IBM TXSeries(CICS)

800-810-1818

IBM 中国软件部

引言

各大企业都渐渐认识到，要在当今不断变化的全球市场上，保持或提高自身的竞争能力，必须要用强有力的交易中间件来武装他们的企业，最大限度地提高企业的整体效益。

目录

第一章

IBM TXSeries(CICS)
基本概念与介绍

什么是中间件？

- 定义：中间件是介于应用与操作系统之间的系统软件
- 功能：应用以中间件为开发、运行的基准平台

什么是交易？

- 定义：交易是对某一应用操作序列的一个工作单元
- 特点(ACID)：
  - 原子性(Atomicity)
  - 一致性(Consistency)
  - 独立性(Isolation)
  - 永久性(Durability)

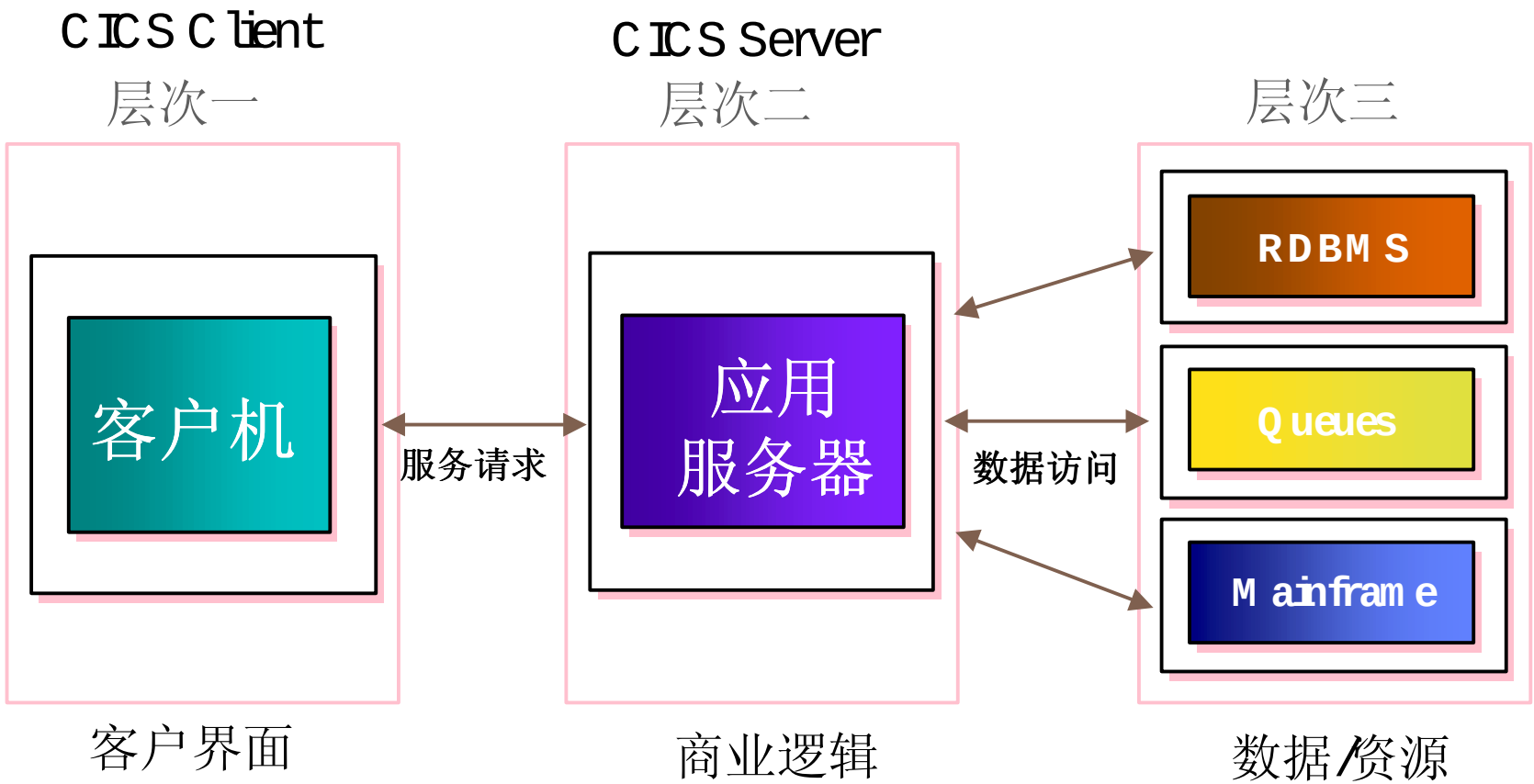
什么是联机交易系统？

- 定义：提供即时、在线的交易服务
- 特点：
  - 提供用户实时的交易请求与响应
  - 提供软硬件故障的交易系统恢复

什么是分布式交易系统?

- 定义：一个交易存取了不同逻辑上或物理上的数据或应用
- 特点：
  - 实时性
  - 多个数据库
  - 异种数据库
  - 分布式协同应用

什么是三层次客户/服务?

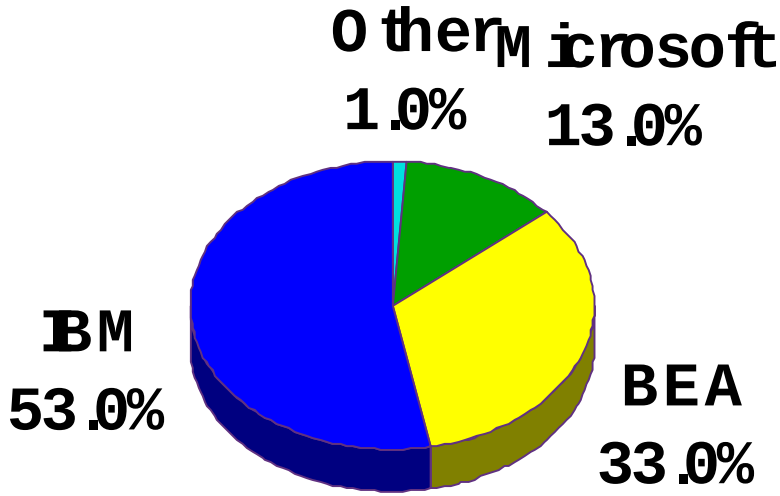

CICS Client
层次一
CICS Server
层次二
层次三
客户机
服务请求
应用
服务器
数据访问
RDBMS
Queues
Mainframe
客户界面
商业逻辑
数据/资源

IBM TXSeries(CICS)的历史

- 1969年IBM CICS第一版发布
- 1993年IBM推出了UNIX平台的CICS产品
- 1998年IBM发布了TXSeries(CICS)系列产品

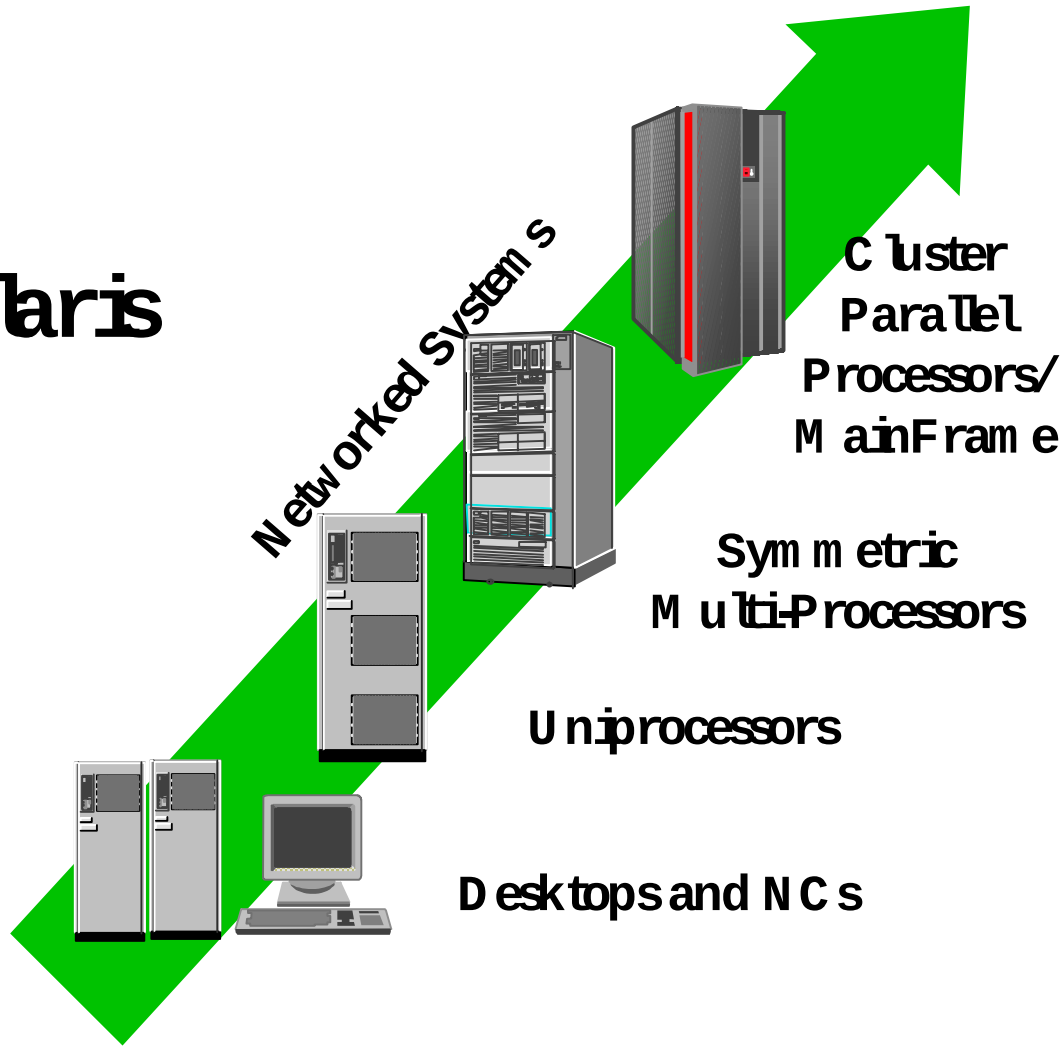
IBM TXSeries(CICS)业界领导

IBM TXSeries市场占有率
(1998年12月 Forrester Research)



IBM TXSeries(CICS)家族

- TXSeries for WinNT
- TXSeries for Aix
- TXSeries for Sun Solaris
- TXSeries for HP-UX
- CICS for OS/400
- CICS for VSE
- CICS for MVS
- CICS for OS/390



CICS Application Server

TXSeries包装

- CICS v4.2
- Encina v4.2
- CICS Clients
  - CICS Clients v2.0.2
  - CICS Internet Gateway v2.0.2
  - CICS Gateway for Java v1.1.3
  - CICS Link for Lotus Notes v2.0.2
- DE-Light v2.1
- MQSeries v5.0
- Domino.Go Webserver v4.6
- DCE Base or Runtime Services
  - v1.1 for Solaris
  - v2.0 for Windows NT or Gradient DCE
- DCE Cell Directory Server and Security Server
  - v2.1 for AIX
  - v1.1 for Solaris
  - v2.0 for Windows NT

CICS Application Server

IBM 承诺

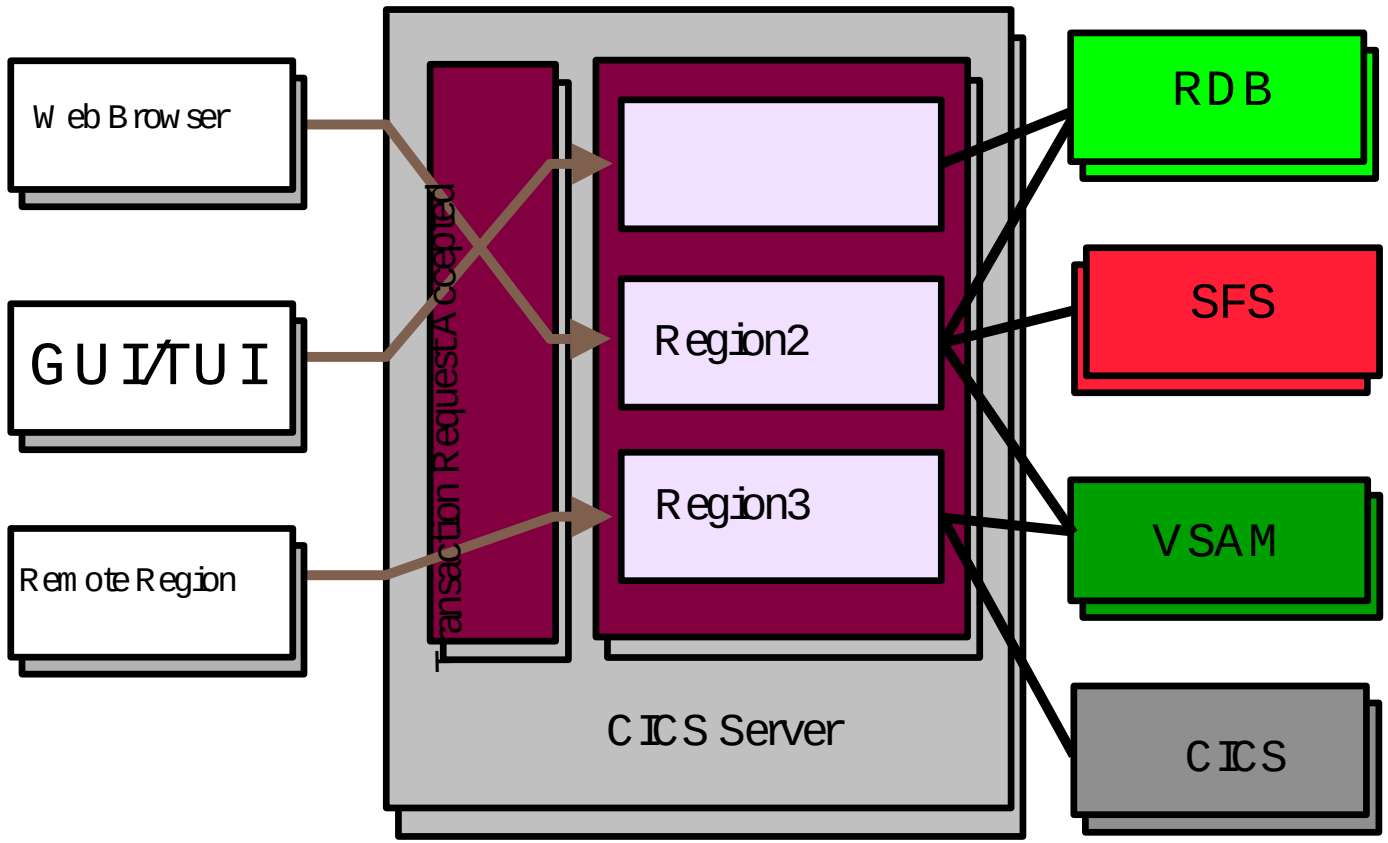
- IBM 支持和遵循大多数重要的工业标准和标准技术：
  - X/OPEN交易处理
  - OSF/DCE
  - TCP/IP
  - SNA
  - XPG
  - POSIX

