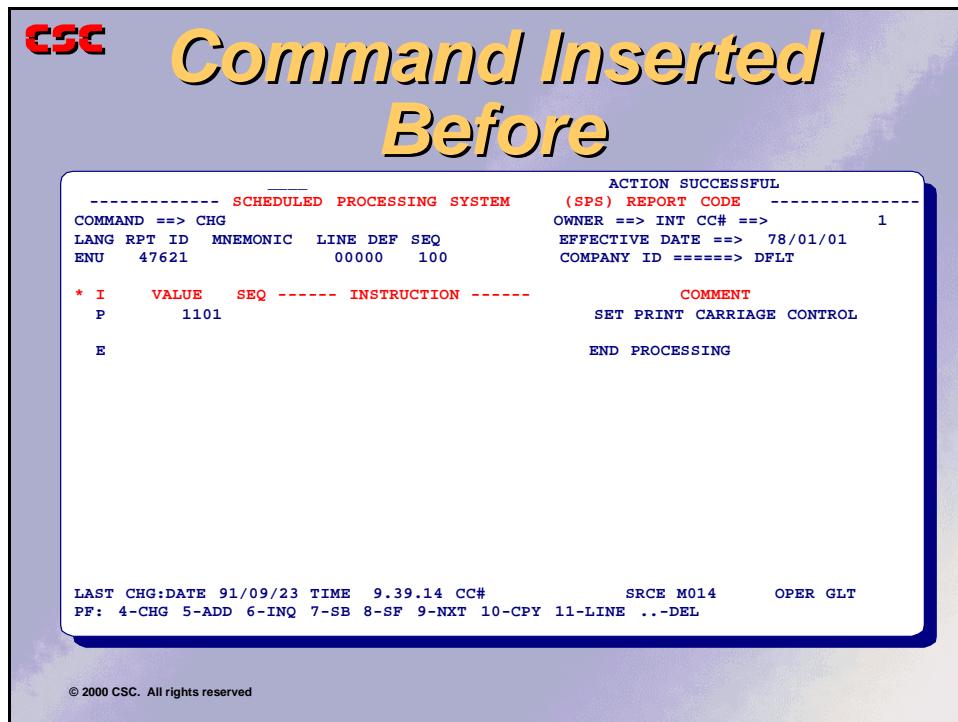
Notes:



Umbrella Programming

SPS Code Batch Print

The new command is inserted before the existing command.



Adding a Shared Format Definition to a Line Definition requires two steps. First, the S Line Definition command is added as shown on the top of the next page.

Notes:





Adding S Line Definition Command

SCHEDULED PROCESSING SYSTEM				48859 PRIMARY CMND & ACTION CONFLICT			
COMMAND ==> CHG				(SPS) REPORT CODE -----			
LANG RPT ID	MNEMONIC	LINE DEF	SEQ	OWNER ==> INT CC# ==>	1		
ENU 47621		00000	100	EFFECTIVE DATE ==>	78/01/01		
				COMPANY ID =====>	DFLT		
* I VALUE SEQ ----- INSTRUCTION -----				COMMENT			
P 1101				SET PRINT CARRIAGE CONTROL			
A S 50							

LAST CHG:DATE 91/09/23 TIME 9.40.18 CC# SRCE M014 OPER GLT
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL

© 2000 CSC. All rights reserved



Command Inserted Before

SCHEDULED PROCESSING SYSTEM				ACTION SUCCESSFUL			
COMMAND ==> CHG				(SPS) REPORT CODE -----			
LANG RPT ID	MNEMONIC	LINE DEF	SEQ	OWNER ==> INT CC# ==>	1		
ENU 47621		00000	100	EFFECTIVE DATE ==>	78/01/01		
				COMPANY ID =====>	DFLT		
* I VALUE SEQ ----- INSTRUCTION -----				COMMENT			
P 1101				SET PRINT CARRIAGE CONTROL			
S 50							
E							

LAST CHG:DATE 91/09/23 TIME 9.39.14 CC# SRCE M014 OPER GLT
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL

© 2000 CSC. All rights reserved



Umbrella Programming

SPS Code Batch Print

The second step is to extend the Shared Format Definition to allow coding of Format Definition Commands. The CHG command is used with the X action.

The screenshot shows a window titled "Extending Shared Format Definition/Subroutine" from the CSC SPS system. The window contains a command entry form:

SCHEDULED PROCESSING SYSTEM					ACTION SUCCESSFUL	
					(SPS)	REPORT CODE
COMMAND ==> CHG	OWNER ==> INT CC# ==>	1				
LANG RPT ID MNEMONIC LINE DEF SEQ	EFFECTIVE DATE ==>	78/01/01				
ENU 47621 00000 100	COMPANY ID =====>	DFLT				
* I VALUE SEQ ----- INSTRUCTION -----	COMMENT					
P 1101	SET PRINT CARRIAGE CONTROL					
X S 50						
E						

At the bottom of the window, there is a status message:

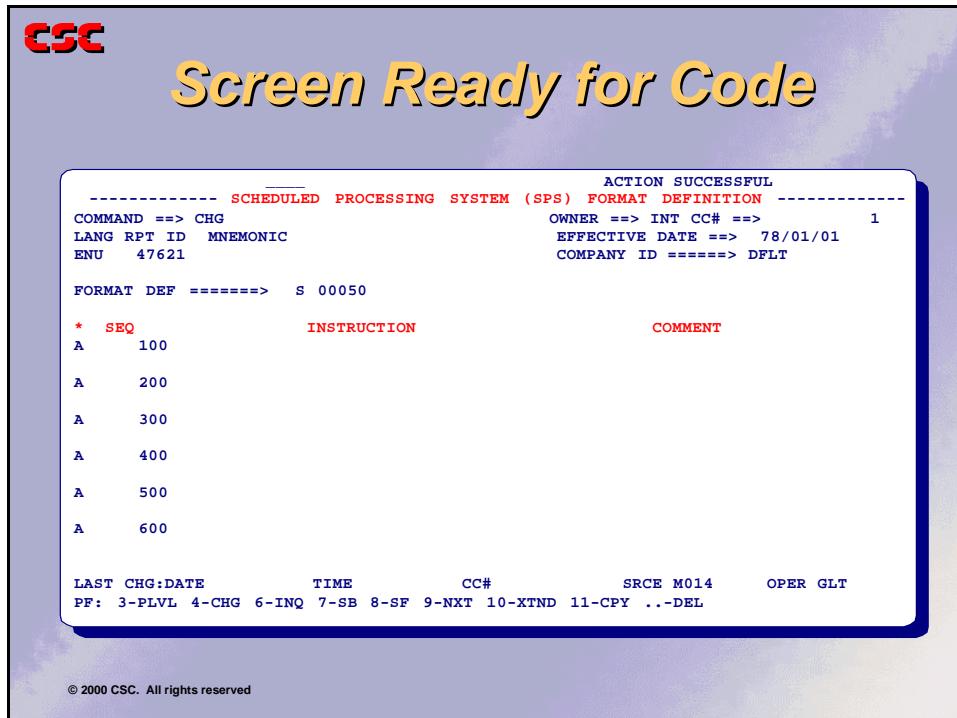
LAST CHG:DATE 91/09/23 TIME 9.39.14 CC# SRCE M014 OPER GLT
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL

© 2000 CSC. All rights reserved

Notes:



The Format Definition is opened and initialized, ready for Format Definition Commands.



Notes:



Umbrella Programming

SPS Code Batch Print

The instructions are coded in the area provided by the screen. Comments can be coded. If a comment is not coded SPS will take the SPS Description on the Element Definition if an Element ID was coded in the instruction. The unused code lines are not added to the Format Definition. After the instructions have been coded, press ENTER.

CSC Coding Format Definition Commands

```
ACTION SUCCESSFUL
----- SCHEDULED PROCESSING SYSTEM (SPS) FORMAT DEFINITION -----
COMMAND ==> CHG                               OWNER ==> INT CC# ==>      1
LANG RPT ID MNEMONIC                         EFFECTIVE DATE ==> 78/01/01
ENU     47621                                 COMPANY ID =====> DFLT

FORMAT DEF ======> S 00050

*   SEQ           INSTRUCTION          COMMENT
A    100          MOVE 'UMB' TO -010/003  MOVE APPLICATION ID
A    200
A    300
A    400
A    500
A    600

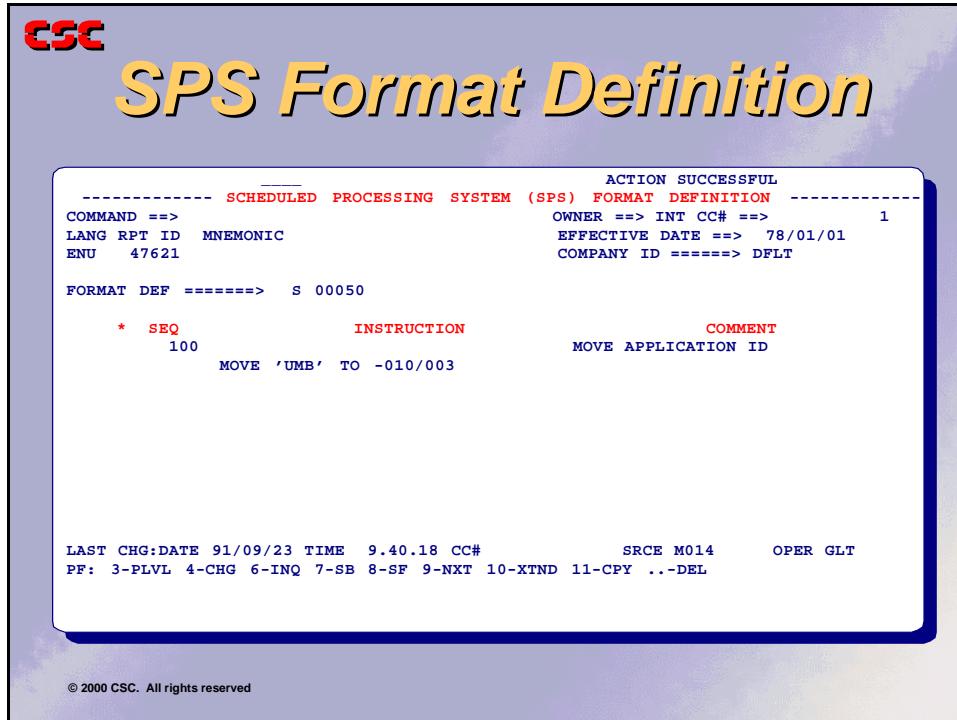
LAST CHG:DATE        TIME       CC#        SRCE M014      OPER GLT
PF: 3-PLVL 4-CHG 6-INQ 7-SB 8-SF 9-NXT 10-XTND 11-CPY ..-DEL

© 2000 CSC. All rights reserved
```

Notes:



The code now appears as:



The screenshot shows a CSC application window titled "SPS Format Definition". The window displays a successful action message and a detailed command definition.

ACTION SUCCESSFUL

----- SCHEDULED PROCESSING SYSTEM (SPS) FORMAT DEFINITION -----

COMMAND ==> OWNER ==> INT CC# ==> 1
LANG RPT ID MNEMONIC EFFECTIVE DATE ==> 78/01/01
ENU 47621 COMPANY ID =====> DFLLT

FORMAT DEF =====> S 00050

*	SEQ	INSTRUCTION	COMMENT
	100	MOVE 'UMB' TO -010/003	MOVE APPLICATION ID

LAST CHG:DATE 91/09/23 TIME 9.40.18 CC# SRCE M014 OPER GLT
PF: 3-PLVL 4-CHG 6-INQ 7-SB 8-SF 9-NXT 10-XTND 11-CPY ..-DEL

© 2000 CSC. All rights reserved

PF3 returns to the "Scheduled Processing System (SPS) Report Code" screen. To add code to the Shared Format Definition, the **X** action is used again.

Notes:



Umbrella Programming

SPS Code Batch Print

CSC

SPS Report Code

```
ACTION SUCCESSFUL
----- SCHEDULED PROCESSING SYSTEM (SPS) REPORT CODE -----
COMMAND ==> INQ OWNER ==> INT CC# ==> 1
LANG RPT ID MNEMONIC LINE DEF SEQ EFFECTIVE DATE ==> 78/01/01
ENU 47621 00000 100 COMPANY ID =====> DFLT

* I VALUE SEQ ----- INSTRUCTION ----- COMMENT
P 1101 SET PRINT CARRIAGE CONTROL

S 50 SHARED FORMAT DEFINITION

100 MOVE 'UMB' TO -010/003 MOVE APPLICATION ID

E END PROCESSING

LAST CHG:DATE 91/09/23 TIME 9.42.38 CC# SRCE M014 OPER GLT
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL
```

© 2000 CSC. All rights reserved

CSC

Extending Shared Format Definition

```
ACTION SUCCESSFUL
----- SCHEDULED PROCESSING SYSTEM (SPS) REPORT CODE -----
COMMAND ==> CHG OWNER ==> INT CC# ==> 1
LANG RPT ID MNEMONIC LINE DEF SEQ EFFECTIVE DATE ==> 78/01/01
ENU 47621 00000 100 COMPANY ID =====> DFLT

* I VALUE SEQ ----- INSTRUCTION ----- COMMENT
P 1101 SET PRINT CARRIAGE CONTROL

X S 50 SHARED FORMAT DEFINITION

100 MOVE 'UMB' TO -010/003 MOVE APPLICATION ID

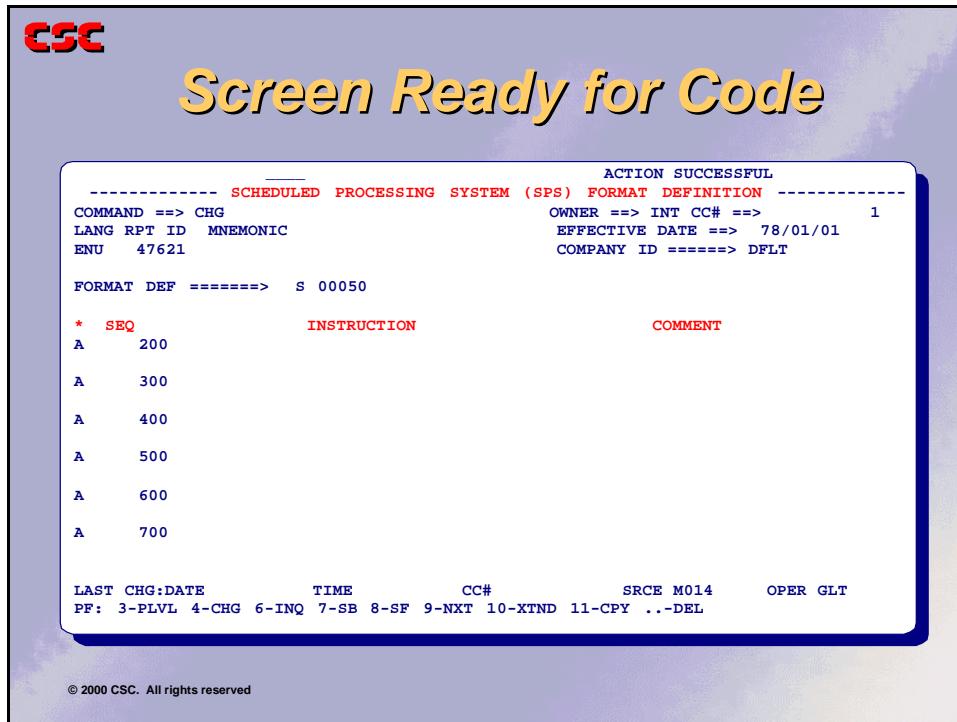
E END PROCESSING

LAST CHG:DATE 91/09/23 TIME 9.43.02 CC# SRCE M014 OPER GLT
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL
```

© 2000 CSC. All rights reserved



The Format Definition is opened with the next available sequence number.



Notes:



Umbrella Programming

SPS Code Batch Print

The additional Format Definition Commands are added to the Format Definition and the Enter key is pressed.

PF3 returns to the "Scheduled Processing System (SPS) Report Code" screen.

The screenshot shows a terminal window with the CSC logo at the top left. The title of the window is "SPS Format Definition Commands Added". Inside the window, there is a message "ACTION SUCCESSFUL" followed by a command history and a log entry.

```
ACTION SUCCESSFUL
-----
----- SCHEDULED PROCESSING SYSTEM (SPS) FORMAT DEFINITION -----
COMMAND ==> CHG
LANG RPT ID MNEMONIC
ENU 47621
OWNER ==> INT CC# ==> 1
EFFECTIVE DATE ==> 78/01/01
COMPANY ID =====> DFLT

FORMAT DEF =====> S 00050
* SEQ INSTRUCTION COMMENT
  200 SPACE 01 TIMES AFTER
  300 PRINT
  400 TCB-SYS-DATE
    EDIT TCB$DATE INTO -010/008 USING .02
  500 PRINT

LAST CHG:DATE 91/09/23 TIME 9.44.37 CC# SRCE M014 OPER GLT
PF: 3-PLVL 4-CHG 6-INQ 7-SB 8-SF 9-NXT 10-XTND 11-CPY ...-DEL

© 2000 CSC. All rights reserved
```

Notes:



Multiple changes can be made simultaneously as illustrated below.

CSC *Multiple Changes Can Be Made*

```

----- SCHEDULED PROCESSING SYSTEM ----- ACTION SUCCESSFUL
COMMAND ==> CHG (SPS) REPORT CODE -----
LANG RPT ID MNEMONIC LINE DEF SEQ OWNER ==> INT CC# ==> 1
ENU 47621 00000 100 EFFECTIVE DATE ==> 78/01/01
COMPANY ID =====> DFLT

* I     VALUE    SEQ ----- INSTRUCTION ----- COMMENT
P       1101

A A 1279

      100           MOVE APPLICATION ID
      MOVE 'UMB' TO -010/003
A      550
      47915
A      250           MOVE 'IS IPL DATE' TO -020
      400           TCB-SYS-DATE
      EDIT TCB$DATE INTO -010/008 USING .02
      500           PRINT

LAST CHG:DATE 91/09/23 TIME 9.44.37 CC# SRCE M014 OPER GLT
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL

```

© 2000 CSC. All rights reserved

In this example Shared Format Definition Command sequence numbers 200 and 300 were overkeyed using the A action. Because these statements are stored in PCD 2302 and the sequence number is part of the key, changes as illustrated above are allowed.

The final program code follows on the next page.

Notes:



Umbrella Programming

SPS Code Batch Print

CSC

SPS Report Code

```
ACTION SUCCESSFUL
----- SCHEDULED PROCESSING SYSTEM (SPS) REPORT CODE -----
COMMAND ==> INQ OWNER ==> INT CC# ==> 1
LANG RPT ID MNEMONIC LINE DEF SEQ EFFECTIVE DATE ==> 78/01/01
ENU 47621 00000 100 COMPANY ID =====> DFLT

* I     VALUE   SEQ ----- INSTRUCTION ----- COMMENT
P      1101
A      1279      PEM ACTIVITY
S      50       SHARED FORMAT DEFINITION

100      MOVE 'UMB' TO -010/003
200      SPACE 01 TIMES AFTER
250      MOVE 'IS IPL DATE' TO -020/011
300      PRINT

LAST CHG:DATE 91/09/23 TIME 9.46.53 CC#           SRCE M014      OPER GLT
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL
```

© 2000 CSC. All rights reserved

CSC

SPS Report Code

```
ACTION SUCCESSFUL
----- SCHEDULED PROCESSING SYSTEM (SPS) REPORT CODE -----
COMMAND ==> SF OWNER ==> INT CC# ==>
LANG RPT ID MNEMONIC LINE DEF SEQ EFFECTIVE DATE ==> 78/01/01
ENU 47621 00000 100 COMPANY ID =====> DFLT

* I     VALUE   SEQ ----- INSTRUCTION ----- COMMENT
400      EDIT TCB$DATE INTO -010/008 USING .02      TCB-SYS-DATE
500      PRINT
550      ACTIVITY 47915
E
END PROCESSING

LAST CHG:DATE 91/09/23 TIME 9.46.53 CC#           SRCE M014      OPER GLT
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL
```

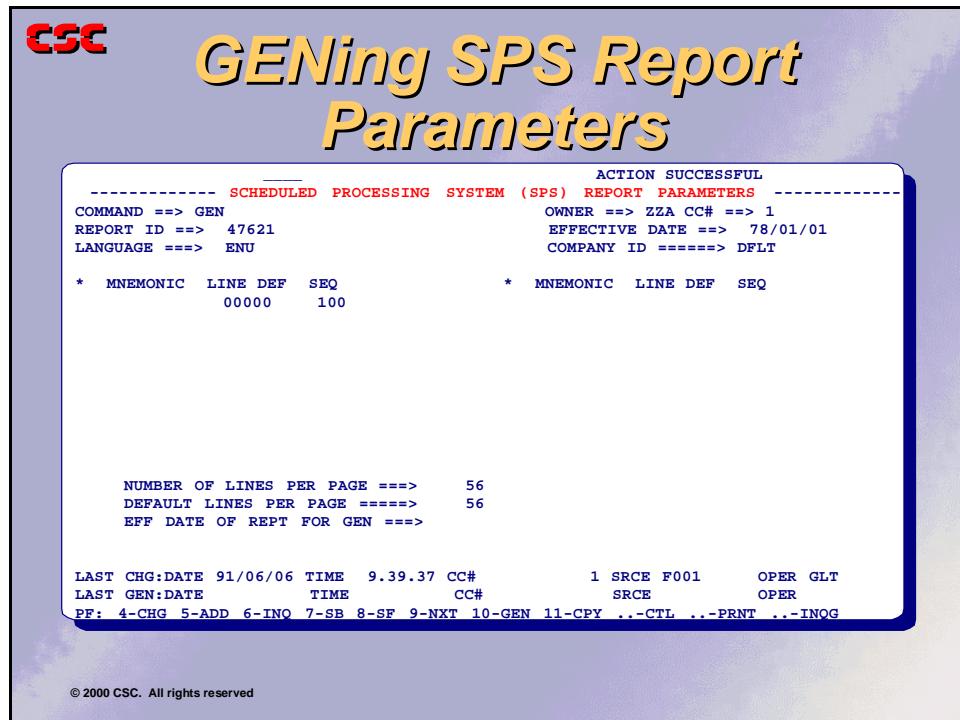
© 2000 CSC. All rights reserved



GENing the SPS Report Parameters

When all modifications to a program have been made, it must be GENed.

PF10 or the command GEN causes SPS to edit the SPS code and if valid update the CDMF1 file with the SPS code.



Notes:



Umbrella Programming

Functions of the SPS Editor

Functions of the SPS Editor

The SPS editor is invoked during online maintenance. The editor program is module I54362. The editor verifies such items as:

- Data element names exist.
- Line Definition Commands are structurally correct.
- Format Definition Commands are structurally correct.

Whenever an error is encountered, a message is displayed and an * indicates the line in error. Multiple errors are flagged simultaneously.

Copy Feature

The CPY command allows the user to copy existing SPS programs (multiple Line Definitions), a single Line Definition, or format definitions. The copy function can be requested from several screens as follows:

- "Scheduled Processing System (SPS) Report Parameters" screen (entire SPS report)
- "Scheduled Processing System (SPS) Line Definition" screen (single SPS line definition)
- "Scheduled Processing System (SPS) Format Definition" screen (single SPS format definition).

Online Example of Copy

A useful feature of SPS is that it is possible to copy an entire program, single Line Definition, or format definition online. A copy is initiated by the CPY command from various screens, including the report parameters, report code, and format definition. These three processes are illustrated on the following pages. You can also copy from the Line Definitions screen although a copy of only the Line Definition commands is less useful than those from the other three screens.

