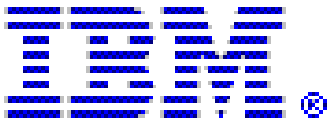


IBM Information Server Suite
Course: DX741
QualityStage 8 Essentials

IBM Software Group
One Rogers Street
Cambridge, MA 02142

Table of Contents

Course contents

- Data quality issues
- Information Server purpose and architecture
- Introduction to DataStage and QualityStage
- Investigation
- Standardization
- Match
- Survivorship
- Special Topics
  - Data quality methodology
  - QualityStage Migration Tool


Data Quality Issues
PLEASE PRINT COURSE MATERIALS.
FOR INDIVIDUAL USE ONLY.
© Copyright IBM Corporation 2007
The course materials are provided for internal use only by the individual receiving the materials.
The materials may not be modified, copied, distributed or transferred without the express prior written consent of IBM.
4.0.3

Unit objectives

- After completing this unit, you should be able to:
  - Describe data quality issues
  - Describe where QualityStage fits into a data cleansing project

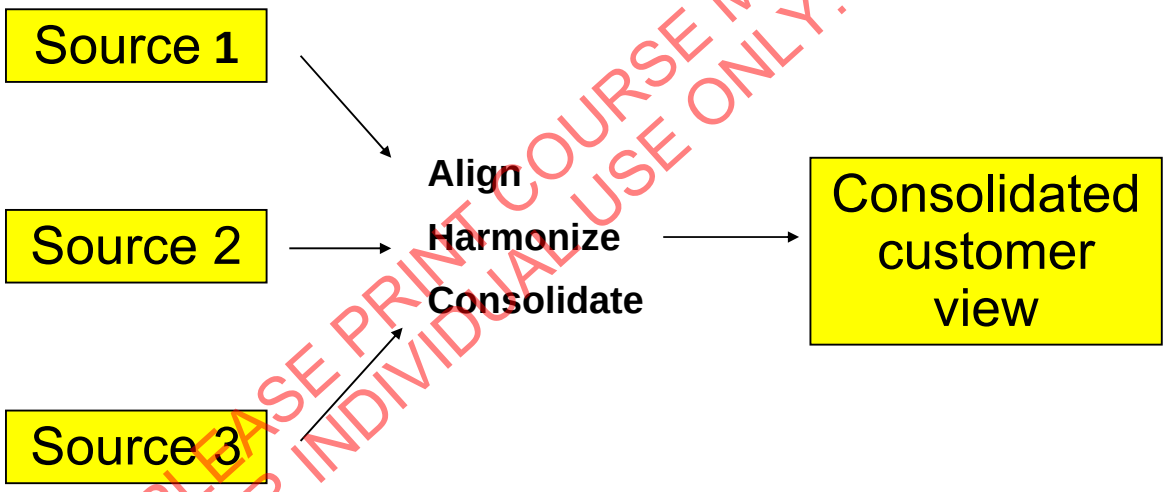
Data quality challenges

- Different or inconsistent standards in structure, format or values
- Missing data, default values
- Spelling errors, data in wrong fields
- Buried information
- Data anomalies

Data quality – why do we care?

- Accurate reports
- Accurate information for support operations
- Support development of applications that go beyond original scope for which data was designed
  - Master Data Management
  - Data Warehouse
  - Analytical applications

Master Data Management



Examples of source systems :

- Orders
- Human resources
- Customer support

The consolidated customer view can represent a data warehouse or a customer relationship management system (CRM).

Different or inconsistent standards

	Name Field	Location
Source 1	MARK DI LORENZO	MA93
	DENIS E. MARIO	CT15
	TOM & MARY ROBERTS	IL21
Source 2	DILORENZO, MARK	6793
	MARIO, DENISE	0215
	ROBERTS, TOM & MARY	8721
Source 3	MARC DILORENZO ESQ	BOSTON
	MRS DENNIS MARIO	HARTFORD
	MR & MRS T. ROBERTS	CHICAGO

- Three sources representing the same four customers, MARK DILORENZO, and DENISE MARIO, TOM ROBERTS and MARY ROBERTS.

- The & in the third record defines a relationship between two of the customers

- The ESQ in the first record from source 3 is a title

- The locations are the same location but coded differently

- MA93 = 6793 = BOSTON

- Different structure for storing data: first name lastname and lastname, firstname

- Sometimes there is a middle initial, sometimes there isn't

- Not all records have a title word: Mr. Mrs, Esq

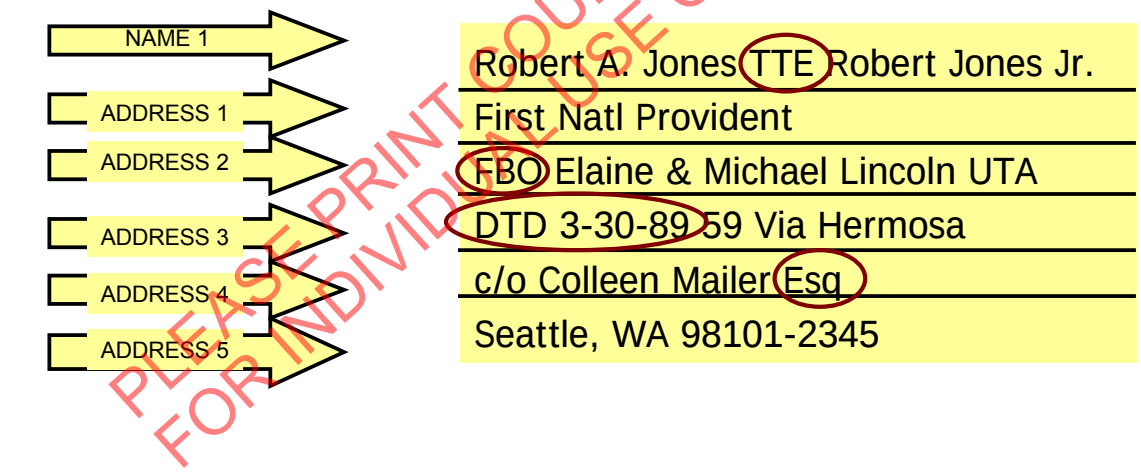
Missing data & default values

Do the field values match the meta data labels?

NAME	SOC. SEC. #	TELEPHONE
Denise Mario DBA	228-02-1975	6173380300
Marc Di Lorenzo ETAL	999999999	3380321
Tom & Mary Roberts	025-37-1888	
First Natl Provident	34-2671434	415-392-2000
	101010101	508-466-1200
Astorial Fedrl Savings	LN#12-756	212-235-1000
Kevin Cooke, Receiver	18-7534216	FAX 528-9825
John Doe Trustee for K	111111111	5436

- Default SSN of 999999999, and 111111111
- Missing phone numbers

Buried information



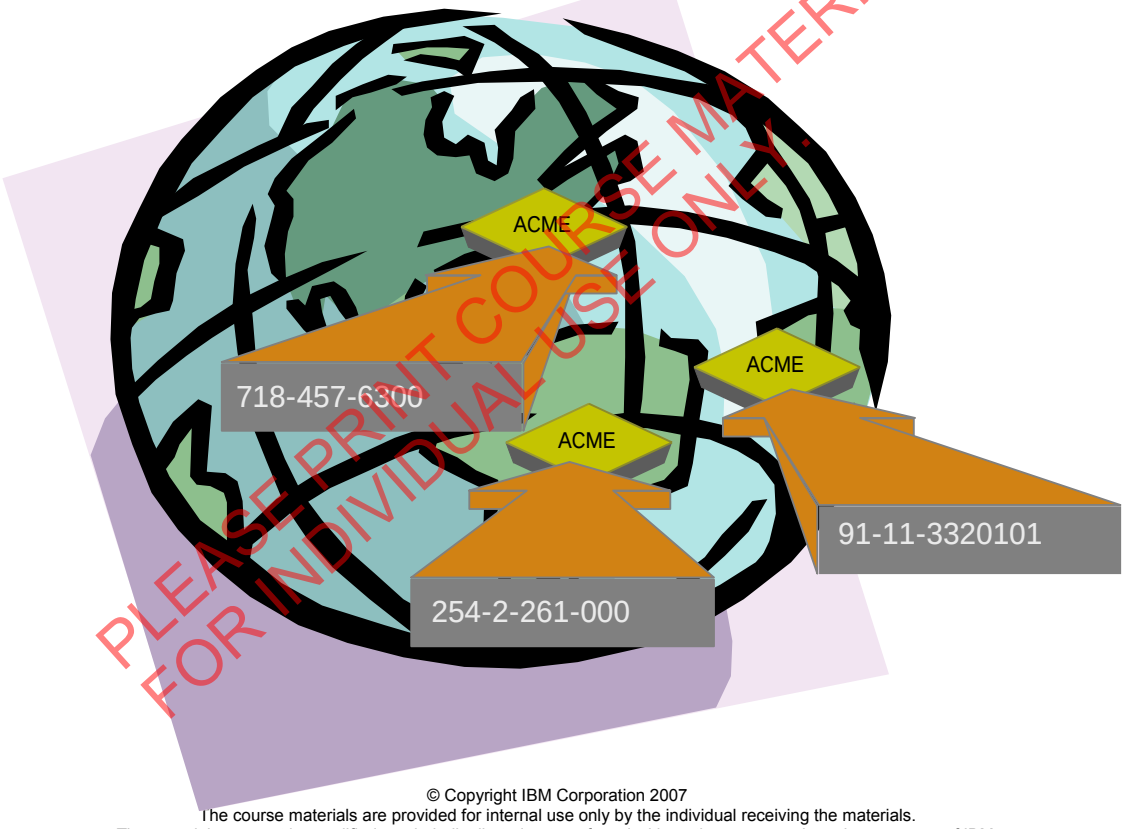
The anomalies nightmare

CUSNUM	NAME	ADDRESS	SALES $
90328574	IBM	187 N.Pk. Str. Salem NH 01456	))
90328575	I.B.M. Inc.	187 N.Pk. St. Sarem NH 01456	))
90238495	International Bus. M.	187 No. Park StSalem NH 04156	))
90233479	Int. Bus. Machines	187 Park Ave Salem NH 04156	))
90233489	Inter-Nation Consults	15 Main St. Andover MA 02341	))
90234889	Int. Bus. Consultants	PO Box 9 Boston MA 02210	))0
90345672	I.B. Manufacturing	Park Blvd. Boston MA 04106	))0

No common key **Anomalies** **Lack of Standards**

Spelling Errors

No consolidated of a single entity



Errors due to:

- Data Entry Errors
- System field limitations
- Mergers and acquisitions
- Feeding legacy data into new systems

What data challenges do you face?

- No consistent naming convention
- Business terms and spillover text
- Missing values or data in the wrong fields
- Buried information
- Misspelling
- No unique key linking records together

Acct #	Name	Address	City	State	Zip	Note
5154155	Peter J. Lalonde	40 Beacon St.	Melrose, Mass		02176	ODP
5152335	LaLonde, Peter	76 George 617-210-0824	Boston	YES	MA	02111
5146261	Lalonde, Sofie	40 Bacon Street	Melrose		MA	CHK ID
87121	Pete & Soph Lalond	76 George Road	Boston	MASS		FR Alert
87458	P. Lalonde FBO	S.Lalonde40 Becon Rd.	Melrose	MA	02	176

The five common data Contaminants or DQ Issues are:

1. Lack of Standards – data coming from disparate systems
2. Spillover and Lack of Domain QualityStage
3. Misspellings and Multiple representations, Missing and invalid data
4. No consolidated key
5. Buried information

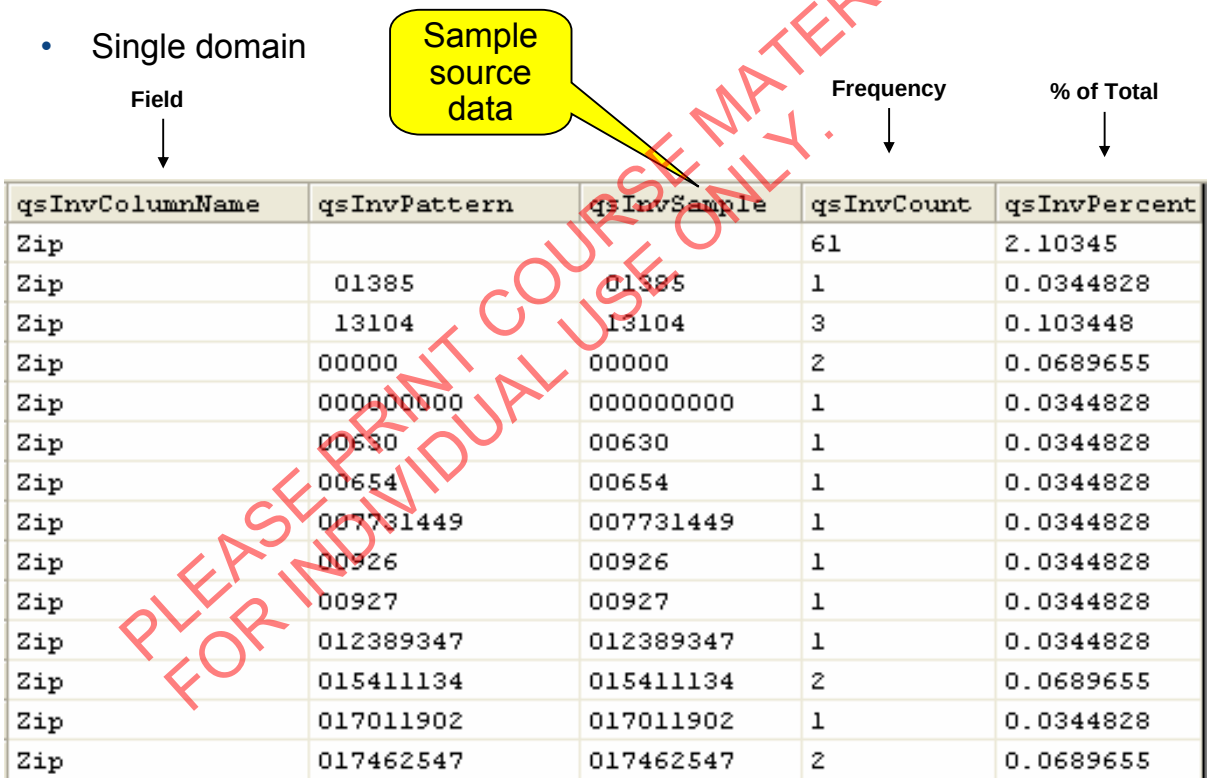
Why investigate?

- Discover trends and potential anomalies in the data
- 100% visibility of single domain and free-form fields
- Identify invalid and default values
- Reveal undocumented business rules and common terminology
- Verify the reliability of the data in the fields to be used as matching criteria
- Gain complete understanding of data within context

Why is Analysis and Assessment important. What does it provide?

Investigate – single domain report

- Single domain



qsInvColumnName	qsInvPattern	qsInvSample	qsInvCount	qsInvPercent
Zip			61	2.10345
Zip	01385	01385	1	0.0344828
Zip	13104	13104	3	0.103448
Zip	00000	00000	2	0.0689655
Zip	000000000	000000000	1	0.0344828
Zip	00630	00630	1	0.0344828
Zip	00654	00654	1	0.0344828
Zip	007731449	007731449	1	0.0344828
Zip	00926	00926	1	0.0344828
Zip	00927	00927	1	0.0344828
Zip	012389347	012389347	1	0.0344828
Zip	015411134	015411134	2	0.0689655
Zip	017011902	017011902	1	0.0344828
Zip	017462547	017462547	2	0.0689655

qsInv – QualityStage Investigate

Investigate – word pattern report

- Freeform text (Word)

Field

Pattern

Sample source data

qsInvColumnName	qsInvPattern	qsInvSample	qsInvCount	qsInvPercent
FullName	?,?	TOWNSEND , PANDA	36	1.24138
FullName	?,??	ODISH , SAMER HERMIZ	5	0.172414
FullName	?,?F	COBB , EARLUST ANN	1	0.0344828
FullName	?,?I	BANGERTER , TRENTYN A	25	0.862069
FullName	?,?I.	GRIDLEY , LEVERITT S.	11	0.37931
FullName	?,F	ANTENNA , SALVATORE	332	11.4483
FullName	?,F&F	KAVANAGH, MARY & OSCAR	2	0.0689655
FullName	?,F-FI	GIACOBBE , SHERRY-LYNN N	2	0.0689655
FullName	?,F?	BANDINI , DENISE WALSH	31	1.06897
FullName	?,FF	HANLY , PETER MICHAEL	56	1.93103
FullName	?,FFF	WYRICK , JIMMY RAY OLIVER	2	0.0689655
FullName	?,FFI	HOLT , JO ANN C	2	0.0689655
FullName	?,FI	BEERMANN , VERNON E	668	23.0345
FullName	?,FI.	BENAVIDES , KALA R.	74	2.55172

Frequency

% of Total

What is standardize?

- Applying business logic to data chaos.
  - Pattern manipulation
- Enforcing business standards on data elements.
  - Standards definition
- Transforming the input to an output which meets the business requirement.
  - Field structuring

How to standardize

- Parse specific data fields into smaller, atomic data elements
  - Atomic data elements are called tokens
  - Categorize identified elements
    - Separate Name, Address, and Area from freeform Name & Address lines
    - Identification of Distinct Material Categories (e.g. Sutures vs. Orthopedic Equipment)
- Refine data elements
  - Example 1
    - Name = 'DR PAUL E JONES' becomes:
      - > Title = 'DR'
      - > First Name = 'PAUL'
      - > Middle Name = 'E'
      - > Last Name = 'JONES'
  - Example 2
    - Part Description = 'BLK LATEX GLOVE' becomes:
      - > Color = 'BLACK'
      - > Type = 'LATEX'
      - > Part = 'GLOVE'

Why

- Invoke enterprise wide standards
  - Common usage of abbreviations across the enterprise
  - Single entity definition/metadata across the enterprise
- Improve searching ability
  - Search by single domain entity: FName, LName, PartName,…
- Improve matching ability
  - Entity level matching NOT mixed domain matching:
  - FName → FName ('JOHN' → 'JOHN'),
  - LName → LName ('SMITH' → 'SMITH'),
  - NOT Name → Name ('MR JON P SMITH' → 'SMITH, JON')
  - Match consistent standardized values not free form variations:
  - 'JAMES' to 'JAMES' not 'JIM' to 'JAMES',
  - 'ST' to 'ST' not 'STREET' to 'STR',
  - 'BLACK' to 'BLACK' not 'BLACK' to 'BLK'
- Enable Categorization through standardized single domain entities
  - OFFICE EQUIP = {PC, FAX, COPIER, PRINTER, PHONE,..)

Why standardize?

- Normalize values in data fields to standard values
  - Transform First Name = 'MIKE' → 'MICHAEL'
  - Transform Title = 'Doctor' → 'Dr'
  - Transform Address = 'ST. Michael Street' → 'Saint Michael St.'
  - Transform Color = 'BLK' → 'BLACK'
- Apply phonetic coding to key words - facilitates record linkage
  - NYSIIS
  - Soundex
  - Typically applied to Name fields (first, last, street, city)

NYSIIS and Soundex are both phonetic coding algorithms.