- IBM 在本地化技术上全力支持
  - 产品的汉化
  - 800-810-1818汉语技术支持
  - 本地工程师远地在线或上点支持

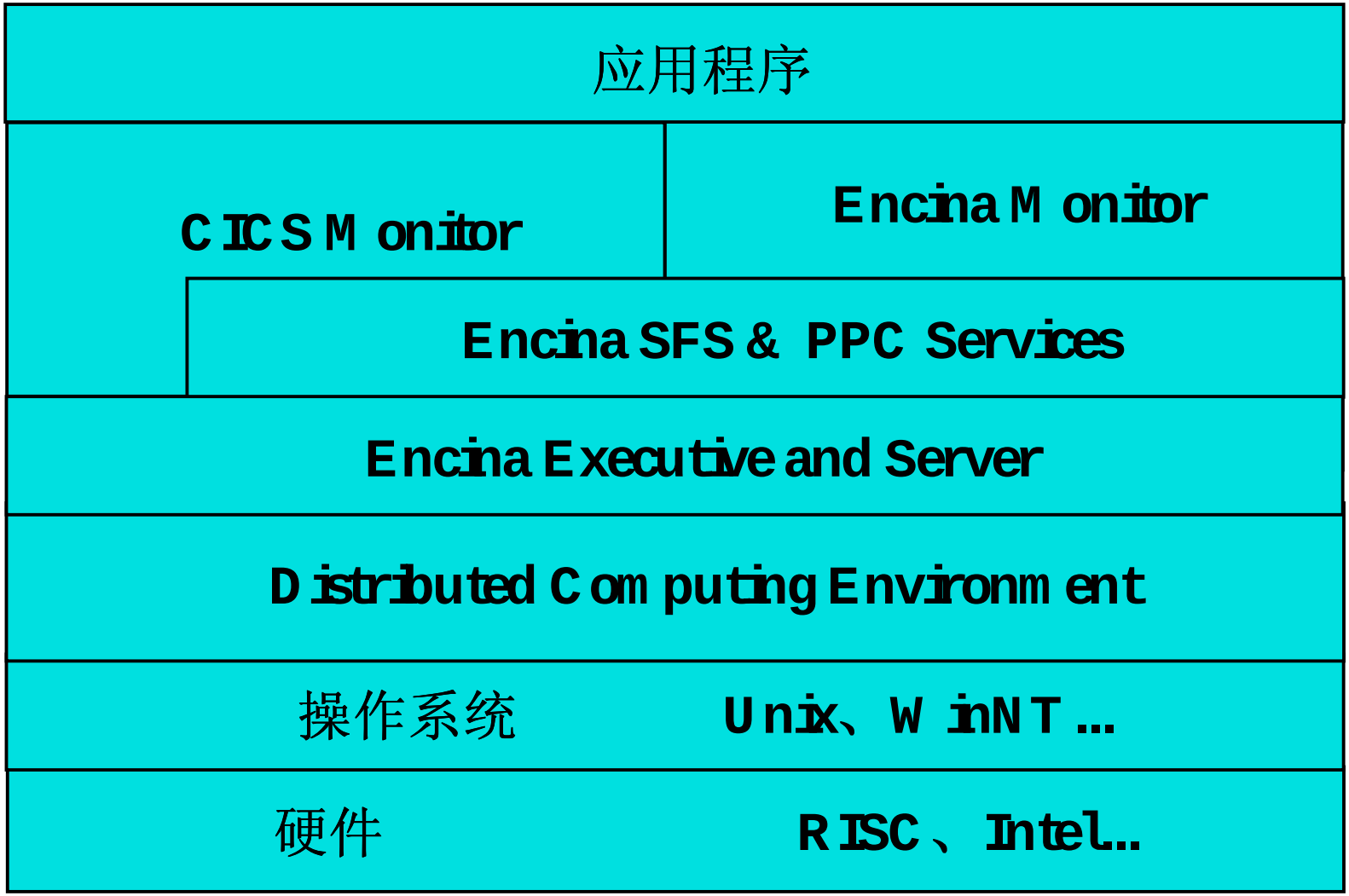
第二章

IBM TXSeries(CICS) Server结构

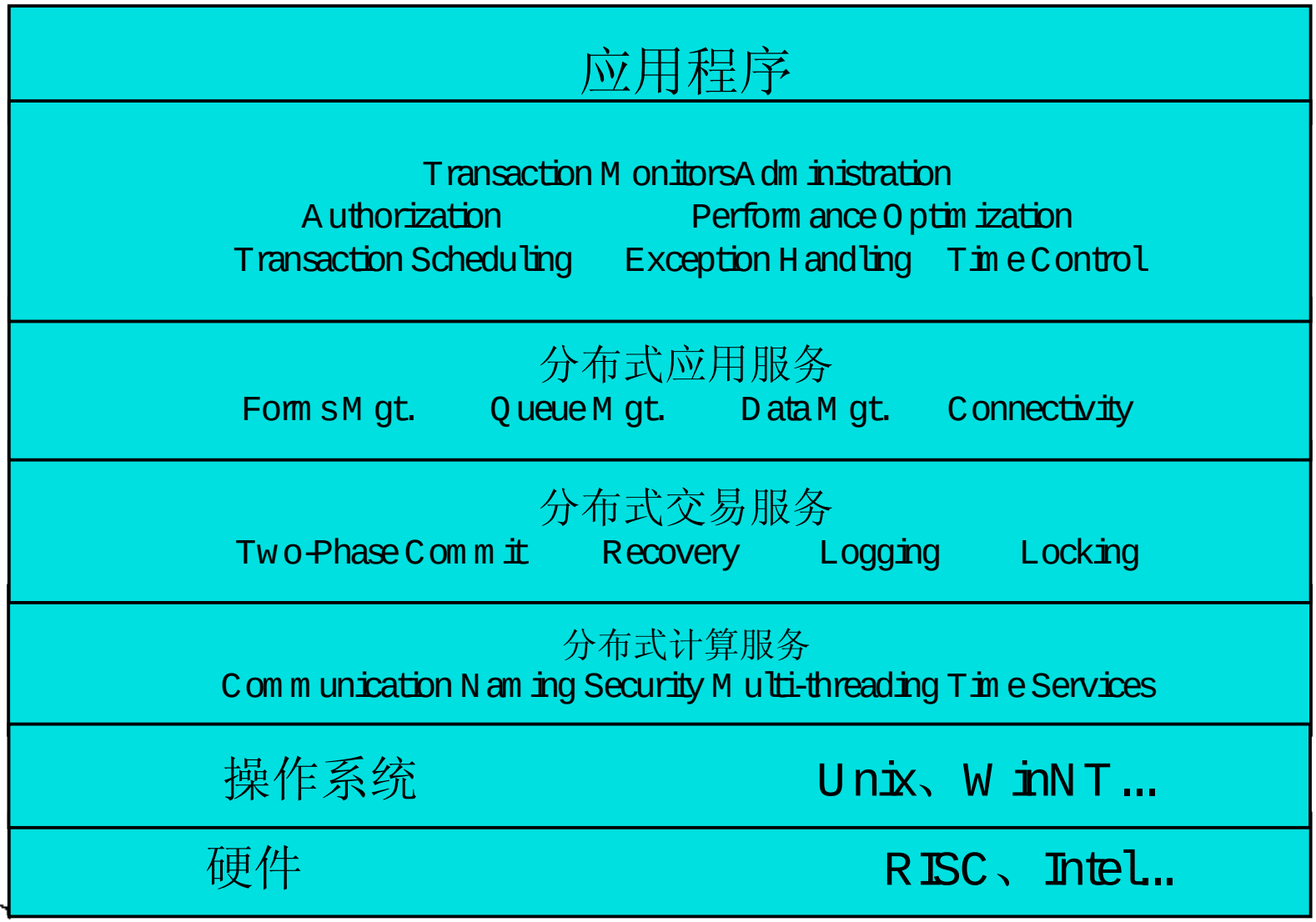
IBM TXSeries典型架构



TX Series Server层次型结构



TX Series Server层次结构



CICS

DCE 服务

Diskless Support Service	Distributed File Service
Time Service	Cell Directory Service
Security Service	Management
Remote Procedure Call	Threads

CICS Application Server

DCE 服务-RPC

- CICS DCE-RPC Only环 境
- DCE-RPC是 DCE CellDirectory Server、 DCE Security Server的 基 础

第三章

IBM TXSeries(CICS)
安装

TXSeries for AIX 硬件需求

	CICS Server	Encina Server	Encina Client
硬盘空间	350 MB	75 MB	48 MB
推荐硬盘交换区	160 MB	150 MB	64 MB
推荐内存	96 MB	64 MB	

注：CICS Server的硬盘空间包括了印刷资料
TXSeries for AIX运行在RS/6000上

CICS Application Server

TX Series for HP-UX硬件需求

	CICS Server	Encina Server	Encina Client
硬盘空间	350 MB	75 MB	48 MB
推荐硬盘交换区	160 MB	150 MB	64 MB
推荐内存	96 MB	64 MB	

注：CICS Server的硬盘空间包括了印刷资料

TX Series for HP-UX运行在HP 9000 Series 800上

CICS Application Server

TXSeries for Solaris硬件需求

	CICS Server	Encina Server	Encina Client
硬盘空间	350 MB	75 MB	48 MB
推荐硬盘交换区	160 MB	150 MB	64 MB
推荐内存	96 MB	64 MB	

注：CICS Server的硬盘空间包括了印刷资料
TXSeries for Solaris运行在Sun SPARC或UltraSPARC上

TXSeriesforWinNT硬件需求

	CICS Server	Encina Server	Encina Client
硬盘空间	350MB	75MB	48MB
推荐硬盘交换区	160MB	150MB	64MB
推荐内存	96MB	64MB	

注：CICS Server的硬盘空间包括了印刷资料
TXSeriesforWindowsNT运行在Intel486(或以上)或兼容(如Cyrix或AMD)上

CICS Application Server

TXSeries软件需求

Product	AIX	HP-UX	Solaris	WindowsNT
Operating System	4.2.1,4.3.1	10.20	2.5.1,2.6	4.0
DCE Base Service	2.1	1.5	1.1	IBM 2.0,Gradient2.1
DCE CellDirectory Service DCE Security Service	2.1	1.5	1.1	IBM 2.0,Gradient2.1
IBM Communication Server	4.2			5.0.1
SNAPlus2		B.10.20		
SunLink SNA/PTP Runtime			9.0	
MicrosoftSNA Server				3.0
IBM C++ forAIX	3.1.4.7			
IBM COBOL	1.1			
IBM PL/1	1.2			
Micro FocusCOBOL	4.1.10	4.1	4.1.6	4.0.26
HP COBOL Softbench		4.0		
HP C/C++ Softbench		5.0		
Sun VisualWorkshop C++			3.0.1	
Sun Workshop ProfessionalC			3.0	
Sun Workshop CompilerC/C++			4.2	
VisualAge forC++ forWindows				4.0
VisualAge COBOL				2.0
MicrosoftVisualC++ forWindowsNT				4.2b,5.0
DB2	2.1.2	2.1.2	2.1.2	2.1.2
Database Server	4.0			5.0
DB2 UniversalDatabase Enterprise Edition	5.0	5.0	5.0	5.0
DB2 UniversalDatabase Enterprise Extended Edition forAIX	5.0			
Informix	7.24	7.2	7.2	
Oracle	7.3.4	7.3.4	7.3.4	7.3.4
Oracle8	8.0.3	8.0.3	8.0.4	8.0.3
Sybase	11.1	11.1	11.1	
SQL Server				

CICS Application Server

TXSeries目录结构

- TX Series for AIX
  - /usr/lpp/cics
    - bin
    - include
    - lib
    - src
      - examples
      - samples
    - utils
    - msg
- TX Series for HP-UX和Sun Solaris
  - /opt/cics
    - bin
    - include
    - lib
    - src
      - examples
      - samples
    - utils
    - msg
- TX Series for Windows NT
  - driver:\opt\cics
    - bin
    - include
    - lib
    - src
      - examples
      - samples
    - utils
    - msg

CICS Application Server

TX Series for AIX 安装(一)

- 设置用户、组

➡ 组：smitty mkgroup

- Group NAME　　cics
- USER list　　root
- ADMINISTRATOR list　　root

- Group NAME　　cicsterm
- USER list　　root
- ADMINISTRATOR list　　root

➡ 用户：smitty mkuser

- User NAME　　cics
- Primary GROUP　　cics
- Group SET　　cicsterm

- User NAME　　SFS_SERV
- Primary GROUP　　cics
- Group SET　　cicsterm

TXSeries for AIX 安装(二)

- 创建jfs和lv

- jfs: smitty crjfs(选择'Add a Standard Journaled File System '和相应得卷组(vg))
  - SIZE of file system 40000(20 mega bytes)
  - MOUNT POINT /var/cics_servers
  - Mount AUTOMATICALLY at system restart? Yes
  - SIZE of file system 80000(40 mega bytes,根据实际交易量)
  - MOUNT POINT /var/cics_regions
  - Mount AUTOMATICALLY at system restart? Yes
  - "mount /var/cics_servers"
  - "mount /var/cics_regions"

- lv: smitty mklv(用F4键选择相应的卷组(vg))
  - Logical volume NAME sfs_SFS_SERV
  - Number of LOGICAL PARTITIONS 8(32 mega bytes)
  - Logical volume NAME log_SFS_SERV
  - Number of LOGICAL PARTITIONS 8(32 mega bytes)

CICS Application Server

TX Series for AIX 安装(三)

- "cd /dev"
- "chown SFS_SERV:cics *SFS*"　　(有四个设备会被授权SFS_SERV用户)

  - 设置环境变量

➡ 编辑"/etc/environment"

- PATH中加入"/usr/lpp/cics/bin:",同时检查数据库"bin"路径是否设置
- "CICSPATH=/usr/lpp/cics"
- "ENCINA_BINDING_FILE=/var/cics_servers/server_bindings
- "RPC_UNSUPPORTED_NETIFS"=en1|fddi0"
  - (把不在"server_bindings"文件中用到的网卡加入，可用"netstat -i")
- "CICSREGION=defaul_region_name"
- "CICS_SFS_SERVER=/.:/cics/sfs/$HOSTNAME"

➡ 编辑?"/etc/services"

- 加入"sfs_port　　8888/udp" (找一未被使用的口(port))

➡ 创建?"/var/cics_servers/server_bindings"

- "/.:/cics/sfs/$HOSTNAME　　ncadg_ip_udp:$host_ip[$sfs_port]
  - ($host_ip可省略)

TX Series for AIX 安装(四)

- 软件安装
- "smitty install"
- "Install and Update Software"
- "Install and Update from LATEST Available Software"
- 选择相应的设备
- SOFTWARE to install(用F4选择安装介质)
  - 用F7选择下面的软件
  - cics.base
  - cics.server
  - cics.client
  - cics.info
  - encina.server
  - encina.client
  - encina.PPC exec
  - encina.SFS
  - encina.info
- 安上面相同的方法安装补丁软件(PTF)

CICS Application Server

TX Series for HP-UX 安装

- 软件安装
- sam
  - swinstall
- 安上面相同的方法安装补丁软件(PTF)

CICS Application Server

TX Series for Solaris安装

- 软件安装

➨ 安上面相同的方法安装补丁软件(PTF)

CICS Application Server

TXSeries for WinNT安装

- 软件安装

- setup
- 安上面相同的方法安装补丁软件(PTF)

CICS Application Server

第四章

IBM TXSeries(CICS)
配置

CICS Region目录结构

- TXSeries for AIX, HP-UX和Solaris
  - /var/cics_regions/<region name>
    - bin
    - data
    - database
    - dumps
      - dir1
    - log
    - maps
- TXSeries for Windows NT
  - <dirver>\var\cics_regions\<region name>
    - bin
    - data
    - database
    - dumps
      - dir1
    - log
    - maps

CICS Application Server

TXSeries for AIX 配置(一)

- "logout" 并以 root用户 "login"
- "cicsdefaultservers"
- "cicssetupclients -m -v"
- 配置DCE
  - "mkdce -o local -n $HOSTNAME rpc" (建立一个DCE RPC-only)
    - 注意: DCE用135口(port),所以当发现135被其它应用(如AIX
      - C Network License Servide)占用时，必需停止它。
      - "stopsrc -s netlsd"
      - "stopsrc -s glbd"
      - "stopsrc -s llbd"

- 生成 SFS 文件系统
  - "smitty cics","Manage Filesystem","Manage Encina SFS Servers",
  - "Define Encina SFS Servers","Create"
    - Model SFS Server Identifier ""
    - SFS Server Identifier "/.:/cics/sfs/$HOSTNAME"
    - Are you using DCE servers "NO"
    - Name Service for advertising server "NONE"

CICS Application Server

TXSeries for AIX 配置(二)

- cold start
  - 生成 SFS 可执行冷启动，以后就用 auto start
  - "cicssfscold /.:/cics/sfs/$HOSTNAME"
  - 可查看"/var/cics_servers/SSD/cics/sfs/$HOSTNAME/msg"启动情况。
  - 如在定义 SFS 时，系统报告该 SFS 已存在时，并用"smitty cics"无法
  - 删除时可用："cicssrcdestroy -s cicssfs.SFS_SERV"
  - 生成CICS REGION
- "smitty cics" "Manage CICS Regions" "Create (Import) a CICS Region"
  - Name of Region to be created "CICSRG1"
  - Force use or no-use of DCE servers? "do not use DCE servers"
  - 如在定义 REGION 时，系统报告该 REGION 已存在时，
  - 并用"smitty cics"无法删除时可用："cicssrcdestroy -r cics.CICSRG1"
- 配置CICS资源到SFS
  - "cicssfsconf -R wc CICSRG1 DefaultFileServer=/.:/cics/sfs/$HOSTNAME"

TXSeries for HP-UX配置

- 生成DCE RPC-only环境
- cicscp -v create dce -R
- 创建结构化文件系统SFS Server
- cicscp -v create sfs_server /.:/cics/sfs/<hostname>
- 启动结构化文件系统SFS Server
- cicscp -v start sfs_server /.:/cics/sfs/<hostname> StartType=cold(默认为auto)
- 创建CICS Region(业务应用系统)
- cicscp -v create region <region name>
- 启动CICS Region(业务应用系统)
- cicscp -v start region <region name> StartType=cold(默认为auto)

TX Series for Solaris配置

- cicscp -v create dce -R
- cicscp -v start region CICSTEST

CICS Application Server

TXSeries for WinNT配置

- cicscp -v create dce -R
- cicscp -v create region CICSTEST

CICS Application Server

TXSeries与DB2XA配置

- 配置2 Phase XA与db2数据库的连接

➡ 生成连接程序(Switch Load File)

- "cd /usr/lpp/db2_02_01/lib"
- "ar -vx libdb2.a"
- "mv shr.o db2.o"
- "cd /usr/lpp/cics/src/examples/xa/"
- 修改db2xa.mk文件中相应的DB2环境变量
- "make -fdb2xa.mk"生成db2xa
  - "mv db2xa /var/cics_regions/$CICSREGION/bin/"

➡ 配置XA

- "smitty cics" "Manage CICS Regions" "Define CICS Resources"
- "XA Configure" "New"
- Identifier: "sample"
- Switch Load File Path Name "db2xa"
  - Resource Manager Initialization String: "dbname,user,password"

➡ 配置环境变量使得root和cics用户可以存取DB2

- "vi /etc/profile",加入"./home/db2/sqllib/db2profile"
- "vi /var/cics_regions/$CICSREGION/environment"加入 "DB2INSTANCE=db2"

CICS Application Server

TXSeries与Informix XA配置

- 配置2 Phase XA与informix数据库的连接

➡ 生成连接程序(Switch Load File)

- "cd /usr/lpp/cics/src/examples/xa/"
- 修改informix7xa.mk文件中相应的 Informix 环境变量
- libcicsshr.o
- "make -f informix7xa.mk" 生成 informxa
  - "mv informxa /var/cics_regions/$CICSREGION/bin/"

➡ 配置 XA

- "smitty cics" "Manage CICS Regions" "Define CICS Resources"
- "XA Configure" "New"
- Identifier: "sample"
- Switch Load File Path Name "informxa"
  - Resource Manager Initialization String: "dbname"

➡ 配置环境变量使得root和cics用户可以存取Informix

- "vi /etc/profile", 加入""
- "vi /var/cics_regions/$CICSREGION/environment" 加入
- "INFORMIXSERSVER=online" "INFORMIXDIR=/home/informix"

➡ 注意：数据库要设成 unbufferd

- "ontape -U database"

CICS Application Server

TXSeries与SybaseXA配置

- 配置2PhaseXA与Sybase数据库的连接
- "isql-Usa-Ppwd","grantallonspt_committabtoprobe","go"
- 生成连接程序(SwitchLoadFile)
  - "cd/usr/lpp/cics/src/examples/xa/"
  - 修改sybasexa.mk文件中相应的Sybase环境变量
  - "makefsybasexa.mk"生成sybasexa
    - "mvsybasexa/var/cics_regions/$CICSREGION/bin/"
- 配置XA
  - "smittycics""ManageCICSRegions""DefineCICSResources"
  - "XAConfigure""New"
  - Identifier: "sample"
  - SwitchLoadFilePathName "sybasexa"
  - ResourceManagerInitializationString: "-Nconn_1-Uuser-Ppw-L/tmp/sybasexa.log"
- 配置环境变量使得root和cics用户可以存取Sybase
  - "cd$SYBASE/scripts""vixa_load"".Xa_load"
  - "vi/home/sybase/xa_config"加入
  - "[xa]
  - lrm=conn_1
  - server=SYBASE"