Notes:





Copy Example SPS Report Parameters

```
ACTION SUCCESSFUL
----- SCHEDULED PROCESSING SYSTEM (SPS) REPORT PARAMETERS -----
COMMAND ==> [CPY] OWNER ==> INT CC# ==> 1
REPORT ID ==> 08813 EFFECTIVE DATE ==> 78/01/01
LANGUAGE ===> ENU COMPANY ID =====> DFLT

* MNEMONIC LINE DEF SEQ * MNEMONIC LINE DEF SEQ
    00013   100     16000   100
    BREAK    06000   100     BREAK    06100   100
    BREAK    06200   100     HEAD     01000   100
    HEAD     02000   100

NUMBER OF LINES PER PAGE ==>      20
DEFAULT LINES PER PAGE =====>    56
EFF DATE OF REPT FOR GEN ==>

LAST CHG:DATE 91/09/23 TIME 9.46.53 CC#           SRCE M014      OPER GLT
LAST GEN:DATE 91/09/23 TIME 9.55.57 CC#           SRCE M014      OPER GLT
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-GEN 11-CPY ..-CTL ..-PRNT ..-INQG
```

© 2000 CSC. All rights reserved



Copy Example SPS Report Parameters

```
48835 ENTER 'COPY-TO' KEY DATA
----- SCHEDULED PROCESSING SYSTEM (SPS) REPORT PARAMETERS -----
COMMAND ==> CPY OWNER ==> INT CC# ==> 1
REPORT ID ==> [08817] EFFECTIVE DATE ==> 78/01/01
LANGUAGE ===> ENU COMPANY ID =====> DFLT

* MNEMONIC LINE DEF SEQ * MNEMONIC LINE DEF SEQ
    00013   100     16000   100
    BREAK    06000   100     BREAK    06100   100
    BREAK    06200   100     HEAD     01000   100
    HEAD     02000   100

NUMBER OF LINES PER PAGE ==>      20
DEFAULT LINES PER PAGE =====>    56
EFF DATE OF REPT FOR GEN ==>

LAST CHG:DATE 91/09/23 TIME 9.46.53 CC#           SRCE M014      OPER GLT
LAST GEN:DATE 91/09/23 TIME 9.55.57 CC#           SRCE M014      OPER GLT
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-GEN 11-CPY ..-CTL ..-PRNT ..-INQG
```

© 2000 CSC. All rights reserved



Umbrella Programming

Functions of the SPS Editor

CSC *Copy Example SPS Report Code*

SCHEDULED PROCESSING SYSTEM					ACTION SUCCESSFUL
COMMAND ==>	CPY	(SPS)	REPORT	CODE	
LANG RPT ID	MNEMONIC	LINE DEF	SEQ	OWNER ==> INT CC# ==>	1
ENU 47621		00000	100	EFFECTIVE DATE ==>	78/01/01
				COMPANY ID =====>	DFLT
* I VALUE SEQ ----- INSTRUCTION ----- COMMENT					
P	1101			SET PRINT CARRIAGE CONTROL	
A	1279			PEM ACTIVITY	
S	50			SHARED FORMAT DEFINITION	
		100		MOVE APPLICATION ID	
		MOVE 'UMB' TO -010/003			
		200	SPACE 01 TIMES AFTER		
		250	MOVE 'IS IPL DATE' TO -020/011		
		300	PRINT		
LAST CHG:DATE 91/09/23 TIME 9.46.53 CC# SRCE M014 OPER GLT					
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL					

© 2000 CSC. All rights reserved

Notes:





Copy Example SPS Report Code

```

----- SCHEDULED PROCESSING SYSTEM ----- ACTION SUCCESSFUL
COMMAND ==> CPY (SPS) REPORT CODE -----
LANG RPT ID MNEMONIC LINE DEF SEQ OWNER ==> INT CC# ==> 1
ENU 47621 00000 100 EFFECTIVE DATE ==> 78/01/01
                                         COMPANY ID =====> DFLT

* I     VALUE    SEQ ----- INSTRUCTION ----- COMMENT
P      1101                               SET PRINT CARRIAGE CONTROL
A      1279                               PEM ACTIVITY
S      50                                SHARED FORMAT DEFINITION

100      MOVE 'UMB' TO -010/003
200      SPACE 01 TIMES AFTER
250      MOVE 'IS IPL DATE' TO -020/011
300      PRINT

LAST CHG:DATE 91/09/23 TIME 9.46.53 CC#           SRCE M014      OPER GLT
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ...-DEL

```

© 2000 CSC. All rights reserved



Copy Example SPS Report Code

```

----- SCHEDULED PROCESSING SYSTEM ----- 48835 ENTER 'COPY-TO' KEY DATA
COMMAND ==> CPY (SPS) REPORT CODE -----
LANG RPT ID MNEMONIC LINE DEF SEQ OWNER ==> INT CC# ==> 1
ENU 47623 00000 100 EFFECTIVE DATE ==> 78/01/01
                                         COMPANY ID =====> DFLT

* I     VALUE    SEQ ----- INSTRUCTION ----- COMMENT
P      1101                               SET PRINT CARRIAGE CONTROL
A      1279                               PEM ACTIVITY
S      50                                SHARED FORMAT DEFINITION

100      MOVE 'UMB' TO -010/003
200      SPACE 01 TIMES AFTER
250      MOVE 'IS IPL DATE' TO -020/011
300      PRINT

LAST CHG:DATE 91/09/23 TIME 9.46.53 CC#           SRCE M014      OPER GLT
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ...-DEL

```

© 2000 CSC. All rights reserved



Umbrella Programming

Functions of the SPS Editor

CSC

Copy Example SPS Format Definition

```
ACTION SUCCESSFUL
----- SCHEDULED PROCESSING SYSTEM (SPS) FORMAT DEFINITION -----
COMMAND ==> CPY                                     OWNER ==> INT CC# ==>      1
LANG RPT ID MNEMONIC                           EFFECTIVE DATE ==> 78/01/01
ENU     47621                                         COMPANY ID =====> DFLT

FORMAT DEF ======> S 00050

*   SEQ           INSTRUCTION           COMMENT
  100    MOVE 'UMB' TO -010/003          MOVE APPLICATION ID
  200    SPACE 01 TIMES AFTER
  250    MOVE 'IS IPL DATE' TO -020/011
  300    PRINT
  400          TCB-SYS-DATE
    EDIT TCB$DATE INTO -010/008 USING .02
  500    PRINT

LAST CHG:DATE 91/09/23 TIME 9.42.38 CC#           SRCE M014      OPER GLT
PF: 3-PLVL 4-CHG 6-INQ 7-SB 8-SF 9-NXT 10-XTND 11-CPY ..-DEL

© 2000 CSC. All rights reserved
```

CSC

Copy Example SPS Format Definition

```
48835 ENTER 'COPY-TO' KEY DATA
----- SCHEDULED PROCESSING SYSTEM (SPS) FORMAT DEFINITION -----
COMMAND ==> CPY                                     OWNER ==> INT CC# ==>      1
LANG RPT ID MNEMONIC                           EFFECTIVE DATE ==> 78/01/01
ENU     47621                                         COMPANY ID =====> DFLT

FORMAT DEF ======> S 00075

*   SEQ           INSTRUCTION           COMMENT
  100    MOVE 'UMB' TO -010/003          MOVE APPLICATION ID
  200    SPACE 01 TIMES AFTER
  250    MOVE 'IS IPL DATE' TO -020/011
  300    PRINT
  400          TCB-SYS-DATE
    EDIT TCB$DATE INTO -010/008 USING .02
  500    PRINT

LAST CHG:DATE 91/09/23 TIME 9.42.38 CC#           SRCE M014      OPER GLT
PF: 3-PLVL 4-CHG 6-INQ 7-SB 8-SF 9-NXT 10-XTND 11-CPY ..-DEL

© 2000 CSC. All rights reserved
```



SPS Error Messages

SPS error messages are stored along with other Hogan error messages on CDMF format 1301. The 54000-54299 number range has been assigned to SPS. Below is an example of one of the SPS error messages:

CSC *SPS Error Messages Are on PCD 1301*

COMMAND ==> NXT	CCP MESSAGE DATA	ACTION SUCCESSFUL
FORMAT ==> 1301	OWNER ==> SPS CC# =====>	
	COMP ==> 1 EFF =====> 94/09/13 HIGH USE? ==> NO	
	FOUND CO: DFLT FOUND EFF: 78/01/01 EXPIRES: 999/12/31	
C C P M E S S A G E S -- FORMAT ID 1301		
KEY FIELDS		
LANG: ENU APPL:	COND CODE: 54271 ENVIRONMENT:	SOURCE: 4
ORIGINATING PROGRAM: I54262		
ACTIVITIES TO EXECUTE IF THIS CONDITION OCCURS (ENTER ACTIVITY ID NUMBER)		
#1	#2 #3 #4 #5	
#6	#7 #8 #9 #10	
.....+....1....+....2....+....3....+....4....+....5+....6....+....7	
MESSAGE LINE ID MUST BE ZERO :		
MEANING LINE ID MUST BE ZERO UNLESS THE ACTIVITY IS 1250, 1251 OR 1255.		
>		
>		
>		
>		
>		
LAST MAINT: DATE 92/06/15 TIME 9.35.40 CC OPER		
PF: 2-XREF 3-PLVLL 4-CHG 5-ADD 6-INQ 7-FMT 8-NXTE 9-NXT 11-ALT ..-DEL ..-NXTA		

© 2000 CSC. All rights reserved

Notes:



SPS Dumps

A PEM formatted dump is produced when a PEM exception is detected, a dump activity is issued, or a system abend condition occurs. Standard dump headers, PSW and registers at the time of abend, the abending program and offset, formatted TCB and CDMF Application Control Block, Trace table, UPCBs, and Data Group Storage are all presented in the dump.

Guidelines for using a PEM formatted dump to debug an SPS error are outlined in this section. The guidelines consist of four suggested steps. Examples are provided to illustrate each step.

Step 1—PEM Exceptions Report

Use the PEM exceptions report to determine the PEM abend message, the transaction being executed, the ID of the executing program, and the activity causing the abend. At times, this information is sufficient to solve the problem.

PAGE 0001
1 49 904 00000630
#905@7674@1128@1012@1468@1@ 00000640
#1009@8@1@SPSCLASS@0@% 00000650
11:49:53 DEMOZZ01 54208 SPS REPORT IS NOT IN SPSGEN FILE 00001 00049 00904 00004 0000001861 0000001288

Notes:



Step 2—System/User Code

Use the system completion codes printed with PSW information to determine the nature of the abend.

If the system code is equal to zeros and the user code is a non-zero value, the executing transaction was terminated by PEM. Next locate the PEM exception message. It is printed with the formatted TCB at the beginning of the dump.

If the system code is a non-zero value when executing SPS, steps 3 and 4 are followed. Step 3 illustrates the use of the flower box when a system abend is detected. Flower boxes are provided for general errors and can identify the line of code in which the S0C7 error has occurred. Step 4 outlines data group storage to be examined when an SPS flower box is not produced or does not yield sufficient information to identify the problem.

```

P E M      FORMATTED STORAGE DUMP          E X C E P T I O N          DATE=90/02/12    TIME=11:49:53    PAGE 0001

EXCEPTION MESSAGE: 54208 SPS REPORT IS NOT IN SPSGEN FILE

FORMATTED USER TCB --
CO: 00001 APP: 00049 FUNC: 00904 S/T: 04 TIME: 114953 DATE: 0900212 SOURCE: DEMOZZ01 OPER:
ACT: 01288 RSLT: 00000 USER: D3C000000000 DG: 03601 EFF DATE: 0900212 DEST:           PARM: 12288
XACT: 0000001288                               XDG: 0000003601

CDMF APPLICATION CONTROL BLOCK
ACT: 00005 RSLT: 00000

KEY FIELDS --
FORMAT ID: 0000001601      COID: 00001      EFF DATE: 0900212      EXP DATE: 9991231
                           FOUND COID: 00001   FOUND EFF DATE: 0840705
FLAGS ----- HI USE: N      OWNER APP:           ITEM FOUND: N
LAST CHANGED -- DATE: 0841214    TIME: 085947    CHG CNTL NUM: 0000018674    SOURCE: FSS8610    OPER: CRW
CHANGE CONTROL NUMBER: 0000000000          SECONDARY KEY:           SUB DGID: 0000000000

USER TRANSACTION CONTROL BLOCK
0011B910 000000 000071D8 A6E9F979 00010031 0388004 05080000 D3C00000 00000E11 30000114 *...Q.Z9.....L.....*  

0011B930 000020 953F0900 212FC4C5 D4D6E9E9 F0F14040 40404040 40404040 40404040 40404040 *....DEMOZZ01*  

0011B950 000040 40400000 00000000 00000000 00E80900 212FC240 00000508 00000E11 00000000 *....Y.B.....*  

0011B970 000060 00050000 00000641 00010900 212C9991 231F0001 0840705F D500D500 00000000 *....N.N.....*  

0011B990 000080 0841214C 0085947C 000048F2 C6E2E2F8 F6F1F040 C3D9E640 40404040 00000000 *....2FSS8610 CRW....*  

0011B9B0 0000A0 00000000 00000000 00000000 *.....*
```

Notes:



Umbrella Programming

SPS Dumps

Step 3—SPS Flower Box

The FLOWER BOX formatted by SPS contains information relevant to the report being generated. It includes the following:

```
COMPANY NUMBER
REPORT ID/MNEMONIC
LINE ID (=LINE DEFINITION NUMBER)
ACTIVITY
LINE ENTRY
FORMAT ENTRY
ERROR CODE, TYPE, AND DESCRIPTION
LAST/CURRENT LINE DEFINITION LOADED
LAST/CURRENT FORMAT DEFINITION LOADED
```

From the following example, identify the problem causing the error condition.

```
*****
*      CO=00001  REPORT=08801          LINE ID=16000
*
*          ACTIVITY=0000001250  LINE ENTRY=000  FORMAT ENTRY=000
*
*      ERROR CODE=54208  ERROR TYPE=SPS    ERROR=REPORT IS NOT IN SPSGEN FILE
*
*      LAST/CURRENT LINE DEFINITION LOADED:
*          CO=08888  PCDID=02300  KEY=08801
*
*      LAST/CURRENT FORMAT DEFINITION LOADED:
*          CO=08888  PCDID=02302  KEY=07627
*****
```

A second flower box is displayed in the following example. Identify the problem that caused the error condition and where the error is occurring.

```
*****
*      CO=00001  REPORT=08801          LINE ID=16000
*
*          ACTIVITY=0000001250  LINE ENTRY=001  FORMAT ENTRY=007
*
*      ERROR CODE=54000  ERROR TYPE=USER    ERROR=0C4 OR 0C7 ABEND DETECTED
*
*      LAST/CURRENT LINE DEFINITION LOADED:
*          CO=08888  PCDID=02300  KEY=08801
*
*      LAST/CURRENT FORMAT DEFINITION LOADED:
*          CO=08888  PCDID=02302  KEY=08801
*****
```

Notes:



Step 4—Data Group Storage

An S0C7 causes the majority of SPS dumps. When the first UPCB (User Program Control Block) in the PEM formatted dump identifies an SPS Program (I542XX), one primary SPS Data Group should be examined to find the location of the S0C7.

The SPS Processor table is contained in data group 2304, and is defined by the copybook I54204E. Elements in this Data Group to be examined are the following:

RT\$XRPT - OFFSET (1E)

THIS HEX VALUE REPRESENTS THE REPORT ID OF THE REPORT BEING GENERATED.

RT\$XMNEM - OFFSET (20)

THIS HEX VALUE REPRESENTS THE MNEMONIC FOR THE REPORT BEING EXECUTED.

RT\$XLINE - OFFSET (26)

THIS HEX VALUE REPRESENTS THE LINE DEFINITION BEING EXECUTED.

RT\$LTLOC - OFFSET (28)

THIS PACKED DECIMAL VALUE REPRESENTS THE NUMERICAL SEQUENCE OF THE LINE DEFINITION COMMAND BEING EXECUTED.

RT\$FTLOC - OFFSET (2A)

THIS PACKED DECIMAL VALUE REPRESENTS THE NUMERICAL SEQUENCE OF THE FORMAT DEFINITION COMMAND BEING EXECUTED.

A copy of a dump of data group 2304 is included in the following example for analysis.

DATA GROUP ID=0000002304													
00105048	000000	00460000	0010C460	028A3E80	00000000	00000000	00000000	00000000	00000000	00012261	*.....D-...../*		
00105068	000020	40404040	40403E80	001C007C	00000000	00000000	00000000	00000000	00000000	**		
00105088	000040	00000000	00000000	00000000	00105E10	00000000	00107FD0	00000000	001080C0	**		
001050A8	000060	00000000	900D9CBA	000D9C60	00045BAC	00108120	00000000	00007AC8	00105048	*-\$.....H...*		
001050C8	000080	00105E88	0049996	004A160	00045AD8	0009C60	0000D850	000DAC60	0009C60	*-..Q...-..Q...-..*		
001050E8	0000A0	001083C4	00108120	00108060	00007AC8	00105048	00105E88	0010BF9C	00108160	*D.....-..H...*		
00105108	0000C0	0010BEF0	0010BF94	00045888	000DAC60	0009C60	00105178	A00D9FAC	00108326	*O.....-..-..*		
00105128	0000E0	00000000	00000000	00003E80	00108120	0010C344	00007AC8	00105048	00105E88	*-.C...H...*		
00105148	000100	0010BF9C	600DBC4C	0010BEF0	0010BF94	00045888	000DAC60	0009C60	00105178	*-..0.....-..*		
00105168	000120	A00DBDA0	00108326	00000000	00000000	000D7E40	00108120	00105DAA	00007AC8	*=.....).H*		
00105188	000140	00105048	00105E88	0010BF98	0010C4C0	0010C4D4	0010BF94	00045888	000DAC60	*D..DM.....*		
001051A8	000160	0009C60	0010BF00	900DA6A6	00000000	000DAC60	000D9C60	00000000	00000000	*-.....-..-..*		
001051C8	000180	00000000	00000000	FF000000	00000000	0008BAA6	00000000	00000000	00000000	**		
001051E8	0001A0	00000000	000015EA	01900000	0010B380	000015EB	01900000	0010B538	000015EC	**		
00105208	0001C0	01900000	0010B6F0	000015E6	01900000	0010B8A8	00000E11	30000000	00042738	*0...W.....*		
00105228	0001E0	0000B810	00960000	0010CF08	00000407	01F40000	00045AD8	0000B806	00C80000	*4...Q...H...*		
00105248	000200	0010CFC8	0000B7FC	00320000	0010D0B8	0000B824	012C0000	0010D110	00000000	*H.....J...*		



Umbrella Programming

SPS Dumps

In the example identify the values in data group 2304:

RT\$XRPT	-	X'2261'	DECIMAL - 8801
RT\$XMNEM	-	X'404040404040'	CHARACTER - SPACES
RT\$XLINE	-	X'3E80'	DECIMAL - 16000
RT\$TLOC	-	(001C)	DECIMAL - 1
RT\$FTLOC	-	(007C)	DECIMAL - 7

Locate the problem in the copy of the SPS code below.

```
REPORT =08801    LINE=16000    SEQ=00100    COMPANY=08888    HEADING ID=000000
S 0000016000
FORMAT=16000    COMPANY=08888
SEQ      INSTRUCTION
00100 MOVE TLE50103 TO WL$LINE
00200 MOVE -018      TO TQE47204
00300 MOVE -        TO TQE47101
00400 MOVE -        TO TQE47102
00500 MOVE -        TO TQE47002
00600 MOVE -        TO TQE47201
00800 MOVE -        TO TQE47403
00900 MOVE -        TO TQE47108
01000 BREAK ON TQE47204(TQE47746) LEVEL=00001 RPT=00000 LID=06000
E 00000000000
```

Notes:



SPS CDMF Structure

Most parts of SPS detail are stored as items on various CDMF Formats (simulated PCDs). As a user of SPS, you are not usually concerned with the CDMF structure. Occasions may arise, however, in debugging problems when you need to know how Line Definitions and Format Definition Commands are stored internally. SPS Line Definitions are stored as individual entries on PCD 2300.

A sample entry is displayed below.

SPS Line Definition—PCD 2300

CSC SPS CDMF Structure
Line Definition PCD 2300

SPS LINE DEFINITION			ACTION SUCCESSFUL	
COMMAND ==> NXT	OWNER ==> FSS	CC# =====>		
FORMAT ==> 2300	COMP ==> 1	EFF =====>	94/09/13 HIGH USE? ==> NO	
		FOUND CO: DFLT	FOUND EFF: 78/01/01 EXPIRES: 999/12/31	
DEFAULT COMPANY = 08888				
LAST MAINT: DATE 94/08/31 TIME 8.50.47 CC 62694			OPER JJH	PCD ID = 02300
***** SPS LINE DEFINITION *****			*****	
< ITEM KEY >				
* LANGUAGE	ENU			
* REPORT ID	00000			
* LINE ID	00010			
* (RESERVED)	00000			
* SEQUENCE NO	00100			
HEADINGS ID 00000				
I VALUE	I VALUE	I VALUE	I VALUE	I VALUE
P 0000001101	A 0000001279	S 0000000010	S 0000000020	P 0000000002
A 0000001279	S 0000000030	A 0000001279	S 0000000040	A 0000001279
S 0000000050	A 0000001279	E 0000000000	0000000000	0000000000
0000000000	0000000000	0000000000	0000000000	0000000000
0000000000	0000000000	0000000000	0000000000	0000000000
0000000000	0000000000	0000000000	0000000000	0000000000

© 2001 CSC. All rights reserved

Notes:



Umbrella Programming

SPS CDMF Structure

Note these points regarding the PCD.

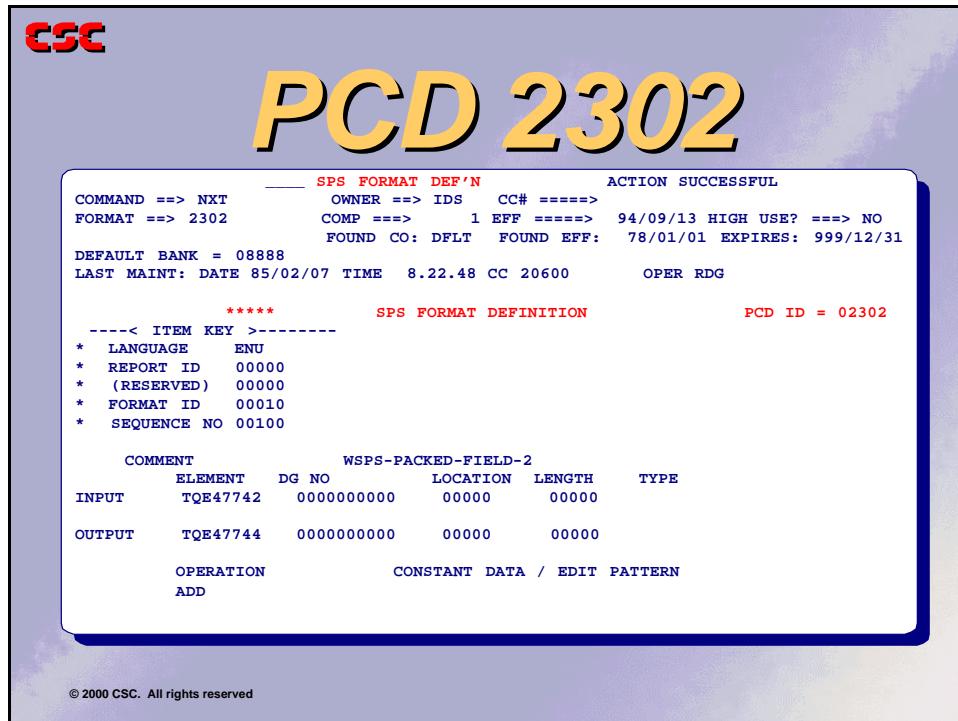
- The Format ID is 2300. FOUND CO is DFLT. FOUND EFF is 78/01/01.
- The Item Key has three parts:
 - USER KEY—Report ID (5 characters)//Mnemonic (6 characters)
 - LINE ID—Line Definition Number
 - SEQ—Arbitrary number (usually 100).
- All of the above fields are key fields and, thus, cannot be changed once the definition is added. A delete and add are required to change the value of the key.
- Line Definition Commands are read right to left top to bottom. There is a maximum of 30 Commands per PCD 2300 entry. SPS builds a new entry with a higher sequence number if more Line Definition Commands are needed.

Notes:



SPS Format Definition—PCD 2302

When an S Line Definition Command is encountered, SPS must identify the Format Definition Commands belonging to the Format Definition. Format Definition Commands are stored as individual items on PCD 2302.



Note these points about the PCD entry.

- FORMAT ID is 2302. FOUND COMPANY is DFLT. FOUND EFF is 78/01/01.
- The item key has three parts:
 - USER KEY—Report ID (5 characters)//Mnemonic (6 characters)
 - FORMAT ID—Format Definition ID Number
 - SEQ—Statement Number for the Format Definition Command.
- All of the above fields are key fields and, thus, cannot be changed once the definition is added. A delete and add are required to change the value of the key.
- Format Definition Commands are executed in numerical sequence within a Format Definition.
- Format Definitions are defined for the user key (Report ID / Mnemonic combination). They belong to the user key, not to a Line Definition. Format Definitions are SHARED among all Line Definitions for a given user key.



Umbrella Programming

SPS CDMF Structure

Using SPS PCD 2300 and PCD 2302

SPS report code and parameters are stored on various PCDs. Users do not normally use the PCDs directly. Their two primary uses are as follows:

- To maintain old reports that reference fields as explicit.
- To correct problems not obvious on the SPS Screens.

It is possible that extraneous PCD 2302 entries exist. They are not actually on reports any longer. If they exist and have errors in structure, a report may fail to GEN. In this case, the PCD entries need to be deleted directly from PCD 2302.

Knowledge of the PCD 2300 and PCD 2302 structure helps in analyzing SPS problems. To verify the actual code that exists, PCD 2300 and PCD 2302 can be reviewed.

SPS PCD are listed below.

PCD 2300

Line Definition (SPS Report ID)

PCD 2302

Format Definition Commands

PCD 2301

Override Lines-Per-Page

PCD 2308

Processing Control Activities

PCD 2379

Print Activities

Notes:



Common SPS Errors

As in any system, there are some errors that will occur frequently. The purpose of this topic is to list some frequent mistakes and their possible causes.

1. The number one error associated with SPS is caused by failure to GEN a report after a change is made. The results of this error are unpredictable because the code that can be viewed online is not that which is being executed from the SPSGEN File. REMEMBER TO GEN!!!!
2. The best debugging technique for SPS is to check all pieces of the SPS code that are generating a report. Because many rules are unbending, you have to follow them. Check all the details.

3. S0C7 ERRORS

- Check that the implosion and explosion fields are in the same order.
- Check for typing errors on a Data Element Name. You may have typed the name of a character field for a numeric one.
- Check for uninitialized fields. For example, the work fields, such as, W01, have to be initialized in simple SPS Reports.

4. BLANK REPORT

- Be sure you did not leave the carriage control byte set to a before printing setting.
- Check that you included the PRINT Command or its equivalent.

5. S001 OR S002

- This error indicates an empty data set.
- It can occur when a Data Base is not read correctly.
- It can occur when a Data String is not written correctly.
- It can occur when a write activity is incorrectly included in an explosion.

6. UNPREDICTABLE RESULTS

- An activity that should link to the SPS Processor Program, Program ID 1850, links to another program.
- Effective dates for pieces of the report are mixed.
- The E Command is left off the Line Definition.
- TLE50107 is not cleared or set in the implosion.



Umbrella Programming

Common SPS Errors

7. TRUNCATED RESULT

- The output length is too short.
- You forgot to count the commas, periods, and negative sign when calculating the length of the edited data.

8. REFERENCES TO AN UNEXPECTED REPORT ID

- Check the Report ID on the Processing Control Entry (SPSCTL).
- This field determines which Report ID is executed.

9. REFERENCES TO PAGE OVERFLOW

- Page overflow refers to the lines per page entry.
- When lines per page field is set on the Report Parameters screen, PCD 2301 entries are added. There should be entries for each print detail and break routine. Verify that all the entries exist.
- In addition, the HEADINGS ID field on the PCD 2300 entry for each detail print and break routine should be set to 1000.

10. 'P02' NOT AN ELEMENT ID

- Previous versions of SPS allowed position references such as P08, P01, and P02. They are no longer allowed. The editor attempts to translate them into an Element Name and fails.
- In GENing an SPS Report, you may discover some entries with these references. You need to delete them from PCD 2302 or correct them to the -008 form.