QualityStage standardize

- Uses a highly flexible pattern recognition language
- Can employ field or domain specific standardization (i.e. unique rules for names vs. addresses vs. dates, etc.)
- Contains customizable classification and standardization tables
- Utilizes results from data investigation

QualityStage standardize report example

Original data

Ind./Org. flag

FullName	NameType	Gender	FirstName	Middle	PrimaryName_USNAME
ECKER , JACOB E	I	M	JACOB	E	ECKER
KNIAT , KENNETH S	I	M	KENNETH	S	KNIAT
MENARD , LYNETTE H	I	F	LYNETTE	H	MENARD
STARBUCK , F DIANE	I	NULL	F	DIANE	STARBUCK
FIRST UNITED	O	NULL	NULL	NULL	FIRST UNITED
FIRSTAR BK	O	NULL	NULL	NULL	FIRSTAR BANK
APPERT LTD LIAB	O	NULL	NULL	NULL	APPERT LTD LIAB
BERTHA L KARRER	I	F	BERTHA	L	KARRER
J. BERNARD	I	NULL	J	NULL	BERNARD
JOHN F WIBLE TRST	O	NULL	NULL	NULL	JOHN F WIBLE TRUST
NELLIE HEALD	I	F	NELLIE	NULL	HEALD
BOROWITZ FAM TRUST	O	NULL	NULL	NULL	BOROWITZ FAM TRUST
FRANCIS BALLMAN TRUST	O	NULL	NULL	NULL	FRANCIS BALLMAN TRUS
OLGA DUEMELAND	I	F	OLGA	NULL	DUEMELAND
EUGENE B BOROWITZ	I	M	EUGENE	B	BOROWITZ
DONALD R HALL	I	M	DONALD	R	HALL

Match

"Conditioned data and QualityStage's matching engine link the previously unlinkable."

- Match Construction:
  - Reliability of input data defines a match result.
- Statistical Analysis & Match Scoring:
  - Linkage probability determined on a sliding scale by field level comparison.
- Report Generation:
  - All business rules applied have easy to understand report structure.

What is match?

- Identifying all records on one file that correspond to similar records on another file
- Identifying duplicate records in one file
- Building relationships between records in multiple files
- Performing statistical and probabilistic matching
- Calculating a score based on the probability of a match

Why match?

- Identify duplicate entities within one or more files
- Perform householding
- Create consolidated view of customer
- Establish cross-reference linkage
- Enrich existing data with new attributes from external sources

How to match

- Single file (Unduplication) or two file (Reference or Geomatch)
- Different match comparisons for different types of data (e.g. exact character, uncertainty/fuzzy match, keystroke errors, multiple word comparison …)
- Generation of composite weights from multiple fields
- Use of probabilistic or statistical algorithms
- Application of match cutoffs or thresholds to identify automatic and clerical match levels
- Incorporation of override weights to assess particular data conditions (e.g. default values, discriminatory elements)

QualityStage match

- A wide variety of match comparison algorithms providing a full spectrum of fuzzy matching functions
- Statistically-based method for determining matches (Probabilistic Record Linkage Theory)
- Field-by-field comparisons for agreement or disagreement
- Assignment of weights or penalties
- Overrides for unique data conditions
- Score results to determine the probability of matched records
- Thresholds for final match determination
- Ability to measure informational content of data

QualityStage match examples

SetID	Record Type	FullName	AddressLine1	City
2126	XA	GEROSA, FRAN X	C/O NANCY C GEROSA	RIDGEFIELD
2126	DA	GEROSA, FRANCIS X	C/O NANCY C GEROSA	RIDGEFIELD
2126	DA	GEROSA, FRANK X	C/O NANCY C GEROSA	RIDGEFIELD
2126	DA	GEROSA, FRANCIS X	C/O NANCY C GEROSA	RIDGEFIELD
62	XA	BIONDI , KATHERINE A.	3142 CENTRAL ST	EVANSTON
62	DA	BIONDI , KATHERINE A.	3142 CENTRAL ST	EVANSTON
254	XA	STEFAN , JOHN R.	11009 AZALEA DR	PITTSBURGH
254	DA	STEFAN , JOHN R.	11009 AZALEA DR	PITTSBURGH
750	XA	RUMMEL , JACK R	640 SUMMERGREEN DRIVE	FRANKENMUTH
750	DA	RUMMEL , JACK R	640 SUMMERGREEN DR	FRANKENMUTH
15	XA	BANGERTER , EDWARD L	2060 CANDLE TREE CV	SANDY
15	DA	BANGERTER , EDWARD L	2060 CANDLE TREE CV	SANDY
389	XA	GOLDBLATT , RICHARD J	6410 TARREGA ST	CORAL GABLES
389	DA	GOLDBLATT , RICHARD J	6410 TARREGA ST	CORAL GABLES
431	XA	COLLINS , THERESA A	3699 CLAY ST APT 2	SAN FRANCISCO
431	DA	COLLINS , THERESA A	3699 CLAY ST APT 2	SAN FRANCISCO
134	XA	GRANT MORROW III	253 N COLUMBIA AVE	COLUMBUS
134	DA	GRANT MORROW III	253 N COLUMBIA AVENUE	COLUMBUS
1954	XA	ELKODSI, SUSAN L	8 SUNBEAM DR	TRUMBULL
1954	DA	ELKODSI, SUE	8 SUNBEAM DRIVE	TRUMBULL

What is survive?

- Creation of best-of-breed “surviving” data based on record or field level information
- Development of cross-reference file of related keys
- Production of load exception reports
- Creating output formats:
  - Relational table with primary and foreign keys
  - Transactions to update databases
  - Cross-reference files, synonym tables
  - Audit trails, exception reports

Why survive?

- Provide consolidated view of data
- Provide consolidated view containing the “best-of-breed” data
- Resolve conflicting values and fill missing values
- Cross-populate best available data
- Implement business and mapping rules
- Create cross-reference keys

How to survive

- Highly flexible rules
- Record or field level survivorship decisions
- Rules can be based upon data frequency, data recency (i.e. date), data source, value presence or length
- Rules can incorporate multiple tests
- QualityStage features
  - Point-and-click (GUI-based) creation of business rules to determine best-of-breed “surviving” data
  - Performed at record or field level

Survive has its own rules (not the same as standardization).

QualityStage survive examples

Example 1:

- The longest populated Middle and Last Name

Matched		
First Name	***Middle Name***	***Last Name***
MARI		**LEMELSON-LAPPNER**
MARI	**S**	LEMELSON

Survived		
First Name	***Middle Name***	***Last Name***
MARI	**S**	**LEMELSON-LAPPNER**

Example 2:

- The longest populated Middle Name, Date of Birth, and SSN

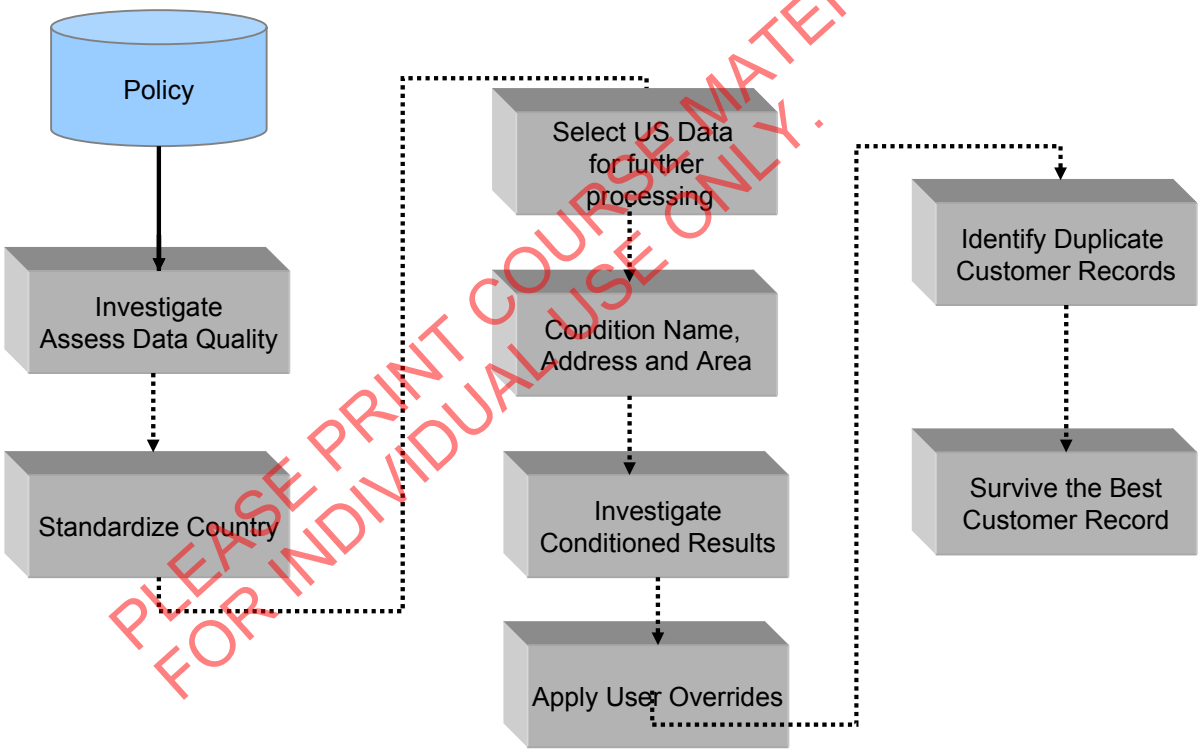
Matched				
First Name	***Middle N***	***Last Name***	***DOB***	***SSN***
DENISE		TRIANO	**19580211**	**98524173**
DENISE	**F**	TRIANO		

Survived				
First Name	***Middle N***	***Last Nam***	***DOB***	***SSN***
DENISE	F	TRIANO	**19580211**	**98524173**

These examples are field-level survivorship. Individual fields are mixed to form an output record.

Record-level survivorship would choose one record over another, perhaps based on source system or date.

Course exercise project design



Checkpoint

1. (T/F) Data quality investigation cleans the source data.
2. (T/F) Standardization modifies the source data so that it can be loaded into the target system.
3. (T/F) Survivorship data can be either record based or field based.

Checkpoint solutions

1. (T/F) (T/F) Data quality investigation cleans the source data.
   Answer: False
2. (T/F) Standardization modifies the source data so that it can be loaded into the target system.
   Answer: False
3. Survivorship data can be either record based or field based.
   Answer: True

Unit summary

Having completed this unit, you should be able to:

- Describe the five common data quality contaminants
  - Different standards
  - Missing and default values
  - Spillover and buried information
  - Anomalies
  - No consolidated view
- Describe each of the following processes:
  - Investigation
  - Standardization
  - Match
  - Survivorship

Exercise 1: Review course project

- Course business case: WINN Insurance CRM project
- See QualityStage Essentials Exercises

Exercise 2: copy student files

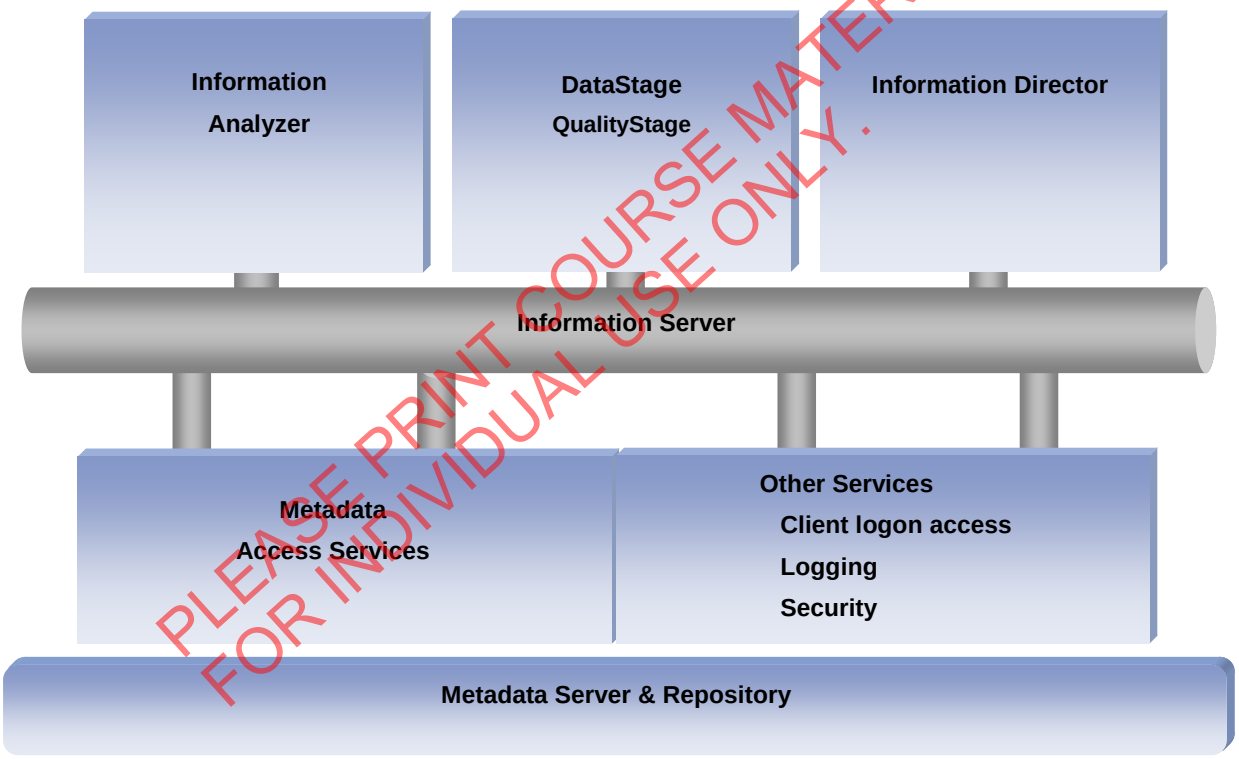
- Copy student files to disk
  - Use C: drive as root for folder


QualityStage 8 Architecture
PLEASE PRINT COURSE MATERIALS.
FOR INDIVIDUAL USE ONLY.
© Copyright IBM Corporation 2007
The course materials are provided for internal use only by the individual receiving the materials.
The materials may not be modified, copied, distributed or transferred without the express prior written consent of IBM.
4.0.3

Unit objectives

- After completing this unit, you should be able to:
  - Describe the Data Quality architecture
  - Identify data quality server and client components
  - Describe the methods of client/server communication

Information Server conceptual architecture



Information Server is really a suite of applications:

- DataStage
- QualityStage
- Information Analyzer
- Business Glossary
- Information Director

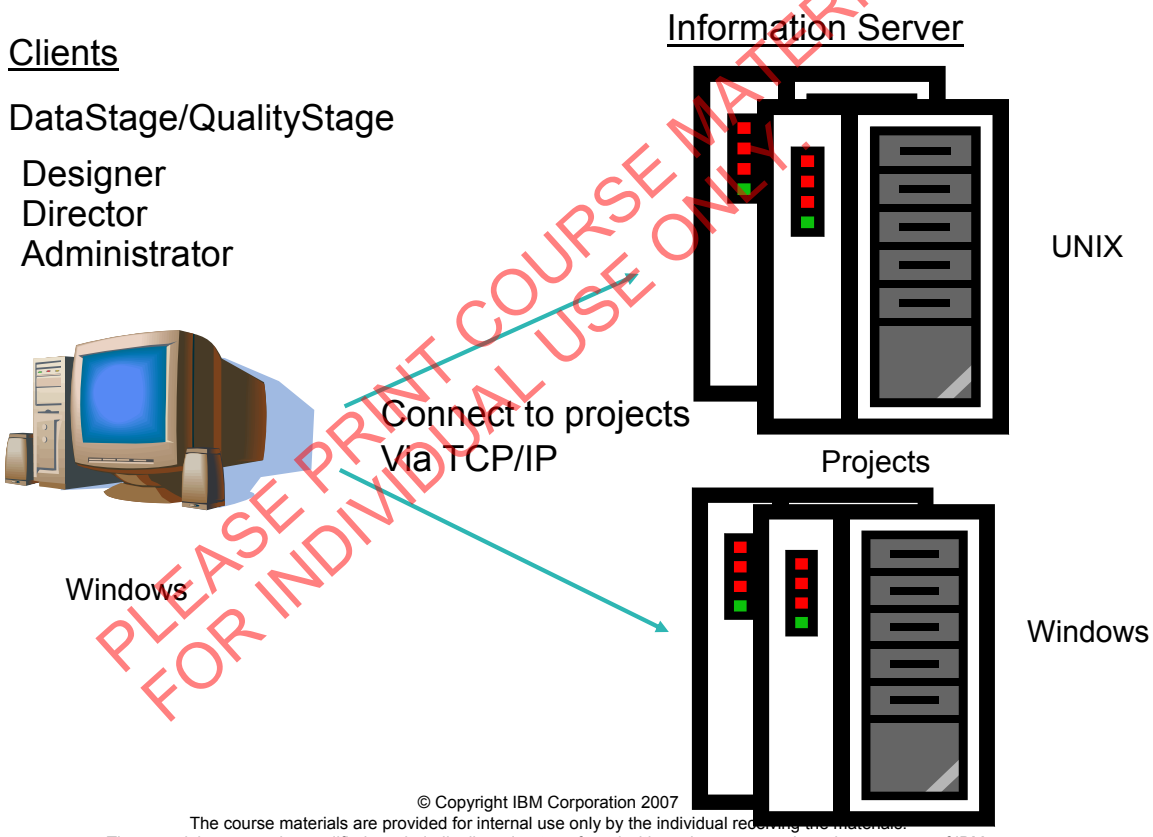
QualityStage is an optional, add-on component to DataStage.

QualityStage technical highlights

- Uses Enterprise level DataStage
  - DataStage design environment
  - Parallel execution engine
  - Stages are native enterprise operators
  - Match designer is embedded in DataStage Designer Client
  - Get DataStage data connectivity by default
    - No need for meta brokers, plug-ins
    - Common meta data
- Legacy (pre-version 8) QS job execution
  - Migration utility available to aid conversion from QS 7.x to QS 8
  - Converted jobs can be compiled and executed in the QS 8 environment

Data Quality stages are only available on Enterprise jobs.