TXSeries与SybaseXA配置

- 配置1PhaseXA与Sybase数据库的连接
  - 生成连接程序(SwitchLoadFile)
    - "cd syb1pc"
    - "cpre -V CS_VERSION_100 sybase1pc.cpre"
    - "vi sybase1pc.c"加入
      - 参照packetsize，加入网络包配置
    - "make" 生成 sybase1pc
  - 配置XA
    - "smitty cics" "Manage CICS Regions" "Define CICS Resources" "XA Configure" "New"
      - Identifier: "samplexa"
      - Switch Load File Path Name "sybase1pc"
      - Resource Manager Initialization String: "SYBASE userid password"
  - 修改环境变量
    - "vi /var/cics_regions/$CICSREGION/environment"加入
      - "DSQUERY=SYBASE
      - SYBASE=/home/sybase"
  - Sybase解库:
    - "su - sybase;cd lib;ar -x libcomn_dce.so.a;ar -x libcs_r.so.a;
    - ar -x libct_r.so.a;ar -x libintl_r.so.a;ar -x libtcl_dce.so.a"

TX Series Listener配置

- 配置TX Series Server Listener

➡ "smitty cics" "Manage CICS Regions" "Define CICS Resources" "Listeners" "Add New"

- Listener Identifier: "TCPIPL1"
- TCP adapter address "194.2.201.254"
- TCP service name "tcpipl1"

➡ "vi /etc/services",

- 加入"tcpipl1 9999/tcp"

CICS Application Server

第五章

IBM TXSeries(CICS)
操作

TXSeries操作

- 启动TXSeries Server for AIX

"cicscp -v start dce　　/*启动DCE*/
cicssfs $CICS_SFS_SERVER　　/*启动SFS*/
cicscp -v start region $CICSREGION StartType=cold"　　/*启动TXSeries Region*/

- 停止TXSeries Server for aix

"cicscp -v stop region $CICSREGION　　/*停止TXSeries Region*/
cicscp -v stop sfs_server $CICS_SFS_SERVER　　/*停止SFS*/
cicscp -v stop dce"　　/*停止DCE*/

- 查看TXSeries Server for AIX状态

"cicstail -r $CICSREGION"

CICS Region的进程组成

CICS Region 系统

cicsas | cicsas | cicsas | cicsas

Application Server Processes

所有交易都是在CICS的应用服务器(cicsas)中运行

CICS Region控制进程

cicsam	Application Server Manager
cicsrl	RPC Listener
cicsip	Common Client Listener
cicssl	LocalSNA Listener
cicsrm	Recovery Manager
cicsrs	Recovery Server
cicsld	Log Daemon
cicsic	IntervalControl Manager

第六章

IBM TXSeries(CICS)
管理

Region资源管理

- Communication Definition(CD)
- Journal Definition(JD)
- Monitor Definition(MD)
- File Definition(FD)
- Program Definition(PD)
- Transaction Definition(TD)
- Temporary Storage Definition(TSD)
- Transient Data Queue Definition(TDD)
- XA Definition(XAD)
- User Definition(UD)

Region资源数据库

- 运行数据库(Runtime Database)
- 永久数据库(Permenanant Database)
- RSLKey
- TSLKey

通讯定义(CD)

- 功能:
- 参数:

用户定义(UD)

- 功能:
- 参数:

日志定义(JD)

- 功能:
- 参数:

监控定义(MD)

- 功能:
- 参数:

文件定义(FD)

- 功能:
- 参数:

程序定义(PD)

- 功能:
- 参数:

交易定义(TD)

- 功能:
- 参数:

零时存储队列定义(TSD)

- 功能:
- 参数:

零时队列定义(TDD)

- 功能:
- 参数:

数据库接口定义(XAD)

- 功能:
- 参数:

第七章

IBM CICS Common Client
安装、配置与操作

TXSeriesClients硬件需求

平台	硬盘	内存
Windows	3.5MB	400KB
Windows95	3.5MB	400KB
WindowsNT	3.5MB	500KB
OS/2	3.5MB	500KB
DOS	1MB	200KB
Macintosh	1MB	500KB
AIX	24MB	7MB
Solaris	24MB	7MB
HP-UX	24MB	7MB
InternetGateway	4.5MB	700KB

CICS Application Server

CICS Client for SCO 安装

- 以root用户登入安装
- pkgrm SCOCC (删除旧版本)
- mv pkg.Z /usr/spool/pkg.Z
- uncompress /usr/spool/pkg.Z
- pkgadd
- 修改 semaphores
- "/etc/conf/bin/idtune -f SEMMNS 400 -max 500"
- "/etc/conf/bin/idtune MSGSSZ 32"
- 修改 Semaphores 和 MessageQueuesue
- "scoadmin","HardwareKernelManager" "Tune Parameters" "Semaphores"
  - SEMMAP 400 SEMMNI 400
  - SEMMNU 100 XSEMMAX 100
- "scoadmin","HardwareKernelManager" "Tune Parameters" "Message Queues"
  - MSGMAP 512 MSGMAX 32767
  - MSGMNB 65532 MSGSEG 4096
- "scoadmin","HardwareKernelManager" "Relink Kernel"
- "reboot"

CICS Application Server

CICS Client for SCO 配置

- "cd /opt/K/SCO/cics/bin"
- "vi CICSCLI.INI" 加入以下内容

"Server = CICSRG1 (建议Server名与REGION名相同)
Description = TCP/IP Server
Protocol = TCPIP
NetName = 194.2.201.254
Port = 1435"

CICS Application Server

CICS Client for SCO 操作

- 启动CICS for SCO的一个Server

"cicscli /S=CICSRG1"

- (注意：请在ksh中启动CICS Client)

或编辑/etc/rc.d/0/sysinit文件加入

- ksh -c "cicscli /S=CICSRG1"

- 停止CICS for SCO的一个Server

"cicscli /X=CICSRG1"

- 查看 CICS for SCO 状态

"cicscli /L"

- 停止CICS for SCO Client

"cicscli /i"

- (注意：在改变过CICSCLI.INI后必须用此命令停止CICS Client,
- 然后再启动CICS Client

CICS Application Server

CICS Client for AIX 安装

- 以root用户登入安装

- uncompress /tmp/cics-302.tar
- tar xvf /tmp/cics-302.tar
- ksh -m kcicscli
- ksh -m kclmsgsus

CICS Client for AIX 配置

- "cd /usr/lpp/cicscli/bin"
- "vi CICSCLI.INI" 加入以下内容

"Server = CICSRG1 (建议Server名与REGION名相同)
Description = TCP/IP Server
Protocol = TCPIP
NetName = 194.2.201.254
Port = 1435"

CICS Client for AIX 操作

- 启动CICS for AIX的一个Server

"cicscli /S=CICSRG1"

- (注意：请在ksh中启动CICS Client)

或编辑/etc/rc.d/0/sysinit文件加入

- ksh -c "cicscli /S=CICSRG1"

- 停止CICS for AIX的一个Server

"cicscli /X=CICSRG1"

- 查看 CICS for AIX 状态

"cicscli /L"

- 停止CICS for AIX Client

"cicscli /t"

- (注意：在改变过CICSCLI.INI后必须用此命令停止CICS Client,
- 然后再启动CICS Client

CICS Application Server

cicsterm 操作

- cicsterm (选择CICS Region)
- CESN (登录)
- CESF (签退)
- CEMT (CICS Region管理)
- CECI (联机CICS API)
- CSTD (联机统计)

第八章

IBM TXSeries(CICS)
问题诊断与系统恢复

系统恢复

- Crash恢复

➡不可预知的软件或硬件故障
➡异常SHUTDOWN

- 介质恢复

➡磁盘故障

- 灾难恢复

➡核战争
➡地震
➡火灾
➡.....

恢复数据类型

- CICS

- 交易状态(committed, aborted, prepared 等等)
- 永久配置数据库
- 运行配置数据库

- SFS

- 交易状态(SFS可以是协作者)
- 应用数据(CICS文件)
- 应用主数据(文件和索引的定义)

CICS恢复

- 文件和队列
- 热启动与冷启动
- 恢复策略
- 灾难恢复

CICS文件

- Region重起
- dump和statsfile信息
- /var/cics_regions/<cics region>/region_restart
- 交易日志
- 存储交易状态信息
- 一个应用服务器(cicsas)，一个日志
- /var/cics_regions/<cics region>/log/*
- CICS运行数据库
- 存储CICS配置信息(热启动)
- /var/cics_regions/<region>/database/XX/XX.<region>/XX.auto
- XX=CD FD GD JD LD MD PD RD TD TDD TSD UD WD XAD
- /var/cics_regions/<region>/database/XAD/XAD.<region>/XAD.emer
- CICS永久数据库
- 存储CICS配置信息(冷启动)
- /var/cics_regions/<region>/database/XX/XX.stanza

热启动与冷启动

- 热启动
  - 配置改变不会丢失
  - 在SFS、Mainframe或Unix数据库中，处于*prepared*状态的交易会被合理解决
  - 可恢复文件与队列，会被恢复
  - 不可恢复的存储数据，会被恢复
    - 除非crash后的紧急重起(emergency restart)
  - 受保护的STARTS/AIDS，会被恢复
  - CICS内存存储会丢失(包括MAIN TSQs)
- 冷启动
  - 运行数据库的修改会丢失(被永久数据库替换)
  - 在SFS、Mainframe或Unix数据库中，处于*prepared*状态的交易不会被自动解决，需管理员来处理

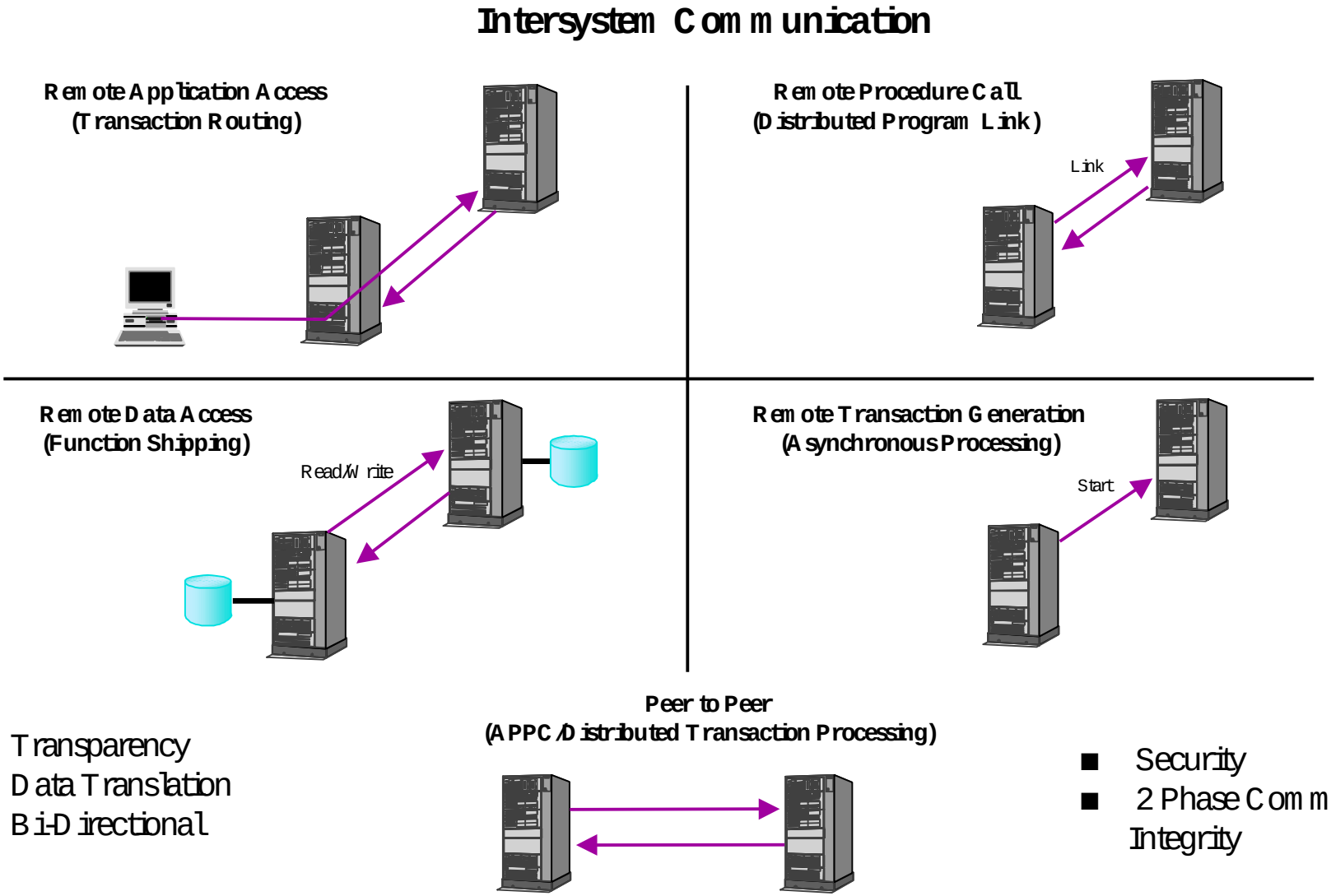
CICS恢复策略

- CICS日志

第九章

IBM TXSeries(CICS)
系统间通讯

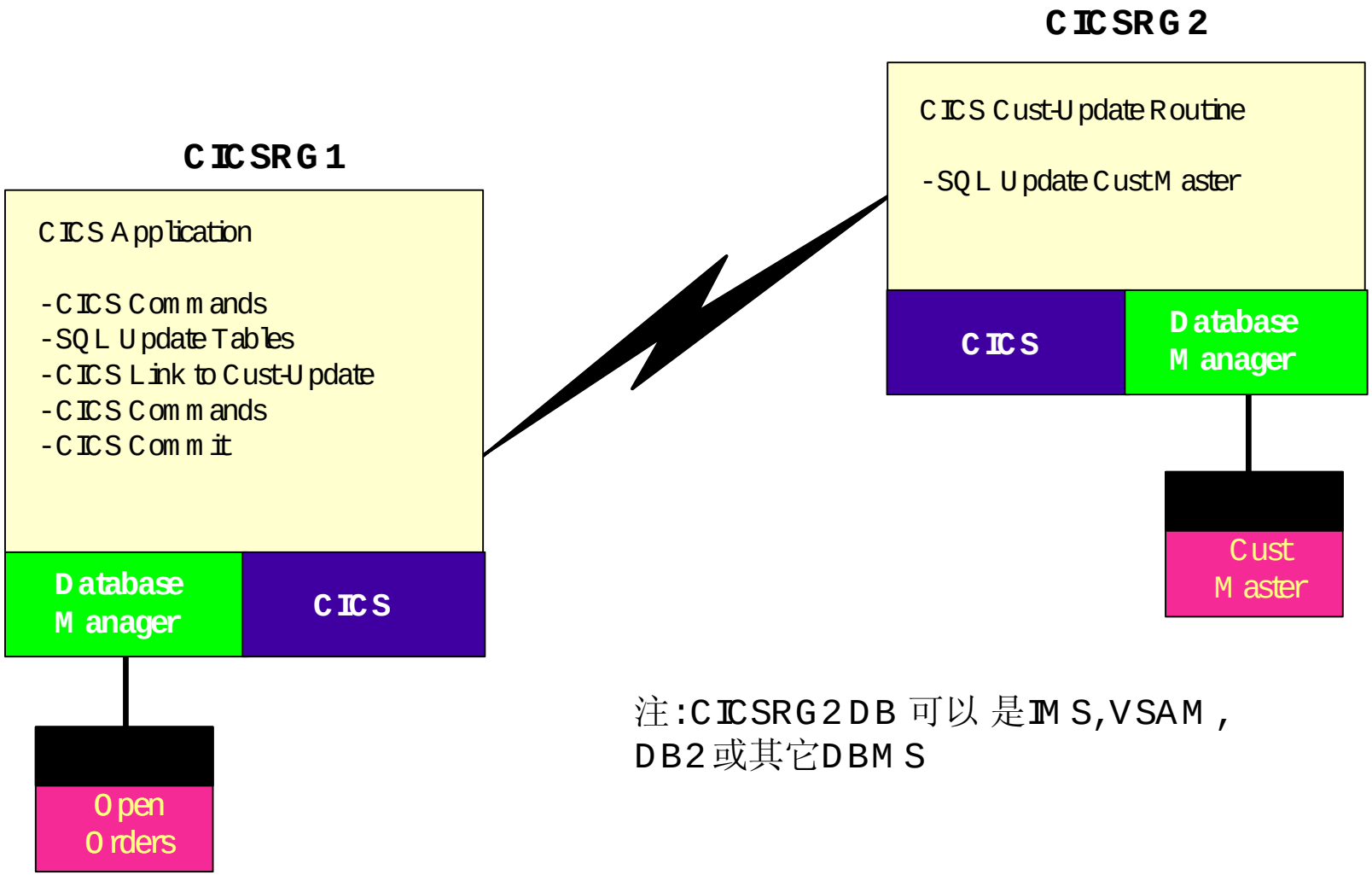
CICS系统间通讯(ISC)



- Transparency
- Data Translation
- Bi-Directional

- Security
- 2 Phase Commit Integrity

分布式程序连接(DPL)