Notes:



Summary

CSC

Summary



- SPS Report Code useful for inquiring or modifying
- Line Definitions and Format Definitions contain needed commands
- SPS Report Parameters lists various reports
- SPS generates reports
- Report generating parameters maintained online

© 2000 CSC. All rights reserved

CSC

Summary



- Phases in report production
 - Implosion
 - Sort
 - Explosion
 - Output
- SPS report parameters
 - Line Definitions
 - Format Definitions
 - Activities associated with explosion and output
- SPS invoked by linking to program ID 1850

© 2000 CSC. All rights reserved



Umbrella Programming

Summary

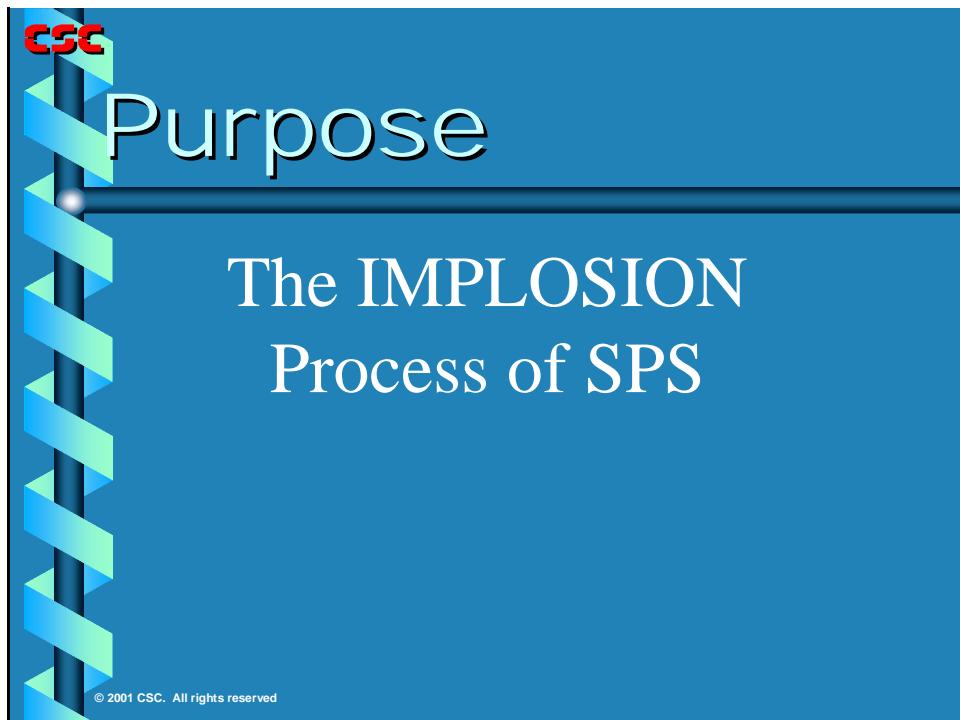
- The "Scheduled Processing System (SPS) Report Code" screen provides a useful means for inquiring on or modifying an SPS report.
- Line Definitions contain the commands needed for processing the lines on a report.
- Format definitions contain the commands for processing at the element level on a report.
- The "Scheduled Processing System (SPS) Report Parameters" screen lists the various reports for a Report ID and allows that Report ID to be GENed.
- SPS is an Umbrella tool for generating reports.
- Reports are generated through parameters maintained online in PCDs.
- There are four phases in the production of a report. They are:
 - Implosion
 - Sort
 - Explosion
 - Output
- SPS report production parameters include:
 - Line Definitions, which are equivalent to SPS programs.
 - Format definitions, which are equivalent to subroutines for the manipulation of data elements
 - Activities to be executed in the explosion and output phases of report production.
- SPS is invoked by linking to program ID 1850. The link activity indicates the type of processing to be performed.



SPS Implosion

27

Purpose



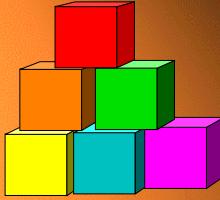
Notes:



Topics



Topics



- Implosion role in SPS
- PCD 1618 and PCD 15006
- Data group 3601

© 2000 CSC. All rights reserved

Notes:



Objectives

CSC

Objectives



- Identify the purpose of an SPS Implosion
- Define a reportable incident in SPS
- Describe the use of PCDs 1618 and 15006
- Explain the relationship among data groups in an Implosion
- Illustrate the structure of data group 3601 and explain its use in an Implosion
- Code an SPS Implosion routine

© 2000 CSC. All rights reserved

Notes:



Overview of the Implosion Phase

CSC

Implosion Occurs

When an Application Program:

- Identifies need to capture data for deferred reporting
- Issues link activity with ID greater than 1255

© 2000 CSC. All rights reserved

The IMPLOSION phase of SPS occurs when an application program identifies a need to capture data for reporting and issues a link activity, with an activity ID greater than 1255, to the SPS Processor Control program 1850.

Implosion begins with data fields, identified by data element names, from one or more data groups. Such fields are moved to data group 3601 to create SPS Data Strings. Entire data group(s) can be moved to the SPS Data String. By storing data during the normal processing cycle as these data strings, the desired information is available for later inclusion in a report.

Notes:



CSC

Data Group 3601 Parts

- VARIABLE RECORD CONTROL
- HOGAN KEY 
- USER-DEFINED KEY and REPORT DATA

© 2000 CSC. All rights reserved

Data group 3601 is used to store each data string as it is created. It has three parts: VARIABLE RECORD CONTROL, the HOGAN KEY, and USER-DEFINED KEY and REPORT DATA.

As each data string is created, it is written from data group 3601 to a sequential file. The sequential file serves as input to the actual production of the report.

Notes:



Umbrella Programming

Overview of the Implosion Phase



Implosions



- Coded as SPS Line Definitions
- Initiated from application program or transaction
- By a link activity to SPS
 - Must point to program ID 1850: SPS Processor
 - Builds key to SPS data
 - Executes SPS implosion program code

© 2000 CSC. All rights reserved.

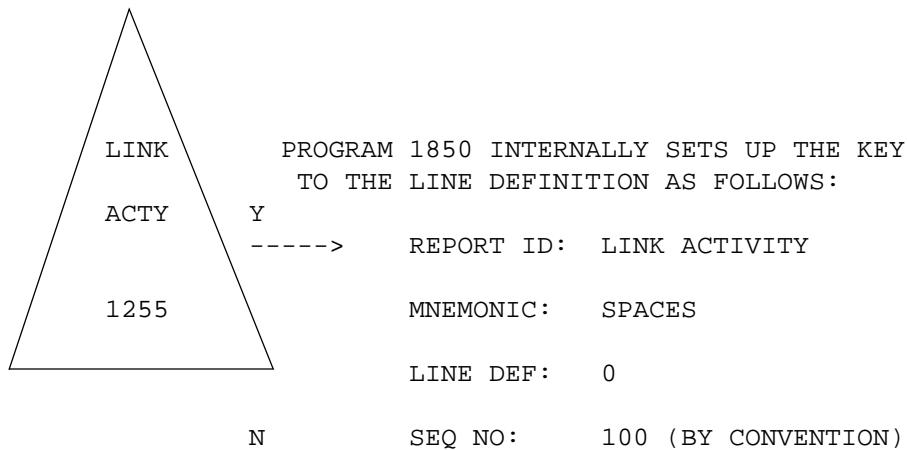
Implosions are coded as SPS Line Definitions. They are initiated from an application program or transaction. Whenever a reportable condition occurs, a link activity to SPS is issued. The link activity must point to program ID 1850, the SPS Processor. The SPS Processor builds the key to the SPSGEN data and executes the SPS implosion program code.

Notes:



Link Activity to Trigger an Implosion

Implosion code in SPS is invoked by a link activity to program 1850, the SPS Processor. The Link activity ID must be a number greater than 1255. The following diagram describes the processing decision and key built by program 1850.



SPS Implosion activities are assigned specific ranges for an application.

Several applications use the pattern 0NNRR:

WHERE	0	ZERO
NN	72 OR 76 FOR DDA	
	73 OR 78 FOR TDA	
	85 OR 87 FOR RPM	
	58 FOR OTP AND CDS	
RR	REPORT NUMBER	

Other applications use the following numbering patterns.

LOANS :	40RRR
PCS :	14RRR

WHERE RRR IS A UNIQUE REPORT NUMBER



Identifying a Reportable Incident

In designing reports to meet user requirements, each data field to appear on the report must be specified. The time at which such data is created in the application environment is termed a REPORTABLE INCIDENT in SPS.

Within application processing, there would be certain logical conditions to check to identify such reportable incidents. These conditions would depend on the design of a given report.

A reportable incident might be a test during a posting run on a non-sufficient-funds (NSF) condition. This condition may need to be included on any or all of the following type of reports:

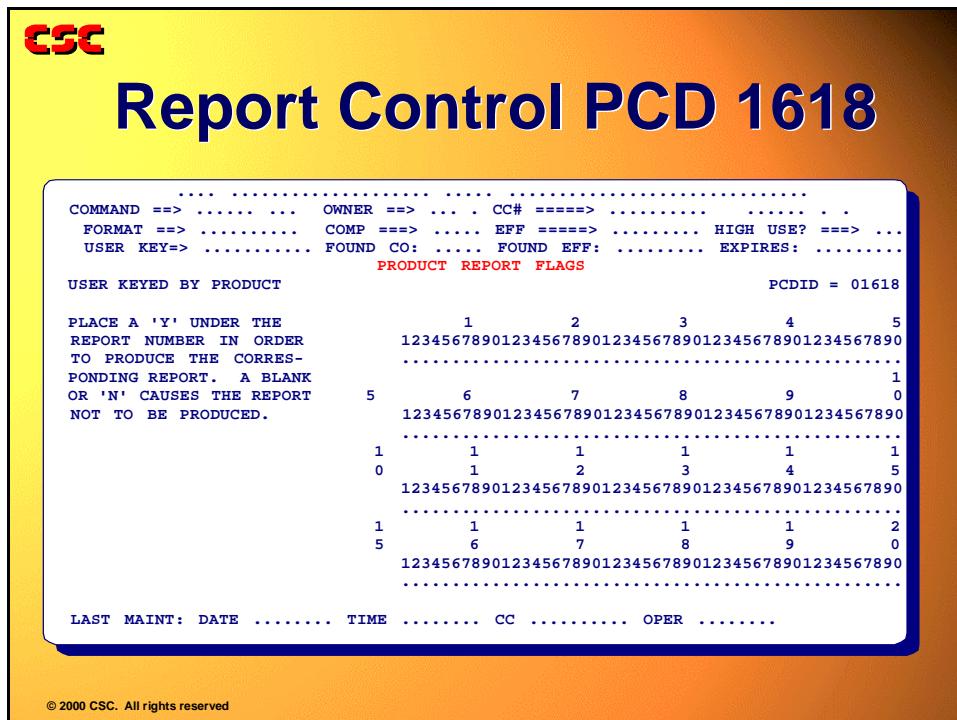
Notes:

The flowchart on the following page summarizes the processing flow in the application environment. If a reportable incident arises, control is passed to SPS program 1850 for an SPS Implosion.



PCD 1618

In the delivered reports, the reportable incidents are predetermined. A customer may not wish to produce every delivered report. There are, therefore, PCDs to allow customers to suppress any delivered reports for any specified company. For most applications, PCD 1618 is used.



PCD 1618 houses flags for up to 200 reports per application. PCD 15006 houses 40 flags. The typical coding to check on the flag to produce a report is shown by the example below:

BB060-POST-JOURNAL .

IF BCTL-RPT-FLAGS (1) EQUAL CC-Y
MOVE ACTIVITY-07201
TO TCB-LONG-ACTIVITY
PERFORM CA000-CALL-PEM

BB060-EXIT

Activity 07201 is a link activity to program 1850, which initiates the Implosion of data for DDA report 1. If report 1 is not desired, the flag 1 on PCD 1618 is set to N.



PCD 15006

ODS uses its own table, PCD 15006. This table contains a flag and the Implosion activity ID to be used with the report.

The screenshot shows a yellow-themed application window titled "Report Control PCD 15006". At the top left is the CSC logo. The main area displays a command-line interface for managing the PCD 15006 table. The commands shown are:

```
U140
COMMAND ==> ..... OWNER ==> ... CC# =====> ....
FORMAT ==> ..... COMP ==> .... EFF =====> .... HIGH USE? ==> ...
USER KEY=> ..... FOUND CO: .... FOUND EFF: .... EXPIRES: ....
```

Below the commands is a section titled "ODS REPORT FLAGS/ACTIVITIES" which lists user keys from 1 to 32. The "PCDID = 15006" parameter is also present. At the bottom, there is a section for last changes and a help menu:

```
LAST CHG:DATE ..... TIME ..... CC# ..... SRCE ..... OPER .....
PF: 1-HELP 3-PLVL 4-CHG 5-ADD 6-INQ 9-NXT ..-DEL
```

At the very bottom of the window, it says "© 2000 CSC. All rights reserved".

Notes:



Illustration of the Relationships Among Data Groups in SPS

The schematic on the next page describes the interplay among data group fields and data group 3601.

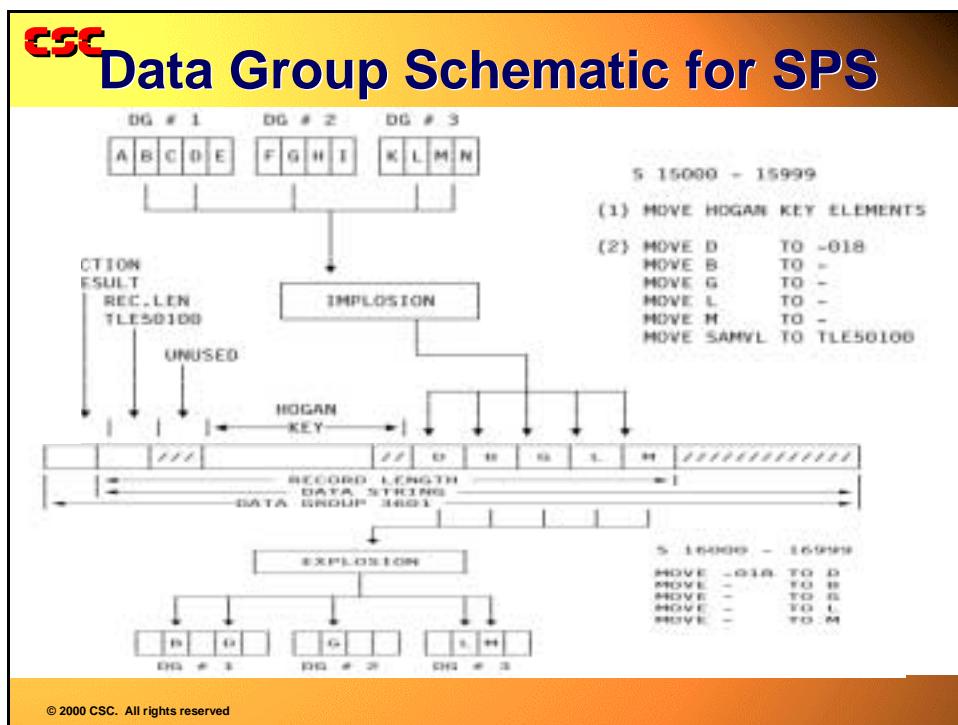
Format Definitions in the 15000 range (15000-15999) are required for SPS Implosion code. Program 1850 is hard coded to move data to data group 3601 for moves in Format Definitions in this range using the - notation.

Notes:



Umbrella Programming

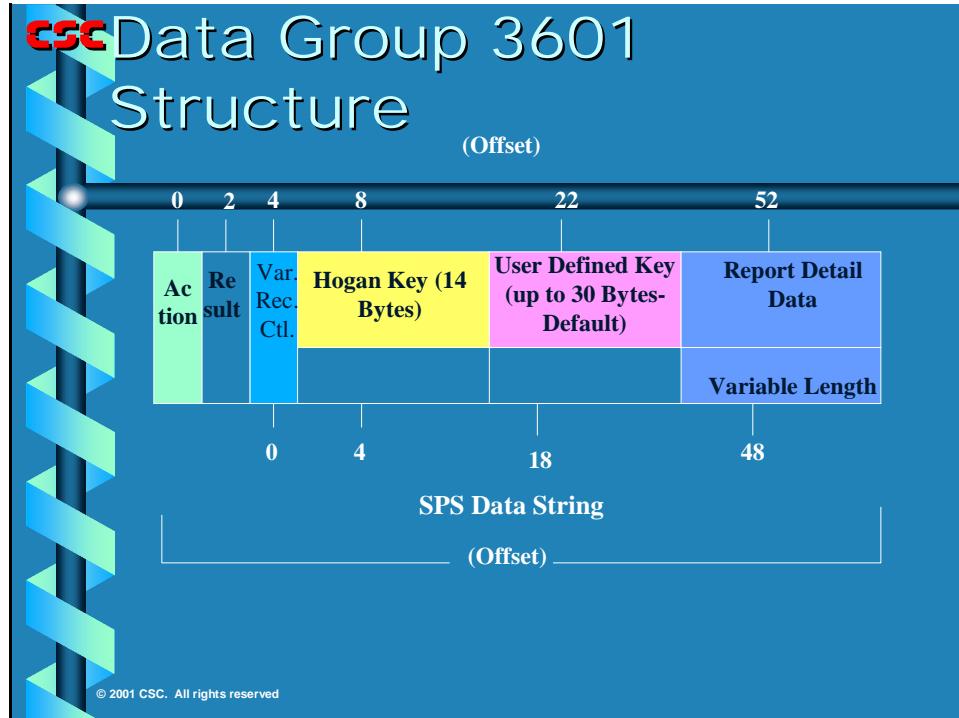
Illustration of the Relationships Among Data Groups in SPS



Notes:



Structure of Data Group 3601



Required by the operating system for variable length records.

ACTN3601 ACTION CODE	2 BYTES BINARY
TLE50100 RECORD LENGTH	2 BYTES BINARY
TLE50101 UNUSED	2 BYTES BINARY

HOGAN KEY

Used to group data strings for the same report. The Data String ID is used along with the Pass Number to access the SPS Control Entry for reports production.

ELEMENT	NAME	DEFINITION	USAGE
TLE50102	DATA STRING ID	2 BYTES BINARY	IMPLOSION ID#
TLE50103	SUB STRING ID	2 BYTES BINARY	DETAIL LINE PRINTER
TLE50104	FORMS ID	2 BYTES BINARY	
TLE50105	COMPANY ID	2 BYTES BINARY	
TLE50106	REPORT ID	2 BYTES BINARY	DETAIL PRINT SPS ID
TLE50107	BRANCH NUMBER	3 BYTES PACKED DECIMAL	
TLE50124	LANGUAGE CODE	1 BYTE CHARACTER	

USER-DEFINED KEY AND REPORT DATA



Umbrella Programming

Structure of Data Group 3601

User-selected fields are moved to the Data String starting at offset 18 of the Data String. The first 30 bytes are the user part of the sort key to control the sequence of data within a report.

Notes:



SPS Implosion Considerations



Requirements for Implosion



- ✓ May need to load 3-byte language code
- ✓ Move write action (02) to action field of data group
- ✓ Move data fields to data group 3601
- ✓ Begin user data fields at offset 018
- ✓ Specify length of data group:
 - ✓ Move SAMVL to TLE 50100
- ✓ Issue sequential data base write activity

© 2001 CSC Software International

There are several key points required to perform an Implosion:

- In shops that have upgraded to Umbrella 2.0, it may be necessary to load the TCB field TCB-SPS-IMPLODE-LANG with the 3-byte language code.
- In order to create a data string, a write action (02) must be moved to the action field of data group 3601.
- Data fields are moved to data group 3601 through move commands. Data Element Names are required. The entire data group can be moved to the Data String.
- The user data fields begin at offset 018 of the data string. The first move is to -018. The rest of the moves are to -. Implosions are required to be coded in Format Definitions within the 15000 range. Program 1850 moves the data to data group 3601 for code within this 15000 range.
- Because the user portion of the data string is variable, the length of the data group must be specified in the variable control record RDW.
- SAMVL is an internal field that totals the length of the Hogan key and user fields. It must be moved to TLE50100 after all data elements are



Umbrella Programming

SPS Implosion Considerations

moved to data group 3601.

- A sequential data base write activity must be issued. The SDB write activity can be issued directly from the SPS Implosion code or from a program linked to from the SPS Implosion routine.

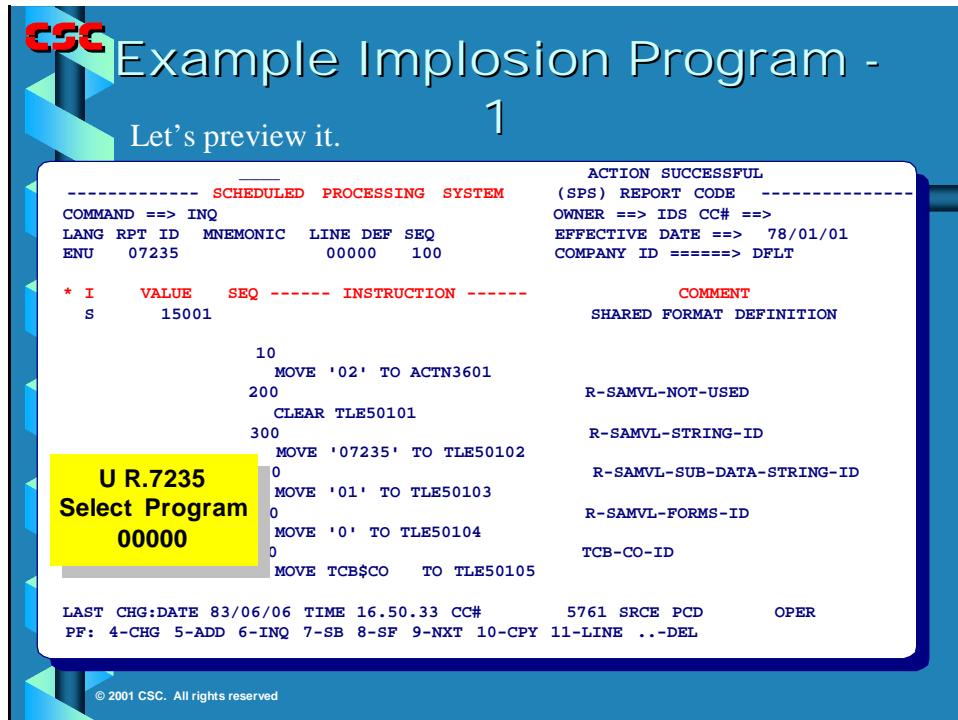
Notes:



Example Implosion Program

U R.7235

SELECT PROGRAM 00000



Notes:



Umbrella Programming

Example Implosion Program

SCROLL FORWARD

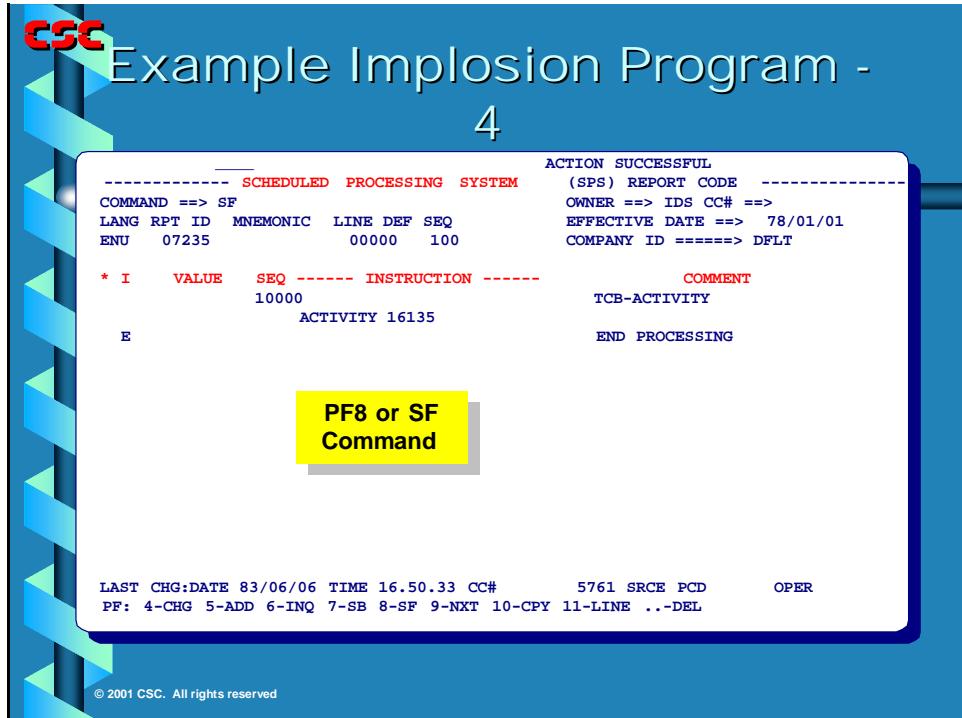
The screenshot shows a computer screen with a blue header bar containing the CSC logo and the title "Example Implosion Program - 2". Below the header is a white rectangular window titled "SCHEDULED PROCESSING SYSTEM". Inside the window, there is a command section and a detailed instruction list. A yellow box highlights the text "PF8 or SF Command". The instruction list includes comments such as "R-SAMVL-REPORT-ID", "M-BRANCH-NO", "M-CURR-BAL", etc. At the bottom of the window, status information is displayed: "LAST CHG:DATE 83/06/06 TIME 16.50.33 CC# 5761 SRCE PCD OPER PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL". The footer of the window says "© 2001 CSC. All rights reserved".

SCROLL FORWARD

The screenshot shows a computer screen with a blue header bar containing the CSC logo and the title "Example Implosion Program - 3". Below the header is a white rectangular window titled "SCHEDULED PROCESSING SYSTEM". Inside the window, there is a command section and a detailed instruction list. A yellow box highlights the text "PF8 or SF Command". The instruction list includes comments such as "M-PRDCT-CODE", "M-SUB-PRDCT-CODE", "M-LINE-AMT", etc. At the bottom of the window, status information is displayed: "LAST CHG:DATE 83/06/06 TIME 16.50.33 CC# 5761 SRCE PCD OPER PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ...-DEL". The footer of the window says "© 2001 CSC. All rights reserved".



SCROLL FORWARD



Notes:



Problem Specifications—SPS Implosion



The report your program has been printing is using only a fraction of the available SPS functions. This method is very limited in that only one report can be produced and in the same sequence as the master file.

This task involves adding an SPS Line Definition for the Implosion routine. Be sure to use the correct Format Definition range. Use activity ID 16135 to write the Data String. Use an Effective Date of 780101 and Company DFLT.

A Link Activity Definition to SPS 1850 will need to be added to the Process Dictionary. The Implosion activity ID is 478yy, where yy is your group number + 20.

Each group produces its own report. The Report ID is 088yy, where yy is your group number + 20.

The Data String to be written is identified by 478yy and the Report ID is 088yy, where yy is your group number + 20. When building the HOGAN KEY, be sure to move your group number to the SUB-DATA-STRING-ID field (TLE50103) and Zeros must be moved to the branch number (TLE50107).

Write the SPS code to implode the data elements listed below. The elements are listed in sorted order. The order for the sort is DEPARTMENT, LAST NAME, and FIRST NAME.

DD-ID	ATTRIBUTES	FIELD NAME	DATA GROUP
TQE47204	PIC X(3)	JOB-STAT-DEPT	47120
TQE47101	PIC X(15)	EMP-L-NAME	47110
TQE47102	PIC X(15)	EMP-F-NAME	47110
TQE47002	PIC 9(11) COMP-3	EMP-KEY-ID	47100
TQE47206	PIC XX	JOB-STAT-CLASS	47120
TQE47207	PIC XX	JOB-STAT-POSITION	47120
TQE47902	PIC X(40)	EMP-POS-DESCRIPTION	47190
TQE47303	PIC S9(7)V99 COMP-3	EMP-C-EARN-TOT	47130
TQE47403	PIC S9(7)V99 COMP-3	EMP-Y-EARN-TOT	47140



Testing of Implosion

Use the Online Activity Driver to read the Employee Data Base, link to your SPS Implosion Routine, and dump.