DataStage/QualityStage physical architecture

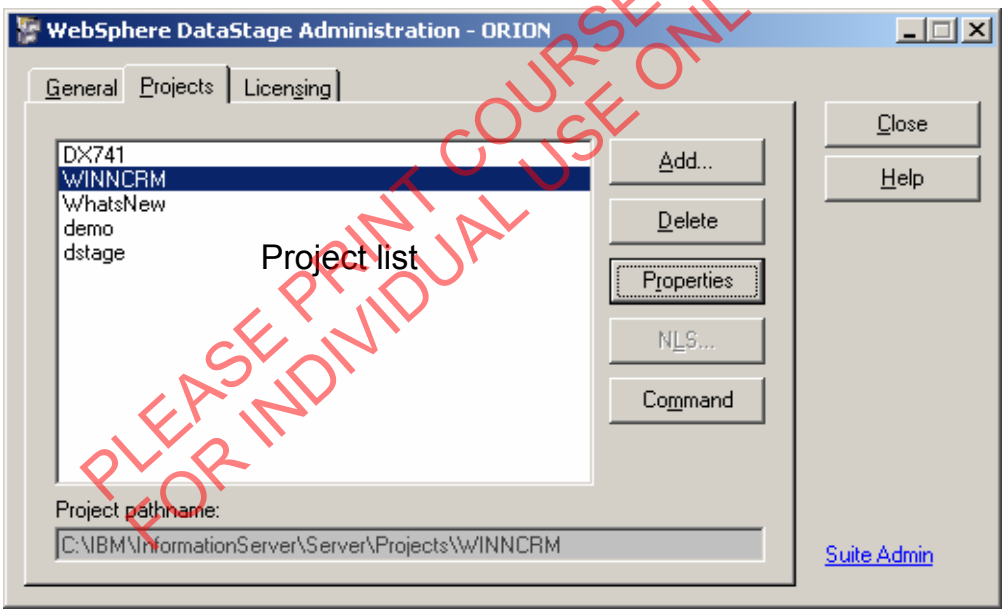


DataStage clients

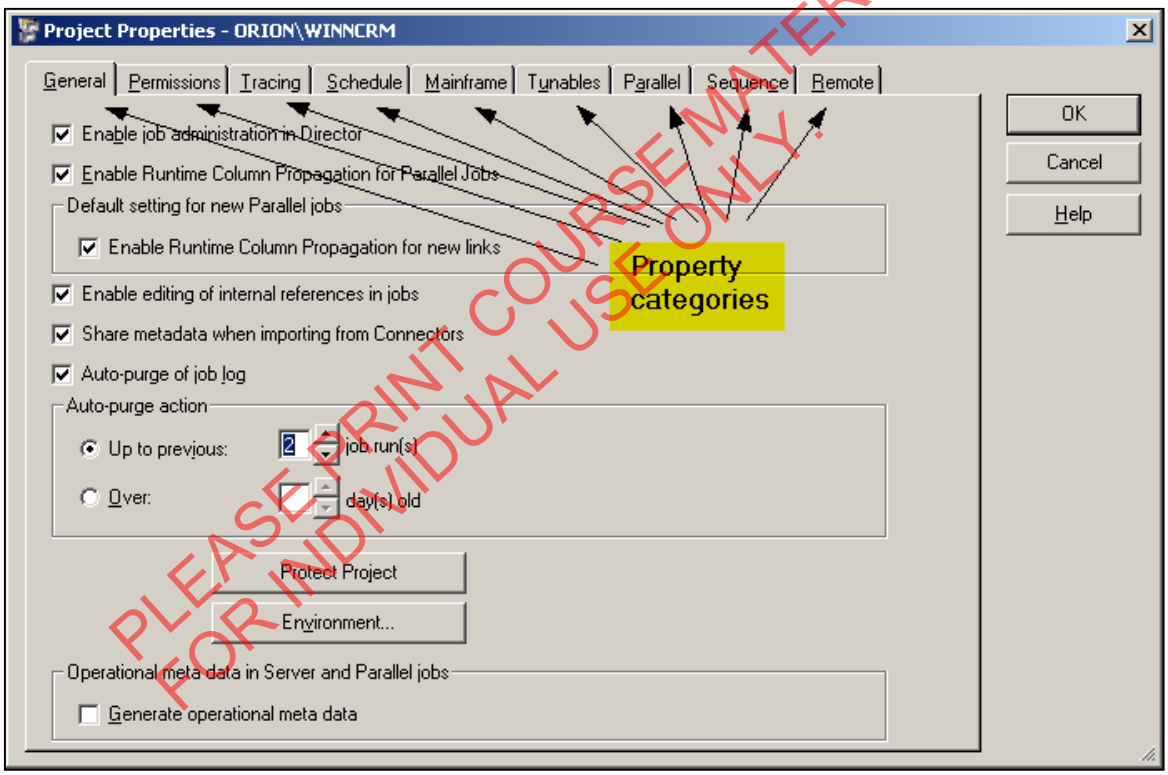
- Administrator
  - Add and delete projects
  - Set project defaults
  - Set project environment parameters
- Designer
  - Maintain data definitions
  - Add, modify, and delete jobs
  - Add, modify, and delete match specifications
  - Manage rule sets
  - Compile jobs
  - Run jobs
  - Provision rule sets and match specifications
- Director
  - Run jobs
  - Review job log
  - Schedule jobs

DataStage Administrator

- Administrator
  - Create or delete projects
  - Set project defaults
  - Apply security



Project property defaults

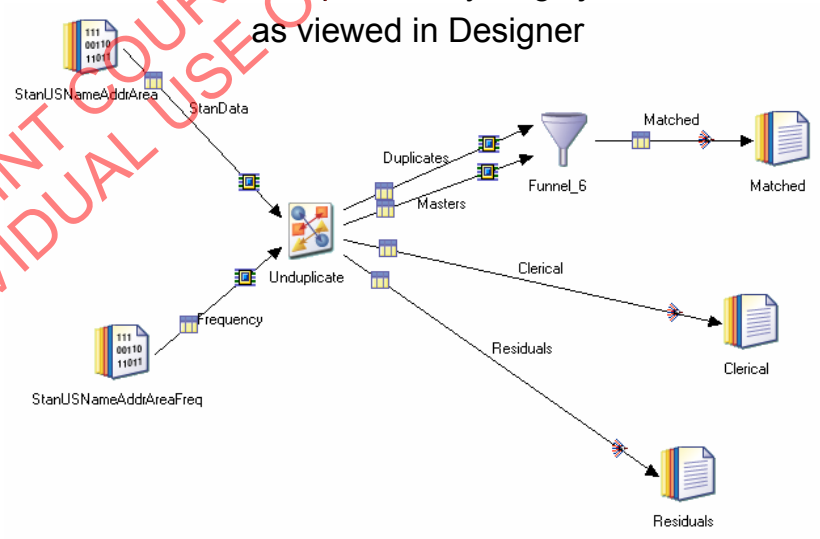


Project properties are divided into categories and accessed with the GUI tabs.

DataStage Designer

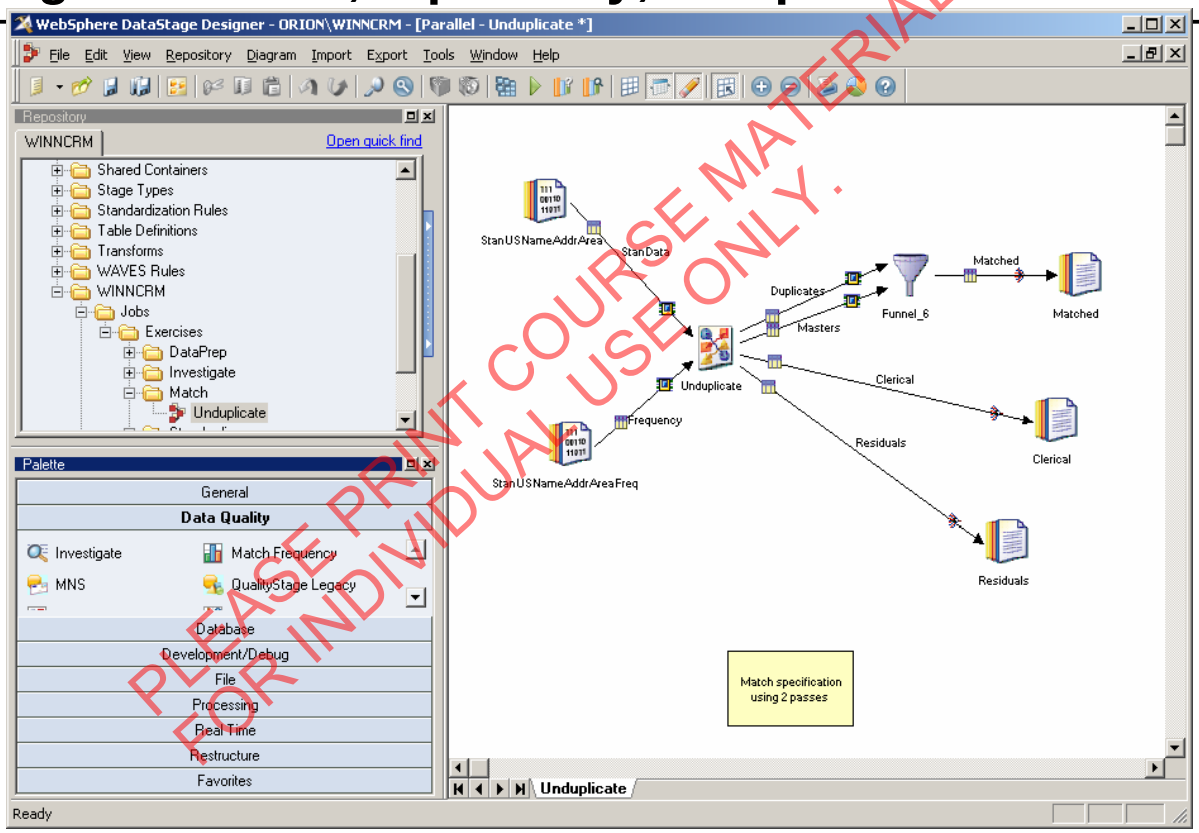
- Designer
  - Client GUI for designing jobs
    - Windows 2000+, XP
    - Build meta data
    - Build Jobs
    - Modify Standardization Rules
    - Build match specifications
  - Designer Repository
    - Database

Sample QualityStage job as viewed in Designer



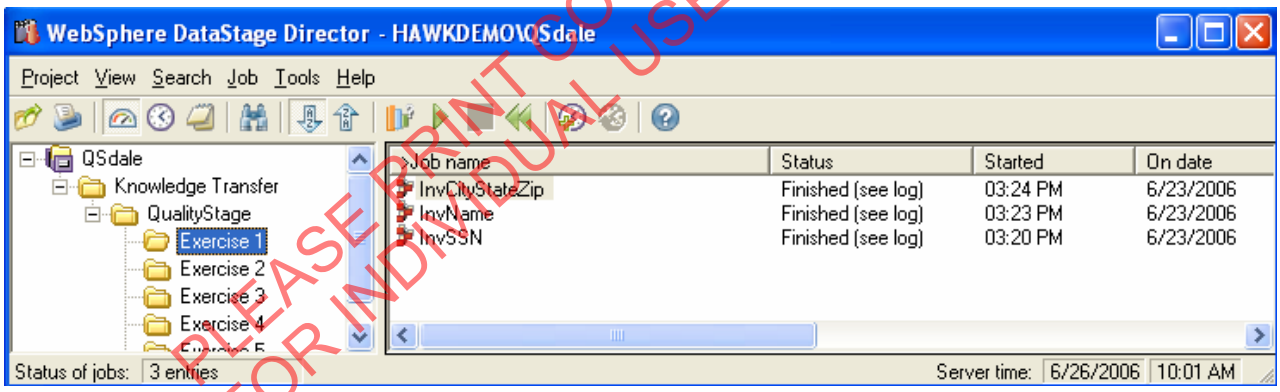
QualityStage jobs are data flow oriented; the direction of the arrows shows the flow of data. Stages process the data.

Designer canvas, repository, and palette



DataStage Director

- Director
  - Client GUI for managing job execution
  - Windows 2000+, XP
  - Run jobs – set job options and parameters
  - View job log
  - Schedule job execution



QualityStage jobs now produce standard DataStage job log. The log should be viewed for statistics.

Job log viewed in Director

12:23:06 PM	12/7/2006	Control	Starting Job Unduplicate.
12:23:09 PM	12/7/2006	Info	Environment variable settings: (...)
12:23:09 PM	12/7/2006	Info	Parallel job initiated
12:23:09 PM	12/7/2006	Info	OSH script (...)
12:23:10 PM	12/7/2006	Info	main_program: IBM WebSphere DataStage Enterprise Edition 8.0.0 (...)
12:23:12 PM	12/7/2006	Info	main_program: orchgeneral: loaded (...)
12:23:13 PM	12/7/2006	Info	Unduplicate: Creating sub-operator: <QSmatStats -workDir ./RT_QS23/W0S7 -f Unduplicate.txt> (...)
12:23:13 PM	12/7/2006	Info	main_program: APT configuration file: C:/IBM/InformationServer/Server/Configurations/default.apt (...)
12:23:13 PM	12/7/2006	Warning	StanUSNameAddrAreaFreq: When checking operator: When binding output schema variable "outRec": When binding ...
12:23:13 PM	12/7/2006	Warning	StanUSNameAddrAreaFreq: When checking operator: When binding output schema variable "outRec": When binding ...
12:23:13 PM	12/7/2006	Warning	Residuals: When checking operator: When validating export schema: At field "StanZip3": "null_field" length (4) must ma...
12:23:13 PM	12/7/2006	Warning	Residuals: When checking operator: A sequential operator cannot preserve the partitioning (...)
12:23:13 PM	12/7/2006	Warning	Clerical: When checking operator: When validating export schema: At field "StanZip3": "null_field" length (4) must matc...
12:23:13 PM	12/7/2006	Warning	Clerical: When checking operator: A sequential operator cannot preserve the partitioning (...)
12:23:13 PM	12/7/2006	Warning	Matched: When checking operator: When validating export schema: At field "StanZip3": "null_field" length (4) must mat...
12:24:05 PM	12/7/2006	Warning	Matched: When checking operator: A sequential operator cannot preserve the partitioning (...)
12:24:06 PM	12/7/2006	Info	Unduplicate,0: Variable: GenderCode_USNAM (...)
12:24:06 PM	12/7/2006	Info	Unduplicate,0: 0126366747 3 0 0 0.90 0.00 D 9.73 -3.32 (...)
12:24:06 PM	12/7/2006	Info	Unduplicate,0: Frequency table(s) will be used
12:24:06 PM	12/7/2006	Info	Unduplicate,0: Default weights calculated for values OUTSIDE table (...)
12:24:06 PM	12/7/2006	Info	Unduplicate,0: <Pass 1> Blocks processed: 1275 (...)
12:24:07 PM	12/7/2006	Info	Unduplicate,0: Variable: GenderCode_USNAM (...)
12:24:07 PM	12/7/2006	Info	Unduplicate,0: 0126366747 3 0 0 0.90 0.00 D 9.73 -3.32 (...)
12:24:07 PM	12/7/2006	Info	Unduplicate,0: Frequency table(s) will be used
12:24:07 PM	12/7/2006	Info	Unduplicate,0: Default weights calculated for values OUTSIDE table (...)
12:24:07 PM	12/7/2006	Info	Unduplicate,0: <Pass 2> Blocks processed: 121 (...)
12:24:09 PM	12/7/2006	Info	Unduplicate,0: *** Output Statistics For UNDUPLICATE *** (...)
12:24:09 PM	12/7/2006	Info	Unduplicate,0: 2843 data records & 1599 match records joined
12:24:09 PM	12/7/2006	Info	Residuals,0: Export complete; 1244 records exported successfully, 0 rejected.
12:24:09 PM	12/7/2006	Info	Clerical,0: Export complete; 0 records exported successfully, 0 rejected.
12:24:10 PM	12/7/2006	Info	Matched,0: Export complete; 1599 records exported successfully, 0 rejected.
12:24:10 PM	12/7/2006	Info	main_program: Step execution finished with status = OK.
12:24:12 PM	12/7/2006	Info	main_program: Startup time, 0:34; production run time, 0:26.
12:24:16 PM	12/7/2006	Info	Contents of phantom output file (...)
12:24:17 PM	12/7/2006	Info	Contents of phantom output file (...)
12:24:17 PM	12/7/2006	Info	Parallel job reports successful completion
12:24:18 PM	12/7/2006	Control	Finished Job Unduplicate.

Checkpoint

1. (T/F) DataStage Administrator executes jobs.
2. (T/F) DataStage Designer configures projects.
3. Which DataStage component displays objects in the designer database?

Checkpoint solutions

1. (T/F) DataStage Administrator executes jobs.
   Answer: False
2. (T/F) DataStage Designer configures projects.
   Answer: False
3. Which DataStage component displays objects in the designer database.
   Answer: the repository view

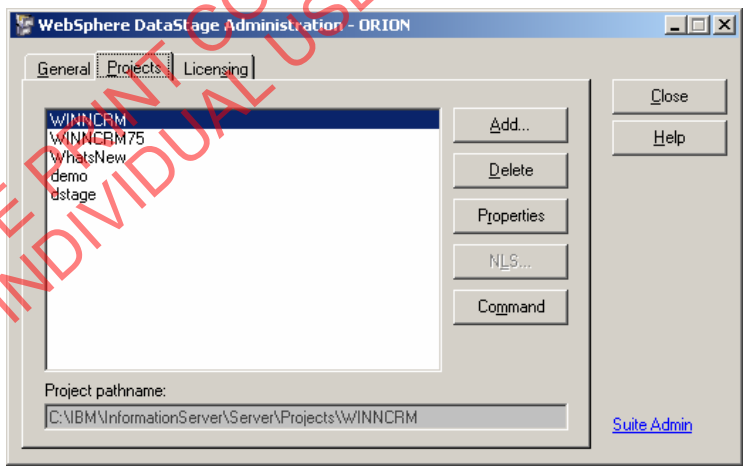
Unit summary

Having completed this unit, you should be able to:

- Describe the Information Server components
- List the DataStage clients
- Describe a typical DataStage configuration

Exercise 3: configure QualityStage project

- Create a project using Administrator
- Set project properties
  - General defaults
  - Environment variables
  - Security groups and roles



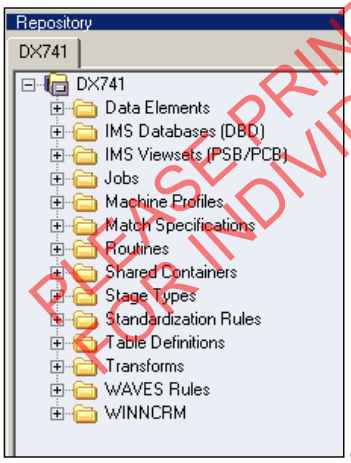

Developing with QualityStage
PLEASE PRINT COURSE MATERIALS.
FOR INDIVIDUAL USE ONLY.
© Copyright IBM Corporation 2007
The course materials are provided for internal use only by the individual receiving the materials.
The materials may not be modified, copied, distributed or transferred without the express prior written consent of IBM.
4.0.3

Unit objectives

- After completing this unit, you should be able to:
  - Define data files and field definitions
  - Build DataStage Jobs
  - Deploy and run jobs
  - Locate and review results

QualityStage application

- Could comprise one or more projects
- Project components
  - Jobs
  - Stages
  - Data File Definitions
    - Meta data
  - Designer repository view shows project components



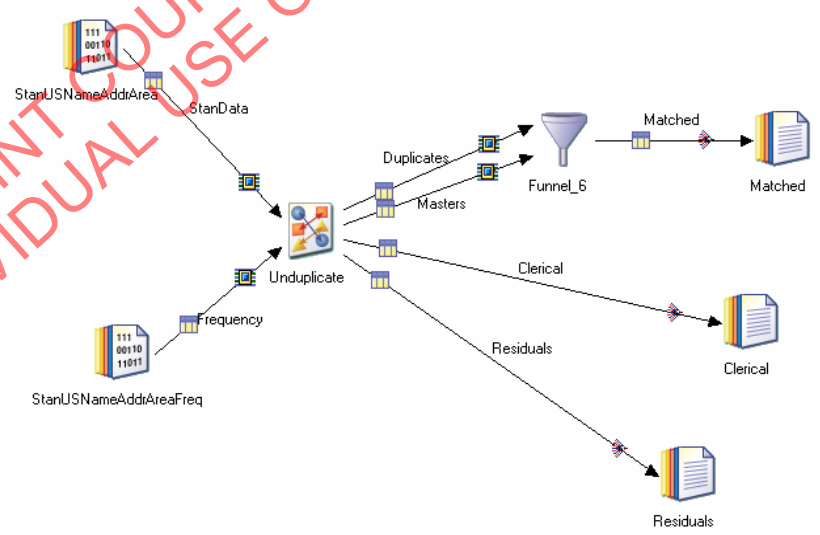
Project components are displayed in the repository view as a hierarchy of folders.