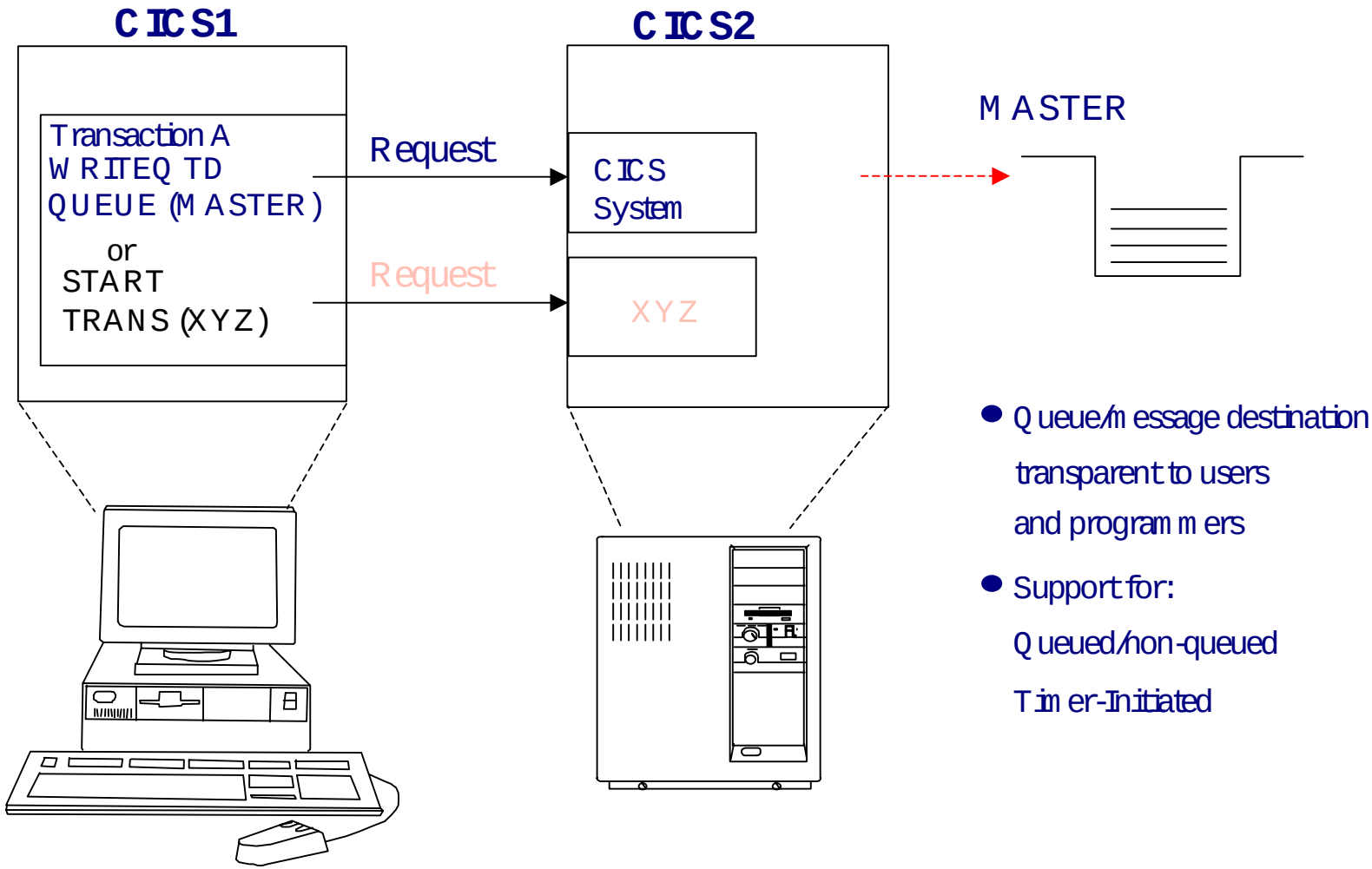


注:CICSRG2DB 可以 是IMS,VSAM,
DB2或其它DBMS

分布式程序连接(DPL)

- EXEC CICS LINK ...
- 方式:
  - 在PD定义中设定:
    - New Program Identifier [TESTPROG]
    - Remote system on which to run program [ISC1]
  - EXEC CICS LINK PROGRAM ('TESTPROG') SYSID(ISC1)

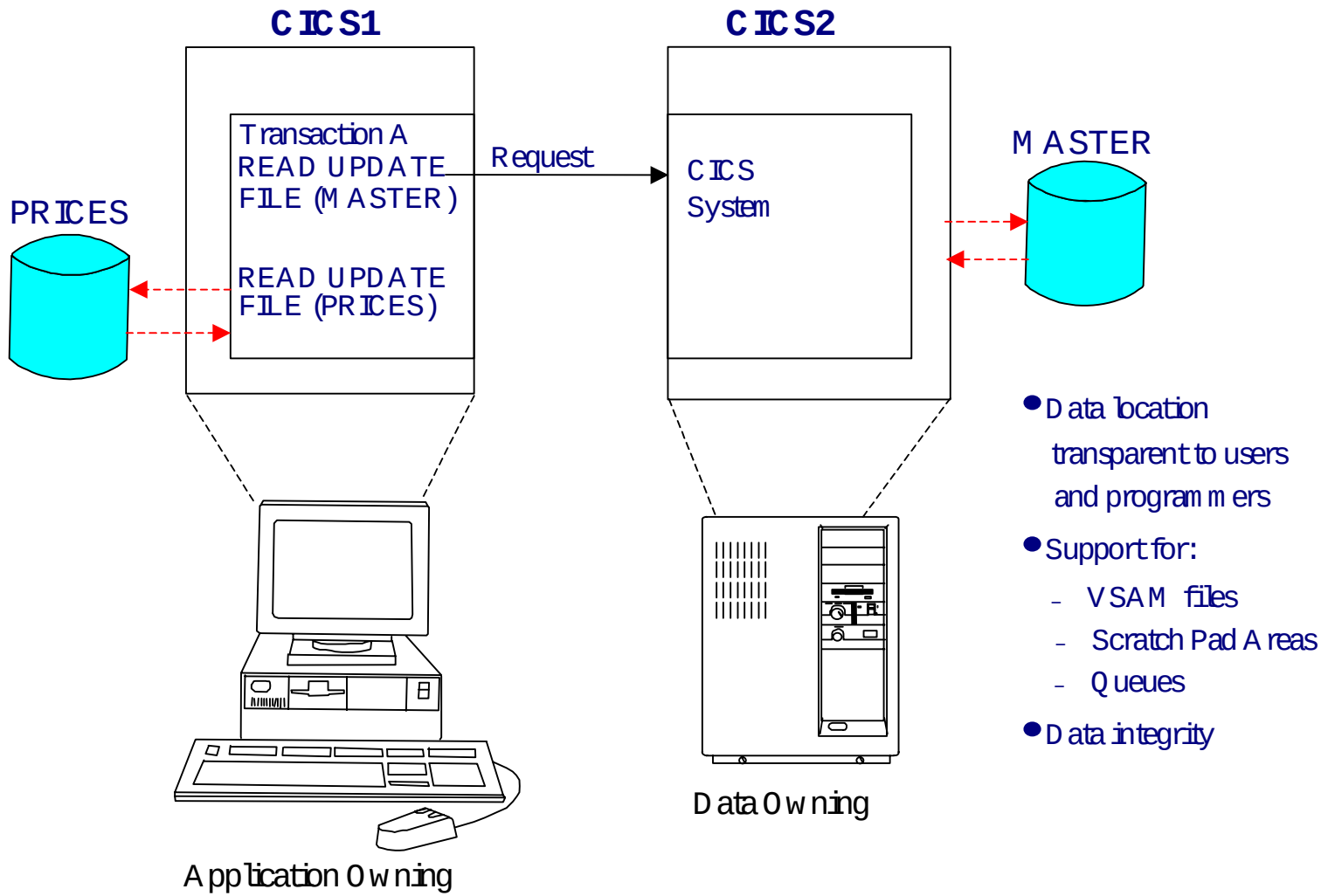
交易异步调用(ATI)



- Queue/message destination transparent to users and programmers
- Support for:
  - Queued/non-queued
  - Timer-Initiated

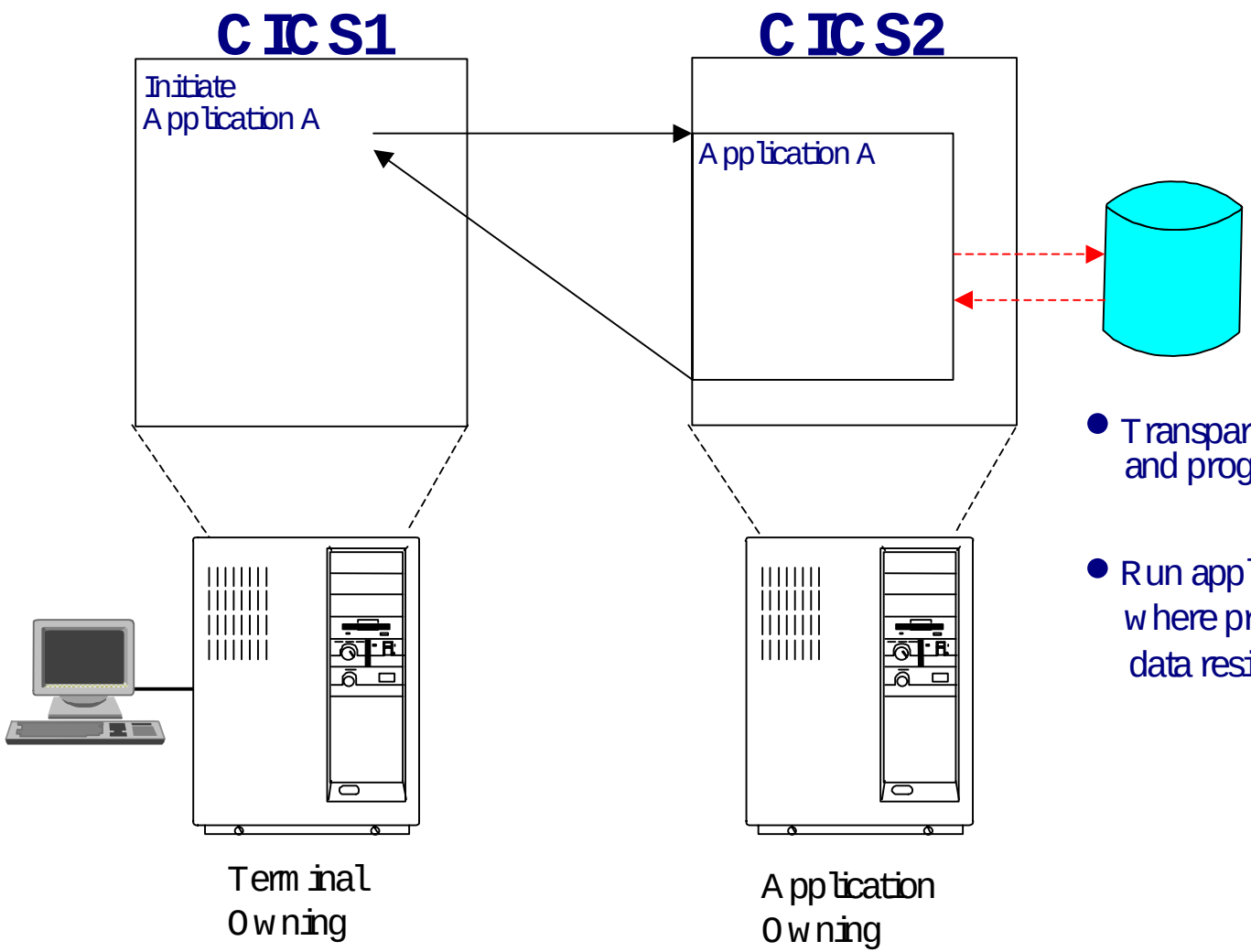
CICS Application Server

功能转移(FS)



- Data location transparent to users and programmers
- Support for:
  - VSAM files
  - Scratch Pad Areas
  - Queues
- Data integrity

交易路游(TR)



- Transparent to users and programmer
- Run application where programs and data reside

CICS Application Server

分布式交易处理(DTP)

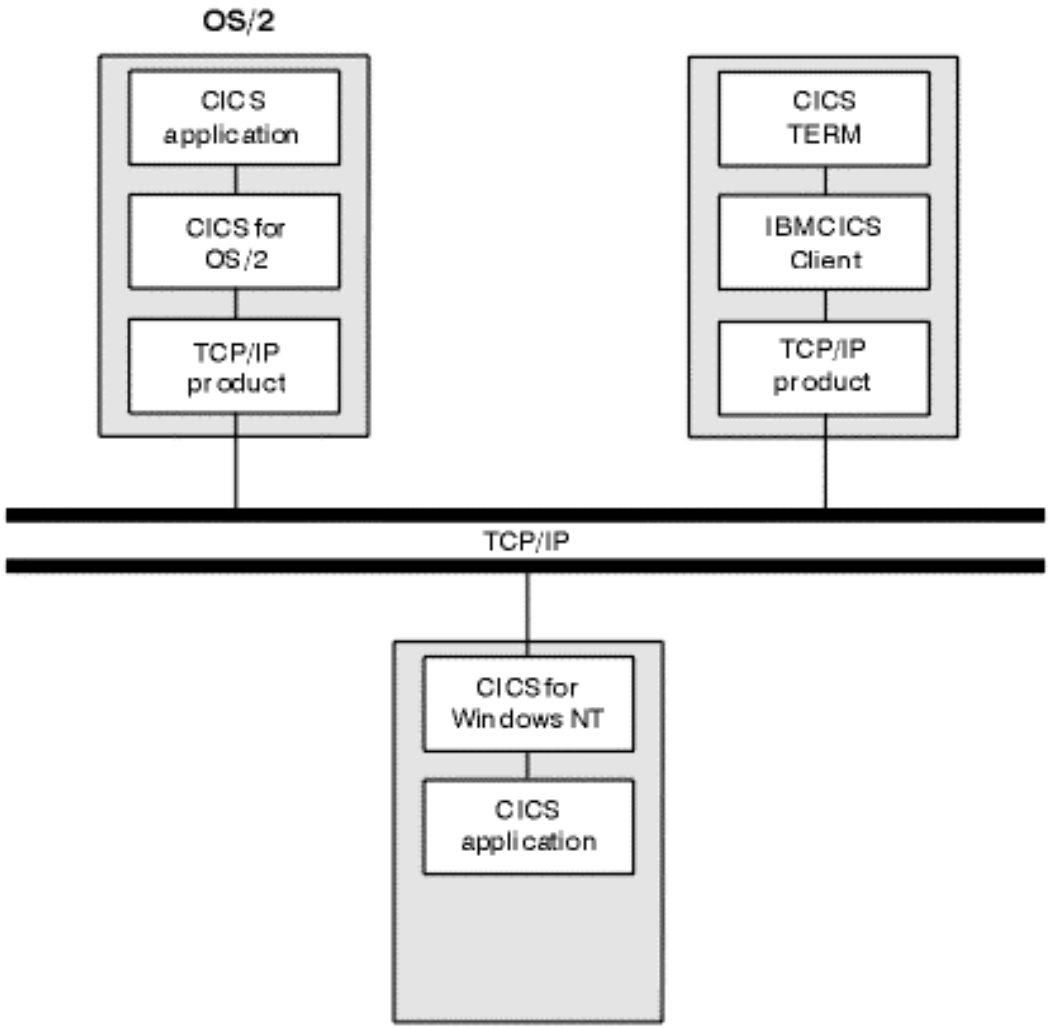
同步级别(SyncLevel)

- SyncLevel0
  - 提交(Commit)

- SyncLevel1
  - 提交(Commit)
  - 提交返回(CommitReturn)

- SyncLevel2
  - 准备(Prepare)
  - 准备返回(PrepareReturn)
  - 提交(Commit)
  - 提交返回(CommitReturn)

CICS family TCP/IP



SyncLevel:

0

1

支持：

功能转移

交易路游

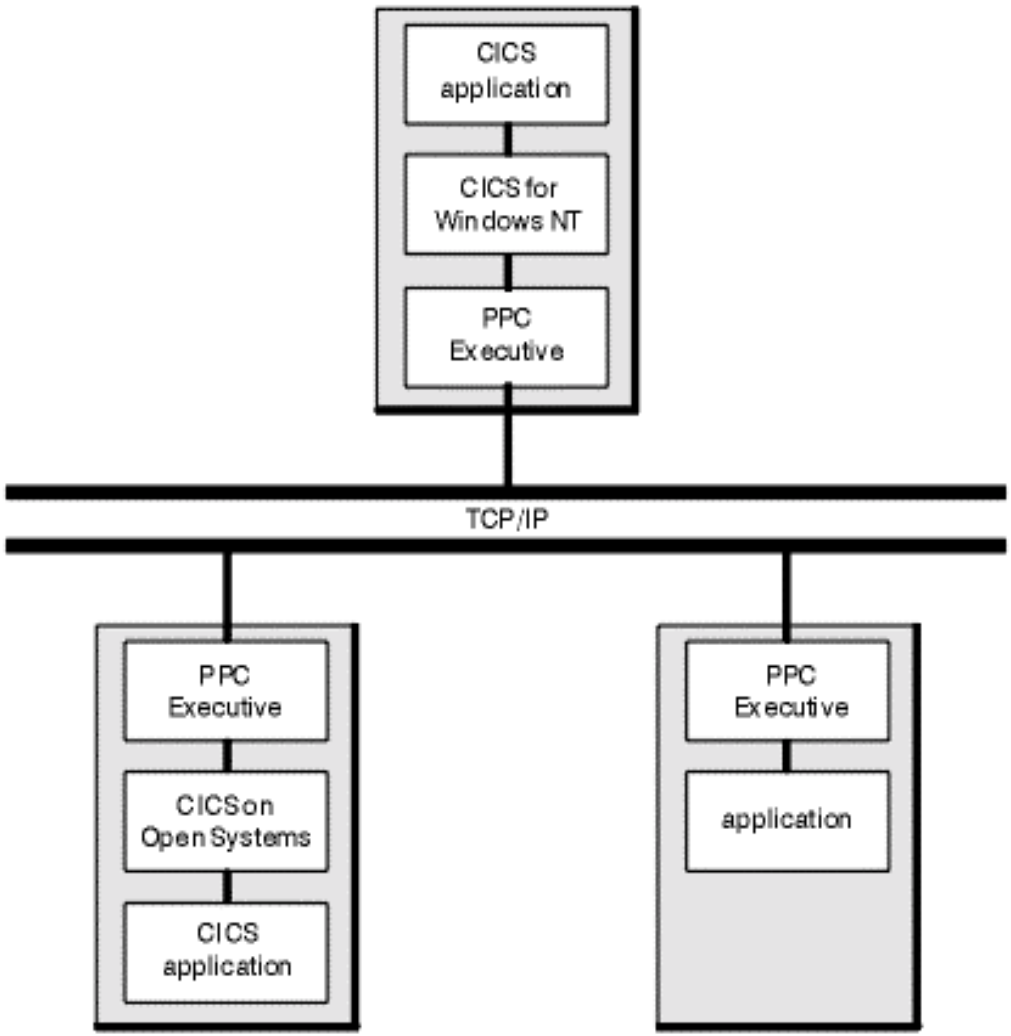
分布式程序连接 (DPL)