1. From a cleared screen, enter TEST and press ENTER.
2. Up to 20 activities may be keyed into this screen. Key an activity ID and press the tab key to position your cursor on the next field. After all activities have been keyed, press the ENTER key.
 - 9996xx, where xx is your group number, to read the EMP Data Base to provide test data.
 - 478yy, where yy is your group number + 40, to invoke the SPS IMPLOSION program you just coded.
 - 12, this activity ID will cause an S0C7 ABEND and invoke SMART.
3. In the SMART Command, key FDG=3601
4. Check the fields in the Data String DG3601 against your SPS Implosion code.
5. The RDW should have a hexadecimal 6F, which equals 111 bytes of data.

Reportable Incident

You will need to change your application program 9994xx, where xx is your group number, to test for Company 1 and invoke your SPS Implosion activity ID 478yy, where yy is your group number + 40.

Just compile/linkedit your program. Testing the program will be done after the next SPS Lab Exercise.

Notes:



Summary



Summary



- Implosion = capturing, storing, writing
- Report is produced when desired
- SPS Line Definition performs Implosion
- Implosion invoked by link activity
- Reportable incidents occur when report data is needed
- Delivered reports can be suppressed
- Data strings save data
- SAMVL totals length of Data String
- Implosion routines move to 3601 and code in 15000 range

© 2000 CSC. All rights reserved.

- SPS captures data at the time it is normally processed and stores the data in a Data String. The Data String is written to a sequential file. This process is called IMPLOSION.
- The file becomes input to producing a report. The production is scheduled when desired.
- The Implosion is performed by an SPS Line Definition.
- SPS Implosion is invoked by a link activity to the SPS Processor, program ID 1850 with the Link activity ID greater than 1255.
- Reportable incidents occur at the time the data needed for a report exists. Logic considerations control the creation of Data Strings.
- Hogan applications are set up so delivered reports can be suppressed. Most applications use PCD 1618. ODS uses PCD 15006.
- The data is saved by building Data Strings on data group 3601. The Data String consists of the RDW, HOGAN KEY, USER-DEFINED KEY, and REPORT DETAIL.
- SAMVL is an internal field that totals the length of the Data String as it is



Umbrella Programming

Summary

being built.

- Moves to - in an Implosion routines are moves to data group 3601. Implosion routines must be coded in the 15000 range.

Notes:



Umbrella Programming

Summary

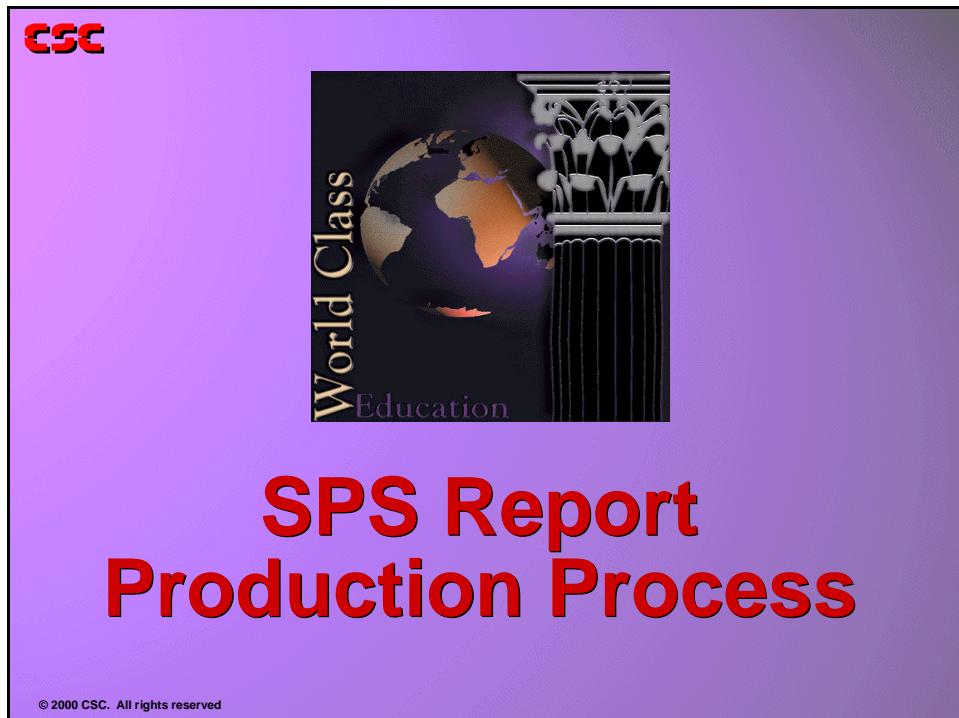


27-24

SPS Implosion

SPS Reports Production

28



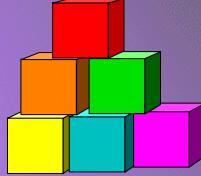
Notes:



Topics



Topics



- SPS Processing Control Program - Program ID
1698
- Sort Procedures
- Processing Control Entry
- Explosion Phase
- Output (PRINT) Phase
- SPS Online Code - Explosion and Output

© 2000 CSC. All rights reserved

Notes:



Objectives

CSC

Objectives



- List functions of SPS Processing Control Program
- Describe Sort Phase in SPS
- Explain structure and use of Processing Control Entry
- Define explosion in SPS
- Explain relationship among data groups in Explosion and Output Phases
- Code an SPS Explosion and Output Line Definition Online

© 2000 CSC. All rights reserved

Notes:



SPS Processing Control Program

When application processing is complete, all the Data Strings created through the Implosion Phase are stored in one or more sequential files. A job is executed to process the sequential file(s) through the Sort, Explosion, and Output Phases. The following is an example job stream.

```
//  
TRNNGRPTS JOB (HOGN,058,5005),'PRINT.RPTS',REGION=2000K  
//PRINT      EXEC HOGNBPEM  
//TID1T21   DD DSN=&&STRINGS,DISP=(OLD,DELETE)  
//SYSIN     DD *  
1 49 904          ACTIVITY DRIVER  
#905@  
7674@          ACTY TO ALLOCATE DG 3601  
1128@          ACTY TO ALLOCATE DG 1009  
1012@          DC ACTY TO DBLK DG 1009  
1468@          LINK ACTY TO SPS PROCESSING CONTROL PROGRAM  
I54260  
1@          END PROGRAM  
#1009@1@1@SPSCLASS@0@%      CONTROL CARD
```

The PEM Transaction shown above deblocks the control card #1009 and then links to the SPS Processing Control Program, program ID 1698, which drives Reports Production. Link activity ID 1468 invokes the program that initiates the Sort Phase. During execution, the sequential file(s) are sorted in ascending order. One record at a time is released into data group 3601 for processing.

The SPS Processing Control Program identifies the Processing Control Entry through the Pass Number in data group 1009 and the Data String ID, field TLE50102 data group 3601. This entry contains the activities to be issued for the Explosion and Output Phases. When the Processing Control Program has processed the last Data String Record, the Report Production Job Step is complete.

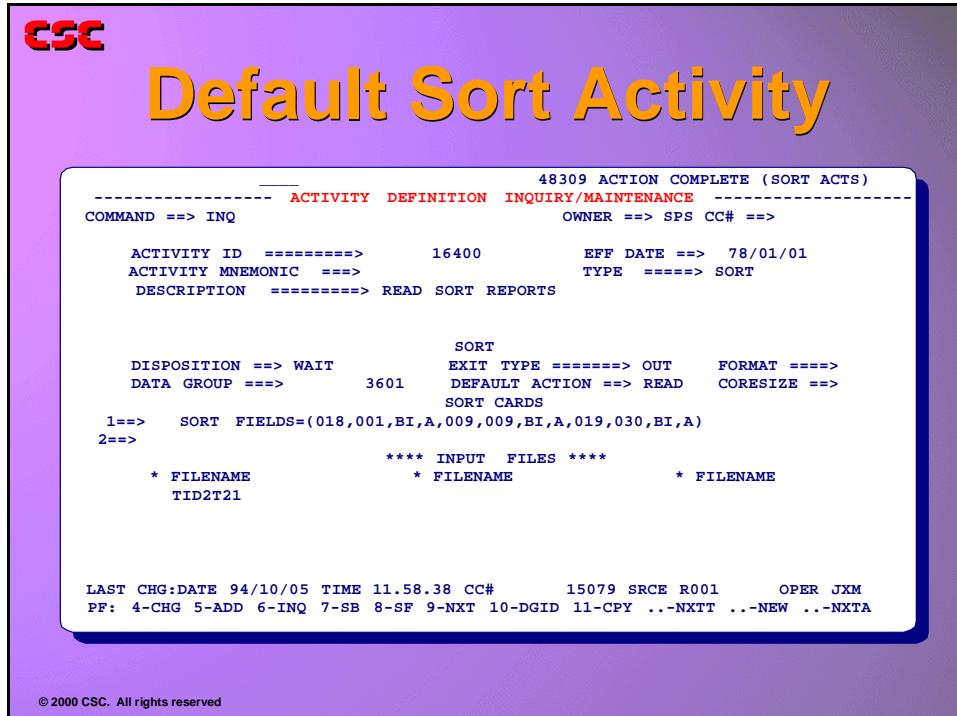
Notes:



Sort Phase

Default Sort Activity

1. From a cleared screen, enter U ACT 16400 and press ENTER.



Notes:



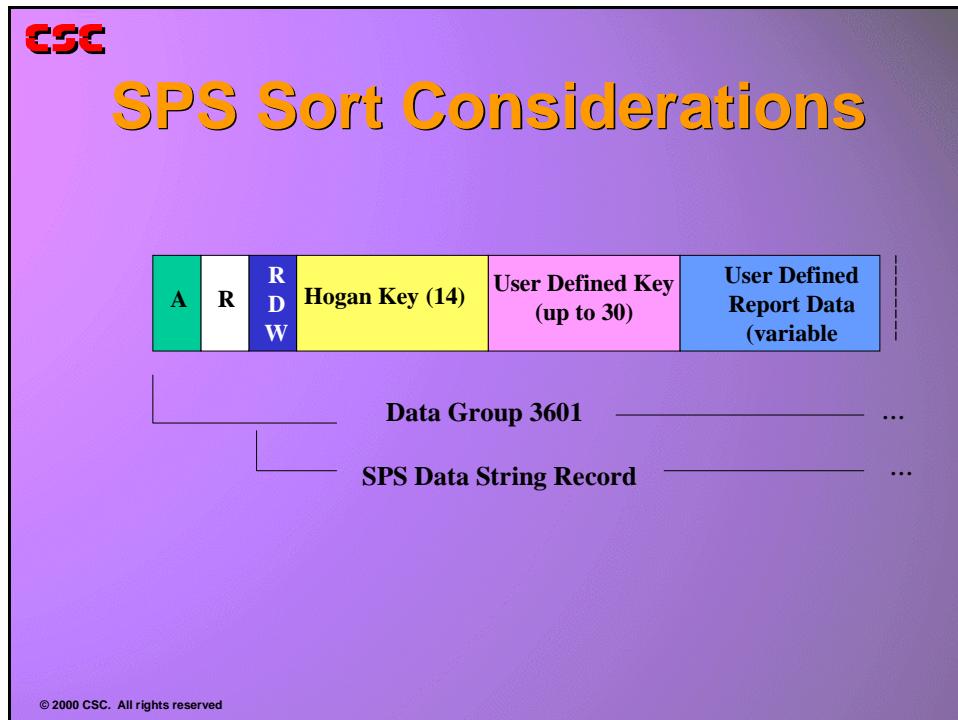
Umbrella Programming

Sort Phase

SPS Sort Considerations

The Data String File(s) created by Implosion processing must be sorted before the actual reports can be produced. The default Sort activity ID is 16400. It is defined to sort position 9 (offset 8 of the Data String) for a length of 9 and position 19 for a length of 30.

As a review, the structure of data group 3601 is as follows:



The fields of the HOGAN KEY are:

DATA STRING ID	2 BYTES
SUB DATA STRING ID	2 BYTES
FORMS ID	2 BYTES
COMPANY ID	2 BYTES
REPORT ID	2 BYTES
BRANCH NUMBER	3 BYTES
LANGUAGE CODE	1 BYTE

The fields of the HOGAN KEY included in the 16400 Sort are:

LANGUAGE CODE	1 BYTE	SORT FIELD=(018,001,A,BI)
FORMS	2 BYTES	SORT FIELD=(009,009,A,BI)
COMPANY ID	2 BYTES	
REPORT ID	2 BYTES	
BRANCH NUMBER	3 BYTES	
USER DEFINED	30 BYTES	SORT FIELD=(019,030,A,BI)



Umbrella Programming

Sort Phase

When an override of the Sort activity 16400 is needed, the override Sort Activity is specified as a parameter in the control card, data group 1009.

FIELDS WITHIN DATA GROUP 1009 ARE DESCRIBED IN THE EXAMPLE BELOW:

1 49 904	
#905@7674@1128@1012@1468@1@	
#1009@	
1@	PASS NUMBER
1@	COMPANY NUMBER
SPSCLASS@	EVENT NAME
0@	EFFECTIVE DATE
@	REPRINT REPORT
17401@	OVERRIDE SORT ACTIVITY
@%	OPTIONAL COMPANY EXIT

Notes:



Umbrella Programming

The Processing Control Entry

The Processing Control Entry

The Pass Number from data group 1009 deblocked from SYSIN and Data String ID placed in TLE50102 in data group 3601 during the Implosion routine is used by the SPS Reports Processor program 1698 as the key to the SPS Processing Control Entry.

Pass Number field is an arbitrary value used to categorize reports. For instance in DDA, Pass 1 Reports process the Implosion Data Strings from the Daily Application Programs. The Data Strings that are imploded from Pass 1 Reports (piggybacked) are processed in Pass 2 Reports. The DDA Pass Numbers are as follows:

1	AND	2	DAILY
3			WEEKLY
4			MONTHLY AND YEAR-END

The REPORT ID is used by SPS in the explosion and output phases. It is the numeric identifier for the SPS Line Definition for report production. For Hogan applications, there is a relationship between Report ID and Report Number. For example, DDA Report CR0035 has a Report ID 35.

The SPS Processing Control Entry contains three activity list areas.

BEG OF REPT (BOR) PROCEDURES.

Activity(ies) issued once at the Beginning of the Report.

Beginning of Report activities normally set up work fields for a report, such as, accumulators and switches. Activities 7627 and 17627 are the standard housekeeping activities. Customers can modify either to meet special needs.

PROCESSING PROCEDURES.

Activities issued for each data string of a report.

These activities define the processing requirements for each data string. At a minimum, each data string must have the element fields exploded back into the original data groups and the output line created. In addition, calculations may need to be performed upon the data elements before the report is printed.

Link activity, 1250, to the SPS Processor generally handles this processing.

END OF REPT (EOR) PROCEDURES.

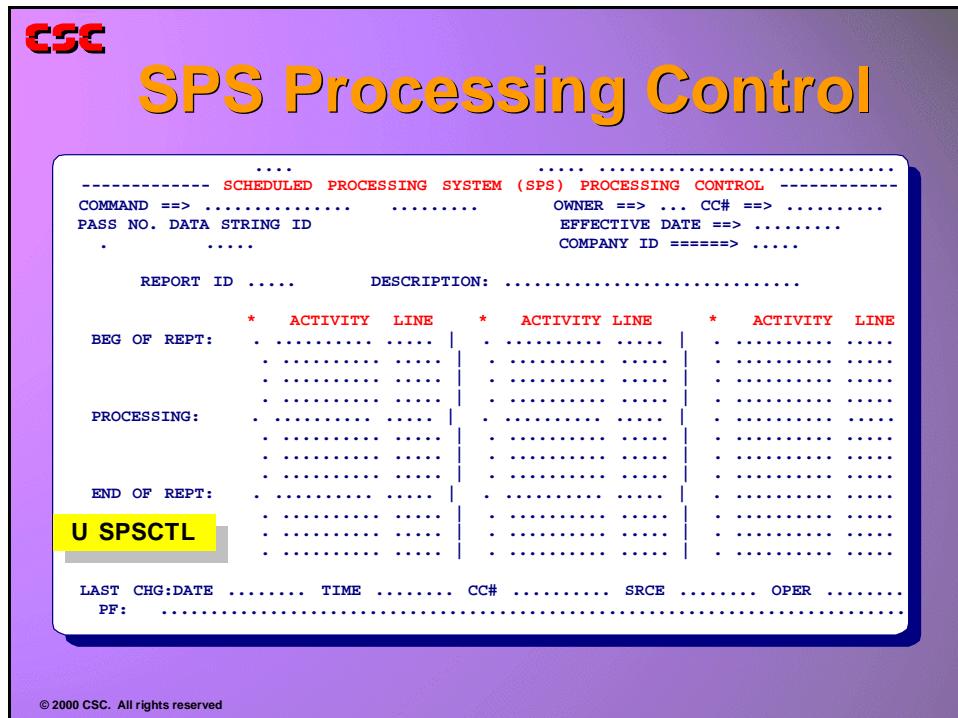
Activities issued once at the end of a report.

These activities generally include those that print company totals. All reports must issue activity 7628 last to release storage required by each report. Activity 7630 can be issued prior to activity 7628. It prints END-OF-REPORT.



SPS Processing Control Screen

The "Scheduled Processing System (SPS) Processing Control" screen can be displayed by using the direct prompt =SPSCTL from any Umbrella screen, or by entering U SPSCTL from a cleared screen. It can also be displayed from the "Scheduled Processing System (SPS) Reports Parameters" screen using the CTL command.



- Standard Command field. BYREPT, the Report Number is used as the search key. BYKEY, the Pass No and Data String ID is used as the search key.
- * (Action code) for each activity entry.
- Last change information.
- PF Keys documented on line 24.

Each task consists of an activity ID and a Line Definition Identifier. The activity is usually a link activity.

When the first Data String is read, the Processing Control Entry for the new Report is retrieved and its contents are tabled. The Activities associated with beginning of report are executed.



Umbrella Programming

SPS Processing Control Screen

Next the Activities associated with Processing are issued. They are executed consecutively for each Data String read. When the last Processing Activity has been processed, the next Data String Record is read. The loop continues until a Data String for a different Report, Company, or End of File condition is recognized.

When one of the above conditions arises, the End of Report section is executed. After the EOR activities have been executed and if EOF was not reached, the new control entry is retrieved and the process begins again.

Notes:



Example Processing Control Entry

To display this item online:

1. Key U SPSCTL from a cleared screen and press ENTER.
2. Key INQ in the COMMAND field, 1 for the Pass Number, and 07235 for the Data String ID and press ENTER.

CSC Example Processing Control Entry

ACTION SUCCESSFUL									
----- SCHEDULED PROCESSING SYSTEM (SPS) PROCESSING CONTROL -----									
COMMAND ==> INQ		OWNER ==> IDS CC# ==>							
PASS NO. DATA STRING ID		EFFECTIVE DATE ==> 78/01/01							
1 07235		COMPANY ID =====> DFLT							
REPORT ID 35 DESCRIPTION: MGMT OD REPORT									
* ACTIVITY LINE * ACTIVITY LINE * ACTIVITY LINE									
BEG OF REPT:	7627								
PROCESSING:	1250 16000			1250					
END OF REPT:	1255 6700			7216					7630
LAST CHG:DATE 85/09/16 TIME 10.09.27 CC# 30442 SRCE F019 OPER KSP									
PF: 4-CHG 5-ADD 6-INQ 9-NXT 10-BYKEY 11-BYREPT ..-DEL									

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

SPSCTL Beg of Rept Section

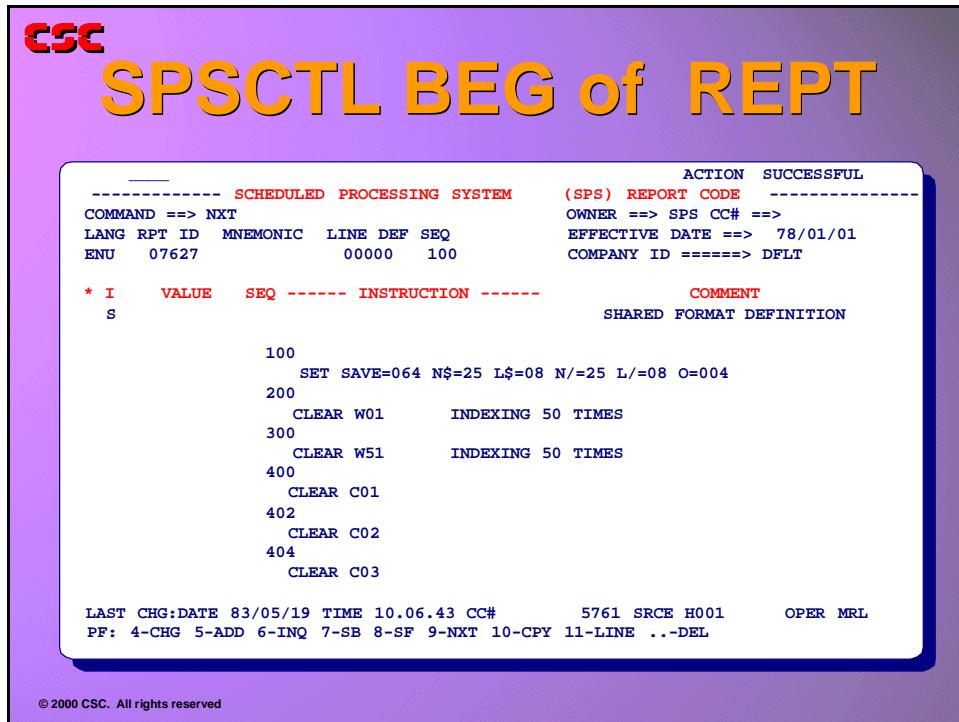
SPSCTL Beg of Rept Section

Beginning of Report work can include such items as printing banner or alignment pages or linking to a program to retrieve special data requirements for the report being generated. Most frequently, these activities are used to set up and initialize work fields or one time housekeeping functions.

In the example, there is only one activity ID in the Beginning of Report section. Activity ID 7627 is a link to the SPS Processor program 1850. Because the activity ID is greater than 1255, the SPS Processor Program builds a search key of 07627//spaces//000000//100. Report 07627, Mnemonic of Spaces, Line Definition zero, and Sequence Number 100. The Line Definition Commands are executed. The E (END) Line Definition Command causes control to return to the SPS Processor Program.

To view the SPS Code for 7627 online:

1. Key U SPS from a cleared screen and press ENTER.
2. Key INQ in the COMMAND field and Report ID 07627 and press ENTER.



The screenshot shows a terminal window with a purple background. At the top left is the CSC logo. The title bar reads "SPSCTL BEG of REPT". The main area displays SPS code. The code includes command definitions and a series of line definition commands (100 through 404) followed by a comment indicating they are part of a shared format definition. The bottom of the window shows system status information.

```
SCHEDULING PROCESSING SYSTEM          ACTION SUCCESSFUL
----- (SPS) REPORT CODE -----
COMMAND ==> NXT                         OWNER ==> SPS CC# ==>
LANG RPT ID MNEMONIC LINE DEF SEQ        EFFECTIVE DATE ==> 78/01/01
ENU    07627      00000   100             COMPANY ID =====> DFLT

* I     VALUE     SEQ ----- INSTRUCTION ----- COMMENT
* S                               SHARED FORMAT DEFINITION

100      SET SAVE=064 N$=25 L$=08 N/=25 L/=08 O=004
200      CLEAR W01      INDEXING 50 TIMES
300      CLEAR W51      INDEXING 50 TIMES
400      CLEAR C01
402      CLEAR C02
404      CLEAR C03

LAST CHG:DATE 83/05/19 TIME 10.06.43 CC#      5761 SRCE H001      OPER MRL
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL
```



Umbrella Programming

SPSCTL Beg of Rept Section

CSC

```

----- SCHEDULED PROCESSING SYSTEM ----- ACTION SUCCESSFUL
COMMAND ==> SF (SPS) REPORT CODE -----
LANG RPT ID MNEMONIC LINE DEF SEQ OWNER ==> SPS CC# ==>
ENU 07627 00000 100 EFFECTIVE DATE ==> 78/01/01
                                         COMPANY ID =====> DFLT

* I     VALUE    SEQ ----- INSTRUCTION ----- COMMENT
      406      CLEAR C04
      408      CLEAR C05
      410      CLEAR C06
      412      CLEAR C07
      414      CLEAR C08
      416      CLEAR C09
      418      CLEAR C10

LAST CHG:DATE 83/05/19 TIME 10.06.43 CC#      5761 SRCE H001      OPER MRL
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ...-DEL

```

© 2000 CSC. All rights reserved

CSC

```

----- SCHEDULED PROCESSING SYSTEM ----- ACTION SUCCESSFUL
COMMAND ==> SF (SPS) REPORT CODE -----
LANG RPT ID MNEMONIC LINE DEF SEQ OWNER ==> SPS CC# ==>
ENU 07627 00000 100 EFFECTIVE DATE ==> 78/01/01
                                         COMPANY ID =====> DFLT

* I     VALUE    SEQ ----- INSTRUCTION ----- COMMENT
      500      CLEAR WRK.BOR
      550      CLEAR PAGENO      CLEAR PAGE NUMBER
      600      ACTIVITY 02901      LINK TO BANK NAME/ADDR
      700      MOVE TLE50107 TO SAVBRCH      BRANCH NUMBER
      E          END PROCESSING

LAST CHG:DATE 83/05/19 TIME 10.06.43 CC#      5761 SRCE H001      OPER MRL
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ...-DEL

```

© 2000 CSC. All rights reserved



SPSCTL Processing Section

The minimum processing needs of each data string are:

- (1) EXPLOSION** The restoration of data to application data groups from the SPS Data String, data group 3601.
- (2) OUTPUT** General calculations, editing and outputting data, checking for control breaks and page headings.