Job definition

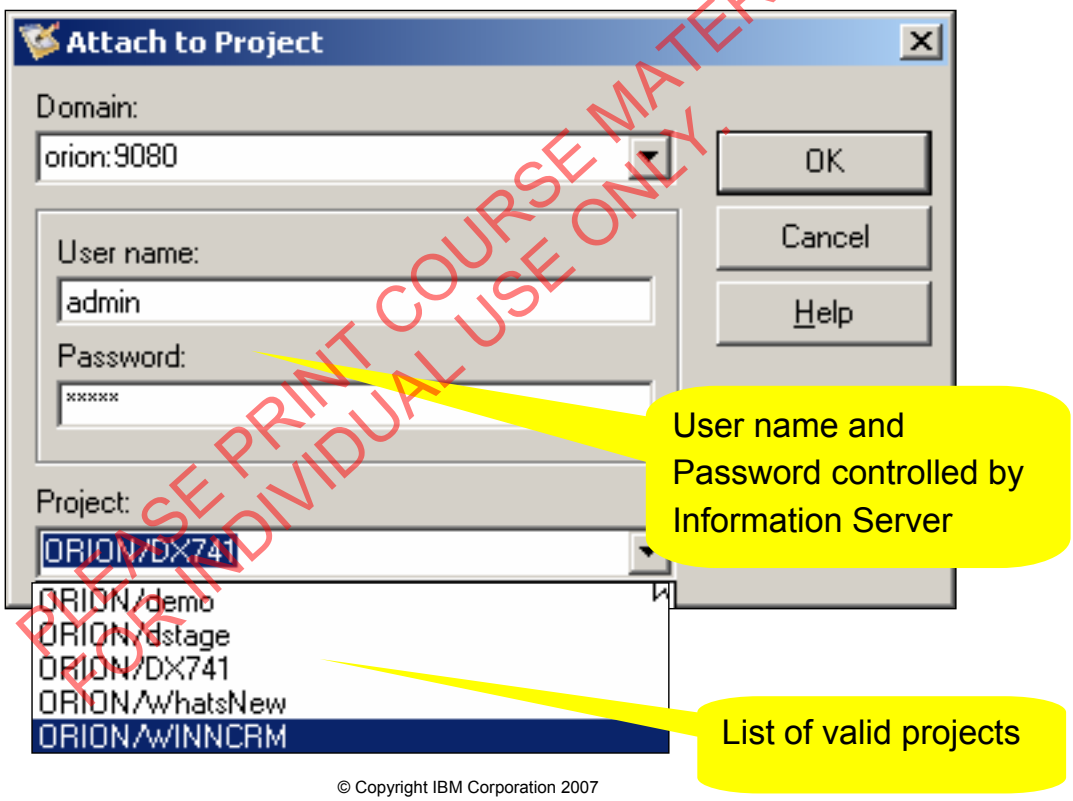
- A job is an executable DataStage/QualityStage program
- Created by job compilation
- Jobs can be run in batch or in real time

Job development overview

- Designer
  - Import or enter file definitions and meta data defining your sources and targets
  - Add stages and links defining the process or task
  - Compile the job
  - Run the job
  - Review results files
- Server
  - Runs the job
  - View job log



Log onto project in Designer or Director



Logon service is provided by the Information Server backbone.

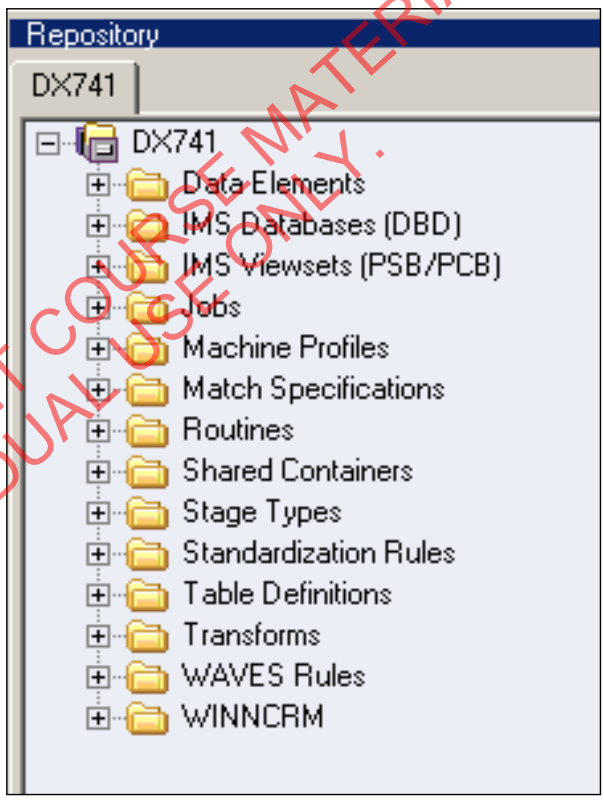
Designer repository components

- Database which stores
  - Data file definitions
  - Job designs
  - Standardization rules
  - Callable components such as buildops
  - Containers
  - Data connection objects

Project structure

Repository view

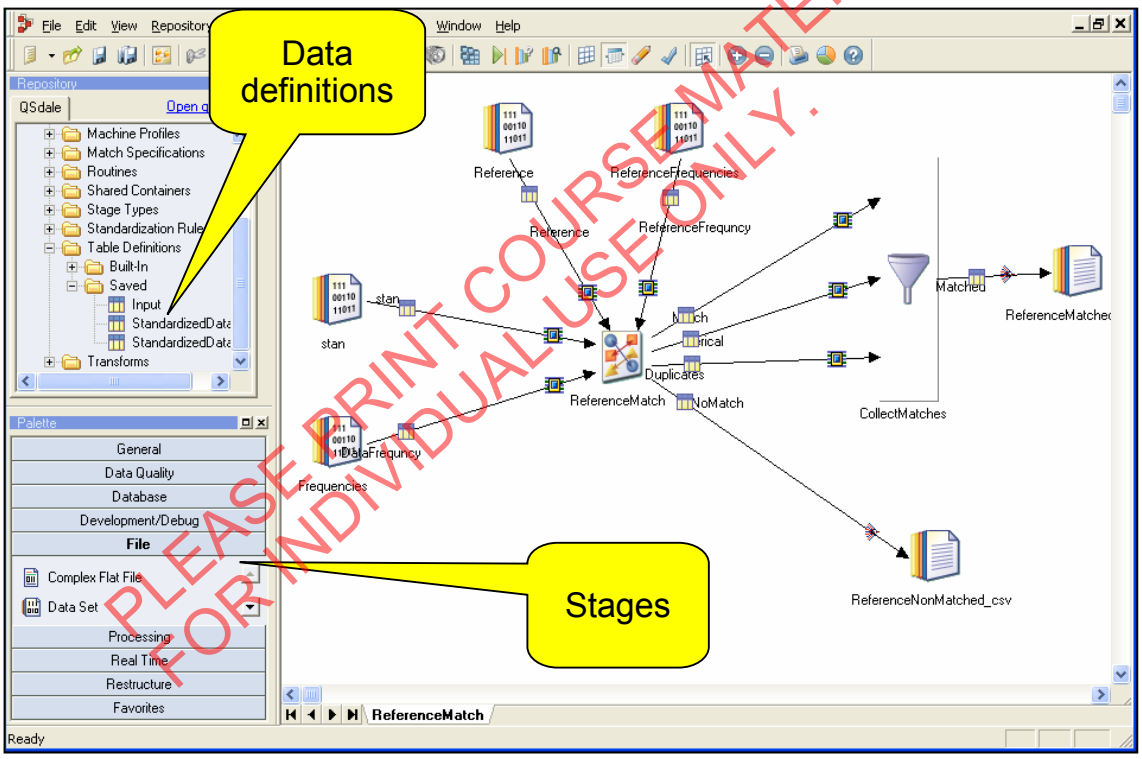
In Designer



DataStage components are arranged in a folder structure.

User can create new folders.

DataStage design environment



The DataStage Designer has full import/export features – both meta data and DataStage components – as well as job designs.

QualityStage jobs have a DataStage visual representation. The above job is a two file reference match.

Jobs are comprised of stages and links.

Stages are functional units and links indicate the flow of data.

To build a job:

Drag stages from palette

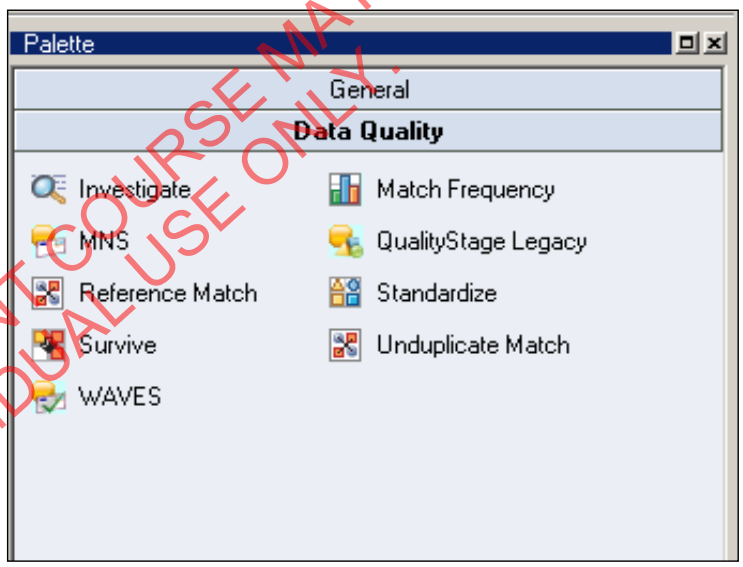
Draw links between the stages

Data definitions

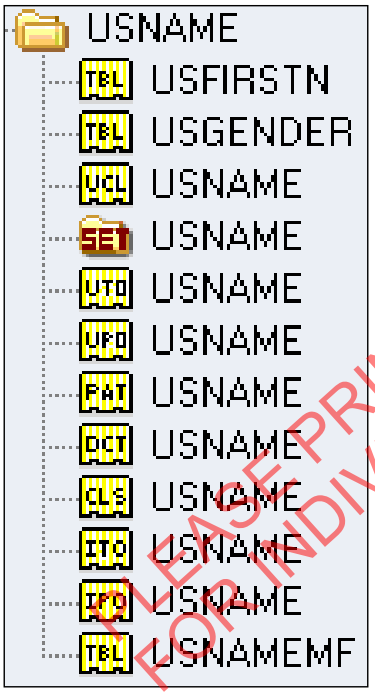
- Entered or loaded via DataStage import mechanisms
  - Sequential file
  - ODBC
  - Plug-ins
  - MetaBrokers
- New and redefined columns can be added on the data flow via Transformer stage

Data Quality folder

- Stages are the building blocks
- Focused in function
- All phases of data quality:
  - Investigate
  - Standardize
  - Match Frequency
  - Match
    - Unduplicate Match
    - Reference Match
  - Survive



Standardization rule sets

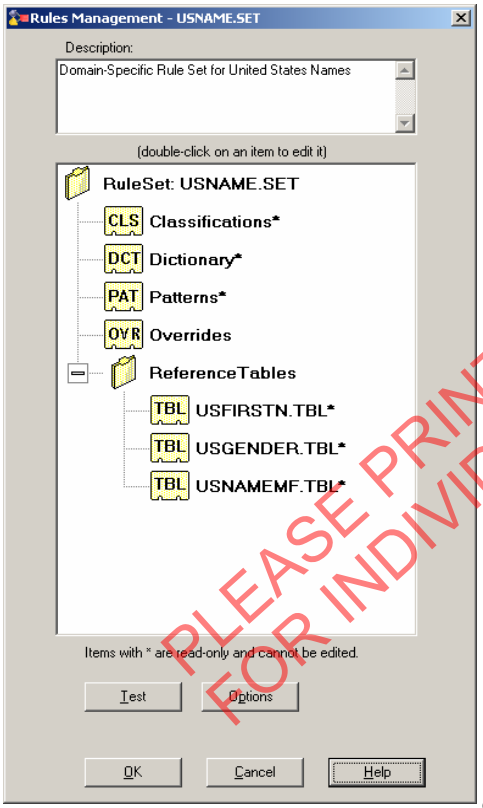


Rule set for USNAME

- Pre-defined rules for parsing and standardizing:
  - Name
  - Address
  - Area (City, State and Zip)
- Multi-national address processing
- Validate structure:
  - Tax ID
  - US Phone
  - Date
  - Email
- Append ISO country codes
- Rule sets are stored in the repository and provisioned to the job execution area

Provisioning copies the rule set from the repository to the job execution area.

Rule set components

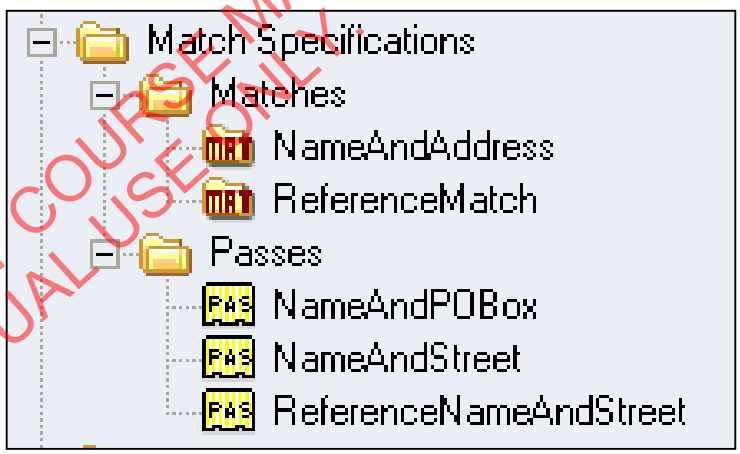


- Can modify some rule set components
- Test rule sets
- Copy rule sets

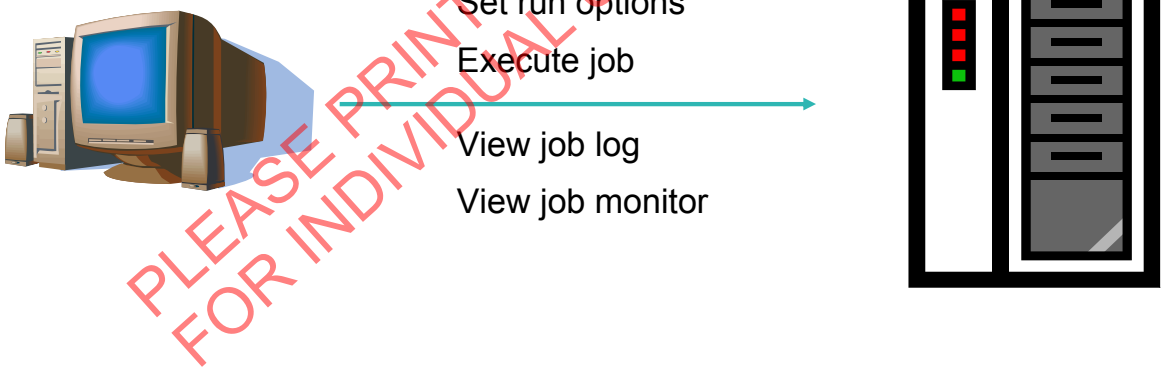
Rule sets are a group of control files that determine the standardization process.

Match Specifications in the DataStage Repository

- Created using the Match Designer
- Allows online testing of match criteria



Executing a job via Director

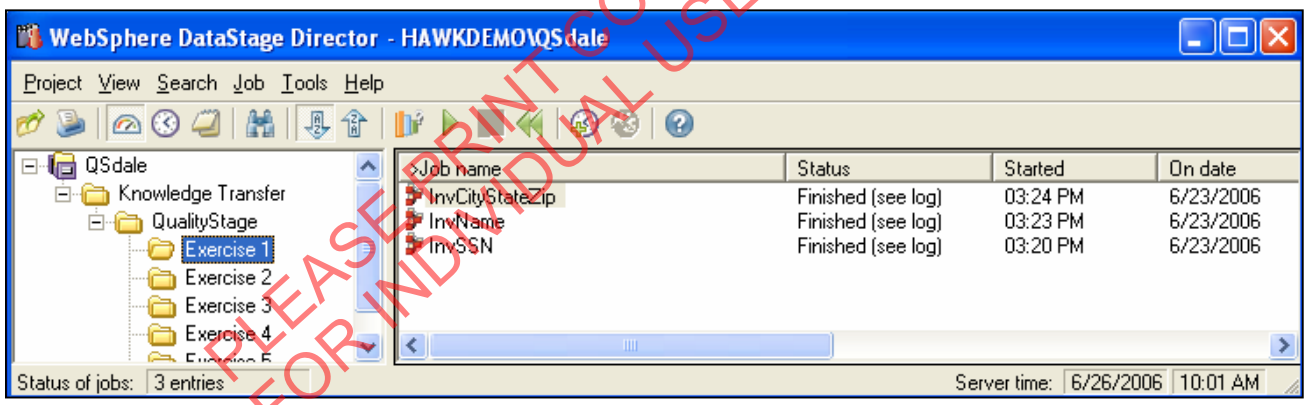


A job can also execute via a shell script independent of the Director.