交易异步调用

不支持：

分布式交易处理(DTP)

Encina PPC TCP/IP



SyncLevel:

0

1

2

支持：

功能转移

交易路游

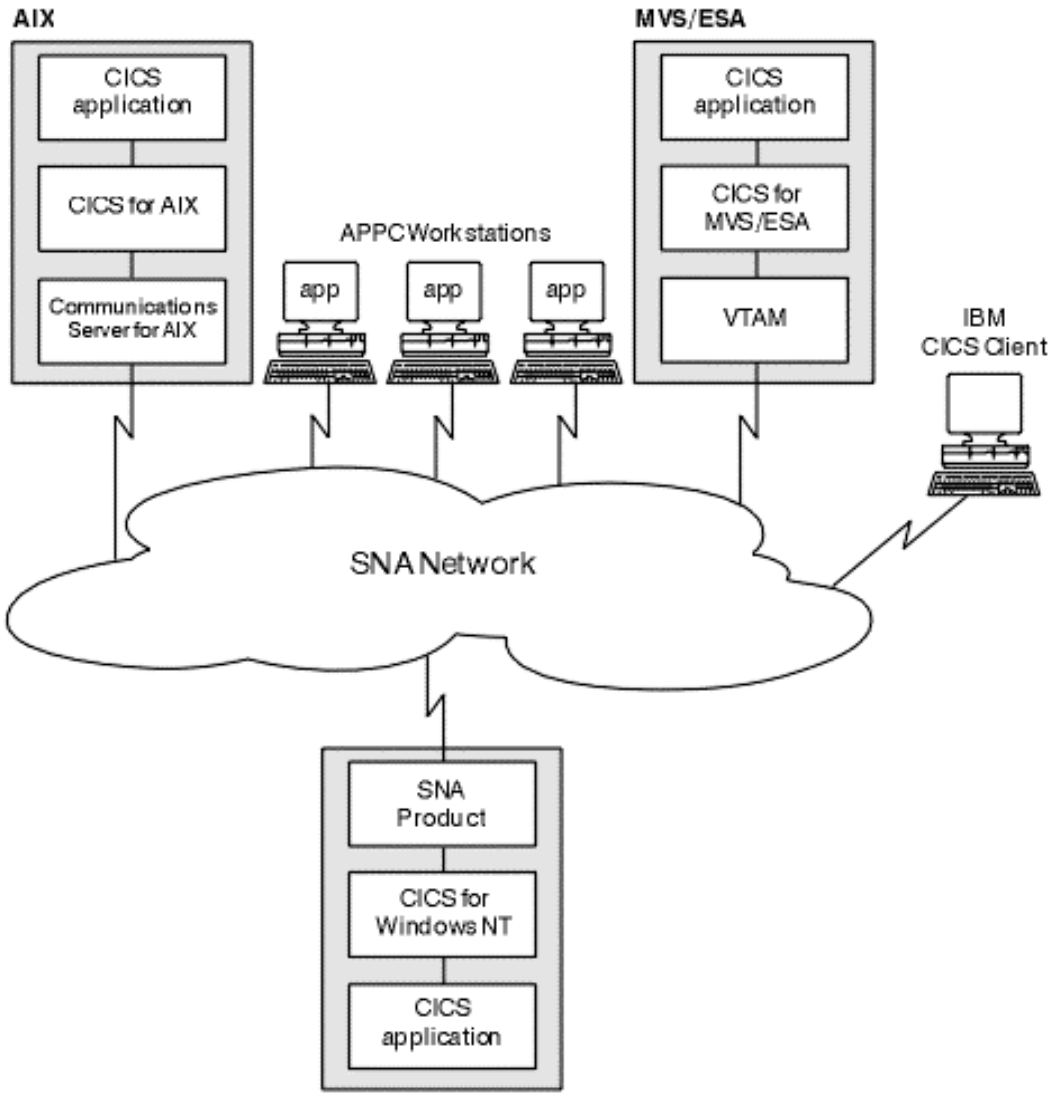
分布式程序连接 (DPL)

交易异步调用

分布式交易处理(DTP)

CICS Application Server

LocalSNA



SyncLevel:

0

1

支持：

功能转移

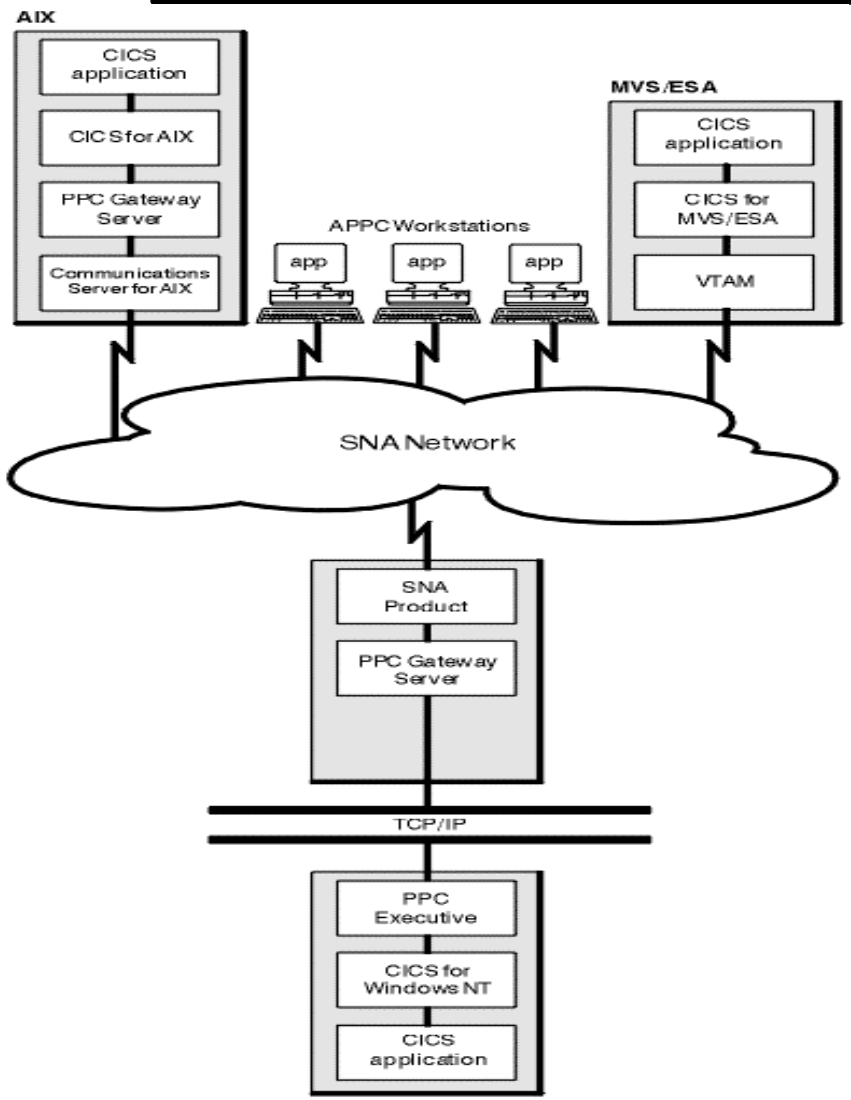
交易路游

分布式程序连接 (DPL)

交易异步调用

分布式交易处理(DTP)

PPC Gateway server



SyncLevel:

0

1

2

支持：

功能转移

交易路游

分布式程序连接 (DPL)

交易异步调用

分布式交易处理(DTP)

CICS Application Server

几种CICS系统间通讯方式的比较

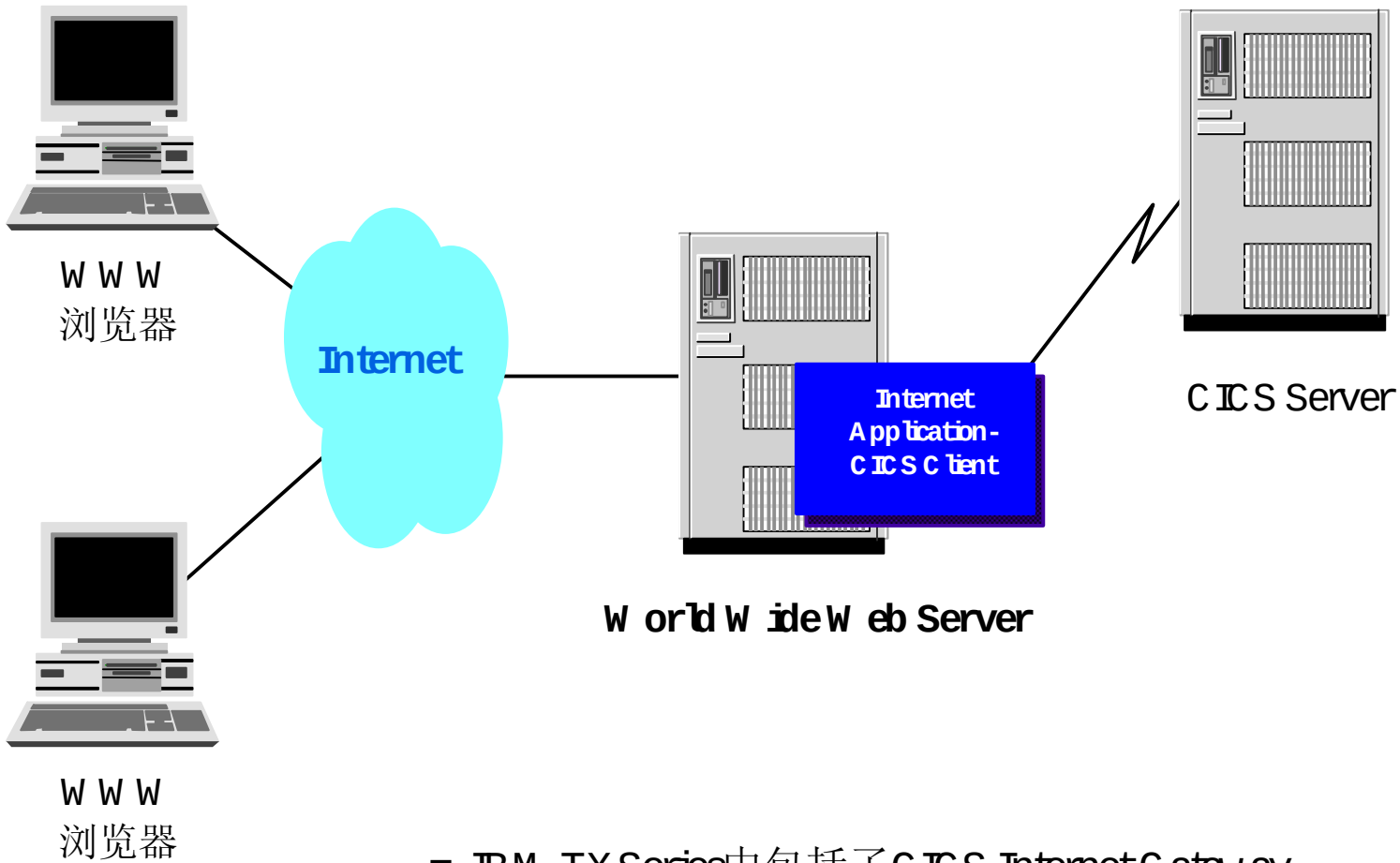
通讯方式	优点	限制
CICS family TCP/IP	Communicating at synchronization level 0 or 1 with CICS on Open Systems, CICS for Windows NT, CICS for OS/2, IBM CICS Clients, and RPC-only regions across TCP/IP	Distributed Transaction Processing (DTP) is not supported. CICS user security must be configured with care because it is not possible to reliably authenticate (identify) the remote system
Encina PPC TCP/IP	Communicating at synchronization level 0, 1 or 2 with other CICS on Open Systems, CICS for Windows NT regions or Encina PPC applications	RPC-only is not supported. Must be in the same DCE cell
Local SNA	Fast synchronization level 0 or 1 communication with remote LU 6.2 (APPC) systems. These connections can be used to connect to any CICS product	A supported SNA product must be installed on the same machine as the CICS region
PPC Gateway/SNA	Synchronization level 0, 1 and 2 communication with remote LU 6.2 (APPC) systems. These connections can be used to connect to any CICS product	When using Communications Server for AIX, HP-UX SNAplus2 and Transit(UNIX), the PPC gateway product must be installed on a machine along with a supported SNA product. Solaris and Windows NT do not support PPC Gateway server

CICS Application Server

第十章

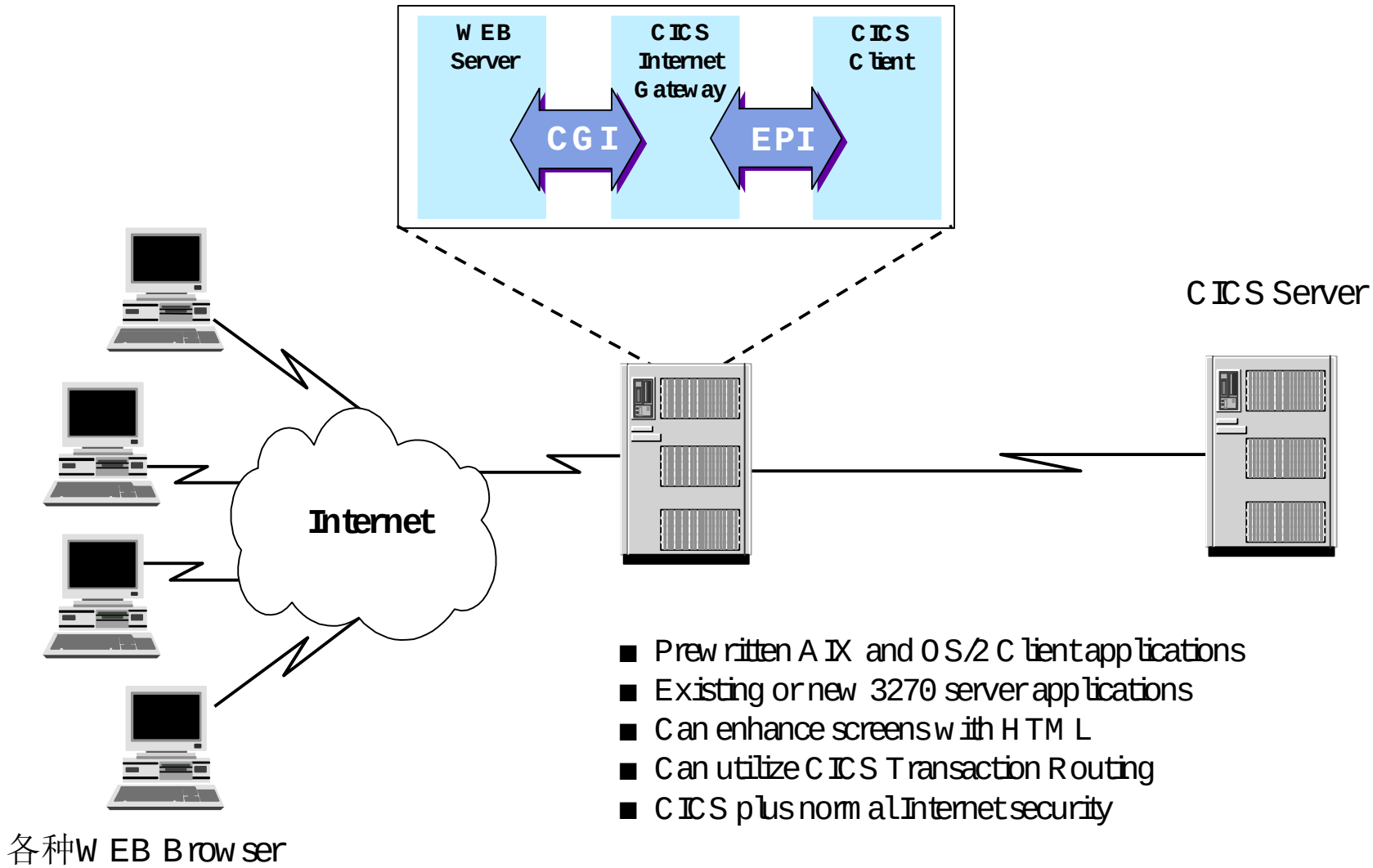
IBM TXSeries(CICS)
CICS Transaction Gateway