CSC **SPSCTL Processing Section**

```
----- SCHEDULED PROCESSING SYSTEM (SPS) PROCESSING CONTROL -----
COMMAND ==> INQ          OWNER ==> IDS CC# ==>
PASS NO. DATA STRING ID   EFFECTIVE DATE ==> 78/01/01
1          07235           COMPANY ID =====> DFLT

REPORT ID      35      DESCRIPTION: MGMT OD REPORT
BEG OF REPT:    * ACTIVITY LINE      * ACTIVITY LINE      * ACTIVITY LINE
                 7627

PROCESSING:     1250 16000 |           1250
END OF REPT:    1255 6700 |           7216
                 7628

LAST CHG:DATE 85/09/16 TIME 10.09.27 CC# 30442 SRCE F019 OPER KSP
PFF: 4-CHG 5-ADD 6-INQ 9-NXT 10-BYKEY 11-BYREPT ..-DEL
```

•Explosion
•Output

© 2000 CSC. All rights reserved

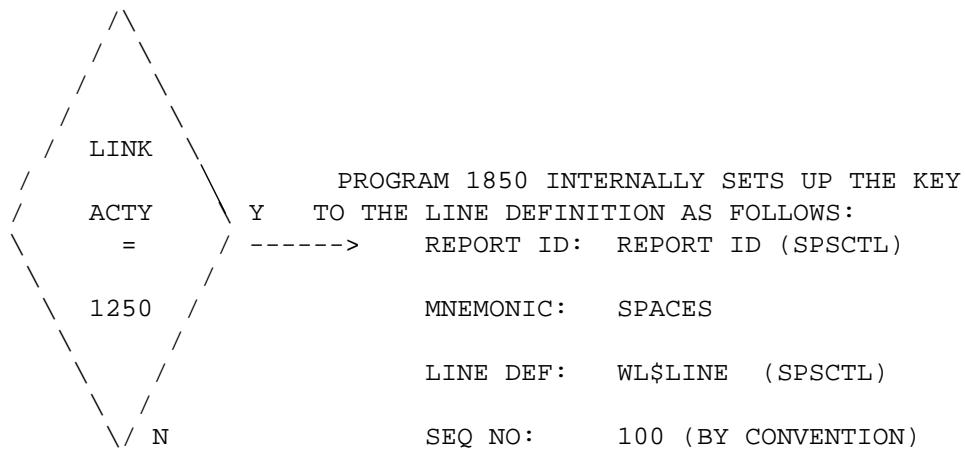
Notes:



Explosion Phase

The Explosion Phase involves the execution of an SPS Line Definition that moves data from a Data String Record (data group 3601) back to the original Data Groups from which they were Imploded. A Explosion is the reverse of an Implosion. The Explosion is executed for each Data String returned from the sort. The 1250 activity ID is coded on the SPS Processing Control Entry in the Processing Section with the Line Definition ID 16000.

Activity 1250 is specially defined to SPS to perform these two functions. It links to program 1850, the SPS Processor. The SPS Program builds the key to the SPSGEN data as follows:



There are three major requirements for Explosion Routines:

1. THE LINE DEFINITION MUST BE IN THE 16000 RANGE. IF ACTIVITY 1250 IS ISSUED, 16000 IS NORMALLY USED.
2. FORMAT DEFINITIONS USED ARE IN THE 16000 RANGE. ANY 'MOVE -' COMMANDS ARE FROM DATA GROUP 3601 RATHER THAN TO DATA GROUP 1031.
3. THE FIELDS MUST BE MOVED FROM DATA GROUP 3601 IN THE SAME ORDER AS THE IMPLOSION.

Note: At least the first Format Definition must be in the 16000 Range. PCS uses a large explosion SPS Line Definition 14800. The first Format Definition is 16010. It sets the pattern for the moves from data group 3601. Subsequent Format Definitions are 14001, 14002, 14004, and so on. These 14000 range numbers tie the implosion and explosion programs.



Umbrella Programming

SPSCTL Processing Section

Example Explosion Program

Line Definition 16000 for Report ID 35 contains the Explosion routine to move the data from the Data String Record back to the original data group(s). Field TLE50103, Sub Data String ID, is moved to WL\$LINE. This field was set to 1 in the Implosion routine.

CSC

Example Explosion Program

```
ACTION SUCCESSFUL
----- SCHEDULED PROCESSING SYSTEM (SPS) REPORT CODE -----
COMMAND ==> SF OWNER ==> IDS CC# ==>
LANG RPT ID MNEMONIC LINE DEF SEQ EFFECTIVE DATE ==> 78/01/01
ENU 00035 16000 100 COMPANY ID =====> DFLT

* I     VALUE   SEQ ----- INSTRUCTION ----- COMMENT
S      16001                               SHARED FORMAT DEFINITION

          100           BRANCH NUMBER
                  BREAK ON TLE50107(BRKBRCH) LEVEL=10 RPT=0 LID=06700
          200           R-SAMVL-SUB-DATA-STRING-ID
          MOVE TLE50103 TO WL$LINE
          300           M-CURR-BAL
          MOVE -018     TO TCE50015   M-DATE-OD
          400           MOVE -         TO TCE50051   M-DATE-LAST-DR-CR
          425           MOVE -         TO TCE50056   M-DATE-LAST-DEP
          450           MOVE -         TO TCE50060

LAST CHG:DATE 78/06/07 TIME 9.53.38 CC# SRCE HOGNJIM1 OPER
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ...-DEL
```

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

SPSCTL Processing Section



SCHEDULED PROCESSING SYSTEM				ACTION SUCCESSFUL
				(SPS) REPORT CODE
COMMAND ==> SF				OWNER ==> IDS CC# ==>
LANG RPT ID MNEMONIC LINE DEF SEQ				EFFECTIVE DATE ==> 78/01/01
ENU 00035 16000 100				COMPANY ID =====> DFLT
* I	VALUE	SEQ	INSTRUCTION	COMMENT
		500	MOVE - TO TCE50006	M-KEY
		600	MOVE - TO TCE50000	M-PRDCT-CODE
		650	MOVE - TO TCE50001	M-SUB-PRDCT-CODE
		700	MOVE - TO TCE50047	M-LINE-AMT
		800	MOVE - TO TCE50035	M-OFFICER
		850	MOVE - TO TCE50036	M-SECONDARY-OFFICER
		900	MOVE - TO TCE50004	M-SORT-NAME
LAST CHG:DATE 78/06/07 TIME 9.53.38 CC# SRCE HOGNJIM1 OPER				
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL				

© 2000 CSC. All rights reserved



SCHEDULED PROCESSING SYSTEM				ACTION SUCCESSFUL
				(SPS) REPORT CODE
COMMAND ==> SF				OWNER ==> IDS CC# ==>
LANG RPT ID MNEMONIC LINE DEF SEQ				EFFECTIVE DATE ==> 78/01/01
ENU 00035 16000 100				COMPANY ID =====> DFLT
* I	VALUE	SEQ	INSTRUCTION	COMMENT
	E			END PROCESSING
LAST CHG:DATE 78/06/07 TIME 9.53.38 CC# SRCE HOGNJIM1 OPER				
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL				

© 2000 CSC. All rights reserved



Umbrella Programming

SPSCTL Processing Section

Output Phase

After the Explosion Routine has completed, control is returned to the SPS Processing Control Program to issue the next activity ID coded in the Processing Section of the SPSCTL entry for this report. Once again, activity ID 1250 is issued and control is passed to the SPS Processor program 1850.

This time, however, there is no Line Definition value specified next to the 1250 activity ID. The Report ID is still 00035 and the MNEMONIC is SPACES. Because there is no Line Definition value, program 1850 uses the value of field WL\$LINE as the Line Definition. During the Explosion Routine TLE50103 was moved to WL\$LINE. The value in WL\$LINE is 1. The SPSGEN data is searched to retrieve the Line Definition 000001. This Line Definition contains the code to format and print a detail line(s) of the report.

CSC

Output Phase

ACTION SUCCESSFUL									
----- SCHEDULED PROCESSING SYSTEM (SPS) PROCESSING CONTROL -----									
COMMAND ==> INQ					OWNER ==> IDS CC# ==>				
PASS NO. DATA STRING ID					EFFECTIVE DATE ==> 78/01/01				
1 07235					COMPANY ID =====> DFLT				
REPORT ID 35 DESCRIPTION: MGMT OD REPORT									
* ACTIVITY LINE * ACTIVITY LINE * ACTIVITY LINE									
BEG OF REPT: 7627									
PROCESSING: 1250 16000 1250									
END OF REPT: 1255 6700 7216 7630									
7628									
LAST CHG:DATE 85/09/16 TIME 10.09.27 CC# 30442 SRCE F019 OPER KSP									
PF: 4-CHG 5-ADD 6-INQ 9-NXT 10-BYKEY 11-BYREPT ..-DEL									

•1250 16000
•1250 WL\$LINE

© 2000 CSC. All rights reserved

Notes:



Example Output(Detail) Program

CSC

Output(Detail) Program Example

```

----- SCHEDULED PROCESSING SYSTEM           ACTION SUCCESSFUL
COMMAND ==> SF          (SPS) REPORT CODE -----
LANG RPT ID MNEMONIC   LINE DEF SEQ      OWNER ==> IDS CC# ==>
ENU     00035            00001  100      EFFECTIVE DATE ==> 78/01/01
                                         COMPANY ID =====> DF LT

* I    VALUE   SEQ ----- INSTRUCTION ----- COMMENT
D      2
S      1
100
MOVE TCE50035 TO -001/005
200
MOVE TCE50036 TO -008/005
300
MOVE TCE50006 TO -015/015
400
MOVE TCE50000 TO -031/003
450
MOVE TCE50001 TO -035/002
LAST CHG:DATE 83/06/06 TIME 16.50.33 CC# 5761 SRCE PCD      OPER
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ...-DEL

```

© 2000 CSC. All rights reserved

CSC

```

----- SCHEDULED PROCESSING SYSTEM           ACTION SUCCESSFUL
COMMAND ==> SF          (SPS) REPORT CODE -----
LANG RPT ID MNEMONIC   LINE DEF SEQ      OWNER ==> IDS CC# ==>
ENU     00035            00001  100      EFFECTIVE DATE ==> 78/01/01
                                         COMPANY ID =====> DF LT

* I    VALUE   SEQ ----- INSTRUCTION ----- COMMENT
500
MOVE TCE50004 TO -040/025
600
EDIT TCE50015 INTO -073/015 USING $
700
DIVIDE TCE50047 BY *100      (GIVING RT$DP)
750
EDIT RT$DP     INTO -089/011 USING ZZZ,ZZZ,ZZ9
800
EDIT TCE50051 INTO -101/008 USING .02
820
EDIT TCE50060 INTO -116/008 USING .02
840
EDIT TCE50056 INTO -125/008 USING .02
LAST CHG:DATE 83/06/06 TIME 16.50.33 CC# 5761 SRCE PCD      OPER
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ...-DEL

```

© 2000 CSC. All rights reserved



Umbrella Programming

SPSCTL Processing Section

CSC

SCHEDULED PROCESSING SYSTEM			ACTION SUCCESSFUL	
COMMAND ==> SF	LANG RPT ID	MNEMONIC LINE DEF SEQ	(SPS) REPORT CODE	
ENU 00035		00001 100	OWNER ==> IDS CC# ==>	
			EFFECTIVE DATE ==>	78/01/01
			COMPANY ID =====>	DFLT
* I VALUE SEQ ----- INSTRUCTION -----			COMMENT	
850			DCB-DIFF-AMT	
	CLEAR THE48004			
860		MOVE TCB\$EFF TO THE48002	TCB-EFFECTIVE-DATE	
870		MOVE TCE50051 TO THE48001	M-DATE-OD	
875		MOVE '50' TO THE48000	LOAD DATE SERVICE ACTION	
880		ACTIVITY 01900	TCB-ACTIVITY	
890		ADD *1 TO THE48004	# OD DAYS +1	
895			PRINT # OD DAYS	
		EDIT THE48004 INTO -111/003 USING /		
LAST CHG:DATE 83/06/06 TIME 16.50.33 CC# 5761 SRCE PCD OPER				
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ...-DEL				

© 2000 CSC. All rights reserved

CSC

SCHEDULED PROCESSING SYSTEM			ACTION SUCCESSFUL	
COMMAND ==> SF	LANG RPT ID	MNEMONIC LINE DEF SEQ	(SPS) REPORT CODE	
ENU 00035		00001 100	OWNER ==> IDS CC# ==>	
			EFFECTIVE DATE ==>	78/01/01
			COMPANY ID =====>	DFLT
* I VALUE SEQ ----- INSTRUCTION -----			COMMENT	
900			M-CURR-BAL	
	ADD TCE50015 TO \$01			
1000		ADD *1 TO /01		
1100		ADD IDS81901 TO \$02	W819-OD-GLOBAL-AVAIL-BAL	
P	1		SET PRINT CARRIAGE CONTROL	
A	1279		PEM ACTIVITY	
S	2			
		100	M-SORT-NAME	
		MOVE TCE50004 TO -015/025		
LAST CHG:DATE 89/07/25 TIME 7.59.31 CC# SRCE PCD OPER KSP				
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ...-DEL				

© 2000 CSC. All rights reserved



SPSCTL End of Rept Section

When all the Data Strings for Report 35 have been processed, the END of REPORT Activities are executed. These activities could invoke final branch totals, company totals, grand totals, printing a matrix, and Piggybacking routines. In this example, Activities 7216, 7630, and 7628 are executed. They are all links to program ID 1850.

07216 Piggyback implosion for reports distribution list.

07630 Causes 'END OF REPORT' to print.

07628 Causes SPS to release all internal data groups, incore tables, and address pointers.

CSC SPSCTL End of Report Section

----- SCHEDULED PROCESSING SYSTEM (SPS) PROCESSING CONTROL -----				ACTION	SUCCESSFUL
COMMAND ==> INQ	PASS NO.	DATA STRING ID		OWNER ==> IDS CC# ==>	
	1	07235		EFFECTIVE DATE ==>	78/01/01
				COMPANY ID =====>	DFLT
REPORT ID	35	DESCRIPTION: MGMT OD REPORT			
BEG OF REPT:		* ACTIVITY LINE		* ACTIVITY LINE	
		7627			
PROCESSING:		1250 16000		1250	
END OF REPT:		1255 6700		7216	
		7628			7630
LAST CHG:DATE 85/09/16 TIME 10.09.27 CC# 30442 SRCE F019 OPER KSP					
PF: 4-CHG 5-ADD 6-INQ 9-NXT 10-BYKEY 11-BYREPT ..-DEL					
07216 Piggyback Reports Distribution					
07630 Print 'END OF REPORT'					
07628 SPS clean up					

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

Problem Specification—SPS Explosion and Output



Problem Specification—SPS Explosion and Output

This problem provides practice in the building of an Explosion routine and the creation of an output print routine using SPS. The result is a detail listing of the Employee Data Base. The print line layout can be of your own design.

Create the Processing Control Entry to process the Data Strings from your Implosion. The report will be for Pass Number 8. Remember the SPSCTL Key is the Pass Number//Data String ID.

Code the SPS Line Definition to Explode the data from the Data Strings you Imploded.

Code the SPS Line Definition to move the data to print and output the detail print line. Single space the report, and ignore page headings and control breaks.

Use Effective Date 780101, and Company DFLT.

DO NOT USE ANY FORMAT DEFINITION COMMAND OR LINE DEFINITION COMMAND THAT HAS NOT BEEN DISCUSSED IN CLASS.

Worksheets for the exercise are provided after the JCL listing.

JCL has been set up for testing. No modifications to the JCL are necessary. Following is a listing of the JCL

Notes:



Execute JCL for SPS Exercise



```

MODULE NAME ZUPCXXRP

//ZUP{J}RP JOB (HOGN,{B},BEF), 'UPC SPS PROBLEM', MSGCLASS=9,
//                      TIME=(00,09), REGION=4M, NOTIFY=&SYSUID
//*
//P$$SLIB  JCLLIB ORDER=( {TL}.PROCLIB)
//*
//***** UMBRELLA PROGRAMMERS CLASS BATCH PROGRAM EXECUTION JCL
//** FOR SPS REPORT LAB PROBLEM
//*****
//JS010    EXEC HOGNBPEM,
//          TEMPLIB=' {L}.TESTLIB',
//          VSAM=' {V}'
//-----
//TOD1D15   DD DSN=&&STRINGS,DISP=(NEW,PASS,DELETE),
//          UNIT=SYSDA,SPACE=(TRK,(1,1),RLSE),
//          DCB=(RECFM=VB,LRECL=2004,BLKSIZE=19069)
//EMP       DD DSN=&VSAM..EMP,DISP=SHR,
//          AMP=( 'BUFNI=05' , 'BUFND=10' )
//LSRINPT   DD *      STANDARD VSAM LSR TUNING CONTROL CARD INPUT
DD=CDMF1  BUFNI=6   BUFND=6
DD=CDMF2  BUFNI=6   BUFND=6
DD=CDMF3  BUFNI=6   BUFND=6
//SYSIN    DD *
1 99 99\\\
//*
//JS020    EXEC HOGNBPEM,
//          TEMPLIB=' {L}.TESTLIB',
//          VSAM=' {V}'
//-----
//SORTWK01  DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTWK02  DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTWK03  DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//TID2T21   DD DSN=&&STRINGS,DISP=(OLD,DELETE)
//LSRINPT   DD *      STANDARD VSAM LSR TUNING CONTROL CARD INPUT
DD=CDMF1  BUFNI=6   BUFND=6
DD=CDMF2  BUFNI=6   BUFND=6
DD=CDMF3  BUFNI=6   BUFND=6
//SYSIN    DD *
1 49 904
#905~7674~1128~1012~1468~1~
#1009~8~1~UPCCLASS~0~%
//*
//JS030    EXEC HOGNBPEM,
//          TEMPLIB=' {L}.TESTLIB',
//          VSAM=' {V}'
//-----
//*
//**      PRINT THE SPS SOURCE ON CDMF - PCD 2300, 2302, 2308
//*
//*****
//LSRINPT   DD *      STANDARD VSAM LSR TUNING CONTROL CARD INPUT
DD=CDMF1  BUFNI=6   BUFND=6
DD=CDMF2  BUFNI=6   BUFND=6
DD=CDMF3  BUFNI=6   BUFND=6
//SYSIN    DD *
1 54 21           IMPLOSION
#1031~ ~478{4}~478{4}~%
1 54 20           PROCESSING CONTROL
#2309~8~8~478{4}~478{4}~%
1 54 21           EXPLOSION AND REPORTS

```



Umbrella Programming

Problem Specification—SPS Explosion and Output

```
#1031-- ~088{4}~088{4}~%  
/*  
*/
```

Notes:



Umbrella Programming
Problem Specification—SPS Explosion and Output

Worksheet for Planning SPS Reports

IMPLOSION PROGRAM: LINK ACTIVITY = _____

WHERE ISSUED? _____

REPORT ID =
MNEMONIC =
LINE DEFN =
SEQ NUMBER =
FORMAT DEFN =

PROCESSING CONTROL ENTRY: PASS NUMBER = _____ //
DATA STRING ID _____

REPORT ID _____

EXPLOSION PROGRAM LINK ACTY = _____

OUTPUT PROGRAM LINK ACTY = _____

WHERE ISSUED? _____ HERE ISSUED? _____

REPORT ID =
MNEMONIC =
LINE DEFN =
SEQ NUMBER =
FORMAT DEFN =

REPORT ID =
MNEMONIC =
LINE DEFN =
SEQ NUMBER =
FORMAT DEFN =

HEADING PROGRAM LINK ACTIVITY = _____

DEPT BREAK LINK ACTIVITY = _____

WHERE ISSUED? _____ WHERE ISSUED? _____

REPORT ID =
MNEMONIC =
LINE DEFN =
SEQ NUMBER =
FORMAT DEFN =

REPORT ID =
MNEMONIC =
LINE DEFN =
SEQ NUMBER =
FORMAT DEFN =



Umbrella Programming

Problem Specification—SPS Explosion and Output

COMPANY BREAK LINK ACTIVITY = _____

WHERE ISSUED? _____

REPORT ID =

MNEMONIC =

LINE DEFN =

SEQ NUMBER =

FORMAT DEFN =

Notes:



Umbrella Programming

Problem Specification—SPS Explosion and Output

Worksheet for Processing Control Entry

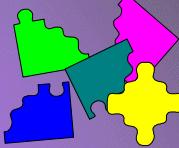
Notes:



Summary

CSC

Summary



- 1698 controls production of reports
- SORT, EXPLOSION, OUTPUT = phases or Reports Production
- 16400 is default Sort Activity
 - Begins at position 18 and sorts for 1 byte
 - Goes to position 9 and sorts for 9 bytes of Hogan Key and 30 bytes of user defined fields
- Each report has Processing Control Entry
- Activities divided into 3 sections

© 2000 CSC. All rights reserved

CSC

Summary



- BEGINNING of REPORT Activities = 7627 and 17627
- Activity 1250 issued twice for each Data String
- Activity 7628 issued as END of REPORT activity
- 16000 range builds Line Definition and Format Definition
- Reports produced as separate JOB STEPS or JOBS
- Sub Data String field provides flexibility

© 2000 CSC. All rights reserved



Umbrella Programming

Summary

- The SPS Processing Control program 1698 controls the production of reports.
- The three phases of Reports Production are SORT, EXPLOSION, and OUTPUT.
- The default Sort Activity is 16400. To override the sort, place the override Sort activity ID in Field 6 of data group 1009.
- The 16400 sort begins at position 18 and sorts for 1 byte language code, then at position 9 and sorts for 9 bytes of the Hogan Key and then 30 bytes of the user defined fields.
- Each report has a Processing Control Entry. The entry contains the activities that are issued to produce the report. The entry is keyed by Pass Number and Data String ID.
- The Activities are divided into 3 sections: BEGINNING of REPORT, EACH DATA STRING, and END of REPORT.
- The two standard BEGINNING of REPORT activities are 7627 and 17627.
- Activity 1250 is generally issued twice for each Data String. The first time for the explosion; the second time to format and output the report.
- When activity 1250 is issued, the SPS Processor identifies the SPS Program to invoke by the Report ID, Mnemonic of spaces, and Line Definition equal to the value of WL\$LINE. The WL\$LINE value can be entered on the Processing Control Entry or set prior to the issuance of the 1250 activity.
- Activity 7628 must be issued as the last END of REPORT activity.
- The Line Definition and Format Definitions for Explosion routines must be built using the 16000 range.
- SPS Reports are produced as separate JOB STEPS or JOBS.
- The Sub Data String field in the Hogan key provides flexibility for processing SPS Data Strings.



Umbrella Programming

Summary



28-30

SPS Reports Production

SPS Heading Processing

29

Heading Process in SPS

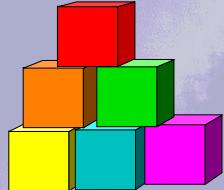
Notes:



Topics



Topics



- Heading Logic
- D Line Definition Command
- Page Overflow Line Definitions
- Special D Command Values
- DEC Format Definition Command
- Number of lines per page

© 2000 CSC. All rights reserved

Notes:



Objectives

CSC

Objectives



- Explain headings logic under SPS
- Use the D Command to control page overflow
- List special values for D Command and explain their use
- Describe use of DEC Command
- Code SPS Heading Routine online printing 20 lines per page

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

Heading Processing

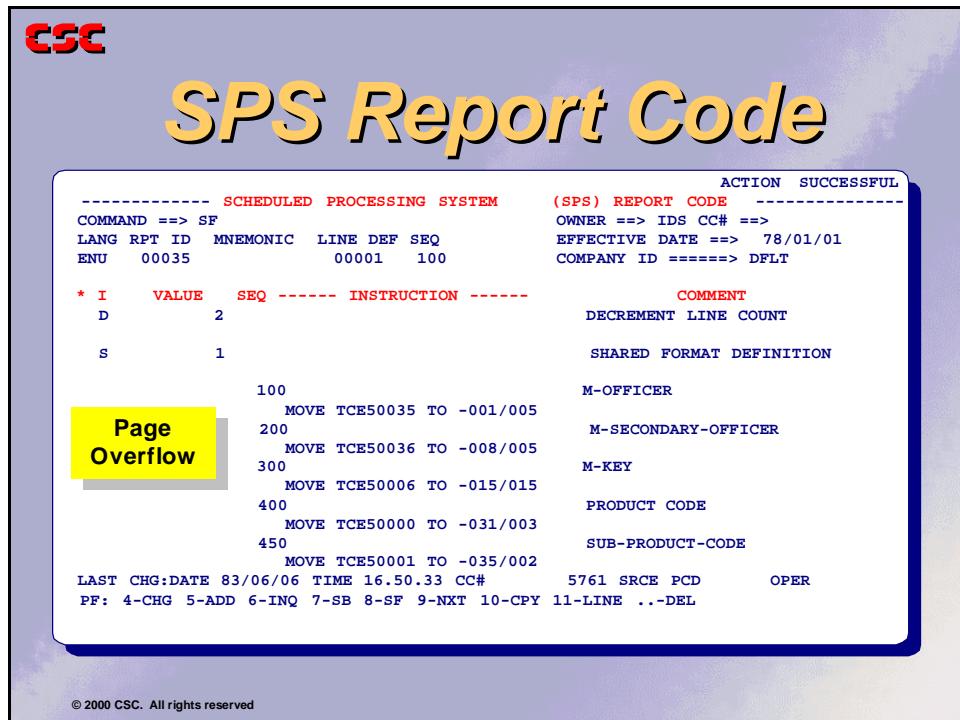
Heading Processing

Page Overflow

Page overflow conditions are controlled in an SPS Line Definition through the use of the D Line Definition Command. This Command decrements a value from an internal line counter, whose default value is 56 lines per page. The SPS Report code issues the D Command and an associated value to reflect the number of lines that have been printed.

When the line counter becomes negative, program 1850 internally issues Link activity 1251. This activity links back to program 1850 and signals that a new page is needed.

In the following SPS detail line print program, the first Line Definition command is a D 2.



The image shows a screenshot of a computer screen displaying an SPS Report Code window. The window has a blue header bar with the CSC logo and the text "SPS Report Code". Below the header is a table with two columns: "SCHEDULED PROCESSING SYSTEM" and "(SPS) REPORT CODE". The table contains several lines of data, including command definitions like "COMMAND ==> SF", "OWNER ==> IDS CC# ==> 1234567890", and "EFFECTIVE DATE ==> 78/01/01". In the middle of the table, there is a yellow box containing the text "Page Overflow". The table ends with a row of numbers (100, 200, 300, 400, 450) followed by their corresponding descriptions: "MOVE TCE50035 TO -001/005", "MOVE TCE50036 TO -008/005", "MOVE TCE50006 TO -015/015", "MOVE TCE50000 TO -031/003", and "MOVE TCE50001 TO -035/002". At the bottom of the table, there is a row of text: "LAST CHG:DATE 83/06/06 TIME 16.50.33 CC# 5761 SRCE PCD OPER" and "PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL". The bottom of the window has a footer with the text "© 2000 CSC. All rights reserved".