Running a job in Director

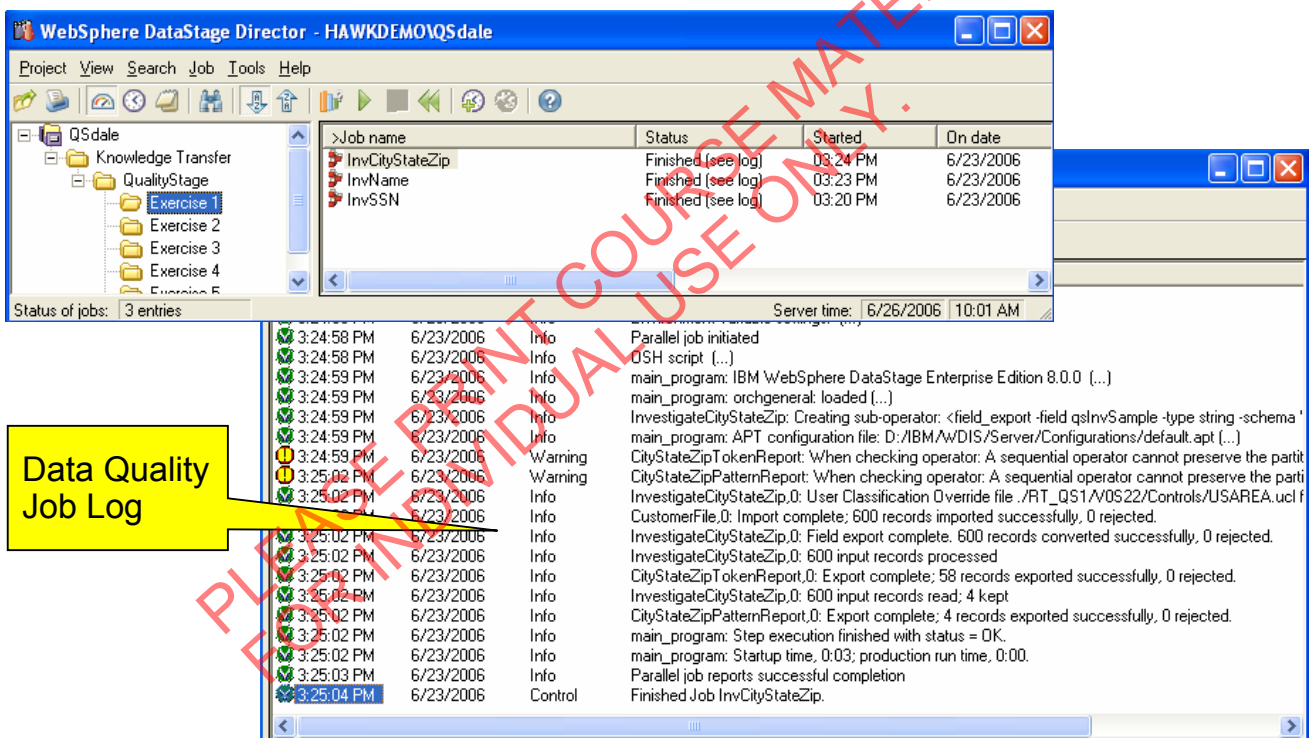
- Director
  - Client GUI for running jobs
    - Windows 2000+, XP
    - View job logs and monitor
    - Job scheduling

Job status view



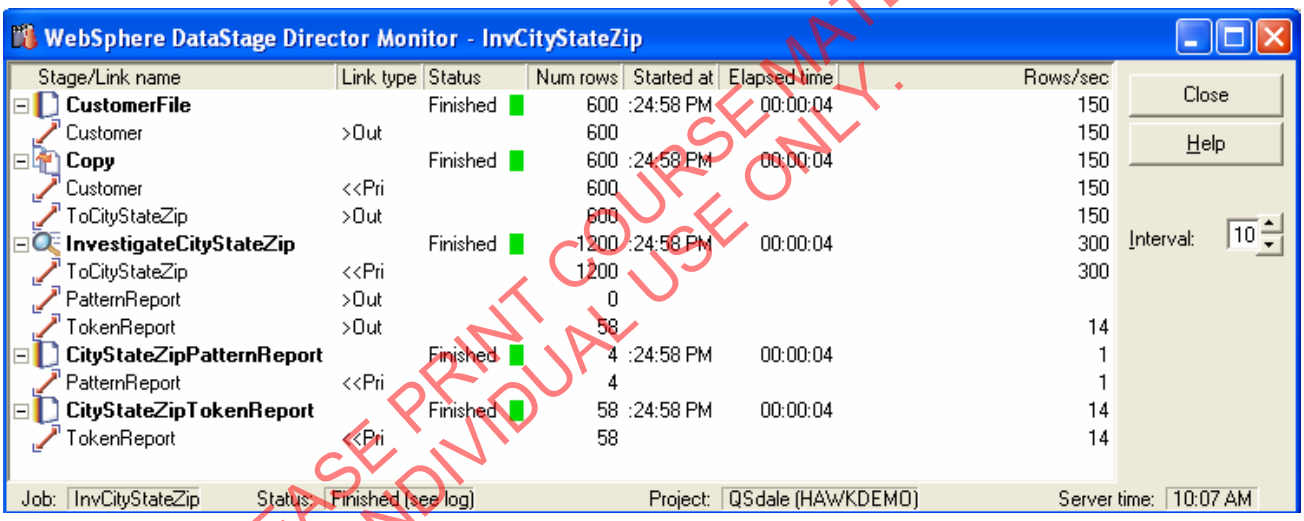
QualityStage jobs are data flow oriented; the direction of the arrows shows the flow of data. Stages process the data.

Execution environment



QualityStage jobs now produce the standard DataStage job log. The log should be viewed for statistics.

Job Monitor statistics



Job development process

- Define data files
  - Enter or import meta data
- Define job
  - Draw stages and links
  - Set stage properties
  - Compile
- Run the job
- Review results

Checkpoint

1. (T/F) The job monitor displays link statistics.
2. (T/F) The job log is viewed in DataStage Designer.
3. What protocol is used for communication between the DataStage clients and server?

Checkpoint solutions

1. (T/F) The job monitor displays link statistics.
   Answer: True
2. (T/F) The job log is viewed in DataStage Designer.
   Answer: False
3. What protocol is used for communication between the DataStage clients and server?
   Answer: TCPIP

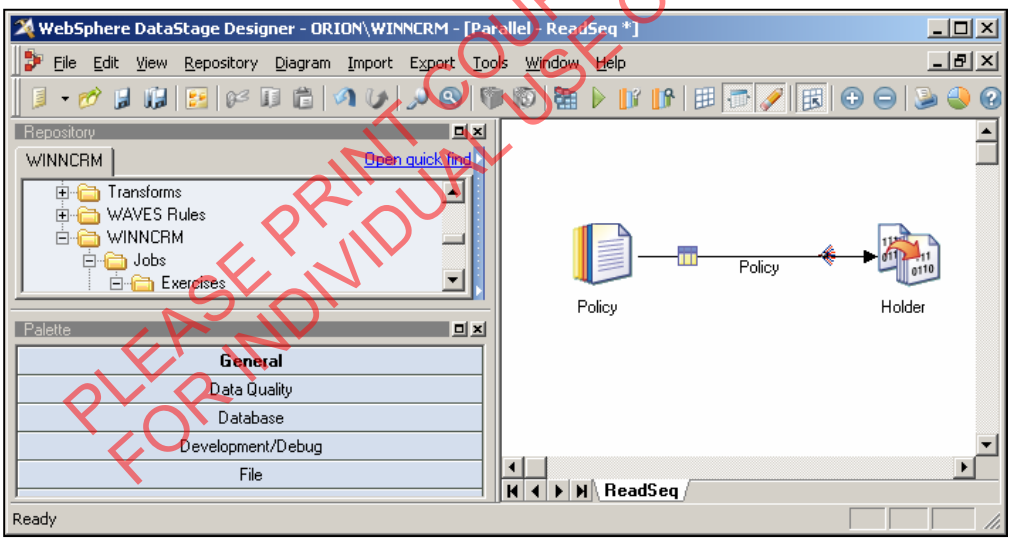
Unit summary

Having completed this unit, you should be able to:

- Define data files and field definitions
- Design jobs
- Deploy and run jobs
- Locate and review results

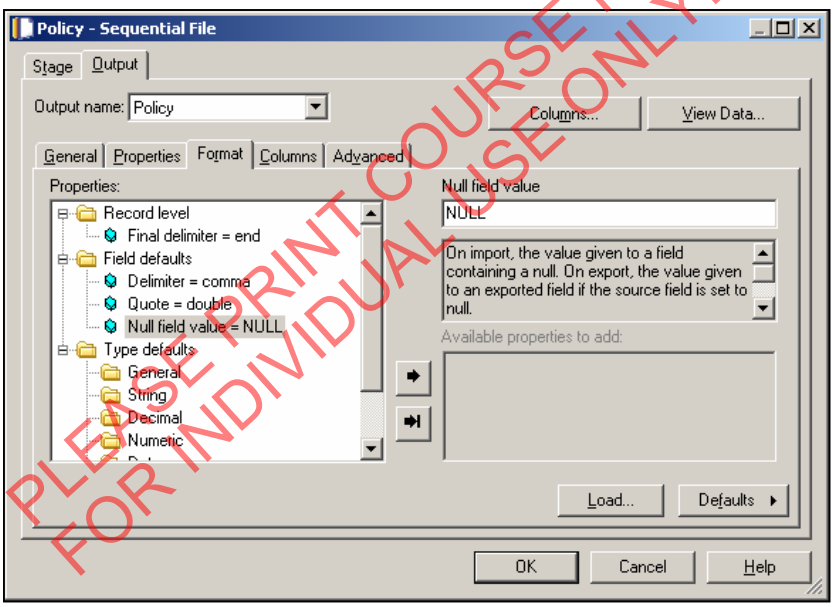
Exercise 4: Import meta data

- DataStage import mechanisms
  - DataStage components
    - Any object built in DataStage, such as jobs, table definitions, match specifications



Exercise 5: Build and run DataStage job

- Read sequential file
  - Must use format tab to handle nulls




IBM
Investigation
PLEASE PRINT COURSE MATERIALS.
FOR INDIVIDUAL USE ONLY.
© Copyright IBM Corporation 2007
The course materials are provided for internal use only by the individual receiving the materials.
The materials may not be modified, copied, distributed or transferred without the express prior written consent of IBM.
4.0.3

Unit objectives

- After completing this unit, you should be able to:
  - Define data investigation
  - Build Investigate stages
  - Use character discrete, concatenate, and word investigations to analyze data fields
  - Locate and review results

Investigation

- Verify the domain
  - Review each field and verify the data matches the meta data
- Identify data formats, missing and default values
- Identify data anomalies
  - Format
  - Structure
  - Content
- Discover “unwritten” business rules
- Identify data preparation requirements

Investigate stage

- Features
  - Analyze free-form and single domain fields
  - Provide frequency distributions of distinct values and patterns
- Investigate methods
  - Character Discrete
  - Character Concatenate
  - Word

Four Common Methods:

Character discrete – Inv multiple single-domain fields independently

Type C – View the character values

Type T – View the field format or “Template”

Type X – Ignore characters

Investigate methods

Method	Why
Character Discrete	Analyzing field values, formats, and domains
Character Concatenate	Cross-field correlation, checking logic relationships between fields
Word Investigation	Identifying free-form fields that may require parsing and discovery of key words for classification

Investigate terminology

Field Masks → Options that represent the data. Options: Character (C), Type (T), Skipped (X)

Tokens → Individual units of data

Character Mask	Usage
C	For viewing the actual character values of the data
T	For viewing the pattern of the data
X	For ignoring characters

Examples of tokens:

string – 914 Josephine St

Token 1 – 914

Token 2 – Josephine

Token 3 - St

Field mask examples

Token	Mask	Result
02116	CCCCC	02116
02116	CCCXX	021
01832-4480	TTTTTTTTT	nnnnn-nnnn
XJ2 6EM	TTTTTT	aanbnaa
(617) 338-0300	CCCCCCCCCCCCCC	(617) 338-0300
617-338-0300	TTTTTTTTTTT	nnn-nnn-nnnn
6173380300	CCCXXXXXXXXX	617
(617)3380300	CCCXXXXXXXXX	(61

Character discrete: field mask (C)haracter

- Usage: Domain quality
  - View the contents of each field to verify that the data values match the field labels
- Mechanism: Investigate stage
  - Generates Reports for frequency and pattern references

These reports provide the quantitative understanding of data values that will permit correlation of the various spellings, misspellings, abbreviations or other representation of data values

▲Also note any anomalies (anything suspect: out of range or defaults values), and how often each anomaly occurs?

▲Percent Populated per field: Note how often the field is populated

▲How many formats “templates” exist for the data?

▲The cardinality of the field: The number of distinct values

▲The frequency distribution: How often does each format occur?

▲How often does “data in the wrong domain” occur?

Character discrete - character results

qsInvColumnName	qsInvPattern	qsInvSample	qsInvCount	qsInvPercent
SourceSystem	A	A	1584	52.8966
SourceSystem	H	H	366	12.6207
SourceSystem	L	L	1000	34.4828
PolicyNumber	003668461	003668461	1	0.0344828
PolicyNumber	003775219	003775219	2	0.0689655
PolicyNumber	004281148	004281148	1	0.0344828
PolicyNumber	004793986	004793986	1	0.0344828
PolicyNumber	004804210	004804210	1	0.0344828

Investigates one or more single-domain fields

Each field is treated independently for frequency count and pattern reporting

Report names

- jobp.FRQ - sorted by frequency in descending order
- jobp.SRT - sorted alphabetical in ascending order
- job.PAT - reference file

Character discrete: field mask (T)ype

- Usage: Data formats (patterns):
  - View the format of field which contain that you suspect may follow or conform to a specific format, e.g., dates, PIN, Tax ID, account numbers.
- Generates reports for frequency and pattern references

These reports provide the quantitative understanding of data values that will permit correlation of the various spellings, misspellings, abbreviations or other representation of data values

▲Also note any anomalies (anything suspect: out of range or defaults values), and how often each anomaly occurs?

▲Percent Populated per field: Note how often the field is populated

▲How many formats “templates” exist for the data?

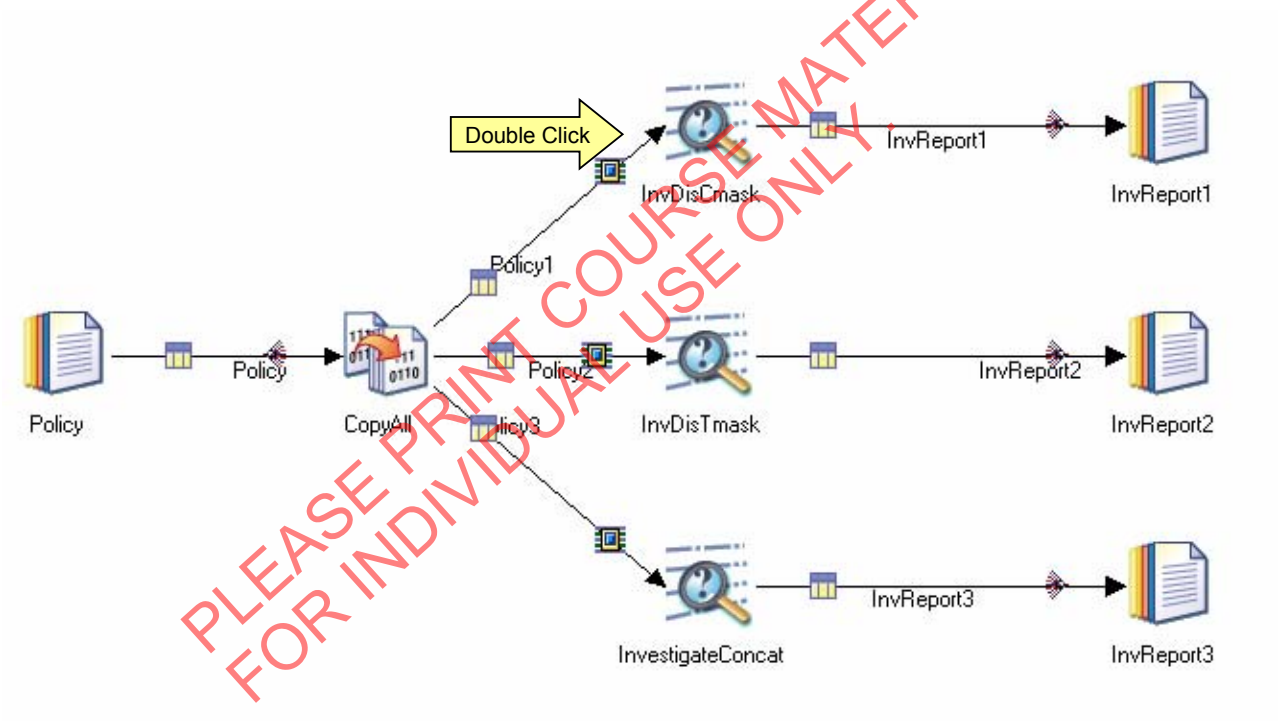
▲The cardinality of the field: The number of distinct values

▲The frequency distribution: How often does each format occur?

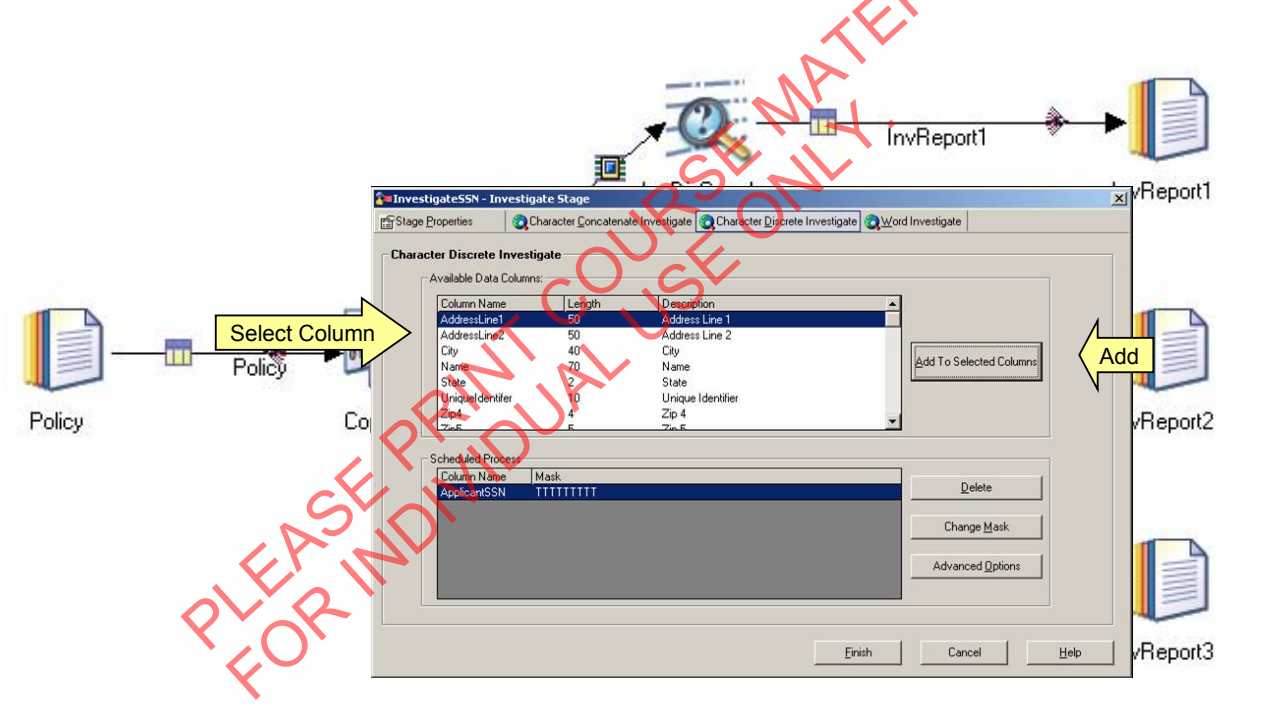
▲How often does “data in the wrong domain” occur?