Internet存取CICS



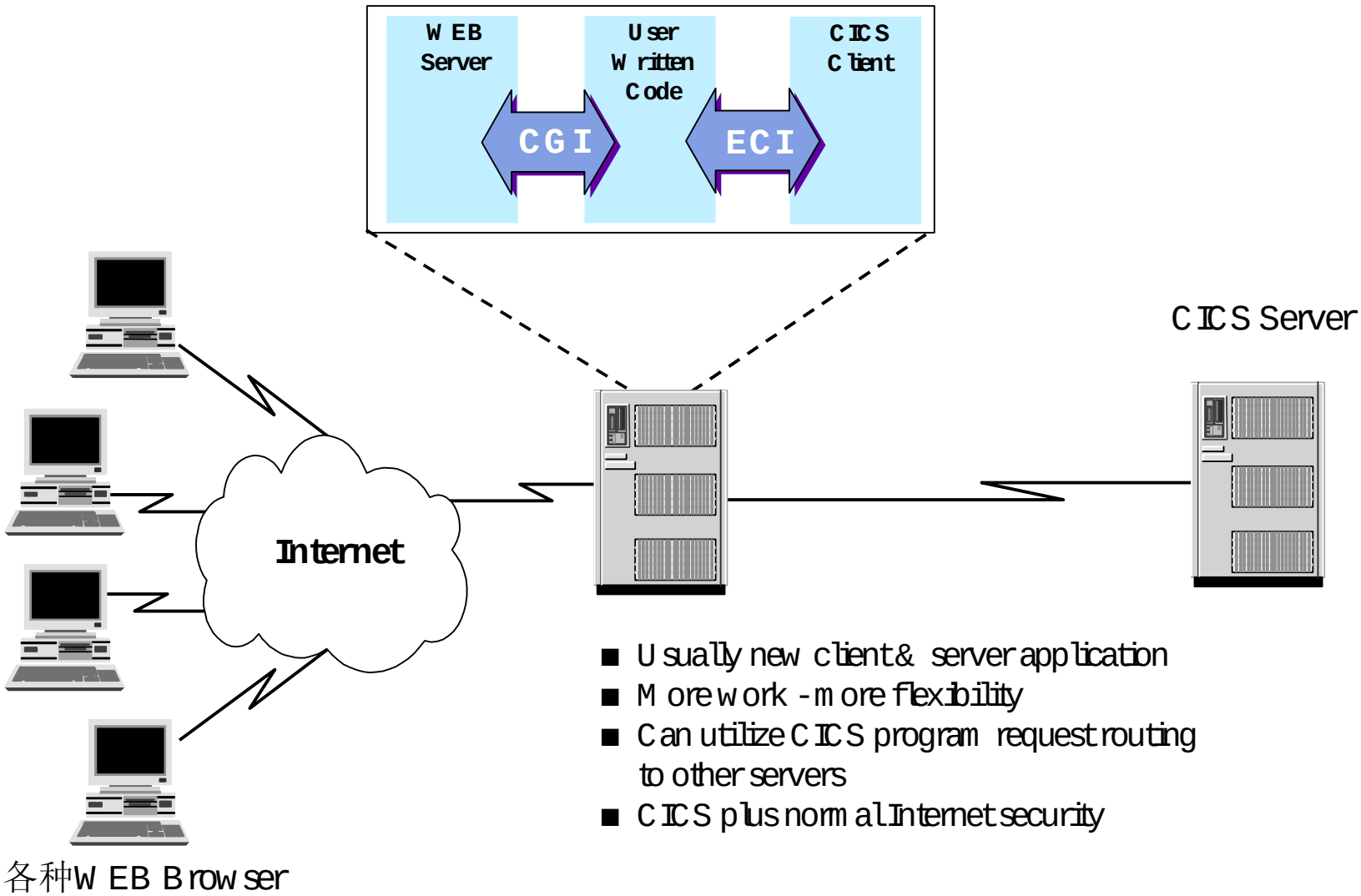
- IBM TXSeries中包括了CICS InternetGateway

CICS Internet Gateway



- Prewritten AIX and OS/2 Client applications
- Existing or new 3270 server applications
- Can enhance screens with HTML
- Can utilize CICS Transaction Routing
- CICS plus normal Internet security

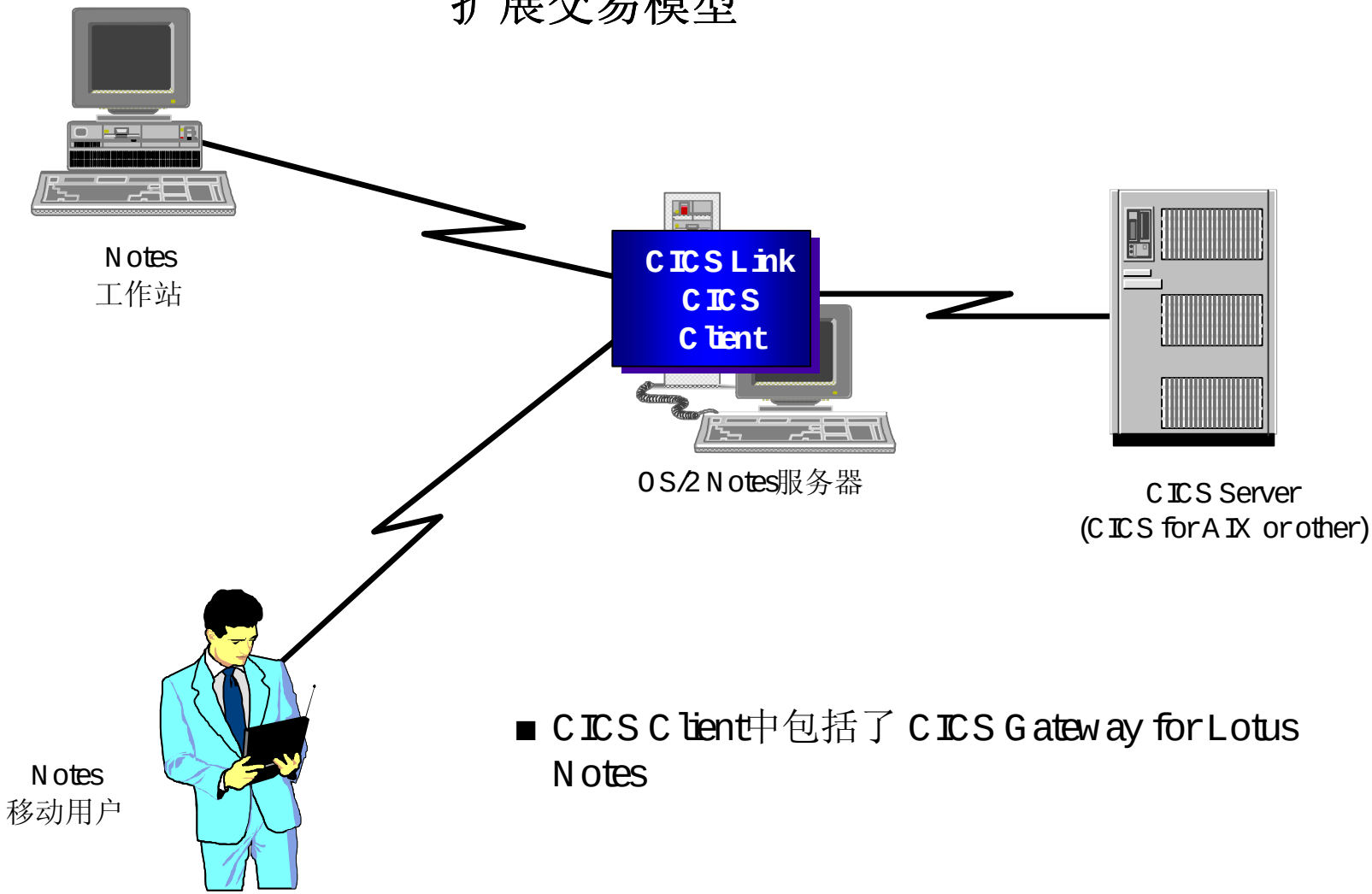
建立CICS Internet应用程序



- Usually new client& server application
- More work - more flexibility
- Can utilize CICS program request routing to other servers
- CICS plus normal Internet security

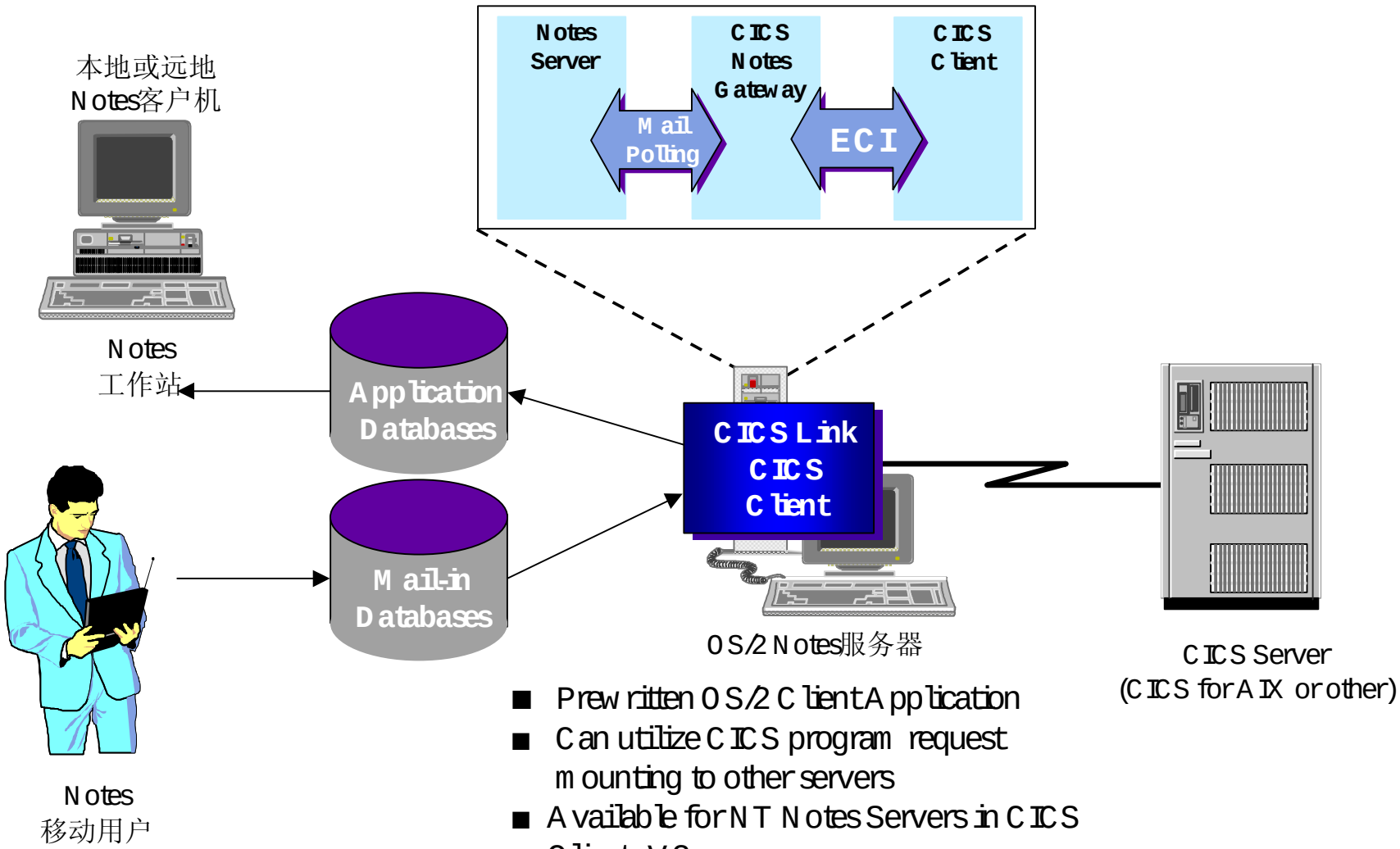
Lotus Notes存取CICS

扩展交易模型



- CICS Client中包括了 CICS Gateway for Lotus Notes

CICS Gateway for Lotus Notes

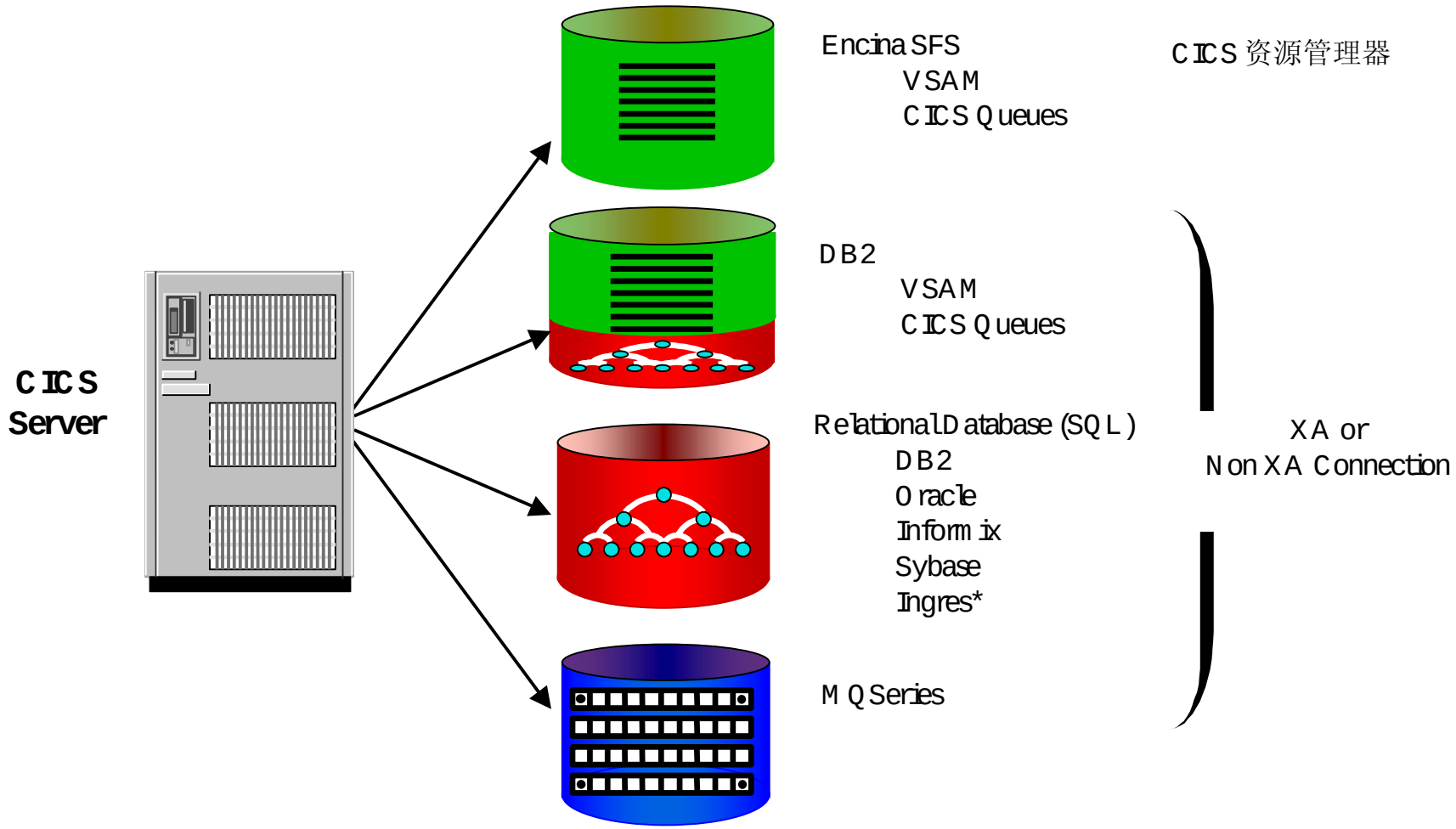


- Prewritten OS/2 Client Application
- Can utilize CICS program request mounting to other servers
- Available for NT Notes Servers in CICS Clients V2

第十一章

IBM TXSeries(CICS)
应用程序开发

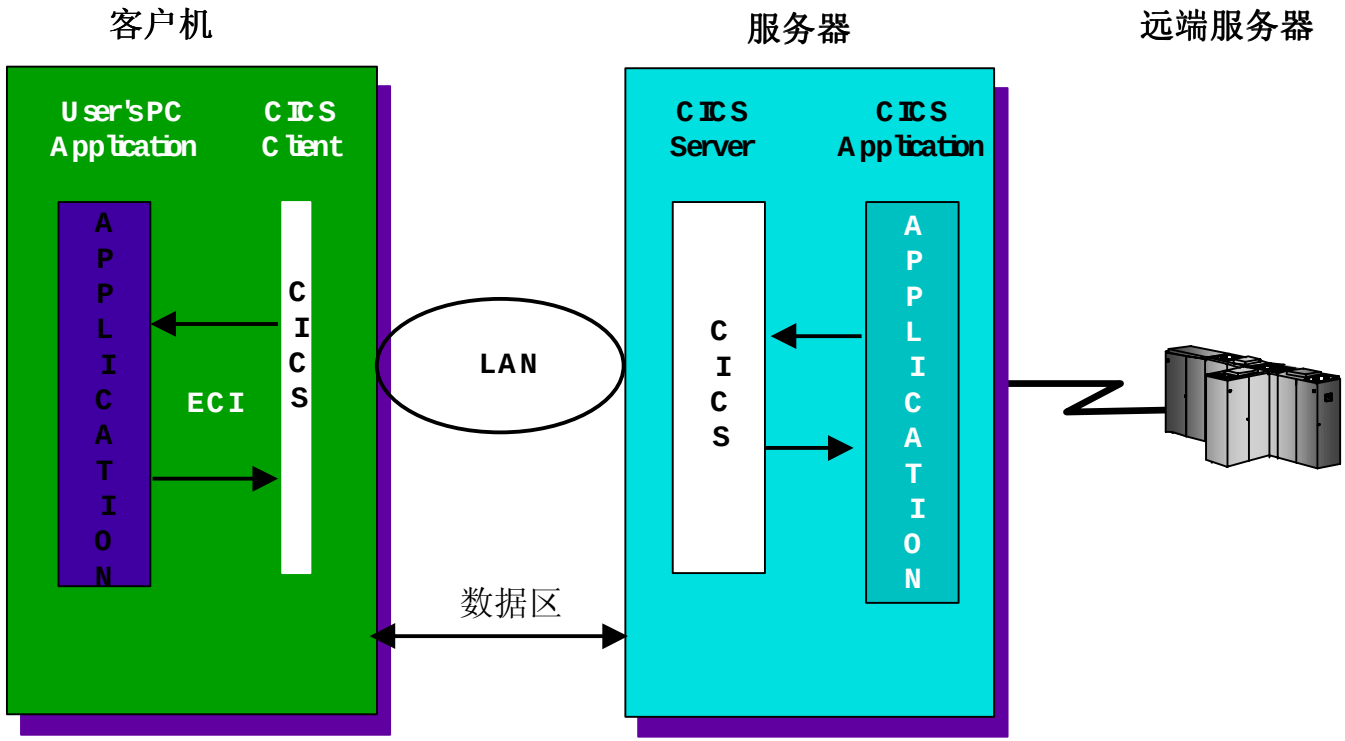
本地资源管理器


CICS
Server
Encina SFS
VSAM
CICS Queues
DB2
VSAM
CICS Queues
Relational Database (SQL)
DB2
Oracle
Informix
Sybase
Ingres*
MQSeries
CICS 资源管理器
XA or
Non XA Connection