SCHEDULED PROCESSING SYSTEM		(SPS) REPORT CODE		
COMMAND ==> SF		OWNER ==> IDS CC# ==>		
LANG RPT ID	MNEMONIC	EFFECTIVE DATE ==> 78/01/01		
ENU 00035		COMPANY ID =====> DFLT		
* I	VALUE	SEQ	INSTRUCTION	COMMENT
D		2		DECREMENT LINE COUNT
S		1		SHARED FORMAT DEFINITION
		100	MOVE TCE50035 TO -001/005	M-OFFICER
		200	MOVE TCE50036 TO -008/005	M-SECONDARY-OFFICER
		300	MOVE TCE50006 TO -015/015	M-KEY
		400	MOVE TCE50000 TO -031/003	PRODUCT CODE
		450	MOVE TCE50001 TO -035/002	SUB-PRODUCT-CODE
LAST CHG:DATE 83/06/06 TIME 16.50.33 CC# 5761 SRCE PCD OPER				
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL				



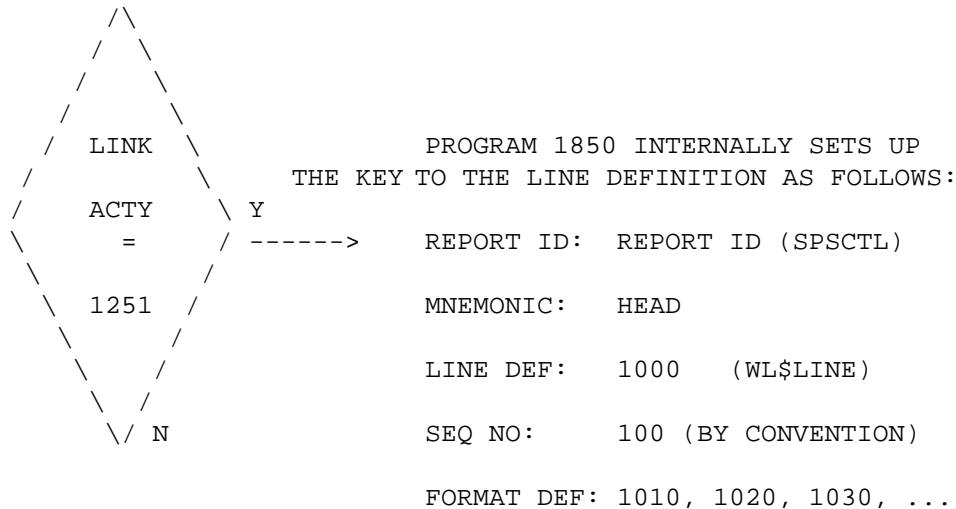
D Line Definition Command

This command is used to control vertical spacing on a report. When SPS executes this command, it performs two functions. The first is to subtract the value of the operand from the internal line counter. The second is to determine if the internal counter is negative.

EXAMPLE: D 00002

The value of 2 is subtracted from the internal line counter. If the result is less than zero, SPS invokes activity 1251 to program 1850. If the result is not negative, processing continues with the next instruction.

Decision Point for Program 1850



Notes:



Umbrella Programming

Heading Processing

Page Overflow Line Definitions

Page headings are produced by SPS Line Definitions. Their Report ID is the same as the report being produced with the mnemonic of HEAD. For example, page heading routines for Report ID 35 would be identified 00035 HEAD.

When SPS automatically invokes these routines, the following numbering standards must be followed. The Line Definition containing the new page processing code must be 1000. Format Definitions within Line Definition 1000 are numbered 1010, 1020, and 1030. Typical processing within these routines includes the following:

- Skipping to channel one, top of page.
- Formatting the report title or heading.
- Incrementing and formatting the page number. PAGENO is a field for accumulating the number of pages.
- Formatting the column headings.
- Decrementing the internal line counter for the number of lines representing the entire page heading.

Notes:



Example Heading Program

In the HEAD routine the report title and column heading text is formatted and printed.

CSC

Heading Program Example

```
----- SCHEDULED PROCESSING SYSTEM -----  
COMMAND ==> SF (SPS) REPORT CODE -----  
LANG RPT ID MNEMONIC LINE DEF SEQ OWNER ==> IDS CC# ==>  
ENU 00035 HEAD 01000 10 EFFECTIVE DATE ==> 78/01/01  
COMPANY ID =====> DFLT  
  
* I VALUE SEQ ----- INSTRUCTION ----- COMMENT  
P 1101 SET PRINT CARRIAGE CONTROL  
  
A 1279 PEM ACTIVITY  
  
P 1 SET PRINT CARRIAGE CONTROL  
  
S 1010 SHARED FORMAT DEFINITION  
  
100 MOVE 'CR0035' TO SAVRPTID  
200 MOVE SAVRPTID TO -001/006  
300 MOVE 'BANK-ID' TO -010/007  
  
LAST CHG:DATE 78/06/07 TIME 9.53.38 CC# SRCE HOGNJIM1 OPER  
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL
```

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

Heading Processing

Generating the Heading

The following logic illustrates the processing flow of SPS when the decrement Line Definition Command is executed.

Note: The generation of a new report resets the internal line counter to zero. The first D Command causes the line counter to be negative. When the Line Counter is negative, the heading routine is executed and the heading for the first page of the report is printed.

1. IS THIS A NEW REPORT?

YES -

RESET THE LINE COUNTER TO 0

2. SUBTRACT THE VALUE IN THE D COMMAND FROM THE LINE COUNTER.

3. IS THE LINE COUNTER NEGATIVE?

YES -

RESET THE LINE COUNTER TO 56 OR LINES PER PAGE ON REPORT PARAMETERS SCREEN

EXECUTE: REPORT ID = REPORT ID (SPSCTL)

MNEMONIC = HEAD

LINE DEFN = 1000

SEQUENCE = 100 (BY CONVENTION)

GO TO STEP 2

4. CONTINUE PROCESSING.

Special Values of the D Command

The D Line Definition Command may be used for special processing.

D 00000

Reinitializes the internal counter to zero.

D X

Where X is greater than the number of lines per page established for the report. Executing this instruction causes page overflow processing routines to be invoked. A common example of this variation of the D Command is D 99.

D 33000

Tests the internal Line Counter for a zero or negative value. If the condition is true, SPS automatically invokes the page header routine for the report. Otherwise, it executes the next sequential instruction.

Note: The value of 33000 is not subtracted from the Line Counter.



DEC Format Definition Command

Another version of decrement exists as a Format Definition Command.

```
DEC (LINECOUNT) (BY) N
```

This instruction causes the current page Line Count to be decremented by the value of N. If the decrement results in a negative line count, the internal Line Counter is reinitialized to zero. The Line Counter, however, is not tested for a negative value. Regardless of the result of the decrement, page overflow processing routines are not automatically invoked.

The DEC Command should be used in conjunction with the D 33000 Command. The D 33000 Command forces the test of the Line Counter and invokes the heading routine if the internal Line Counter is negative or zero.

Notes:



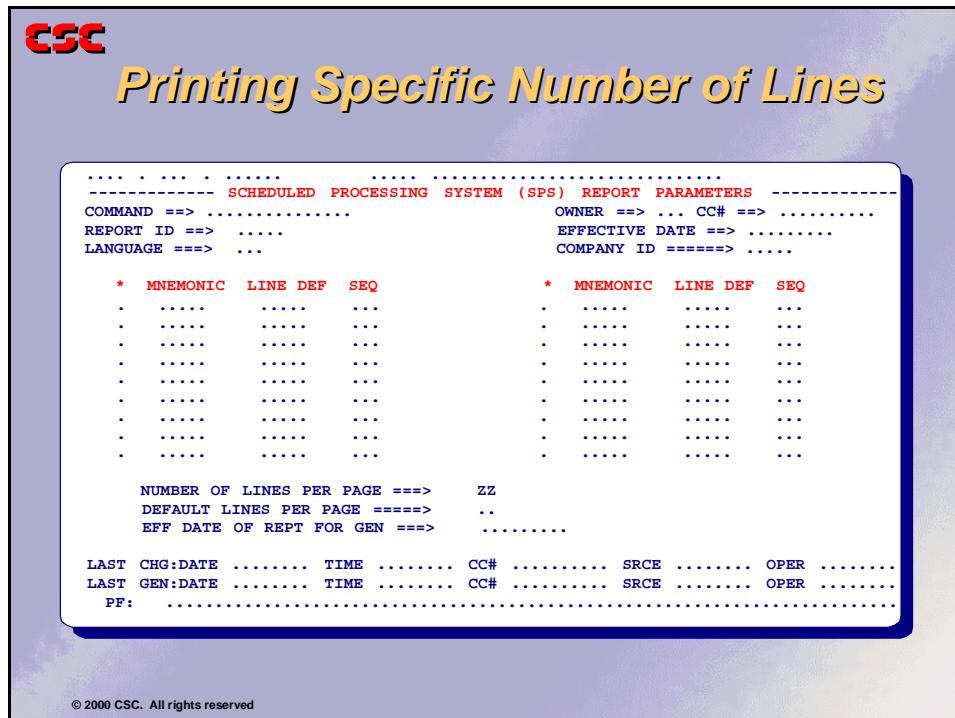
Umbrella Programming

Heading Processing

Printing a Specific Number of Lines

Overriding the default number of lines per page for a report requires modification to the "Scheduled Processing System (SPS) Report Parameters" screen. Note the screen below and the field NUMBER OF LINES PER PAGE.

When a change is made to the number of lines per page, the report must be GENed.



Notes:



Problem Specifications—SPS Head



Each group is to design appropriate page and column headings for the report being produced. This task involves the following:

1. Using the correct Line Definition Command to trigger SPS to issue activity 1251.
2. Add an SPS Line Definition to handle new page processing requirements. Use an Effective Date 780101 and Company DFLT.
3. The report title, include your Group Name and the page number, should print at the top each page. Be sure the first detail line is not printing prior to the first heading for your report.
4. Column heading text for the detail employee data should appear double spaced after the heading.

Notes:



Summary

CSC

Summary

- D Line Definition Command controls page overflow
- Negative internal line counter backs itself with Link activity 1251
- Heading Routines coded as separate SPS Line Definitions
- Special values for D
 - D 0
 - D X where X is a number greater than *lines per page*
 - D 33000

© 2000 CSC. All rights reserved



CSC

Summary

- Lines per page set on SPS Report Parameters screen
- Internal line counter at 0 for new report - 56 otherwise
- Change in number of lines = reGENed report

© 2000 CSC. All rights reserved



Umbrella Programming

Problem Specifications—SPS Head

- Page overflow is controlled by the D Line Definition Command. This command decreases an internal line counter and tests for a negative value.
- When the internal line counter is negative, the SPS Processor links back to itself with the Link activity 1251. This value flags the invocation of a Heading Routine.
- Heading Routines are coded as separate SPS Line Definitions with a Report ID for the report being produced, mnemonic of HEAD, and Line Definition of 1000.
- The three special values for D are:
 - D 0
 - D X where X is a number greater than lines per page
 - D 33000
- The "Scheduled Processing System (SPS) Report Parameters" screen has a field where the number of lines per page is set.
- The internal line counter is set to 0 at the beginning of a new report. Otherwise, it is set to 56 or the number of lines per page on the "Scheduled Processing System (SPS) Report Parameters" screen.
- If the number of lines per page changes, the report must be reGENed.

Notes:



Umbrella Programming

Problem Specifications—SPS Head

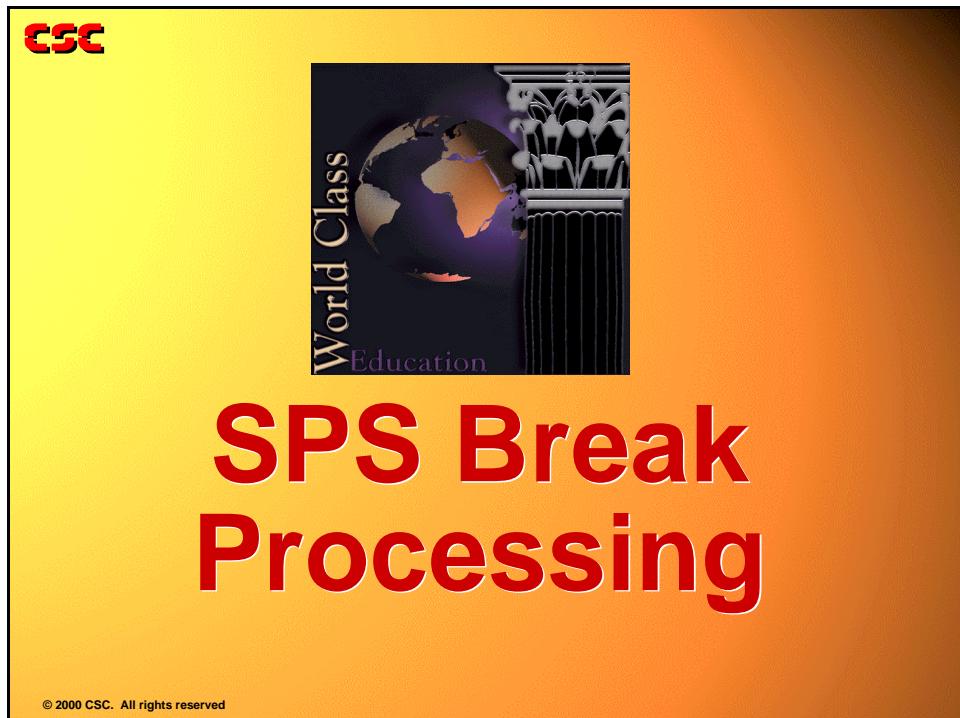


29-14

SPS Heading Processing

SPS Break Processing

30



Notes:

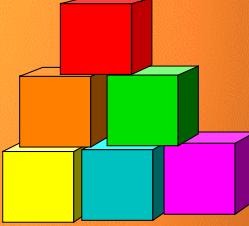


Topics



Topics

- Break Format Definition Command
- Levels within Breaks
- Break Logic
- Processing Flow



© 2000 CSC. All rights reserved

Notes:



Objectives



Objectives



- Explain Break Logic under SPS
- Use Break Format Definition Command
- Describe levels within breaks
- Code SPS break routines online

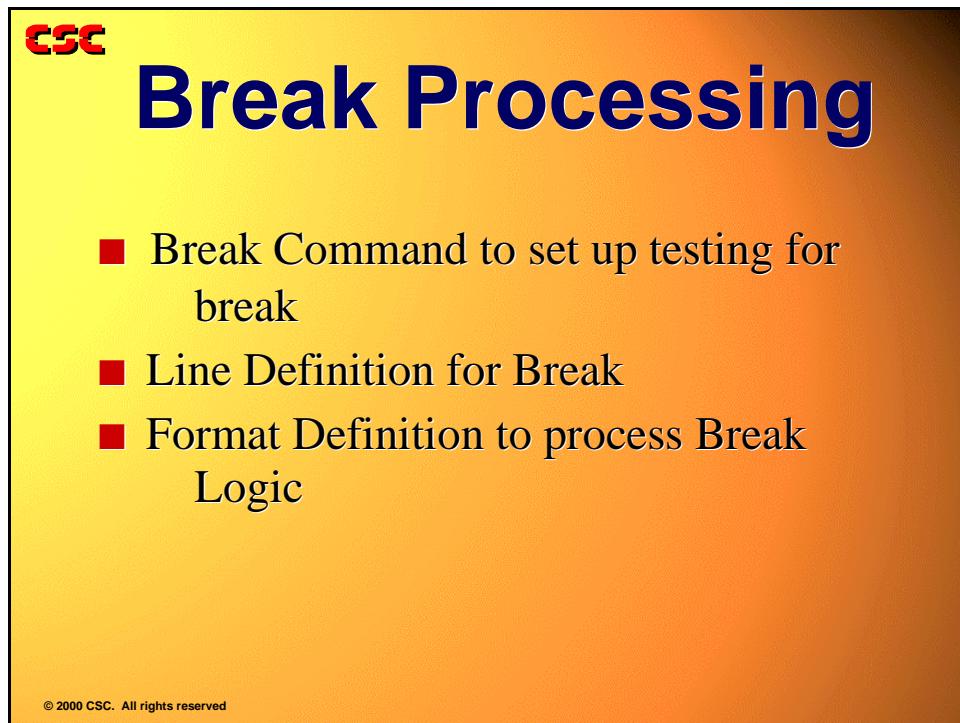
© 2000 CSC. All rights reserved

Notes:



SPS Break Processing

There are three parts to Break Processing:



The slide has a yellow-to-orange gradient background. In the top left corner is the CSC logo. The title 'Break Processing' is centered in large blue text. Below the title is a bulleted list of three items. At the bottom of the slide is a small copyright notice.

CSC

Break Processing

- Break Command to set up testing for break
- Line Definition for Break
- Format Definition to process Break Logic

© 2000 CSC. All rights reserved

Break processing is controlled in an SPS Program by using the Format Definition Command BREAK. This Command prompts SPS to compare a current value in a field to a previous value stored in a save area. User-coded break processing routines are automatically executed when the current value is not equal to the previous value.

The placement of the Break Command within the logic depends on when the data is available. In the example the Break Command is coded at the beginning of the Explosion routine. The data field TLE50107 is BRANCH NUMBER and is defined in the HOGAN KEY portion of the Data String. If the break data field was part of the USER DATA portion of the Data String, the Break Command would have to be coded at the end of the Explosion Routine. The Break Command can also be coded at the beginning of the Detail print routine.

Grand totals are not controlled by the BREAK Command. These totals would be invoked from the SPS Processing Control Entry specified in the End of Report Section.



CSC

Break Processing

SCHEDULED PROCESSING SYSTEM				ACTION SUCCESSFUL	
				(SPS) REPORT CODE	
COMMAND ==> SF				OWNER ==> IDS CC# ==>	
LANG RPT ID	MNEMONIC	LINE DEF	SEQ	EFFECTIVE DATE ==>	78/01/01
ENU 00035		16000	100	COMPANY ID =====>	DFLT
* I	VALUE	SEQ	INSTRUCTION	COMMENT	
S	16001			SHARED FORMAT DEFINITION	
		100	BREAK ON TLE50107(BRKBRCH)	BRANCH NUMBER	
		200	MOVE TLE50103 TO WL\$LINE	LEVEL=10 RPT=0 LID=06700 R-SAMVL-SUB-DATA-STRING-ID	
		300	MOVE -018 TO TCE50015	M-CURR-BAL	
		400	MOVE - TO TCE50051	M-DATE-OD	
		425	MOVE - TO TCE50056	M-DATE-LAST-DR-CR	
		450	MOVE - TO TCE50060	M-DATE-LAST-DEP	
LAST CHG:DATE 78/06/07 TIME 9.53.38 CC#				SRCE HOGNJIM1 OPER	
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ...DEL					

© 2000 CSC. All rights reserved

Notes:



Umbrella Programming

SPS Break Processing

BREAK Format Definition Command

```
BREAK ON INPUT-ELEMENT(OUTPUT-ELEMENT) LEVEL=NNNNN  
LID=NNNNN RPT=NNNNN
```

Where:

INPUT-ELEMENT Is the current value.

OUTPUT-ELEMENT Is the previous value.

LEVEL=NNNNN Used for documentation only unless concatenated elements are required.

LID=NNNNN Line definition containing the break processing logic to be executed if a break occurs.

RPT=NNNNN If the value of NNNNN is 0, the report ID of the current report is combined with the mnemonic break to identify the break routine.

If the value is not 0, the number specified becomes the report ID.

Notes:



Examples of Three Level Breaks

Example 1

```
S nnn
 100
    BREAK ON BRANCH  LEVEL=01 LID=6900 RPT=0
 200
    BREAK ON PRODUCT LEVEL=02 LID=6800 RPT=0
 300
    BREAK ON SUBPROD LEVEL=03 LID=6700 RPT=0
```

These three Break Commands are coded from major to minor levels.

Note: There are no imbedded instructions within these commands. If a control break occurs at a higher level, then subsequent breaks at a minor level(s) are forced by SPS.

When the Break Command is executed and the current value is not equal to the previous value, SPS accesses the Line Definition using the Report ID and mnemonic BREAK as the key.

By convention, Line Definitions containing break processing logic are numbered in the 6000 range. Format Definitions follow the pattern established by the Line Definition. Line Definition 6900 would contain Format Definitions numbered 6910, 6920, and 6930.

Example 2

```
S nnn
 100
    BREAK ON BRANCH(XBRANCH)    LEVEL=01 LID=6900 RPT=0
 200
    BREAK ON PRODUCT(XPRODUCT)  LEVEL=02 LID=6800 RPT=0
 300
    BREAK ON SUBPROD(XSUBPROD)  LEVEL=03 LID=6700 RPT=0
```

The previous value (OUTPUT-ELEMENT) is provided to the user if a save area is referenced in the break statement itself. Both the INPUT-ELEMENT and OUTPUT-ELEMENT must be defined on the Data Dictionary through normal data group maintenance.

First time logic in SPS moves the current value to the save area, and no break checking occurs.



Umbrella Programming

SPS Break Processing

Example 3

```
S nnn
 100
      BREAK ON BRANCH LEVEL=01 LID=6900 RPT=0
 200
      BREAK ON PRODUCT LEVEL=02 LID=6800 RPT=0
 300
      BREAK ON SUBPROD LEVEL=03 LID=6700 RPT=0
 400
      BREAK ON OFFICER LEVEL=03 LID=6700 RPT=0
```

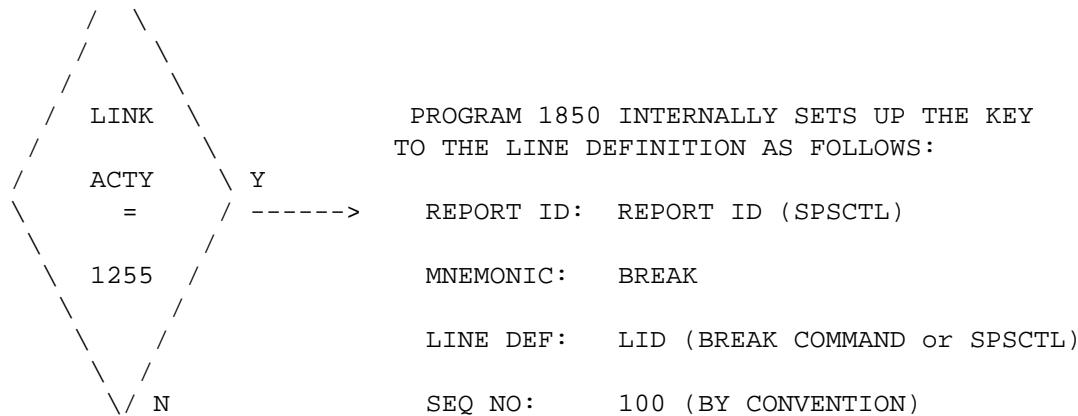
Multiple elements may be concatenated into one level control when the Level Number, Line Definition, and Report Number are the same. This example illustrates the two break statements to code when a change in either the subproduct or officer code is detected by SPS. In each case, Line Definition 6700 is executed.

Notes:



Break Logic

Break routines handle special processing requirements when changes in certain data elements occur. The most common special processing is subtotals and grand totals. For example, a new branch number may indicate a need to print summary totals for the prior branch. In addition, branch level totals could be rolled into company level accumulators. When a break is sensed by the SPS Processor, Link activity ID 1255 is issued to program ID 1850.



Notes:

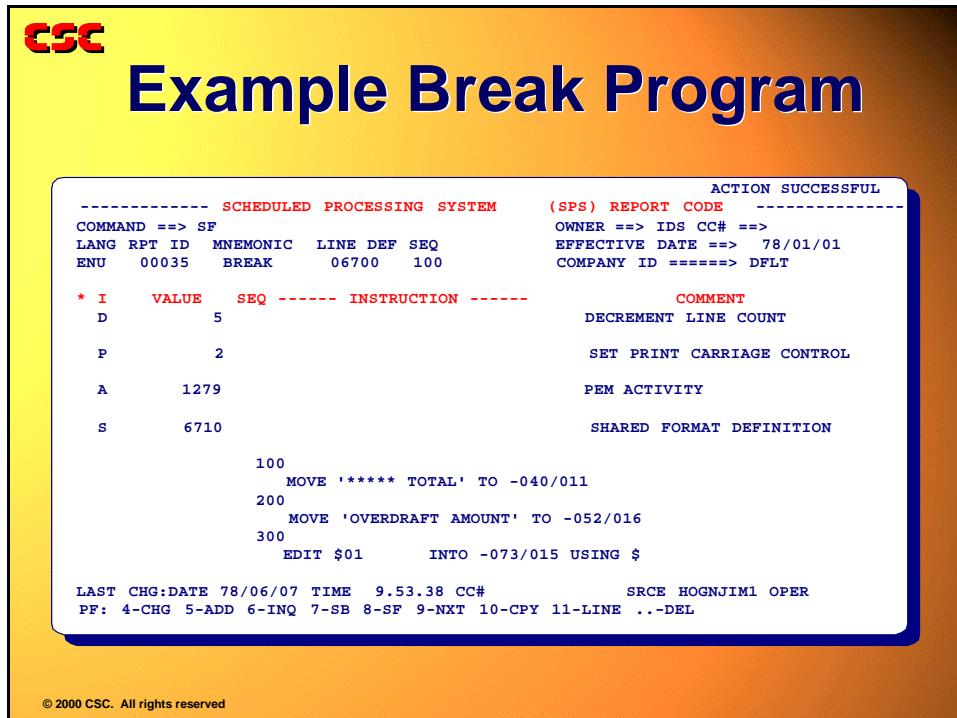


Umbrella Programming

SPS Break Processing

Example Break Program

The Format Definitions within the Break Line Definition follow the numbering convention of the Line Definition plus 10. The first Format Definition is 6710, the second would be 6720. The logic within the Break Line Definition would be to format and print totals, roll the totals to a higher level, and clear the current level totals.



The screenshot shows a CSC application window titled "Example Break Program". The window displays a report from the "SCHEDULED PROCESSING SYSTEM". The report header includes fields like COMMAND, LANGUAGE, REPORT ID, MNEMONIC, LINE DEF, and SEQ. It also shows ownership information (OWNER, EFFECTIVE DATE, COMPANY ID) and a status message "ACTION SUCCESSFUL". The main body of the report contains several sections: a table of instructions with columns for * I, VALUE, SEQ, INSTRUCTION, and COMMENT; a series of MOVE statements; and a summary line at the bottom. The footer of the window includes a copyright notice: "© 2000 CSC. All rights reserved".

```
ACTION SUCCESSFUL
----- SCHEDULED PROCESSING SYSTEM (SPS) REPORT CODE -----
COMMAND ==> SF OWNER ==> IDS CC# ==>
LANG RPT ID MNEMONIC LINE DEF SEQ EFFECTIVE DATE ==> 78/01/01
ENU 00035 BREAK 06700 100 COMPANY ID =====> DFLT

* I     VALUE   SEQ ----- INSTRUCTION ----- COMMENT
D       5
P       2
A       1279
S       6710
DECREMENT LINE COUNT
SET PRINT CARRIAGE CONTROL
PEM ACTIVITY
SHARED FORMAT DEFINITION

100
MOVE '***** TOTAL' TO -040/011
200
MOVE 'OVERDRAFT AMOUNT' TO -052/016
300
EDIT $01      INTO -073/015 USING $

LAST CHG:DATE 78/06/07 TIME 9.53.38 CC#           SRCE HOGNJIM1 OPER
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ..-DEL

© 2000 CSC. All rights reserved
```

Notes:



Accumulations for the Break

If totals are to be printed in a Break Line Definition, the total values must be accumulated when the data is available. In the following detail print line process, fields are being added to the SPS accumulation fields /01, \$01 and \$02.