Investigation Implementation

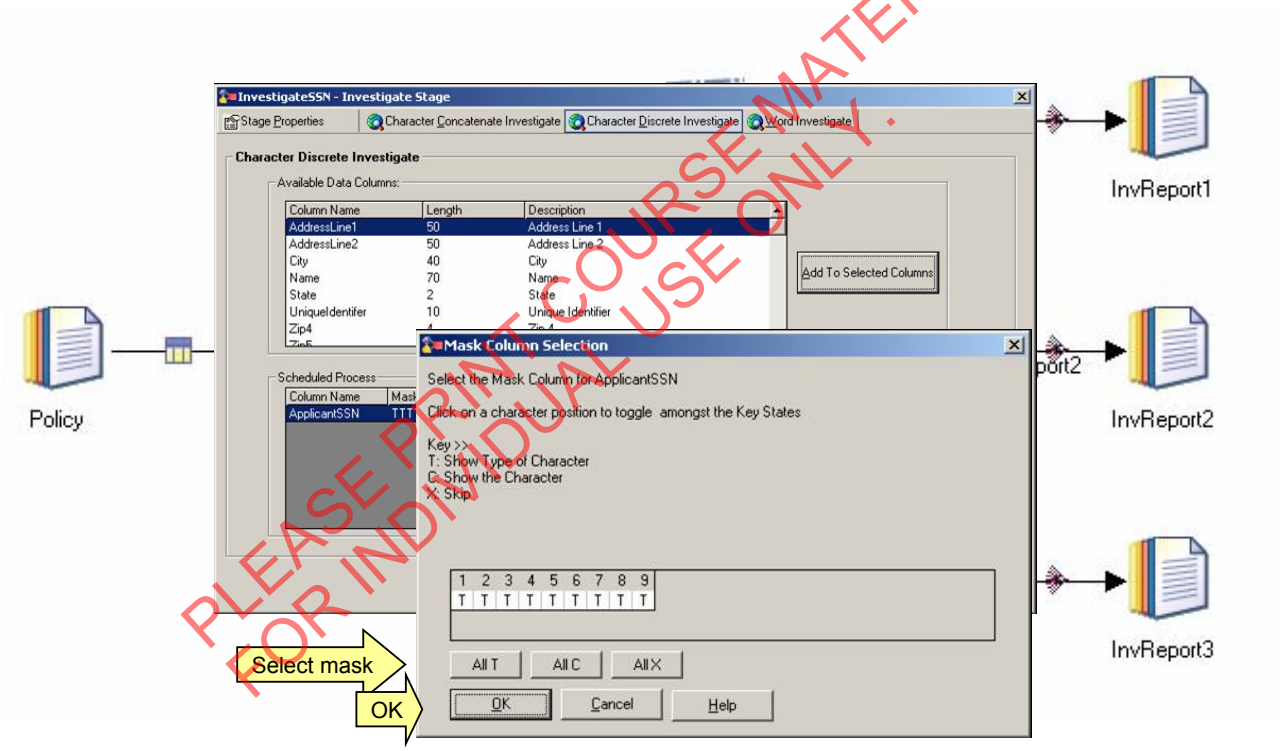
QualityStage Investigation job – character



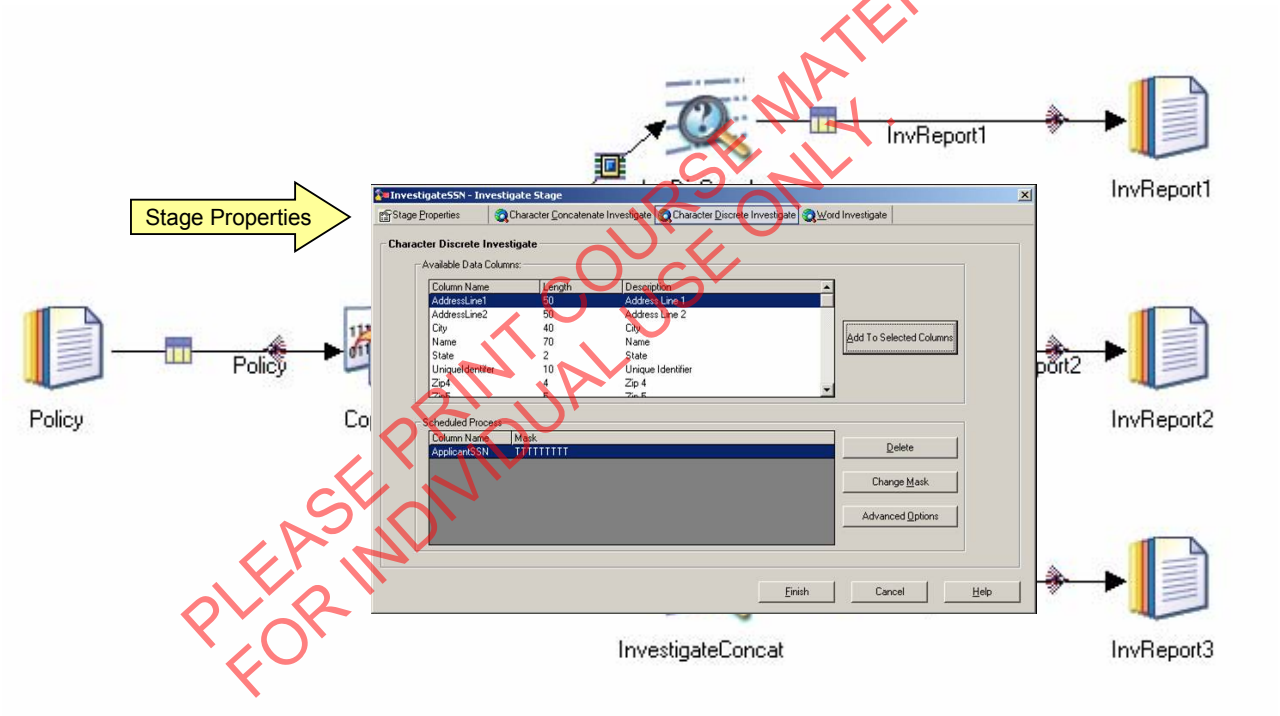
Investigation - Character



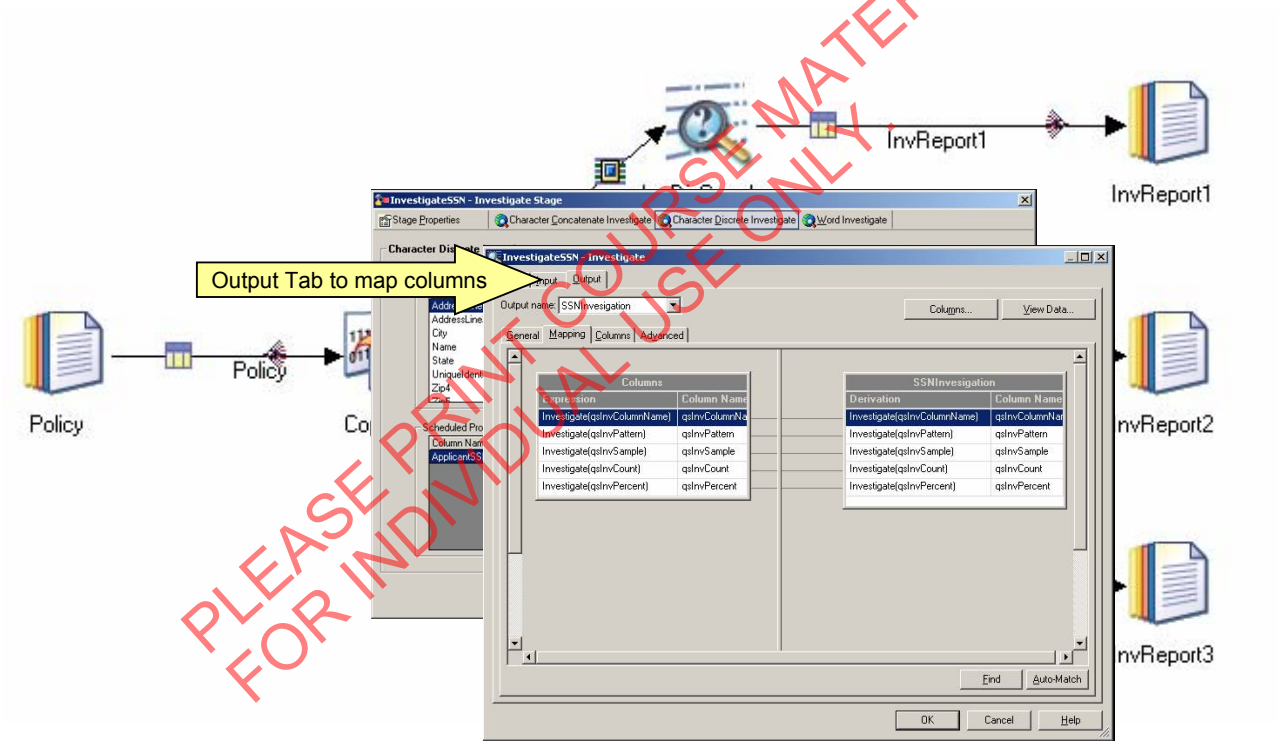
Investigation - Character



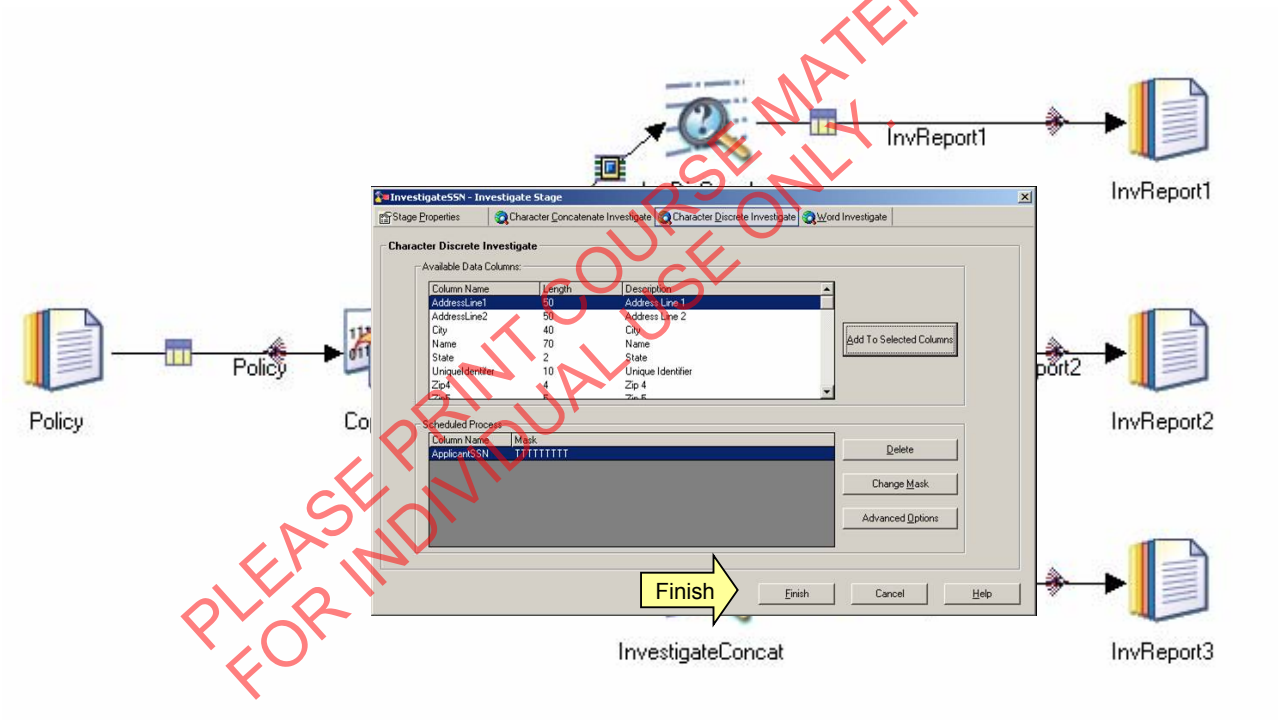
Investigation - Character



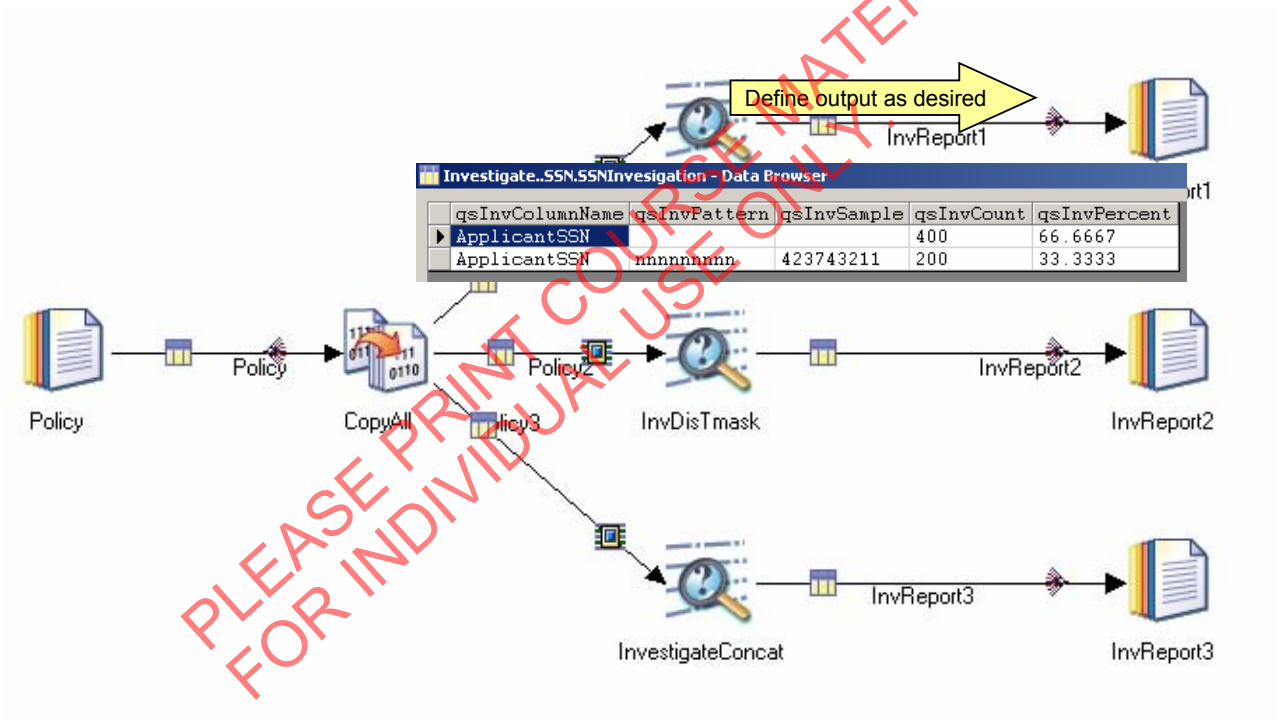
Investigation - Character



Investigation - Character



Investigation - Character



Character concatenate

- Identify Field Relationships
  - Investigate one or more fields to uncover any relationship between the field values.
  - Uses combinations of character masks
  - Generates Reports for frequency and pattern references

These reports provide the quantitative understanding of data values prevalence that will permit correlation of the various spellings, misspellings, abbreviations or other representation of data values

▲Also note any anomalies (anything suspect: out of range or defaults values), and how often each anomaly occurs?

▲Percent Populated per field: Note how often the field is populated

▲How many formats “templates” exist for the data?

▲The cardinality of the field: The number of distinct values

▲The frequency distribution: How often does each format occur?

▲How often does “data in the wrong domain” occur?

Character concatenate results

DOB and DOD Fields

qsInvColumnName	qsInvSample	qsInvCount	qsInvPercent
DOB+DOD		1184	40.8417
DOB+DOD	0000000000000000	2	0.0689893
DOB+DOD	1908121500000000	1	0.0344947
DOB+DOD	1909010100000000	1	0.0344947
DOB+DOD	1914060200000000	3	0.103484
DOB+DOD	1915033000000000	2	0.0689893
DOB+DOD	1915071600000000	1	0.0344947
DOB+DOD	1917022500000000	2	0.0689893
DOB+DOD	1917033100000000	2	0.0689893

Word investigate

- Usage: Pattern free-form fields and lexical analysis
  - To view the pattern of the data within a freeform text field and parse it into individual tokens
- QualityStage process
  - Apply rule sets to free-form fields
  - Discover parsing requirements
  - Discover patterns in data
  - Generate reports for word frequency, pattern frequency distributions, and word classification

These reports provide the quantitative understanding of data values prevalence that will permit correlation of the various spellings, misspellings, abbreviations or other representation of data values

▲Also note any anomalies (anything suspect: out of range or defaults values), and how often each anomaly occurs?

▲Percent Populated per field: Note how often the field is populated

▲How many formats “templates” exist for the data?

▲The cardinality of the field: The number of distinct values

▲The frequency distribution: How often does each format occur?

▲How often does “data in the wrong domain” occur?

Word investigation results

Pattern Reports

```
^D?T                    639 N MILLS AVE
^D?S                    306 W MAIN ST
^D?T                    3142 W CENTRAL AVE
^?T                     843 HEARD AVE
```

Word Frequency Reports

```
0000000869 ST
0000000791 RD
0000000622 STE
0000000566 AVE
```

Word Classification Reports

```
ABBOTT          ABBOTT          ?       ;0000000001
ABERCON         ABERCON         ?       ;0000000001
ABERCORN        ABERCORN        ?       ;0000000007
ABERDEEN        ABERDEEN        ?       ;0000000001
```


Parses free-form data into individual tokens
Tokens are classified to create patterns
Uses a set of rules for parsing and classifying the tokens
Discover tokens (key words) to be added to the classification table such as name prefixes, business terminology, street types, new abbreviations for cities
Create patterns of data tokens with the field context
Identify spelling, misspellings and representations of data
Identify parsing requirements for the conditioning process

Patterns Reports: Distinct patterns within the field
Pattern Reports
List of all patterns sorted by frequency (p.frq)
List of all patterns sorted alphabetical (p.srt)
List of each token and it's associated pattern (.pat)

Word Frequency Reports: The frequency distribution of distinct values
List of all alpha sorted by frequency (c.frq)
List of all alpha sorted alphabetically (a.frq)
May include numerics and mixed tokens

Word Classification Reports: The frequency distribution of "classified" and "unclassified" words
List of classified alpha (u.dlt)
List of not-classified alpha (n.dlt)
All alpha listed in the classification table are considered classified alpha

Rule sets

- Rules for parsing, classifying, and organizing data
- Rule Set Domains
  - Country processing
  - Pre-processing
  - Domain Processing
    - Name: Business and Personal
    - Street Address
    - Area: Locality, City, State and Zip/Postal codes
  - Multinational Address Processing

Parsing

- Parse free-form data with the SEPLIST and a STRIPLIST
  - SEPLIST - Any character in the SEPLIST will separate tokens, and become a token itself
  - STRIPLIST - Any character in the STRIPLIST will be ignored in the resulting pattern
- The SEPLIST is always applied first

There is a default seplist and striplist on the Word investigation: Advanced Options screen

Remember investigation is about discovery, feel free to changes the seplist and strip to experiment and identify the “best” parsing parameters”

The seplist and striplist only allow “simple” parsing (parsing by encountering the presence of a single character. More complex parsing can be done in the next phase, Conditioning.