CICS Client编程

- 定义
  - 从非CICS的客户程序调用CICS交易，
- 方式
  - External Call Interface，简称ECI
  - External Presentation Interface，简称EPI

External Call Interface (ECI)



A method for a non-CICS Application to call a CICS Program as a subroutine

CICS application could be on server or remote server

Call/Parameter Driven

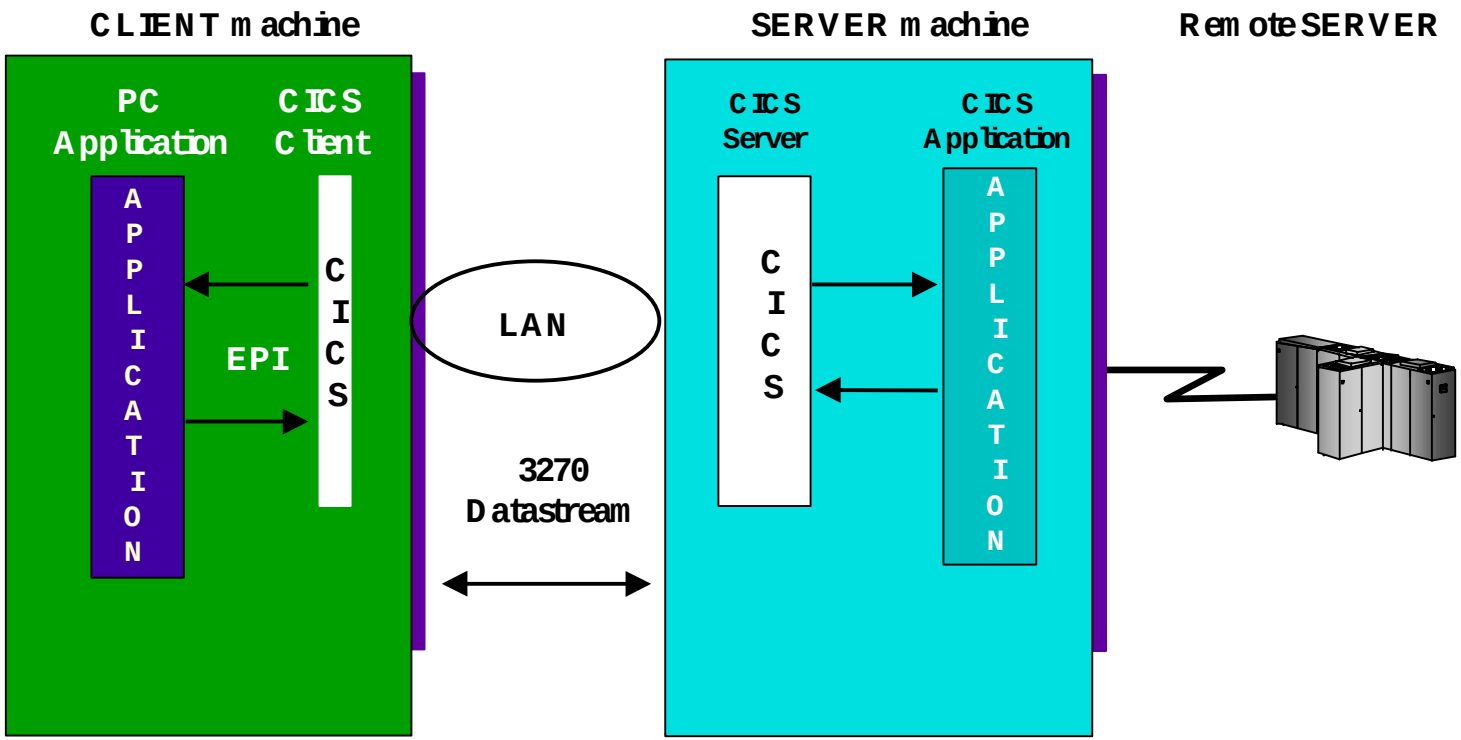
ECI Client端程序例子

```
#include <cics_eci.h>
ECI_PARMS  EciParms;
char commArea[1024];

memset(&EciParms,0,sizeof(ECI_PARMS));
memset(commArea,0,sizeof(commArea));
EciParms.eci_version                    = ECI_VERSION_1A;
EciParms.eci_call_type                  = ECI_SYNC;
memcpy(&EciParms.eci_program_name,  "SERV0001",    8);
memcpy(&EciParms.eci_userid,        "CICSUSER",    8);
memcpy(&EciParms.eci_password,      "CICSUSER",    8);
memcpy(&EciParms.eci_system_name,   "CICSRG01",    8);
EciParms.eci_commarea                   = commArea;
EciParms.eci_commarea_length            = sizeof(commArea);
EciParms.eci_extend_mode                = ECI_NO_EXTEND;
EciParms.eci_luw_token          = ECI_LUW_NEW;
EciParms.eci_timeout                    = 0;
memcpy(EciParms.tpn_name,           "BPMI",        4);
Rc = CICS_ExternalCall(&EciParms);
```

CICS Application Server

External Presentation Interface



- A method for a PC program to pretend to be a 3270 and use a 3270 oriented CICS application
  - Allows a GUI front end to be added to an EXISTING CICS application
    - without changing the CICS application
  - CICS application request could be forwarded to remote server
  - Data Stream or Event Driven
  - Host can initiate applications for connected clients

Server 程序框架 - C

```
main()
{
   unsigned long respCode;
   char *commArea;

   EXEC CICS ADDRESS EIB(dfheiptr) RESP(respCode);
   if (respCode != DFHRESP(NORMAL)) {
       fprintf(stderr, "Error occurred addressing comm area, rc = %d\n", respCode);
       EXEC CICS RETURN;
   }
   EXEC CICS ADDRESS COMMAREA(commArea) RESP(respCode);
   if (respCode != DFHRESP(NORMAL)) {
       fprintf(stderr, "Error occurred addressing comm area, rc = %d\n", respCode);
       EXEC CICS RETURN;
   }
   ...
   EXEC SQL ...
   ...
   EXEC CICS SYNCPOINT;
   strcpy(commArea, "Return from Server.\n");
   EXEC CICS RETURN;
}
```

CICS程序编译

- 数据库预编译
- db2 prep -- db2
- proc -- Oracle
- cpre -- Sybase
  - CICS预编译
- cicstran -lC server.ccs
  - C编译
- cc
- cl
  - 连接
- link
  - cicstcl -lC server.ccs(CCFLAGS=-I -L db2.o/clntshr.a/cs_r.so ct_r.so)

Logical Unit of Work

- CICS 交易中两个CICS 提交/回滚之间的处理为一个LUW
- ECI Client可以控制LUW
  - eci_extend_mode参数
    - ECI_NO_EXTEND结束一个LUW
    - ECI_EXTENDED将LUW 延续至下一次ECI调用
  - eci_luw_token参数
    - 说明LUW 的编号
    - ECI_LUW_NEW 本次CALL 作为一个LUW

CICS 编程接口

- 提供CICS 服务，由服务器端程序调用
- 分类
  - 逻辑控制
  - 数据及存储服务
  - 时间服务
  - 程序跟踪
  - APPC 通讯
  - ...
- 通过CICS API调用-EXEC CICS ...

程序与交易

- 程序

完成一定功能的代码段

- 交易
  - CICS程序运行的特定环境
  - 本身无实际的代码
  - 第一个程序(FirstProgram)，链接其他程序

接口参数块-EIB

- EIB -EXEC Interface Block
- 常用内容
  - EIBCALEN -传输区长度
  - EIBDATE，EIBTIME -程序启动时间
  - EIBREQID -请求编号
  - EIBRESP -回应代码
  - EIBRESP2 -详细回应代码
  - EIBTASKN -任务编号
  - EIBTRNID -交易名称
- 一些可通过CICS API获得

Communication Area

- 通讯区域，由CICS 自动传递
- 使用方式
  - Client/Server通讯
  - Server程序传送数据给被调用程序
  - Server程序传送数据给异步启动交易
  - Server 程序返回数据
- 长度不大于32K
- EXEC CICS ADDRESS COMMAREA

交易内数据共享

- TWA - Transaction Work Area
- 使用CICS私有存储区
  - EXEC CICS GETMAIN SET()
- 使用COMMAREA - LINK 或XTCL
  - EXEC CICS LINK …COMMAREA()
  - EXEC CICS XCTL …COMMAREA()
- 程序自身数据段(如COBOL CALL)
  - CALL USING …

Transaction Work Area

- 同一交易内所有程序共享
- TD中定义大小
  - TWASize (Transaction Work Area Size)
  - 范围0-31767 **Bytes**
- EXEC CICS ADDRESS TWA()

交易间数据共享

- CWA - Common Work Area
- 使用临时存储队列(TSQ)
- 使用暂存数据队列(TDQ)
- 使用VSAM 文件
- 使用CICS RETURN 的COMMAREA
  - EXEC CICS RETURN COMMAREA()
- 共享CICS 存储区
  - EXEC CICS GETMAIN SET()SHARED

Common Work Area

- REGION内所有程序共享
- RD中定义大小
  - CWASize (Common Work Area Size)
  - 范围512 - 3584 **Bytes**
- EXEC CICS ADDRESS CWA()

CICS 内部资源

- 内存资源(包括Private和Shared)
- VSAM 文件
- 临时存储队列(TSQ)
- 暂存数据队列(TDQ)

内存资源

- 由Region定义大小
- 交易私有存储区
  - 交易结束自动释放
  - 交易独享存储量
- 交易共享存储区
  - 需要显式释放
  - 所有交易共享存储总量
- 通过CICS API操作

VSAM 文件

- 结构化文件-含记录结构
- 种类
  - ESDS
    - 以入口为序
    - 只能在尾部增加
  - RRDS
    - 记录长度固定
    - 可以复用删除的记录空间
  - KSDS
    - 以索引为序

临时存储队列(TSQ)

- 序列存放可变长记录的队列,
  - main(在内存中,non-recoverable,在shutdown REGION后丢失)
  - auxiliary(在SFS中,可设成recoverable,在cold启动REGION后丢失)
- 当超过限时inactive状态时,被删除
- 使用之前可不定义;当考虑TSQ为REMOTE,安全考虑或要设成可恢复时,必须先定义
- 可用CEBR来浏览或删除

暂存数据队列(TDQ)

- intrapartition(在SFS文件系统中)
  - trigger
  - recoverable
    - none(不可恢复)
    - physical(当REGION异常中断时,可恢复最后一个READ)
    - logical(根据LUW,可恢复)
  - READ后,自动删除
- extrapartition(在AIX文件系统中)
  - record-oriented AIX文件
- IOMODE(read-only或write-only)

接口存取操作

- EXEC CICS ADDRESS
  - 可获取LINK，XTCL的参数
    - COMMAREA
  - TWA，CWA
- EXEC CICS RETRIEVE
  - 可获取START的参数
    - COMMAREA，TERMID，TRANID
    - RTERMID，RTRANID
- EXEC CICS ASSIGN
  - 可获得环境参数(Region属性等)

内存操作

- EXEC CICS GETMAIN
  - 申请内存，non-shared 或shared
- EXEC CICS FREEMAIN
  - 释放内存
- EXEC CICS LOAD
  - 载入Table or Map
- EXEC CICS RELEASE
  - 释放Table or Map

资源锁操作

- EXEC CICS ENQ
  - 资源加锁
  - NOSUSPEND 遇忙不等待
- EXEC CICS DEQ
  - 释放资源锁

TDQ操作

- EXEC CICS READQ TD
  - Extrapartition TDQ 必须设为input属性
  - Intrapatition TDQ 读出的项被删除
- EXEC CICS WRITEQ TD
  - Extrapartion TDQ必须设为output属性
- EXEC CICS DELETEQ TD
  - 删除TDQ的所有内容
  - 仅对Intrapatition TDQ有效

TSQ 操作

- EXEC CICS READQ TS
  - ITEM 参数指定读某一项
  - NUMITEMS获得队列中项的数量
- EXEC CICS WRITEQ TS
  - 写入一项并从ITEM 参数返回编号
  - 可指定ITEM 对某一项覆盖
  - MAIN |AUXILIRARY指定TSQ在内存或外设
- EXEC CICS DELETEQ TS
  - 删除TSQ的所有项

时间相关操作

- EXEC CICS ASKTIME
  - 获得绝对时间
  - 自1900年1月1日以来的毫秒数
- EXEC CICS FORMATTIME
  - 按要求格式化绝对时间
- EXEC CICS DELAY
  - 使程序延时执行
- EXEC CICS CANCEL
  - 取消延时或取消异步执行的交易

两阶段提交

- CICS作为交易协调服务器
- 阶段一
  - 记录Prepare日志
  - 给所有资源管理器发Prepare命令
  - 收集返回的信息(Ready/Abort)
- 阶段二
  - 如果返回信息都是Ready
    - 记录Commit日志
    - 给所有资源管理器发Commit命令
  - 如果返回信息中含有Abort或超时
    - 记录Abort日志
    - 给返回Ready的资源管理器发Abort命令

访问数据库(non-XA)

- 数据库连接
  - EXEC SQL CONNECT TO DATABASE；数据库操作
  - EXEC SQL ...
- 数据库提交
  - EXEC SQL COMMIT;
  - EXEC SQL ROLLBACK;
- 数据库关闭
  - EXEC SQL DISCONNECT;
- 通过sqlca.sqlcode来判断返回状态

访问数据库(XA)

- 无需数据库连接、关闭
- 数据库操作
  - EXEC SQL ...
- 交易数据提交
  - EXEC CICS SYNCPOINT;
  - EXEC CICS SYNCPOINT ROLLBACK;
- 通过sqlca.sqlcode来判断返回状态

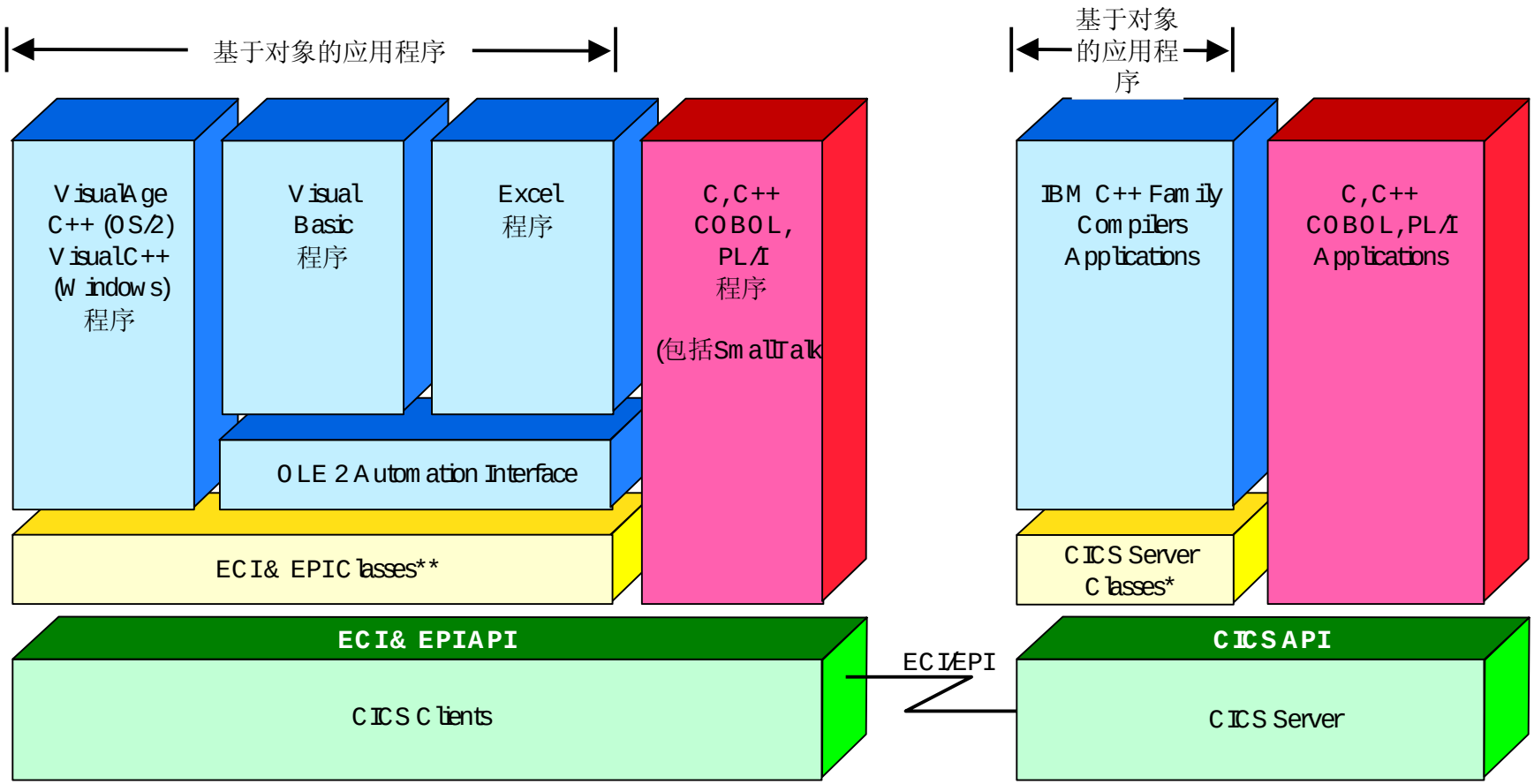
交易管理操作

- EXEC CICS SYNCPOINT
  - 提交CICS 交易，结束一个LUW
  - 在各个资源管理器之间(包括RDBMS)达成两阶段提交
- EXEC CICS SYNCPOINT ROLLBACK
  - 回滚CICS 交易，结束一个LUW

业务流程控制

- EXEC CICS LINK
  - 调用另一个程序
  - 结束返回调用程序
- EXEC CICS XCTL
  - 将控制转给另一个程序
- EXEC CICS START
  - 异步执行另一个交易
- EXEC CICS RETURN
  - 执行返回语句