CSC

Accumulations for Break

----- SCHEDULED PROCESSING SYSTEM -----				ACTION SUCCESSFUL	
COMMAND ==>	SF			(SPS) REPORT CODE	-----
LANG RPT ID	MNEMONIC	LINE DEF	SEQ	OWNER ==> IDS CC# ==>	
ENU 00035		00001	100	EFFECTIVE DATE ==>	78/01/01
* I VALUE SEQ ----- INSTRUCTION -----				COMPANY ID =====>	DFLT
		900			
			ADD TCE50015 TO \$01	COMMENT	
		1000		M-CURR-BAL	
			ADD *1 TO /01		
		1100		W819-OD-GLOBAL-AVAIL-BAL	
			ADD IDS81901 TO \$02		
P	1			SET PRINT CARRIAGE CONTROL	
A	1279			PEM ACTIVITY	
S	2	100		M-SORT-NAME	
			MOVE TCE50004 TO -015/025		
LAST CHG:DATE 89/07/25 TIME 7.59.31 CC#				SRCE PCD	OPER KSP
PF: 4-CHG 5-ADD 6-INQ 7-SB 8-SF 9-NXT 10-CPY 11-LINE ...DEL					

© 2000 CSC. All rights reserved

Notes:

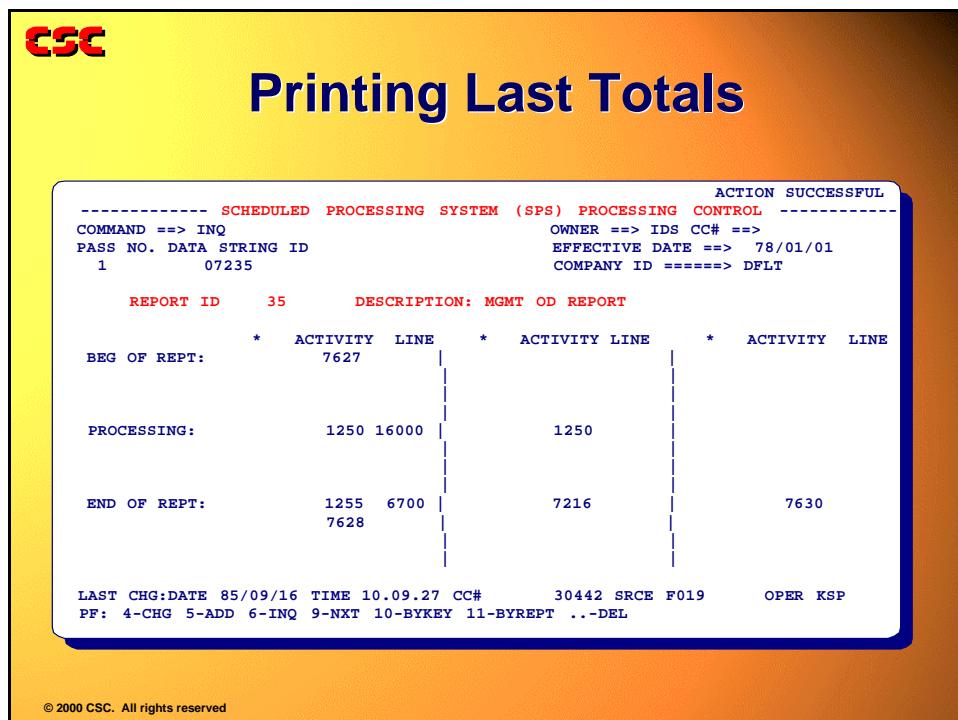


Umbrella Programming

SPS Break Processing

Printing the Last Set of Totals and Grand Totals

The BREAK Line Definition Command will drive the your process to print the Break routine when there is a change in the data value(s) specified in the Break Command. However, when there is a change in the Report ID or end of file on the SPS Data String file is detected, SPS starts processing the End of Report activities. This is where you need to specify the Break routine(s) to print the last total set and grand totals. The 1255 activity ID with the corresponding Line ID causes the SPS Break routines to be processed. The order of the 1255/Line ID must be specified in the order you wish the totals to be printed.



Notes:



Problem Specifications—SPS Break



The report so far contains details and headers. In the exercise code needs to be added to your SPS report to test, accumulate and print department totals. This task involves the following:

1. Accumulate the detail amount and quantity fields. The /nn and \$nn fields defined in the SET Format Definition Command can be used.
2. Add the BREAK Format Definition Command to print department totals when a break on department number is detected. One of the WRK. fields can be used to store the value of the department.
3. The printing of the totals should be two space lines after the last detail line for the department being totaled. The break should include the department number, total number of employees, total current earnings and total year-to-date earnings for each department. The totals lines should appear as follows:

DEPARTMENT XXX TOTALS - EMPLOYEES	ZZZZ9
- CUR EARNINGS	\$ \$\$, \$\$, \$\$9.99
- YTD EARNINGS	\$ \$\$, \$\$, \$\$9.99

4. Start the next department detail on a new page with a page number of 1.
5. A modification to your SPS Processing Control Entry (SPSCTL) must be made in order to print the last department total at the End of the Report.

Notes:



Summary



Summary



- SPS handles processing needs with changes of categories
- SPS breaks on the field(s)
- Separate SPS Programs for printing
- Multiple levels of breaks can be processed
- Break Command can save previous value
- Invoking Break routines for final categories and grand totals handled in End of Report area

© 2000 CSC. All rights reserved

- SPS handles processing needs when changes in categories of data occur through break processing.
- SPS breaks on the field(s) through the processing of the BREAK Format Definition Command.
- There are separate SPS Programs to print out the break information for each break field and the grand total.
- Multiple levels of breaks can be processed. Multiple BREAK Commands are required. They are listed from major to minor levels. Multiple Break routines are needed.
- The BREAK Command can save the previous value for the data.
- Invoking the Break routines for the final categories and grand totals is handled in the End of Report area of the SPS Processing Control Entry.



Summary—Product Usage

A

This summary provides a quick reference for individual product standards.

The following sections identify, by system, the Umbrella components used by delivered products.

Umbrella System (UMB) Component Identification

The components used by the Umbrella System and its subsystems are on the following pages.

CCP—Condition Code Processing System

Component Identifier	Identification Range
Family	UMB
Ownership	CCP
Application	48
Activities	1350–1399, 5000–5099
ASSEMBLER Macros	U48000E–U48999E, T58500E–T58599E
COBOL Copybooks—Data Division	U48000D–U48999D, T48400D–T58599D
COBOL Copybooks—Procedure Division	U48000P–U48999P, T58000P–T58999P
Data Elements	TGE58500–TGE58999
Data Groups	1300–1349
Maps	U48000M–U48999M, T58500M–T58599M
PCDs/Formats	1300–1349
Program Source	U48000–U48999, T58500–T58599
Programs	1200–1399
Reports	UR0000–UR9999
Transactions—CICS/IMS	UAAA–U999



Umbrella Programming

Umbrella System (UMB) Component Identification

CCS—Change Control System

Component Identifier	Identification Range
Family	UMB
Ownership	CCS
Application	59
Activities	5300–5499
ASSEMBLER Macros	R59001E–R59999E
COBOL Copybooks—Data Division	R59000D–R59999D
COBOL Copybooks—Procedure Division	R59000P–R59999P
Data Elements	TRE59000–TRE59999
Data Groups	5400–5412
Maps	R59000M–R59999M
PCDs/Formats	5400–5412
Program Source	R59001–R59999
Programs	5400–5499, 48831, and 48832
Reports	RR0000–RR9999
Transactions—CICS/IMS	RAAA–R999



DTS—Date Services System

Component Identifier	Identification Range
Family	UMB
Ownership	DTS
Application	48
Activities	1900–1949
ASSEMBLER Macros	U48001E–U48999E, T58001E–T58049E
COBOL Copybooks—Data Division	U48000D–U48999D, T58001D–T58049D
COBOL Copybooks—Procedure Division	U48000D–U48999D, T58001P–T58049P
Data Elements	THE48000–THE48999
Data Groups	2000–2049
Maps	U48000M–U48999M, T58001M–T58049M
PCDs/Formats	2000–2049
Program Source	U48000–U48999, T58001–T58049
Programs	1600–1649
Reports	UR0000–UR9999
Transactions—CICS/IMS	UAAA–U999



Umbrella Programming

Umbrella System (UMB) Component Identification

EFS—Edit and Format System

Component Identifier	Identification Range
Family	UMB
Ownership	EFS
Application	55
Activities	1015–1049, 1051, 1140–1199
ASSEMBLER Macros	U48000E–U48999E, I55001E–I55999E
COBOL Copybooks—Data Division	U48000D–U48999D, I55000D–I55999D
COBOL Copybooks—Procedure Division	U48000D–U48999D, I55000P–I55999P
Data Elements	TWE55000–TWE55999
Data Groups	1050–1099, 1113–1199
Maps	U55000M–U55999M
PCDs/Formats	1100–1199
Program Source	U55000–U55999, I55100–I55800
Programs	1030–1049
Reports	UR0000–UR9999
Transactions—CICS/IMS	UAAA–U999



FPS—Function Processing System

Component Identifier	Identification Range
Family	UMB
Ownership	FPS
Application	53
Activities	13551–13599
ASSEMBLER Macros	U48000E–U48999E
COBOL Copybooks—Data Division	U48000D–U48999D, I53000D–I53999D
COBOL Copybooks—Procedure Division	U48000P–U84999P, I53000P–I53999P
Data Elements	TIE5300–TIE55999
Data Groups	13551–13599
Maps	U53000M–U53999M, I53000M–I53999M
PCDs/Formats	1680–1699, 13551–13599
Program Source	U53000–U53999, I53000–I53999
Programs	13551–13599
Reports	UR0000–UR9999
Transactions—CICS/IMS	UAAA–U999



Umbrella Programming

Umbrella System (UMB) Component Identification

PCD—Process Control Data

Component Identifier	Identification Range
Family	UMB
Ownership	PCD
Application	57
Activities	1000–1029, 1500–1549
ASSEMBLER Macros	D57000E–D57999E, I57000E–I57999E
COBOL Copybooks—Data Division	U48000D–U48999D, I57101D–I57105D
COBOL Copybooks—Procedure Division	U48000P–U48999P, I57101P–I57105P
Data Elements	PCDAAAAA–PCD99999, TWE57100–TWE57999
Data Groups	1450–1499
Maps	D57000M–D57999M, I57100M–I57299M
PCDs/Formats	None
Program Source	D57000–D57999, I57100–I57199
Programs	1000–1029
Reports	DR0000–DR9999
Transactions—CICS/IMS	DAAA–D999



PEM—Processing Environment Manager

Component Identifier	Identification Range
Family	UMB
Ownership	PEM
Application	49
Activities	49000–49999
ASSEMBLER Macros	P49000E–P49999E
COBOL Copybooks—Data Division	P49000D–P49999D
COBOL Copybooks—Procedure Division	P49000P–P49999P
Data Groups	49000–49999
Maps	P49000M–P49999M
PCDs/Formats	49000–49999
Program Source	P49000–P49999
Programs	49000–49999
Reports	PR0000–PR9999
Transactions—CICS/IMS	PAAA–P999



Umbrella Programming

Umbrella System (UMB) Component Identification

RCS—Release Control System

Component Identifier	Identification Range
Family	UMB
Ownership	RCS
Application	48, 60
Activities	48623–48625, 48701–48799, 48835–48882, 48938
ASSEMBLER Macros	R60031E–R60999E, U48700E–U48882E
COBOL Copybooks—Data Division	U48700D–U48882D
COBOL Copybooks—Procedure Division	Not used
Data Elements	Not used
Data Groups	48211, 48377, 48378, 48428–48431, 48622–48623, 48700–48716, 48723, 48725, 48751, 48770–48799, 48831, 48835, 48836, 48840–48856, 48860–48879
Maps	Not used
PCDs/Formats	48790, 48840–48855
Program Source	R60000–R60999
Programs	31, 48210, 48211, 48700–48704, 48770–48799, 48835–48882
Reports	UR0000–UR9999
Transactions—CICS/IMS	None



SPS—Scheduled Processing System

Component Identifier	Identification Range
Family	UMB
Ownership	SPS
Application	54
Activities	1200–1299, 1400–1499, 1800–1899
ASSEMBLER Macros	U48000E–U48999E, I54200E–I54399E
COBOL Copybooks—Data Division	U48000D–U48999D, I54200D–I54399D
COBOL Copybooks—Procedure Division	U48000P–U48999P, I54200P–I54399P
Data Elements	TXE54001–TXE54999
Data Groups	1350–1399, 2050–2099, 2300–2399, 2600, 2650–2799
Maps	U54000M–U54999M, I54200M–I54399M
PCDs/Formats	2300–2399
Program Source	U54000–U54999, I54200–I54399
Programs	1400–1499, 1650–1899
Reports	UR0000–UR9999
Transactions—CICS/IMS	UAAA–U999



Umbrella Programming

Umbrella System (UMB) Component Identification

UMB—Umbrella

Component Identifier	Identification Range
Family	UMB
Ownership	Various
Application	48
Activities	48000–49999, 488000endash.499999
ASSEMBLER Macros	U48000E–U49999E
COBOL Copybooks—Data Division	U48000D–U49999D
COBOL Copybooks—Procedure Division	U48000P–U49999P
Condition Codes	Reserved ranges Future use, 2000–2999, 4000–4999, 6000–6999, 7000–7999, 8000–8999, 9000–9999, 18000–20999, 24000–24999
Data Elements	UAAAAAAA–UXX99999
Data Groups	48000–49999
Maps	U48000M–U49999M
PCDs/Formats	48000–49999
Program Source	U48000–U49999
Programs	48000–49999
Reports	None
Transactions—CICS/IMS	U–U999



Budget and Planning System (BPS) Component Identification

The following are the components used by the Budget and Planning System.

BPS—Budget and Planning System (BPS)

Component Identifier	Identification Range
Family	BPS
Application	270
Activities	270000–279999
ASSEMBLER Macros	B270000E–B279999E
COBOL Copybooks—Data Division	B270000D–B279999D
COBOL Copybooks—Procedure Division	None
Condition Codes	Application Code of 270, 1–49999, 60000–65535
Data Elements	None
Data Groups	270000–279999
Maps	None
PCDs/Formats	None
Program Source	B270000–B279999
Programs	270000–279999
Reports	None
Transactions—CICS/IMS	None



Umbrella Programming

Consumer and Merchant Servicing System (CAMS) Component Identification

Consumer and Merchant Servicing System (CAMS) Component Identification

The following are the components used by the Consumer and Merchant Servicing System.

TCS—Consumer and Merchant Servicing System

Component Identifier	Identification Range
Family	TCS
Application	90
Activities	45000–45999, 100000–119999
ASSEMBLER Macros	Q00000E–Q19999E
COBOL Copybooks—Data Division	Q00000D–Q19999D
COBOL Copybooks—Procedure Division	Not used
Condition Codes	Application Code of 90, 1–49999, 60000–65535
Data Elements	Q00000AA–Q9999999
Data Groups	100000–119999
Maps	Q00000M–Q19999M
PCDs/Formats	106000–106999
Program Source	Q00000–Q19999
Programs	100000–119999
Reports	QR15000–QR15999
Transactions—CICS/IMS	TCS, TCSC, TINT



Credit Risk System (CRS) Component Identification

The following are the components used by the Credit Risk System.

RAS—Credit Risk System (CRS)

Component Identifier	Identification Range
Family	RAS
Application	250
Activities	250000–259999
ASSEMBLER Macros	R250000E–R259999E
COBOL Copybooks—Data Division	R250000D–R259999D
COBOL Copybooks—Procedure Division	Not used
Condition Codes	Application Code of 250, 1–49999, 60000–65535
Data Elements	None
Data Groups	250000–259999
Maps	None
PCDs/Formats	None
Program Source	R250000–R259999
Programs	250000–259999
Reports	None
Transactions—CICS/IMS	None



Umbrella Programming

Earnings Analysis System (EAS) Component Identification

Earnings Analysis System (EAS) Component Identification

The following are the components used by the Earnings Analysis System.

EAS—Earnings Analysis System

Component Identifier	Identification Range
Family	EAS
Application	24, 40
Activities	240000–241999
ASSEMBLER Macros	Y40000E–Y41999E
COBOL Copybooks—Data Division	Y40000D–Y41999D
COBOL Copybooks—Procedure Division	Not used
Condition Codes	Application code of 24, 1–49999, 60000–65535
Data Elements	Y41000AA–Y419999
Data Groups	240000–241999
Maps	Y40000M–Y41999M
PCDs/Formats	None
Program Source	Y40000–Y41999
Programs	240000–241999
Reports	YR24000–Y241999
Transactions—CICS/IMS	YAAA–Y999



Financial Information System (FIS) Component Identification

The following are the components used by the Financial Information System.

FIS—Financial Information System

Component Identifier	Identification Range
Family	FIS
Application	36
Activities	36100–36999, 600000–699999
ASSEMBLER Macros	F60000E–F69999E
COBOL Copybooks—Data Division	F60000D–F69999D
COBOL Copybooks—Procedure Division	F60000P–F69999P
Data Elements	FAA00001–FZZ99999
Data Groups	600000–699999
Maps	F60000M–F69999M
PCDs/Formats	600000–699999
Program Source	F36500–F36999, F600000–F699999
Programs	36500–36999, 600000–699999
Reports	FR0000–FR9999
Transactions—CICS/IMS	FAAA–F999



Umbrella Programming

Financial Support System (FSS) Component Identification

Financial Support System (FSS) Component Identification

The following are the components used by the Financial Support System and its subsystems.

CDS—Cycle Data System

Component Identifier	Identification Range
Family	FSS
Ownership	CDS
Application	51
Activities	8800–8950
ASSEMBLER Macros	I51000E–I51999E
COBOL Copybooks—Data Division	I51000D–I51999D
COBOL Copybooks—Procedure Division	I51000D–I51999D
Data Elements	TBE51000–TBE51999
Data Groups	3602, 8800–8950
Maps	I51100M–I51999M
PCDs/Formats	8800–8899
Program Source	I51000–I51999
Programs	8800–8950
Reports	SR0000–SR9999
Transactions—CICS/IMS	KAAA–K999



CWS—Check Writing System

Component Identifier	Identification Range
Family	FSS
Ownership	CWS
Application	58
Activities	5900–5999
ASSEMBLER Macros	I58000E–I58999E
COBOL Copybooks—Data Division	I58000D–I58999D
COBOL Copybooks—Procedure Division	I58000P–I58999P
Data Elements	ICW000A1–ICW99999
Data Groups	5900–5999
Maps	I58000M–I58999M
PCDs/Formats	None
Program Source	I58000–I58999
Programs	5900–5999
Reports	IR0000–IR9999
Transactions—CICS/IMS	IAAA–I999



Umbrella Programming

Financial Support System (FSS) Component Identification

G/L—General Ledger Interface System

Component Identifier	Identification Range
Family	FSS
Ownership	G/L
Application	36
Activities	5500–5599, 36000–36099
ASSEMBLER Macros	G36000E–G36999E
COBOL Copybooks—Data Division	G36000D–G36499D, G50500D–G50599D
COBOL Copybooks—Procedure Division	G36000D–G36499D, G50500P–G50599P
Data Elements	TIC37000–TIC37999
Data Groups	5500–5599, 36000–36499
Maps	G36000M–G36499M, G50500M–G50506M
PCDs/Formats	36000–36499, 5500–5599
Program Source	G36000–G36499, G50001–G50011
Programs	5500–5599, 36000–36499
Reports	GR0000–GR9999
Transactions—CICS/IMS	GAAA–G999



NAF—Name and Address System

Component Identifier	Identification Range
Family	FSS
Ownership	NAF
Application	33
Activities	1979–1998, 2260, 2301–2323
ASSEMBLER Macros	I33000E–I33999E
COBOL Copybooks—Data Division	I33000D–I33999D, T50600D–T50999D
COBOL Copybooks—Procedure Division	I33000D–I33999D, T50600P–T50999P
Data Elements	TTE50100–TTE50500
Data Groups	3311–3322, 3430
Maps	I33000M–I33999M, T50300M–T50329M
PCDs/Formats	None
Program Source	I33000–I33999, T50252, T50300–T50302, T50315–T50324, T50920–T50925, U50330, U50550
Programs	2316–2325
Reports	SR0000–SR9999
Transactions—CICS/IMS	NAAA–N999



Umbrella Programming

Financial Support System (FSS) Component Identification

OTP—Outstanding Transaction Processing and NSF—Insufficient Funds Processing

Component Identifier	Identification Range
Family	FSS
Ownership	OTP
Application	45 (OTP), 95 (NSF)
Activities	9200–9399
ASSEMBLER Macros	I45000E–I45999E, I95000E–I95999E
COBOL Copybooks—Data Division	I45000D–I45999D, I95000D–I95999D
COBOL Copybooks—Procedure Division	I45000D–I45999D, I95000P–I95999P
Data Elements	TGE45000–TGE45999
Data Groups	9200–9299
Maps	I45000M–I45999M
PCDs/Formats	9200–9299
Program Source	I45000–I45999
Programs	9200–9299
Reports	SR0000–SR9999
Transactions—CICS/IMS	NAAA–N999, OAAA–O999



Umbrella Programming
Financial Support System (FSS) Component Identification

RST—Restraints System

Component Identifier	Identification Range
Family	FSS
Ownership	RST
Application	46
Activities	5600–5699
ASSEMBLER Macros	I46000E–I46999E
COBOL Copybooks—Data Division	I46000D–I46999D, T58056D, T58067D, T58600D–T58999D
COBOL Copybooks—Procedure Division	I46000P–I46999P, T58056P, T58067P, T58600P–T58999P
Data Elements	TGE48000–TGE48999
Data Groups	1400–1499, 5800–5899
Maps	I46000M–I46999M, T58650M–T58699M
PCDs/Formats	5800–5899
Program Source	I46000–I46999, T58600–T58699
Programs	5600–5699
Reports	SR0000–SR9999
Transactions—CICS/IMS	RAAA–R999



Umbrella Programming

Financial Support System (FSS) Component Identification

SIM—System Integration Management System

Component Identifier	Identification Range
Family	FSS
Ownership	SIM
Application	38
Activities	38000–38999
ASSEMBLER Macros	I38000E–I38999E
COBOL Copybooks—Data Division	I38000D–I38999D
COBOL Copybooks—Procedure Division	I38000P–I38999P
Data Elements	SIMAAAAA–SIM99999
Data Groups	38000–38999
Maps	I38000M–I38999M
PCDs/Formats	38000–38999
Program Source	I38000–I38999
Programs	38000–38999
Reports	IR0000–IR9999
Transactions—CICS/IMS	IAAA–I999



Hogan Education Center Component Identification

The following are the components used by the Hogan Education Center.

HEC—Hogan Education Center

Component Identifier	Identification Range
Family	HEC
Application	99
Activities	47000–47999
ASSEMBLER Macros	Z47000E–Z47999E
COBOL Copybooks—Data Division	Z47000D–Z47999D
COBOL Copybooks—Procedure Division	Z47000P–Z47999P
Data Elements	Z4700001–Z4799999
Data Groups	47000–47999
Maps	Z47000M–Z47999M
PCDs/Formats	47000–47999
Program Source	Z47000–Z47999
Programs	47000–47999
Reports	ZR0000–ZR9999
Transactions—CICS/IMS	ZAAA–Z999



Deposits System Component Identification

The following are the components used by the Deposits System and its subsystems.



DDA—Demand Deposit Application

Component Identifier	Identification Range
Family	IDS
Ownership	DDA
Application	42
Activities	3000–4999, 7200–7299, 7600–7799, 200000–204999
The following are used for the DDA User Data Group and must not be used for development: 3128, 3328, 4028, 4128, 4228, 4328, 4428, and 4828.	ASSEMBLER Macros
C42000E–C42999E	COBOL Copybooks—Data Division
C42000D–C42999D, C50000D–C50999D	COBOL Copybooks—Procedure Division
C42000P–C42999P, C50000P–C50999P	Condition Codes
42000–42999	Data Elements
TCE50001–TCE50999	Data Groups
1200–1299	Maps
C42000M–C42999M, C50000M–C50999M	PCDs/Formats
1601–1699, 2500–2599	Program Source
C42000–C42999, C50000–C50999	Programs
3000–3999	Reports
CR0000–CR9999	Transactions—CICS/IMS



Umbrella Programming

Deposits System Component Identification

Component Identifier	Identification Range
BAAA-B999, DDAA-DD99	

DEP—Dynamic Environment Processing System

Component Identifier	Identification Range
Family	IDS
Ownership	DEP
Application	47
Activities	8000–8099
ASSEMBLER Macros	B47000E–B47999E
COBOL Copybooks— Data Division	B47000D–B47999D, I47000D–I47999D
COBOL Copybooks— Procedure Division	B47000P–B47999P, I47000P–I47999P
Data Elements	TBE52000–TBE52999
Data Groups	8000–8049
Maps	B47000M–B47999M, I47000M–I47999M
PCDs/Formats	8000–8049
Program Source	B47000–B47999, I47000–I47999
Programs	8000–8049
Reports	BR0000–BR9999
Transactions— CICS/IMS	BAAA–B999



IDS—Deposits System

Component Identifier	Identification Range
Family	IDS
Application	34,36,40,42,44,47,56,60,61,99
Activities	2000–4999, 6000–8999, 11000–14999, 200000–209999
ASSEMBLER Macros	B00000E–B99999E, T50000E–T50999E, U50000E–U50999E
COBOL Copybooks—Data Division	B00000D–B99999D, T50000D–T50999D, U50000D–U50999D
COBOL Copybooks—Procedure Division	B00000P–B99999P, T50000P–T50999P, U50000P–U50999P
Condition Codes	13000–13999, 30000–30999, 38000–38400, 40000–40999, 41000–41999, 42000–42999, 43000–43999, 44000–47999, 50000–50999, 51000–51100, 56000–56999, 60000–61999, 63000–63999
Data Elements	IDSAAAAA–IDS99999, TAANNNNN– TXX50NNN
Data Groups	1200–1299, 1600–1699, 2400–4999, 7000– 8999, 11000–13499
Maps	B00000M–B99999M, T50000M–T50999M, U50000M–U50999M
PCDs/Formats	1600–1699, 2500–2599
Program Source	B00000–B99999, T50000–T50999, U50000– U50999
Programs	2000–4999, 13000–14000, 14700–14758
Reports	BR0000–BR9999
Transactions—CICS/IMS	BAAA–B999, IDSP, IDS, IDST



Umbrella Programming

Deposits System Component Identification

LOC—Line of Credit System

Component Identifier	Identification Range
Family	IDS
Ownership	LOC
Application	44
Activities	2030–2096, 2261–2284, 2500–2586, 3030–3039, 3430–3459, 3530–3559, 3729–3759, 3830–3859, 3930–3959
ASSEMBLER Macros	K44000E–K44999E
COBOL Copybooks—Data Division	K44000D–K44999D, T50031D–T50059D
COBOL Copybooks—Procedure Division	K44000P–K44999P
Data Elements	TKE50000–TKE50999
Data Groups	1231–1259
Maps	K44000M–K44999M, K50001M–K50999M
PCDs/Formats	None
Program Source	K44000–K44999, K50000–K50999
Programs	3030–3039, 3430–3459, 3530–3559, 3729–3759, 3830–3859, 3930–3959
Reports	KR0000–KR9999
Transactions—CICS/IMS	KAAA–K999



PPS—Priority Post System

Component Identifier	Identification Range
Family	IDS
Ownership	PPS
Application	37
Activities	8100–8120
ASSEMBLER Macros	B37000E–B37999E, I37000E–I37999E
COBOL Copybooks—Data Division	B37000D–B37999D, I37000D–I37999D
COBOL Copybooks—Procedure Division	B37000P–B37999P, I37000P–I37999P
Data Elements	TBE54001–TBE54020
Data Groups	8100–8199
Maps	B37000M–B37999M
PCDs/Formats	8100–8199
Program Source	B37000–B37999, I37000, I37011
Programs	8100–8189
Reports	BR0000–BR9999
Transactions—CICS/IMS	BAAA–B999



Umbrella Programming

Deposits System Component Identification

TDA—Time Deposit Application

Component Identifier	Identification Range
Family	IDS
Ownership	TDA
Application	40
Activities	7300–7399, 7500–7599, 7800–7999, 11001–12999, 205000–209999 Note: The following are used for the TDA User Data Group and must not be used for development: 11639, 11689, 11739, 11839, 11939, 12039, 12139, 12239, 12539, 12689, 12739, 12789, 12839, and 12939.
ASSEMBLER Macros	S40000E–S40999E
COBOL Copybooks—Data Division	S40000D–S40999D, T50000D–T50199D
COBOL Copybooks—Procedure Division	S40000P–S40999P
Condition Codes	40000–40999
Data Elements	TDE50001–TDE50999
Data Groups	3500–3599
Maps	S40000M–S40999M, S50000M–S50999M
PCDs/Formats	3500–3599
Program Source	S40000–S40999, S50000–S50999
Programs	11001–11999
Reports	SR0000–SR9999
Transactions—CICS/IMS	TDAA–TD99



Loans System Component Identification

The following are the components used by the Loans System.