If you really aren’t sure how to parse the data then be very conservative, that is separate by a space and only strip out a space. Add in more characters after analysis of the results

The rule is “Whenever in doubt don’t strip out”. If a character sometimes adds context and sometimes does not then DON’T strip out the character. Stripping the character loses context in all cases. Often we will choose the separate by the character but not strip it out:

Examples:
½ if we strip out the / we won’t know if this started as ½ or 12 or two independent digits 1 and 2.

C/O: again if we strip out the / we may not realize that this was an abbrev of “care of” and interpret the token as “company”

Parsing example

Example: **120 Main St. N.W.**

SEPLIST "¬."
STRIPLIST "¬"

Token1	*Token2*	*Token3*	*Token4*	*Token5*	*Token6*	*Token7*	*Token8*
120	**Main**	**St**	**.**	**N**	**.**	**W**	**.**

SEPLIST "¬"
STRIPLIST "¬."

Token1	*Token2*	*Token3*	*Token4*
120	**Main**	**St**	**NW**

SEPLIST "¬."
STRIPLIST "¬"

Token1	*Token2*	*Token3*	*Token4*	*Token5*
120	**Main**	**St**	**N**	**W**

¬ designates a space.

We talk about tokens back in the early parts of investigation. This slide helps make it more clear how we create "tokens"

Example

APT. is a period you can strip BUT $10.00 is not a period you can strip without changing the meaning of the data.

Data typing: classifying tokens

- Identify and type the token in terms of it's business meaning and value

MASK KEY:

N – Numeric token

A – Alpha token

AN – Mixed Token

120	Main	Street	Apt	6C
NNN	AAAA	AAAAAA	AAA	AN

PATTERN KEY(USNAME rule set):

^ – Numeric token

? – Unclassified alpha token

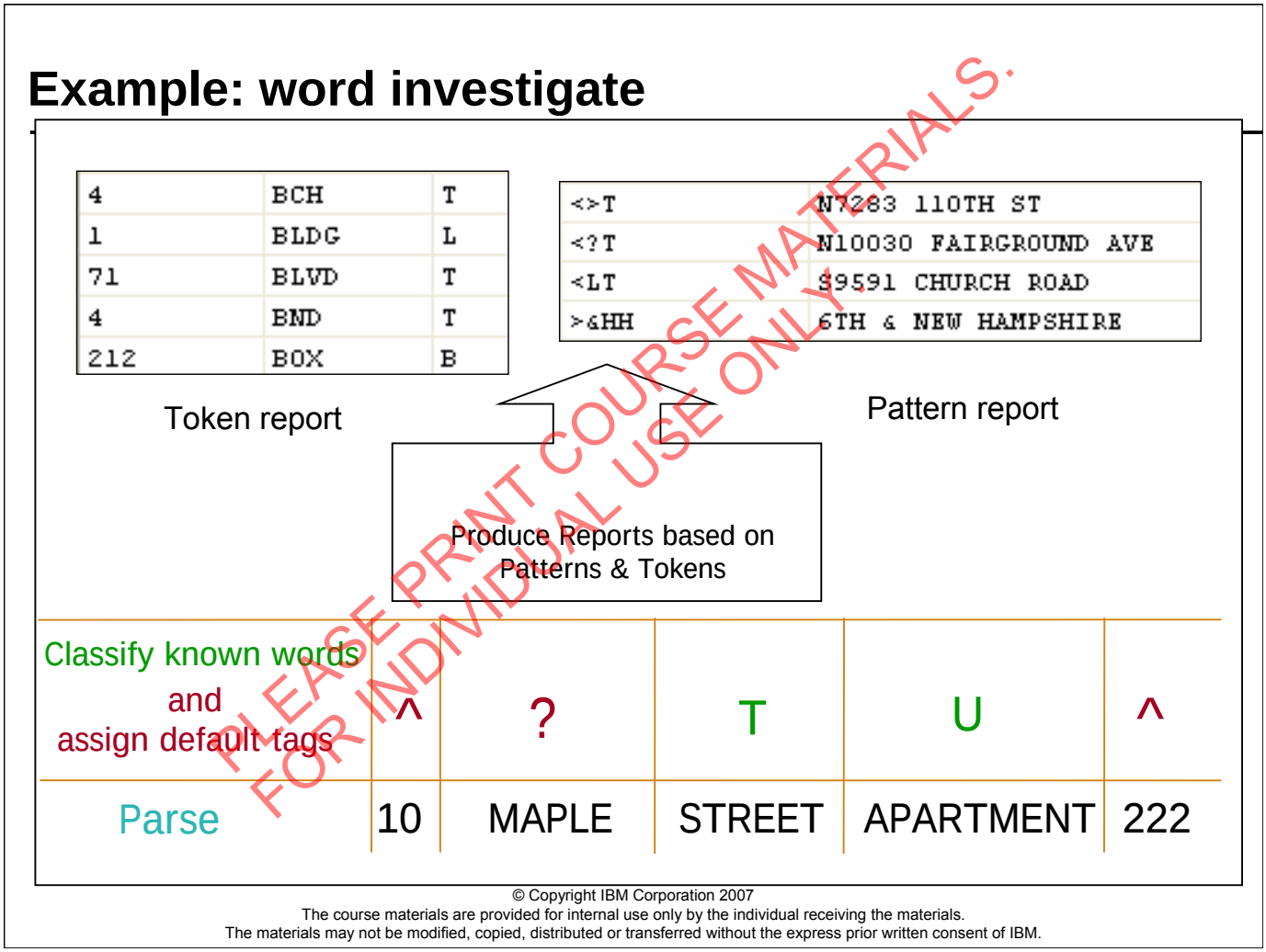
@, <, > – Mixed Token

T – Street Type

U – Unit Type

120	Main	Street	Apt	6C
^	?	T	U	>

Start with an example of simple classification. Is the data Alpha, Numeric or Mixed then introduce more sophisticated classification, the classification table.



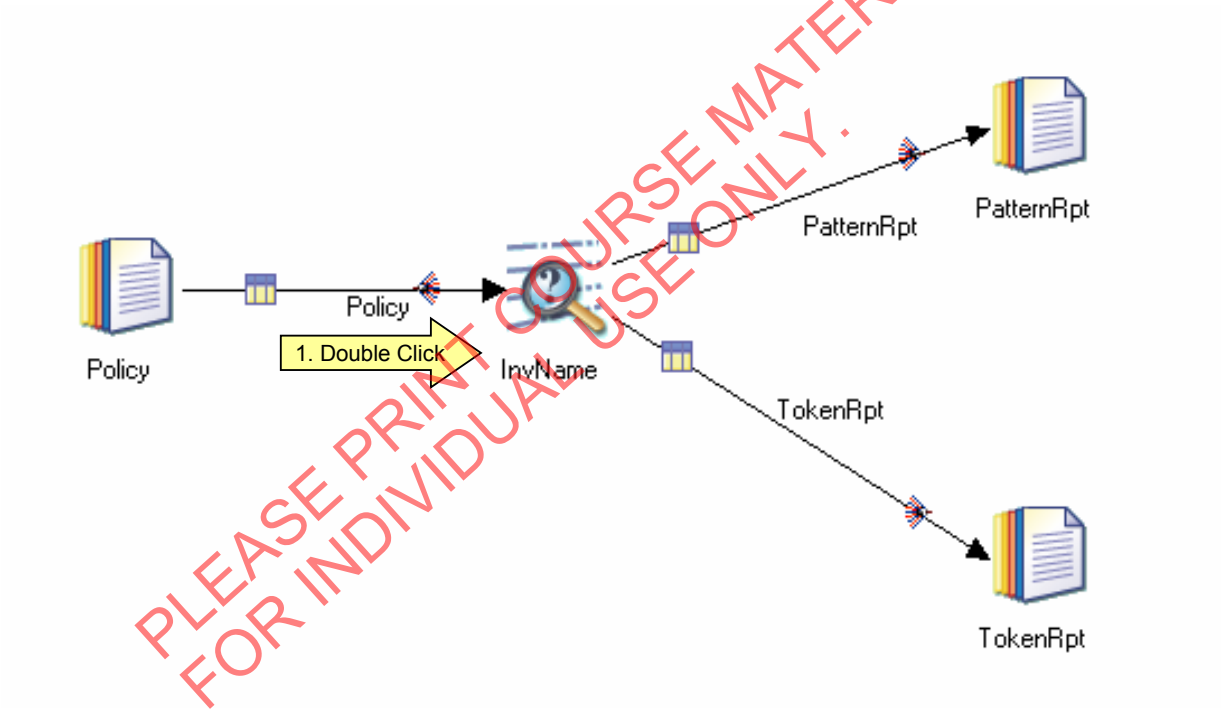
Illustrates the process for Parsing and classifying data tokens to create patterns

Example:

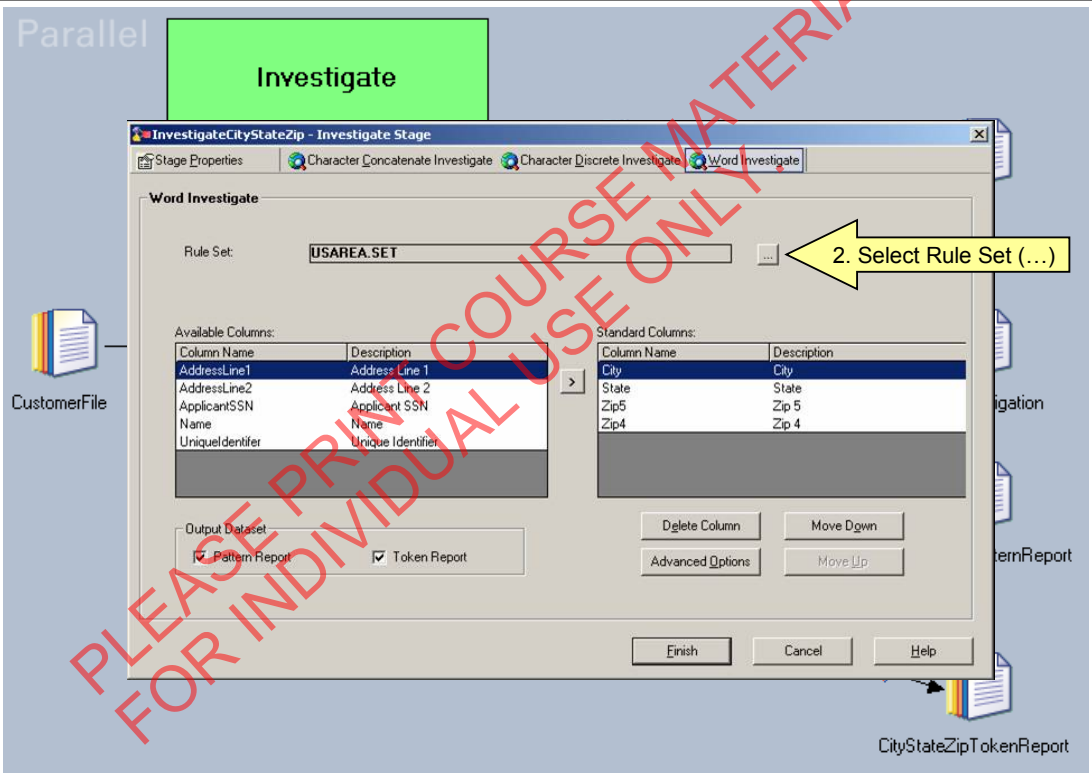
Rebuild	120	Apple	RD	APT	4B
DataType	HN	SN	ST	UT	UV
Pattern	^?TM>				
Classify2	^	?	T	M	>
Classify1	N	A	A	A	M
Parse	120	Apple	Road	Apart	4B

Start with the bottom line and build.

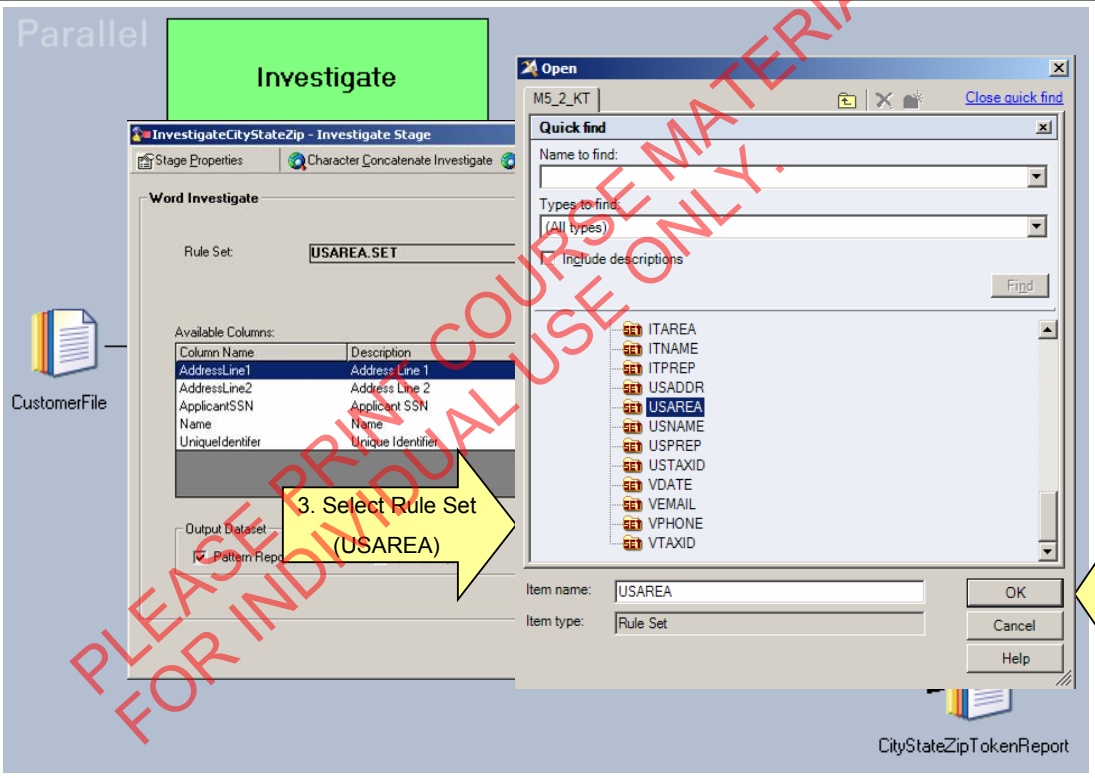
Investigation - Word



Investigation - Word

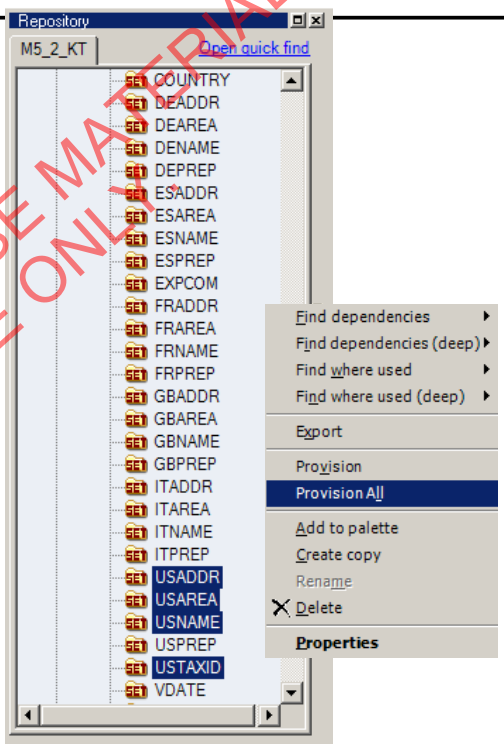


Investigation - Word

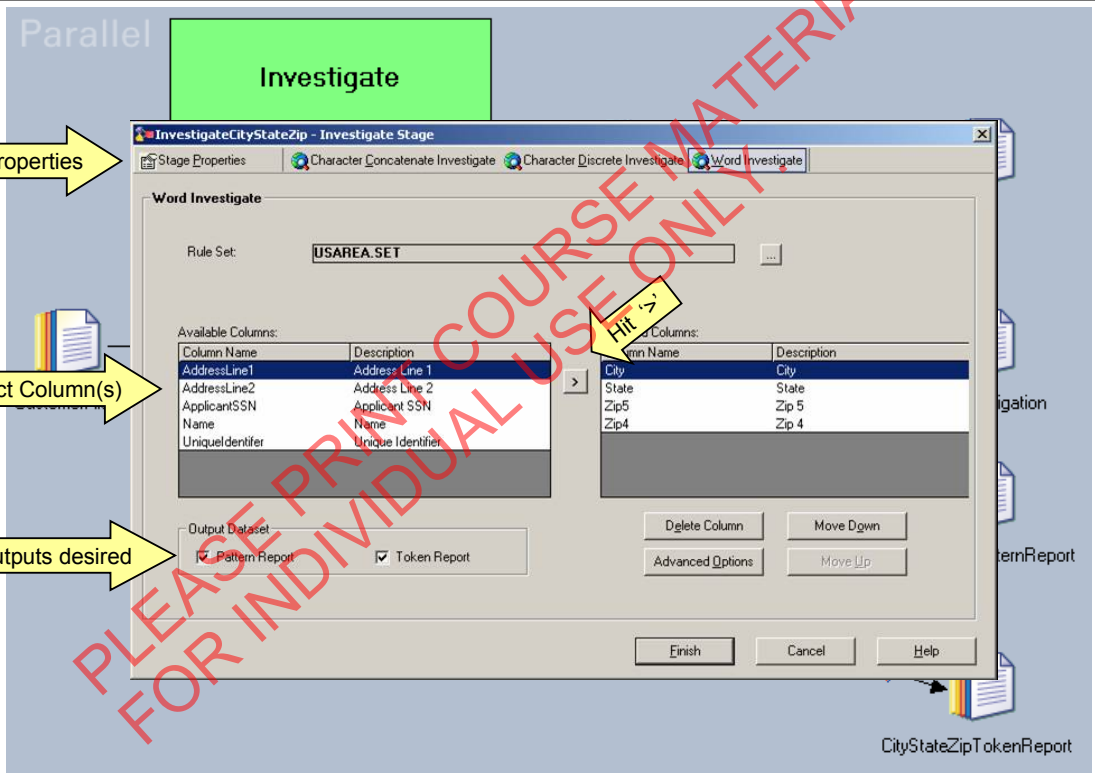


Provision Rules to be used

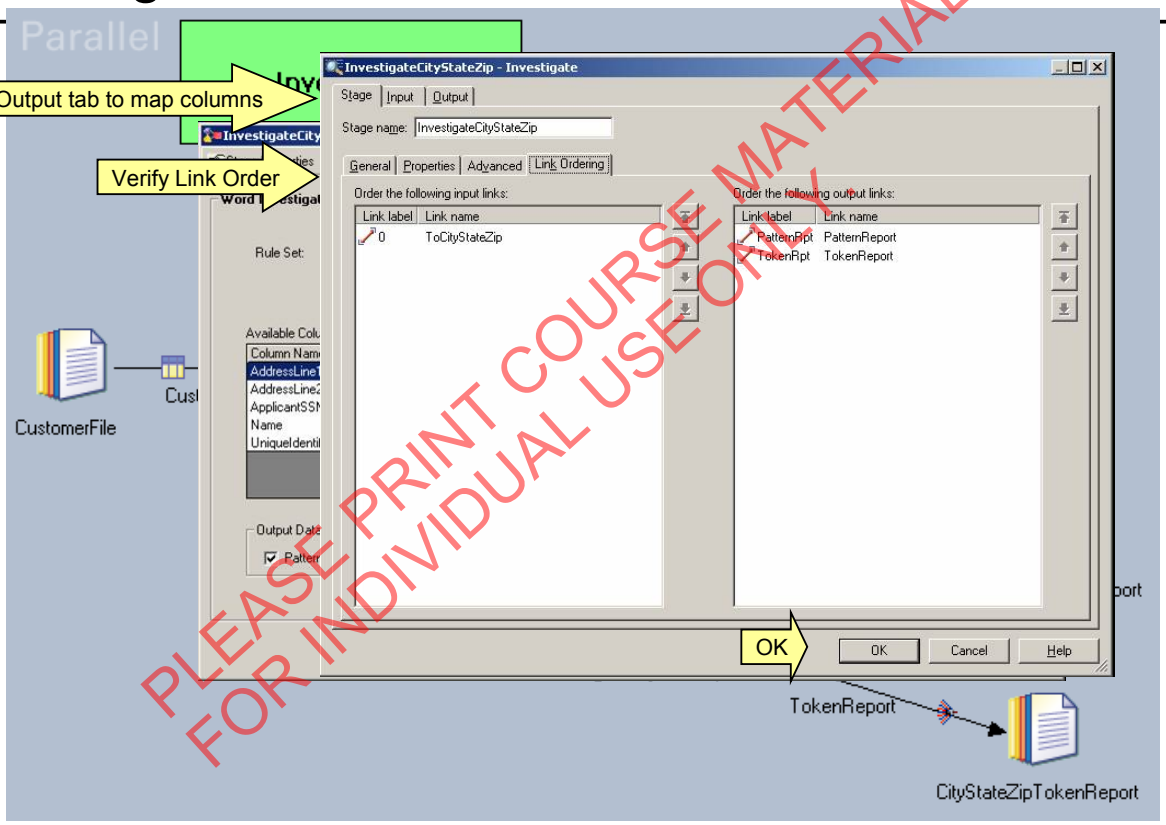
- Provisioning copies rules from repository to execution area
- Use ‘Provision all’