ILP—Loans System

Component Identifier	Identification Range
Family	ILP
Application	10,17,20,26–32,35,39,41,43
Activities	10000–10999, 17000–17999, 20000–20999, 26000–28999, 30000–30499, 31000–32999, 35000–35999, 39000–43999, 46000–46999, 210000–219999
ASSEMBLER Macros	L10000E–L10999E
COBOL Copybooks—Data Division	L10000D–L10999D, L17000D–L17999D, L20000D–L20999D, L26000D–L30499D, L31000D–L31999D, L35000D–L35999D, L41000D–L41999D, L43000D–L43999D, L46000D–L46999D
COBOL Copybooks—Procedure Division	L10000P–L10999P, L17000P–L17999P, L20000P–L20999P, L26000P–L30499P, L31000P–L31999P, L35000P–L35999P, L41000P–L41999P, L43000P–L43999P, L46000P–L46999P
Data Elements	ILPAAAAA–ILP99999, (or prefixes by ownership as follows), LBR000A1–LBR69999, LCA000A1–LCA69999, LCL000A1– LCL69999, LCO000A1–LCO69999, LIL000A1–LIL69999, LIR000A1–LIR69999, LMA000A1–LMA69999, LML000A1– LML69999, LPY000A1–LPY69999, LSM000A1–LSM69999, LSP000A1– LSP69999, LXL000A1–LXL69999
Data Groups	10000–10999, 17000–17999, 20000–20999, 26000–30499, 31000–31999, 35000–35999, 41000–41999, 43000–43999, 46000–46999
Maps	L10000M–L10999M, L17000M–L17999M, L20000M–L20999M, L26000M–L30499M, L31000M–L31999M, L35000M–L35999M, L41000M–L41999M, L43000M–L43999M, L46000M–L46999M
PCDs/Formats	10000–10999, 17000–17999, 20000–20999, 26000–30499, 31000–31999, 35000–35999, 41000–41999, 43000–43999, 46000–46999



Umbrella Programming
Loans System Component Identification

Component Identifier	Identification Range
Program Source	L10000–L10999, L17000–L17999, L20000–L20999, L26000–L30499, L31000–L31999, L35000–L35999, L41000–L41999, L43000–L43999, L46000–L46999
Programs	10000–10999, 17000–17999, 20000–20999, 26000–30499, 31000–31999, 35000–35999, 41000–41999, 43000–43999, 46000–46999
Reports	LR00000–LR99999
Transactions—CICS/IMS	LAAA–L999



Management Support System (MSS) Component Identification

The following are the components used by the Management Support System.

MSS—Management Support System (MSS)

Component Identifier	Identification Range
Family	MSS
Application	260
Activities	260000–269999
ASSEMBLER Macros	M260000E–M269999E
COBOL Copybooks—Data Division	M260000D–M269999D
COBOL Copybooks—Procedure Division	M260000P–M269999P
Condition Codes	Application Code of 260, 1–49999, 60000–65535
Data Elements	None
Data Groups	260000–269999
Maps	None
PCDs/Formats	None
Program Source	M260000–M269999
Programs	260000–269999
Reports	None
Transactions—CICS/IMS	None



Online Collections System (OCS) Component Identification

The following are the components used by the Online Collections System.

OCS—Online Collections System

Component Identifier	Identification Range
Family	OCS
Application	16
Activities	33000–33999
ASSEMBLER Macros	N16000E–N16999E
COBOL Copybooks—Data Division	N16000D–N16999D
COBOL Copybooks—Procedure Division	N16000P–N16999P
Data Elements	OCSAAAAA–OCS99999
Data Groups	16000–16999, 33000–33999
Maps	N16000M–N16999M
PCDs/Formats	16000–16999
Program Source	N16000–N16999
Programs	16000–16999, 33000–33999
Reports	NR00000–NR99999
Transactions—CICS/IMS	NAAA–N999



Online Delivery System (ODS) Component Identification

The following are the components used by the Online Delivery System.

ODS—Online Delivery System

Component Identifier	Identification Range
Family	ODS
Application	15,21,22,23
Activities	15000–15999, 19000–19999, 21000–23999, 220000–220999
ASSEMBLER Macros	M15000E–M15999E, M21000E–M23999E
COBOL Copybooks—Data Division	M15000D–M15999D, M21000D–M23999D
COBOL Copybooks—Procedure Division	M15000P–M15999P, M21000P–M23999P
Data Elements	M1500001–M1599999, M2100001–M2399999
Data Groups	15000–15999, 21000–23999
Maps	M15000M–M15999M, M21000M–M23999M
PCDs/Formats	15000–15999, 21000–23999
Program Source	M15000–M15999, M21000–M23999
Programs	15000–15999, 21000–23999
Reports	MR0000–MR9999
Transactions—CICS/IMS	ATAA–AT99, TTAA–TT99, OSAA–OS99



Platform Automation Support System (PAS) Component Identification

The following are the components used by the Platform Automation Support System.

PAS—Platform Automation Support System

Component Identifier	Identification Range
Family	PAS
Application	81
Activities	5100–5199, 810000–819999
ASSEMBLER Macros	V81000E–V81999E
COBOL Copybooks—Data Division	V81000D–V81999D
COBOL Copybooks—Procedure Division	Not used
Condition Codes	Application code of 81, 1–49999, 60000–65535
Data Elements	V81000AA–V8199999
Data Groups	810100–810999
Maps	V81000M–V81999M
PCDs/Formats	810000–819999
Program Source	V81000–V81999
Programs	810000–819999
Reports	VR81000–VR81999
Transactions—CICS/IMS	P and PAS, PASA–PAS9



Preferred Client Services System (PCS) Component Identification

The following are the components used by the Preferred Client Services System.

PCS—Preferred Client Services System

Component Identifier	Identification Range
Family	PCS
Application	14
Activities	14000–14999, 140000–149999
ASSEMBLER Macros	H14000E–H14999E
COBOL Copybooks—Data Division	H14000D–H14999D, H14000D–H149999D
COBOL Copybooks—Procedure Division	H14000P–H14999P
Data Elements	PCSAAAAA–PCS99999
Data Groups	140000–149999
Maps	H14000M–H14999M
PCDs/Formats	140000–149999
Program Source	H14000–H14999, H140000–H149999
Programs	140000–149999
Reports	HR0000–HR9999
Transactions—CICS/IMS	PCSP, PCSR, PCST



Relationships and Profitability System (RPS) Component Identification

The following are the components used by the Relationships and Profitability System and its subsystems.

CIS—Customer Information System

Component Identifier	Identification Range
Family	PRO
Ownership	CIS
Application	64
Activities	24000–25999
ASSEMBLER Macros	A64000E–A65999E
COBOL Copybooks— Data Division	A64000D–A65999D
COBOL Copybooks— Procedure Division	A64000P–A65999P
Condition Codes	64000–64999
Data Elements	TAE64000–TAE65999
Data Groups	24000–25999
Maps	A64000M–A65999M
PCDs/Formats	24000–25999
Program Source	A64000–A65999
Programs	24000–25999
Reports	AR0000–AR9999
Transactions— CICS/IMS	HCIR, HCIS, HCIT, HCMU



RPM—Relationships and Profitability Manager

Component Identifier	Identification Range
Family	PRO
Ownership	RPM
Application	12, 18, 62
Activities	8200–8799, 13549, 18000–18999
ASSEMBLER Macros	E12000E–E12999E, E18000E–E18999E, E62000E–E62999E
COBOL Copybooks—Data Division	E62000D–E62999D, E12000D–E12999D, U50500D–U50529D, U50535D–U50549D
COBOL Copybooks—Procedure Division	E62000P–E62999P, E12000P–E12999P, U50500P–U50529P, U50535P–U50549P
Data Elements	TEE12000–TEE12999, TEE62000–TEE62999
Data Groups	8240–8799, 13500–13529, 13535–13549, 18000–18999
Maps	E62000M–E62999M, U50500M–U50529M
PCDs/Formats	8300–8500, 18000–18999
Program Source	E62000–E62999, E12000–E12999, U50500–U50529, U50535–U50549
Programs	8240–8799, 13500–13529, 13535–13549, 18000–18999
Reports	ER0000–ER9999
Transactions—CICS/IMS	RPMP, RPMR, RPMT, RPPC



Umbrella Programming

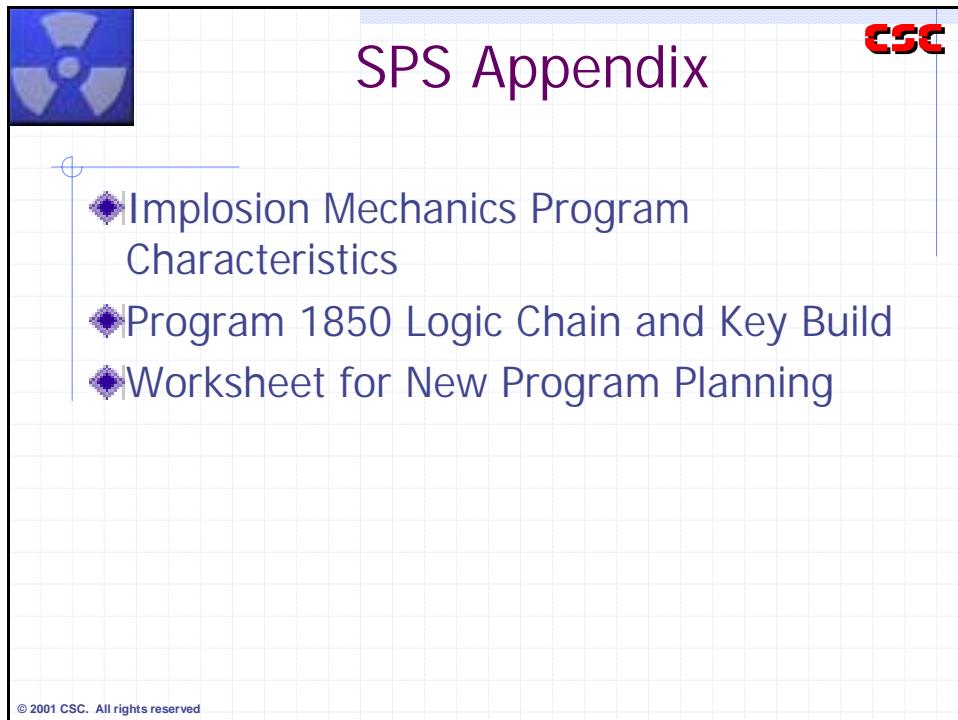
Relationships and Profitability System (RPS) Component Identification



SPS Appendix

B

Introduction



The slide has a blue header bar with a radiation symbol icon on the left and the CSC logo on the right. The main title "SPS Appendix" is in large purple font. Below it is a bulleted list of three items, each preceded by a purple diamond icon. At the bottom left is a small copyright notice.

© 2001 CSC. All rights reserved

- ◆ Implosion Mechanics Program Characteristics
- ◆ Program 1850 Logic Chain and Key Build
- ◆ Worksheet for New Program Planning

Notes:



Implosion Mechanics Program Characteristics



SPS Program Characteristics

The 'Implosion' Hogan Key

CSC

Key= Link Activity # / Spaces / 00000 / 100
S 15000 (- 15999)

```
100 MOVE *2 TO ACTN3601
200 MOVE *linkact# TO TLE50101
300 MOVE *dtailpgm# TO TLE50103
400 MOVE *dtailrptid# TO TLE50106
500 MOVE *comp#      TO TLE50105
600 MOVE *branch#    TO TLE50107
700 MOVE *langcode   TO TLE50124
```

© 2001 CSC. All rights reserved



SPS Program Characteristics

The 'Implosion' Application Data

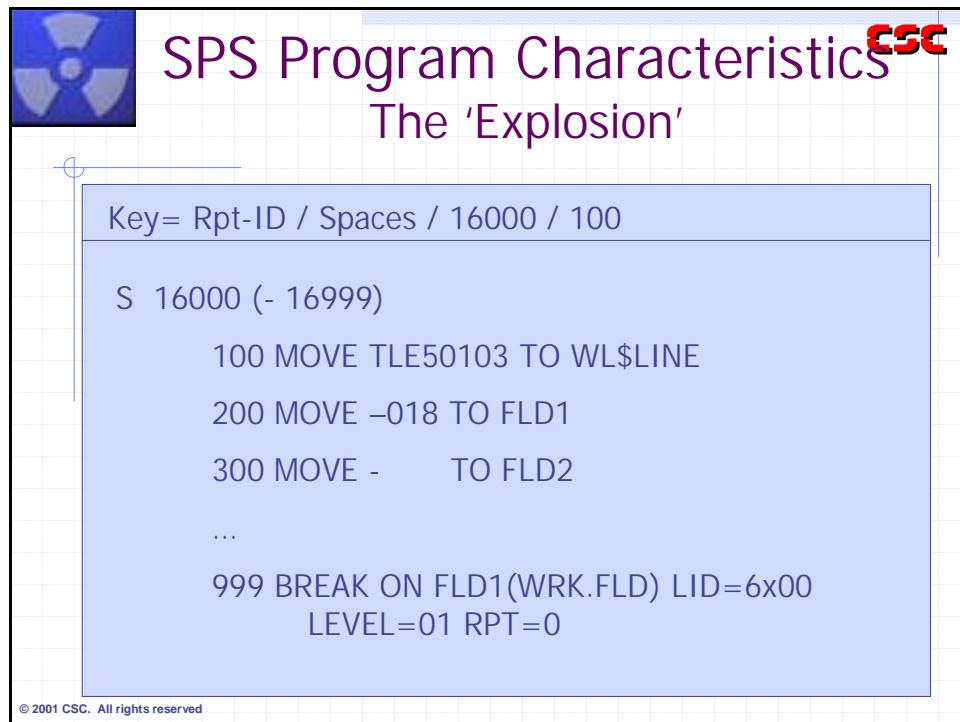
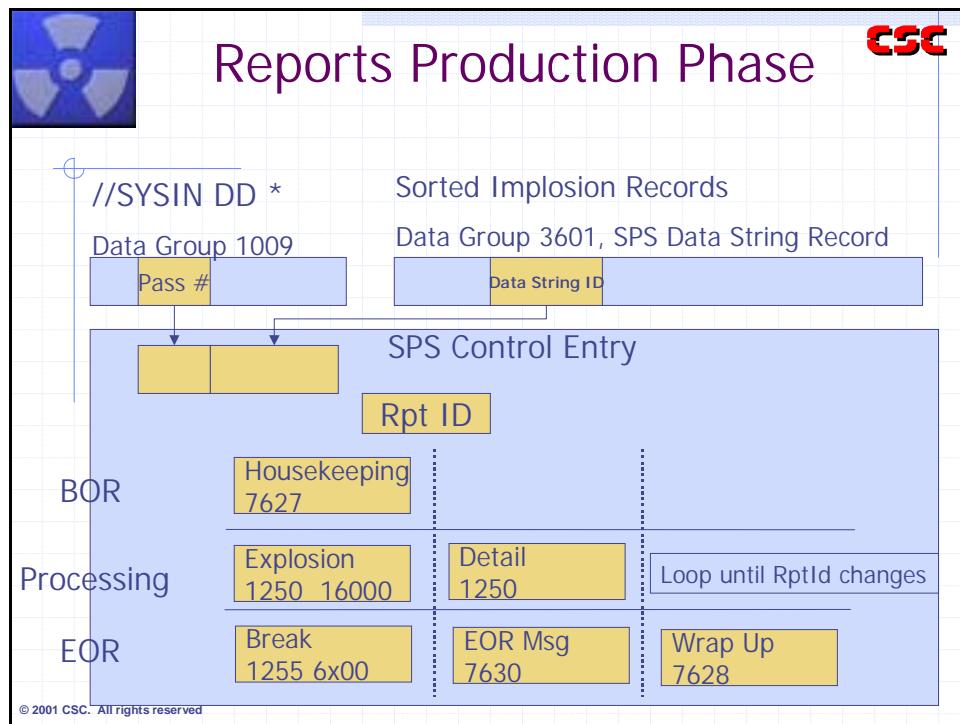
CSC

S 15000 (- 15999)

```
100 MOVE FLD1 TO -018 (for sorted fields)
100 MOVE FLD1 TO -048 (for no sorting)
200 MOVE FLD2 TO -
...
900 MOVE SAMVL TO TLE50100
910 CLEAR TLE50101
ACTIVITY 16135
```

© 2001 CSC. All rights reserved





Umbrella Programming

Implosion Mechanics Program Characteristics



SPS Program Characteristics

The Detail Line Print

CSC

Key= Rpt-ID / Spaces / WL\$LINE / 100
D n
S 10, 20, 30, etc.
100 MOVE FLD1 TO -001/003
200 EDIT FLD2 INTO -009/008 USING \$
250 ADD FLD2 TO \$01
300 EDIT FLD3 INTO -020/015 USING /
400 ADD *1 TO /01
500 SPACE 1 TIMES AFTER
600 PRINT

© 2001 CSC



SPS Program Characteristics

The Heading Program

CSC

Key= Rpt-ID / HEAD / 1000 / 100
P 1101
A 1279
S 1000 (, 1010, 1020,)
050 SPACE 1 TIME AFTER
100 MOVE 'PAGE' TO -001
200 EDIT PAGENO INTO -005/004 USING ZZZ9
300 DEC BY *1
400 PRINT

© 2001 CSC. All rights reserved





SPS Program Characteristics

The Break Program

CSC

Key= Rpt-ID / BREAK / 6x00 / 100

D n

S 6x10 (, 6x20, 6x30,)

100 MOVE 'COUNT' TO -035/006

200 EDIT /01 INTO -041/005 USING /

300 ADD /01 TO /0102

400 CLEAR /01

500 SPACE 3 TIMES AFTER

600 PRINT

D 00000

© 2001 CSC

Notes:



Program 1850 Logic Chain and Key Build



Program 1850 Logic Chain & CSC Key Build

Remember: 1850 always begins by locating the SPS code you want him to execute.

Link Activity > 1255?

Y → Key = RptID- **Link Activity #**
Mnemonic- Spaces
Line Def #- 00000
Seq #- 100

N

Format Def # in 15000-15999 = Implosion. DG3601 will be used.
Else, it's an Ad Hoc, and DG 1031 will be used.



Program 1850 Logic Chain & CSC Key Build

Link Activity = 1250?

Identified as an Execution Request

Y → Key = RptID- **from SPSCTL**
Mnemonic- Spaces
Line Def #- SPSCTL or
WL\$LINE
Seq #- 100

N

Line Def # and Format Def # in 16000-16999 = Explosion.
DG3601 will be used.
Else, it's a print program, and DG 1031 will be used.





Program 1850 Logic Chain & Key Build



Identified as a Heading Request

```
graph TD; A{Link Activity = 1251?} -- Y --> B[Key = RptID- from SPSCTL  
Mnemonic- HEAD  
Line Def #- 1000 or  
WL$LINE  
Seq #- 100]; A -- N --> C[By convention, we use 1000 series  
for the Line Def # & Format Def #]
```

© 2001 CSC. All rights reserved.



Program 1850 Logic Chain and Key Build



Identified as a Break Request

```
graph TD; A{Link Activity = 1255?} -- Y --> B[Key = RptID- from SPSCTL  
Mnemonic- BREAK  
Line Def #- LID= or  
SPSCTL  
Seq #- 100]; A -- N --> C[By convention, we use 6000 series  
for the Line Def # & Format Def #]
```

© 2001 CSC. All rights reserved.



New Program Planning

 Worksheet For New Program Planning
Implosion

© 2001 CSC. All rights reserved

Implosion Program: Link Activity = ?

Where Issued?
Application Program Reportable Incident

You select:
➤ HW GT 1255
➤ Link to 1850
➤ Used as RptId

Report ID	= Link Activity
Mnemonic	= spaces
Line Def #	= 00000
Seq #	= 100

© 2001 CSC. All rights reserved

 Worksheet For New Program
Planning SPSCTL Record

© 2001 CSC. All rights reserved

SPS Processing Control: Pass Number = ?

Primary key fields:

Data String ID=

You select:
➤ Appl. Specific
➤ Read from Ctl Card #1009

Implosion Link Activity # from DG 3601

Secondary key field: RptId= ?

You select. Will be used to house all subsequent SPS Programs.



 Worksheet For New Program Planning 
SPSCTL Record Programs

Explosion Program- Link activity = 1250
Where Issued? SPSCTL

Report ID = SPSCTL RptId
Mnemonic = spaces
Line Def # = 16000 (-16999)
Seq # = 100
Format Def # = 16000 (-16999)

© 2001 CSC. All rights reserved

 Worksheet For New Program Planning 
SPSCTL Record Programs

Detail Print Program- Link activity = 1250
Where Issued? SPSCTL or WL\$LINE

Report ID = SPSCTL RptId
Mnemonic = spaces
Line Def # = NOT 16000-16999
Seq # = 100
Format Def # = NOT 15000-16999

© 2001 CSC. All rights reserved



Umbrella Programming

New Program Planning



Worksheet For New Program Planning SPSCTL Record Programs

CSC

Heading Print Program- Link activity = 1251

Where Issued? Automatic via 'D' command

Report ID	= SPSCTL RptId
Mnemonic	= HEAD
Line Def #	= 1000-1999
Seq #	= 100
Format Def #	= 1000-1999

© 2001 CSC. All rights reserved



Worksheet For New Program Planning SPSCTL Record Programs

CSC

Break Print Program- Link activity = 1255

Where Issued? Automatic via LID=, and on SPSCTL

Report ID	= SPSCTL RptId
Mnemonic	= BREAK
Line Def #	= 6000-6999
Seq #	= 100
Format Def #	= 6000-6999

© 2001 CSC. All rights reserved



**Umbrella Programming
Course**

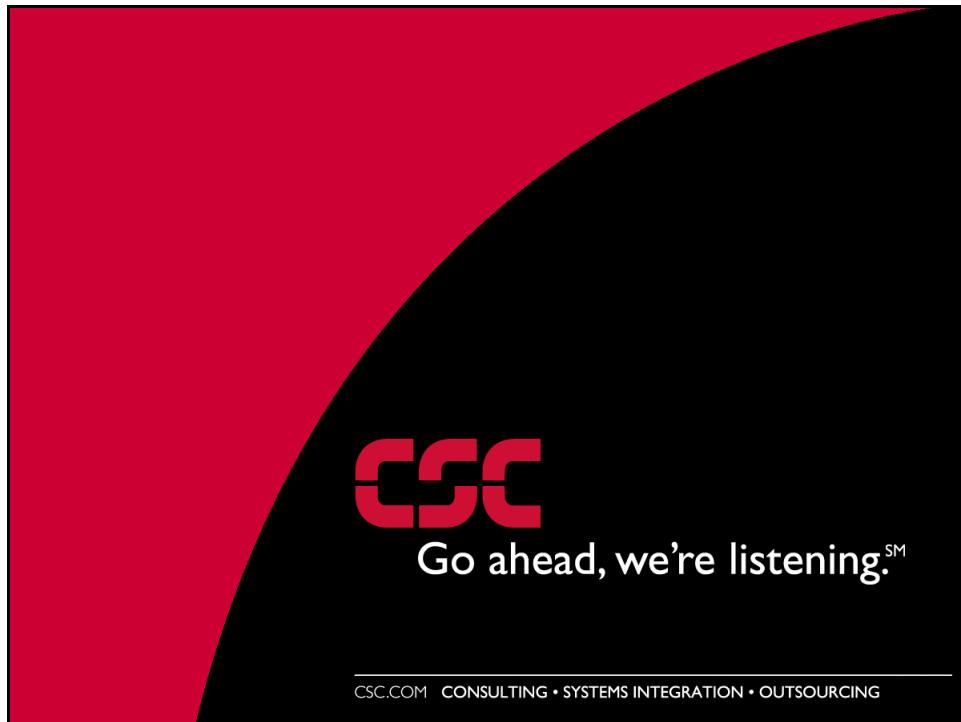
csc

World Class Education

* Hogan's Umbrella System

Thank You!

© 2001 CSC. All rights reserved.



Umbrella Programming

New Program Planning



This documentation contains trade secrets and confidential information which are proprietary to Computer Sciences Corporation. The use, reproduction, distribution, or disclosure of the documentation, in whole or in part, without the express written permission of Computer Sciences Corporation is prohibited. This documentation is also an unpublished work protected under the copyright laws of the United States of America and other countries. If this documentation becomes published, the following notice shall apply:

Copyright © 2001 Computer Sciences Corporation

All Rights Reserved.

All product names and trademarks are the property of their respective owners.