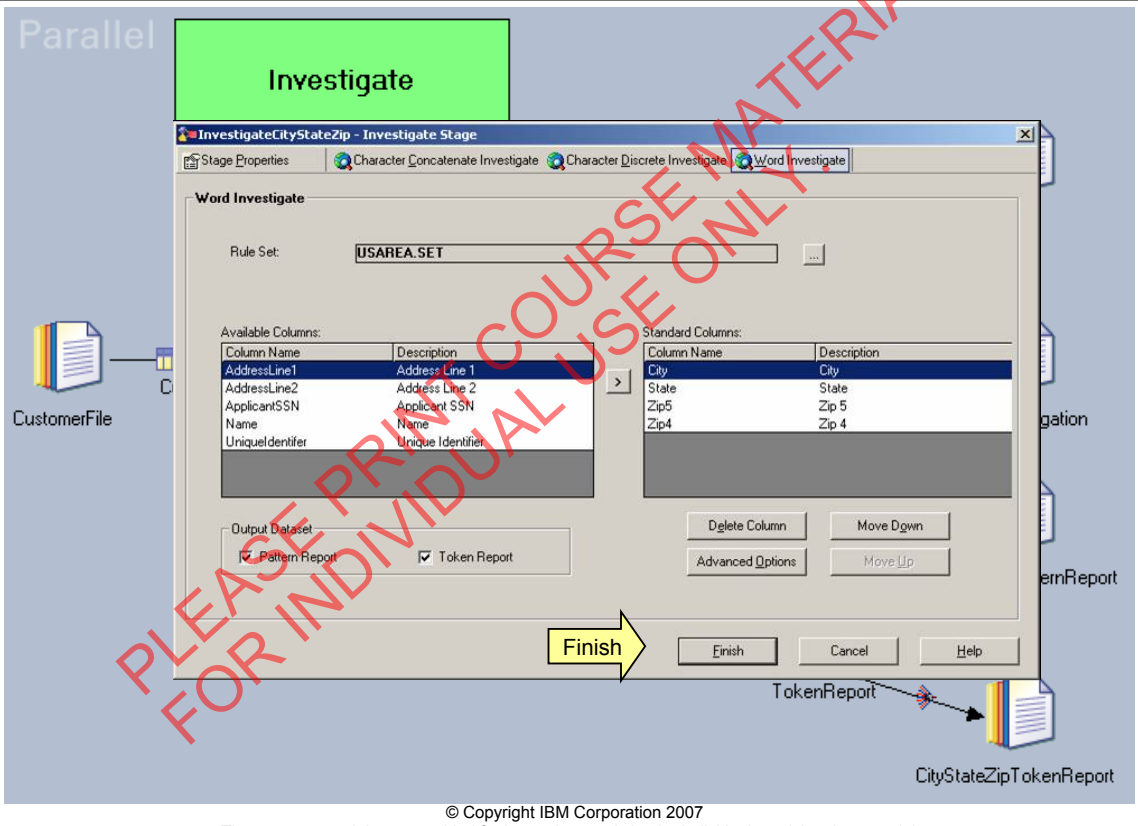
Investigation - Word



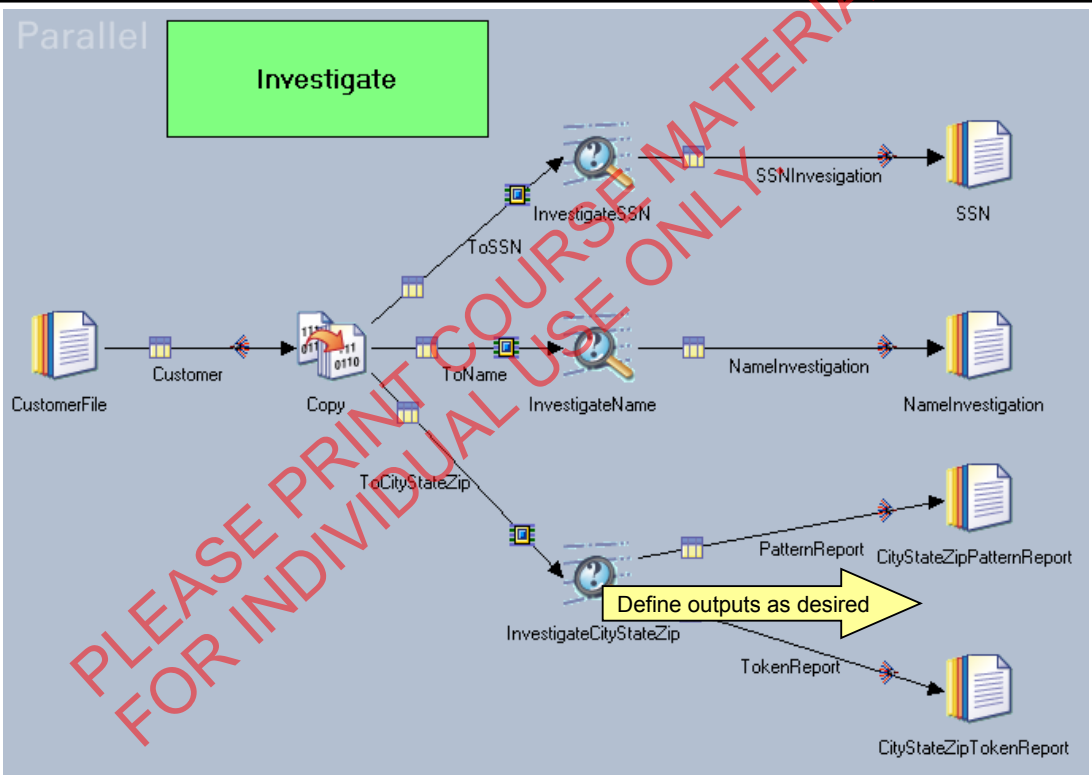
Investigation - Word



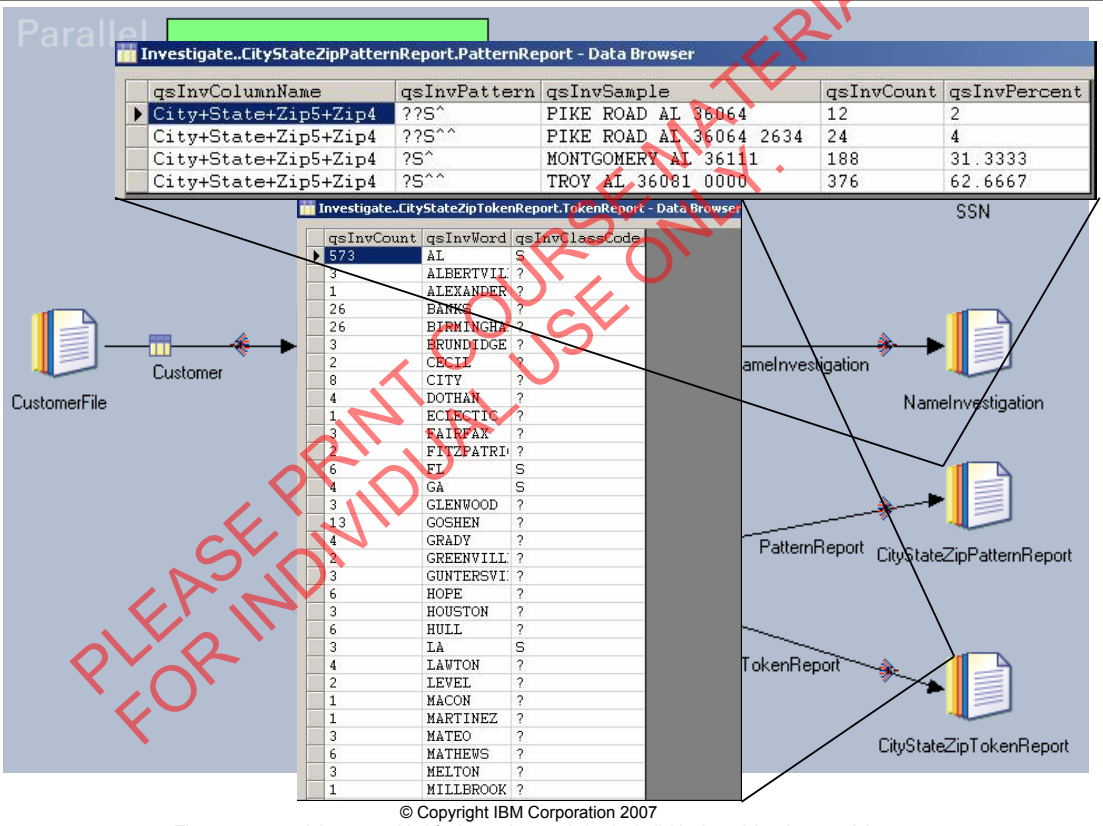
Investigation - Word



Investigation - Word



Investigation - Word



Data quality assessment

- Review and analyze each field for the following information:
  - How often is the field populated?
  - What are the anomalies and out-of-range values? How often does each one occur?
  - How many unique values were found?
  - What is the distribution of the data or patterns?
- Use Investigate results to:
  - Update business requirements
  - Define development plan and application design

Checkpoint

1. (T/F) Character discrete investigation examines a single domain.
2. (T/F) Word investigation examines a single domain.
3. Name the three character masks.

Checkpoint solutions

1. (T/F) Character discrete investigation examines a single domain.
   Answer: True
2. (T/F) Word investigation examines a single domain.
   Answer: False
3. Name the three character masks.
   Answer: C, T, and X

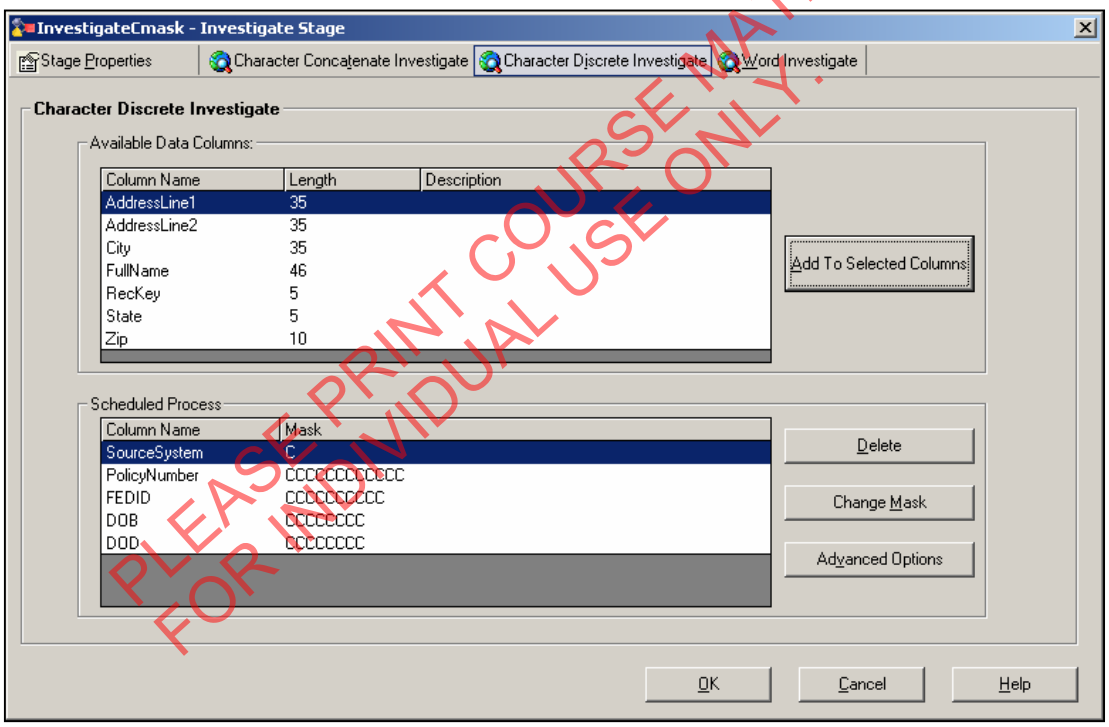
Unit summary

Having completed this unit, you should be able to:

- Define data investigation
- Build Investigate stages
- Use character discrete, concatenate, and word investigations to analyze data fields
- Locate and review results

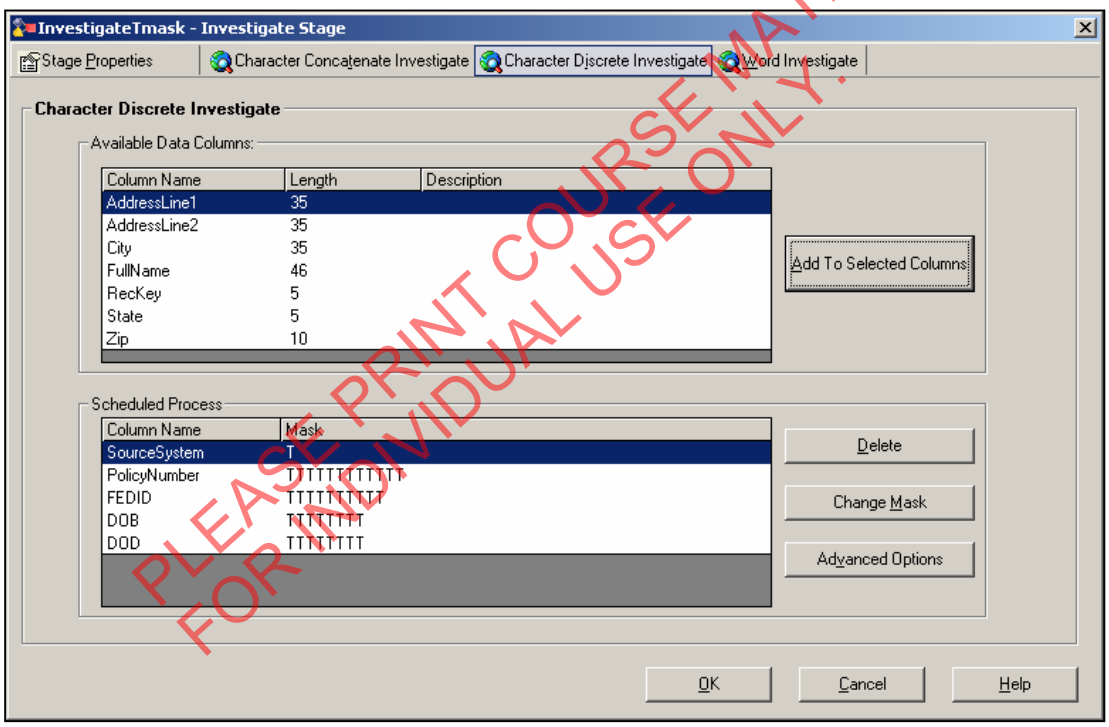
Exercise 6: Build investigate job

- Character with C mask



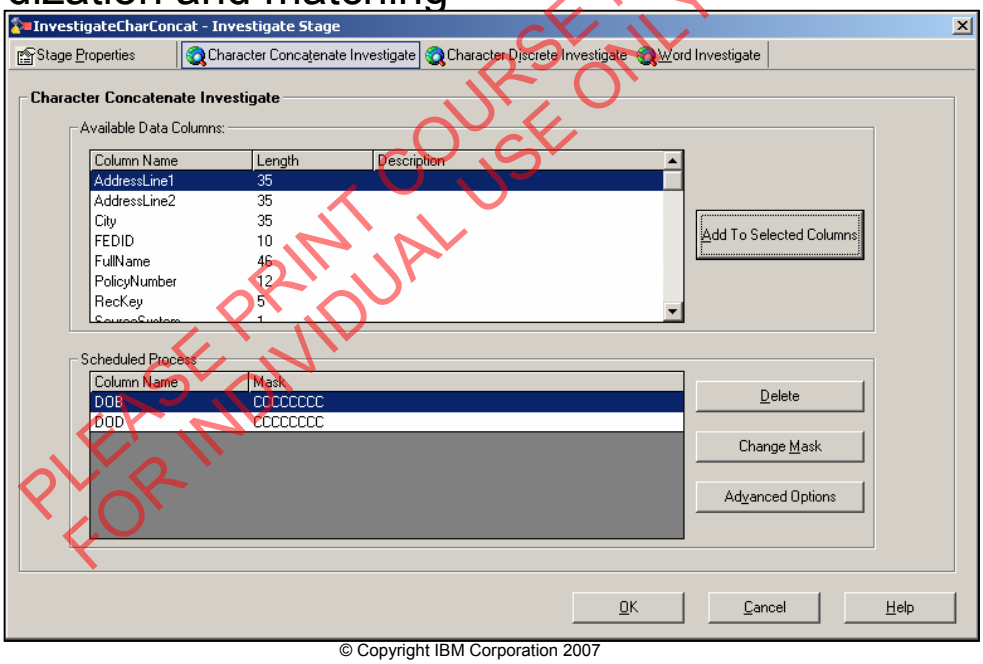
Exercise 7: Build investigate job

- Character with T mask



Exercise 8: Build investigate job

- Character concatenate
- Useful for auditing results of other processes, such as standardization and matching



Exercise 9: Build investigate job

- Word investigation