面向对象应用开发


基于对象的应用程序
基于对象的应用程序
VisualAge C++ (OS/2) VisualC++ (Windows) 程序
Visual Basic 程序
Excel 程序
C,C++ COBOL, PL/I 程序
(包括SmallTalk
OLE 2 Automation Interface
ECI& EPI Classes**
IBM C++ Family Compilers Applications
C,C++ COBOL,PL/I Applications
CICS Server Classes*
ECI& EPI API
CICS Clients
ECI/EPI
CICS API
CICS Server

CICS Foundation Classes

- 新的OO编程接口
  - 针对非CICS程序员
  - 内含C++类库
  - 支持C++异常处理
- Client
  - OS/2, Windows 3.1, 95和NT
  - 含概了所有ECI和EPI函数
  - 在CICS Client V2.0中支持
  - 支持IBM 和Microsoft C++编译器
  - 支持OLE 2
- Server
  - 含概了主要EXEC CICS API(大约40各类库)
  - 有限的3270支持(没有BMS)
  - 无需使用CICS预编译器
  - CICS for OS/2 V3.0, CICS for AIX V2.1, CICS for MVS/ESA V4.1

第十二章

IBM TXSeries(CICS)
开发注意事项

注意事项(一)

- CICS程序不可使用的系统函数
  - fork()、execl()、system()
    - 用EXEC CICS LINK、XCTL、START替代
  - gethostbyname()、gethostbyaddr()、getprotent()、getservbyname();
    - 用gethostbyname_r()、gethostbyaddr_r()、getprotent_r()、getservbyname_r()替代
  - exit()、
    - 用EXEC CICS RETURN替代

注意事项(二)

- CICS中不推荐使用的函数
  - malloc()
    - 用EXEC CICS GETMAIN替代
  - kill()
    - 用EXEC CICS SET TASK PURGETYPE(PURGE |FORCEPURGE)替代

注意事项(三)

- CICS Application Server会保留以下进程状态，因此在交易结束时要注意关闭：
  - open-file descriptors
  - TCP/IP socket descriptors
  - Environment variables
  - Current working directory
  - Process priority
  - Shared memory
  - Dynamically allocated memory

注意事项(四)

- 当CICS程序需要驻留在内存时(PD的Resident=Yes),慎用静态(static)变量

注意事项(五)

- Structure packing
  - struct commAreaStruct
  - {
  - char ch;
  - int i;
  - }
  - sizeof(struct commAreaStruct) = ?
- 编译选项
  - #pragma options align = packed
  - #pragma options align = reset

注意事项(六)

- EXEC SQL DECLARE CURSOR;
- EXEC SQL OPEN CURSOR;
- EXEC SQL CLOSE CURSOR;
- EXEC SQL DEALLOCATE CURSOR;

注意事项(七)

- EXEC SQL PREPARE;
- EXEC DESCRIBE …INTO pSQLDA;
- 替换成
- EXEC …

注意事项(八)

- EXEC SQL SELECT * FROM table1 INTO TEMP tempTable;
- EXEC SQL DROP TABLE tempTable;

第十三章

IBM TXSeries(CICS)
实验

实验预备环境

- 硬件设备(IBM RS/6000,CPU,Memory,Harddisk)
- AIX 4.2.1或以上
- IBM TXSeries for AIX V4.2或以上
- IBM C 3.1.4或以上
- 各种与IBM TXSeries for AIX兼容的数据库及其开发环境(Embedded SQL环境)
- MQSeries for AIX V5.0或以上

实验一:安装CICS Server

- 目标:
  - 安装IBM TXSeries for AIX V4.2
  - 安装补丁
- 步骤:
  - 设置操作系统环境
  - 安装介质

实验二:配置CICS Server

- 目标:
  - 配置CICS Server环境,使得CICS Server可启动并可被本地CICS Client或远地的CICS Client访问
- 步骤:
  - 配置DCE RPC-Only环境
  - 创建Encina SFS
  - 创建CICS Region(默认为CICSRG00)
  - 增加CICS Listener资源(TCP/IP)
  - 增加CICS XA资源(数据库任选)

实验三:安装配置CICS Client

- 目标:
  - 安装CICS Universal Client for AIX V3.0.2
  - 配置CICS Universal Client for AIX V3.0.2
- 步骤:
  - 安装介质
  - 配置

实验四：Hello程序

- 目标：
  - 开发CICS Client/Server Hello程序

实验五:数据库访问程序

- 目标:
  - 开发CICS Client/Server数据库访问程序

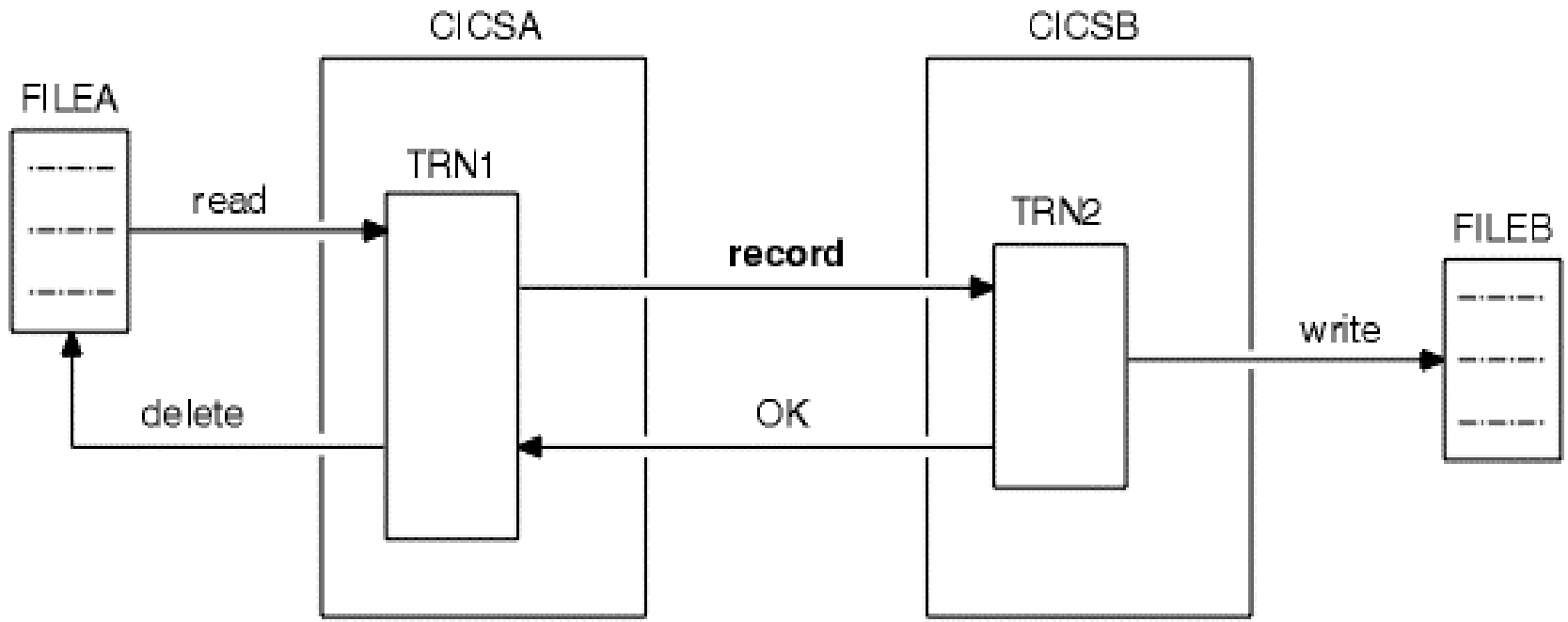
实验六：MQSeries访问程序

- 目标：
  - 开发CICS Client/Server MQSeries访问程序
- 步骤：
  - 配置DCE RPC-Only环境
  - 创建Encina SFS
  - 创建CICS Region(默认为CICSRG00)
  - 增加CICS Listener资源(TCP/IP)
  - 增加CICS XA资源(数据库任选)

CICS Application Server

实验七：通存通兑程序

- 目标：
  - 开发CICS Server通存通兑程序



实验八:单笔多次查询程序

- 目标:
  - 开发CICS Client/Server单笔多次查询程序

实验九:禁止CICS交易程序

- 目标:
  - 开发CICS Server禁止交易程序

第十四章

IBM TXSeries(CICS)
结束语

参考资料和例子程序

CICS/6000 Application Development by Neil Kolban
http://www.software.ibm.com/ts/cics/usr/lpp/cics/src
http://www.software.ibm.com/ts/txseries/library

/usr/lpp/cics/src
/home/sybase/sample/xalibrary/CICS

http://cheng.itgo.com
http://cheng.itgo.com/pub

CICS for AIX用户协会例子

■CICS Internet Gateway to open up student access at University of Florida	G511-3701
■ Bank of China converts to client/server with CICS for AIX	G511-3641
■ CICS for AIX benefits SHL's application rehosting project	G511-3636
■ CICS for AIX takes the credit at JCIC	GC33-1503
■ Cable TV listings company switches on to client/server with CICS for AIX -TVSM	GC11-3639
■ Dun & Bradstreet credits CICS/6000 with faster customer service in promising rightsizing project	GK20-2675
■ CICS for AIX provides mutual benefits at Twentieth Century	G511-3411
■ MEMC (Manufacturer)	GK20-2689
■ Lamonts (Retail)	GK20-2660

Application briefs are available from IBM and are available electronically on the CICS WWW homepage: http://www.hursley.ibm.com/cics/solutions.html

其它CICS书籍

- SC33-1436 : CICS Clients Administration
- SC33-1748 : CICS Clients Gateways
- SC33-1435 : CICS Family - Client/Server Programming
- SC33-1007 : CICS Family - API Structure
- SC33-0824 : CICS Family - Interproduct Communication
- GA23-0059 : IBM 3270 Information Display Programmer Reference
- GC24-2534 : CICS Clients Unmasked (Red Book)
- GC24-4375 : MVS to AIX Application Migration Cookbook (Red Book)
- SG24-4547 : Accessing CICS Business Applications from the World
- Wide Web (Red Book)

第十五章

IBM TXSeries(CICS)

附录

附录一:安装CICS Server(一)

- 设置用户、组
- 组：smitty mkgroup
  - Group NAME　　cics
  - USER list　　root
  - ADMINISTRATOR list　　root

  - Group NAME　　cicsterm
  - USER list　　root
  - ADMINISTRATOR list　　root

- 用户：smitty mkuser
  - User NAME　　cics
  - Primary GROUP　　cics
  - Group SET　　cicsterm

  - User NAME　　SFS_SERV
  - Primary GROUP　　cics
  - Group SET　　cicsterm
  - SoftDATA Segment　　default*2
  - SoftStack Segment　　default*2

附录一:安装CICS Server(二)

- 创建jfs和lv
- jfs:smitty crjfs(选择'Add a Standard Journaled File System '和相应得卷组(vg))
  - SIZE of file system 40000(20 mega bytes)
  - MOUNT POINT /var/cics_servers
  - Mount AUTOMATICALLY at system restart? Yes
  - SIZE of file system 80000(40 mega bytes,根据实际交易量)
  - MOUNT POINT /var/cics_regions
  - Mount AUTOMATICALLY at system restart? Yes
  - "mount /var/cics_servers"
  - "mount /var/cics_regions"
- lv:smitty mklv(用F4键选择相应的卷组(vg))
  - Logical volume NAME sfs_SFS_SERV
  - Number of LOGICAL PARTITIONS 8(32 mega bytes)
  - Logical volume NAME log_SFS_SERV
  - Number of LOGICAL PARTITIONS 8(32 mega bytes)
  - "cd /dev"
  - "chown SFS_SERV:cics *SFS*" (有四个设备会被授权SFS_SERV用户)

附录一:安装CICS Server(三)

- 设置环境变量

编辑"/etc/environment"

- PATH中加入"/usr/lpp/cics/bin:",同时检查数据库"bin"路径是否设置
- "CICSPATH=/usr/lpp/cics"
- "ENCINA_BINDING_FILE=/var/cics_servers/server_bindings
- "RPC_UNSUPPORTED_NETIFS=en1|fddi0"
- "CICSREGION=defaul_region_name"
- "CICS_SFS_SERVER=/.:/cics/sfs/$HOSTNAME"
- "CICS_HOSTS=1.1.1.1,2.2.2.2"

创建?"/var/cics_servers/server_bindings"

- "/.:/cics/sfs/$HOSTNAME ncadg_ip_udp:$host_ip[$sfs_port]

编辑?"/etc/services"

- 加入"sfs_port 8888/udp"

附录一:安装CICS Server(四)

- 软件安装

- "smitty install"
- "Install and Update Software"
- "Install and Update from LATEST Available Software"
- 选择相应的设备
- SOFTWARE to install(用F4选择安装介质)
  - 用F7选择下面的软件
  - cics.base
  - cics.server
  - cics.client
  - cics.info
  - encina.server
  - encina.client
  - encina.PPCexec
  - encina.SFS
  - encina.info

- 安上面相同的方法安装补丁软件(PTF)

附录二:配置CICS Server(一)

- "logout"并以 root用户"login"
- "cicsdefaultservers"
- 配置DCE

➡ "mkdce -o local -n $HOSTNAME rpc"(建立一个DCE RPC-only)

- 生成SFS文件系统

➡ "smitty cics","Manage Filesystem","Manage Encina SFS Servers",
➡ "Define Encina SFS Servers","Create"
- Model SFS Server Identifier ""
- SFS Server Identifier "/.:/cics/sfs/$HOSTNAME"
- Are you using DCE servers "NO"
- Name Service for advertising server "NONE"

- 冷启动SFS文件系统

- "cicssfscold /.:/cics/sfs/$HOSTNAME"

- 生成CICS REGION

➡ "smitty cics" "Manage CICS Regions" "Create (Import) a CICS Region"
- Name of Region to be created "CICSRG01"
- Force use or no-use of DCE servers? "do not use DCE servers"

➡ 配置CICS资源到SFS
- "cicssfsconf -R wc CICSRG1 DefaultFileServer=/.:/cics/sfs/$HOSTNAME"

CICS Application Server

附录二:配置CICS Server(二)

- 配置TX Series Server Listener

➡ "smitty cics" "Manage CICS Regions" "Define CICS Resources" "Listeners" "Add New"

- Listener Identifier "TCPIPL1"
- TCP adapter address "194.2.201.254"
- TCP service name "tcpipl1"

➡ "vi /etc/services",

- 加入"tcpipl1 9999/tcp"

- 配置2 Phase XA与db2数据库的连接

➡ 生成连接程序(Switch Load File)

- "cd /usr/lpp/db2_05_00/lib"
- "ar -vx libdb2.a"
- "mv shr.o db2.o"
- "cd /usr/lpp/cics/src/examples/xa/"
- 修改db2xa.mk文件中相应的DB2环境变量
- "make -f db2xa.mk"生成db2xa
  - "mv db2xa /var/cics_regions/$CICSREGION/bin/"

CICS Application Server

附录二:配置CICS Server(三)

- 配置XA
  - "smitty cics" "Manage CICS Regions" "Define CICS Resources"
  - "XA Configure" "New"
  - Identifier: "sample"
  - Switch Load File Path Name "db2xa"
  - Resource Manager Initialization String: "dbname,user,password"

- 配置环境变量使得root和cics用户可以存取DB2
  - "vi /etc/profile",加入"./home/db2/sqllib/db2profile"
  - "vi /var/cics_regions/$CICSREGION/environment"加入 "DB2INSTANCE=db2"

附录三:安装配置CICS Client

- 以root用户登入安装

- uncompress /tmp/cics-302.tar.Z
- tar xvf /tmp/cics-302.tar
- ksh mkcicscli.
- ksh mkclimsgsus

- "cd /usr/lpp/cicscli/bin"

- "vi CICSCLI.INI" 加入以下内容

- "Server=CICSRG1
- Description=TCP/IP Server
- Protocol=TCPIP
- NetName=194.2.201.254
- Port=1435"

附录四：Hello client.c程序(一)

```
#include <stdio.h>
#include <string.h>

#include <cics_eci.h>

/* GlobalVariables */
ECI_PARMS           EciParms;
char            Server[9] = "CICSRG01";     /* FILL IN YOUR SERVER HERE */
char            UserID[9] = "CICSUSER";     /* FILL IN YOUR USER ID HERE */
char            Passwd[9] = "";             /* FILL IN YOUR PASSWORD HERE */

void                EciSync     (void);

int main(void)
{
  EciSync ();

  return 0;
} /* main */
```

附录四：Hello client.c程序(二)

```
void EciSync(void)        /* Issue a CICS_ExternalCall for an ECI_SYNC */
{
  short       Rc;
  char        CommArea[256];
  char        Name[256] = "Hello From Client";

  memset(CommArea, '0', 256);
  memset(&EciParms, 0, sizeof(ECI_PARMS));

  EciParms.eci_version                = ECI_VERSION_1A;
  EciParms.eci_call_type              = ECI_SYNC;
  memcpy(&EciParms.eci_program_name, "SERVER", 6);
  memcpy(&EciParms.eci_userid,   UserID,  8);
  memcpy(&EciParms.eci_password,  Passwd,  8);
  memcpy(&EciParms.eci_system_name, Server,  8);
  EciParms.eci_commarea               = CommArea;
  EciParms.eci_commarea_length          = 256;
  EciParms.eci_extend_mode            = ECI_NO_EXTEND;
  EciParms.eci_luw_token              = ECI_LUW_NEW;
  EciParms.eci_timeout                = 0;

  Rc = CICS_ExternalCall(&EciParms);
   if(Rc == ECI_NO_ERROR)
  {
    CommArea[(256-1)] = '0';
    printf("CommArea Returned:%s", CommArea);
  } /* endif */
} /* EciSync */
```

CICS Application Server

附录四：Hello server.ccs程序(三)

```
#include <stdio.h>
#include <stdlib.h>

void main(void)
{
   unsigned long respCode;
   char * pCommArea;

   EXEC CICS ADDRESS EIB(dfheiptr) RESP(respCode);
   if (respCode != DFHRESP(NORMAL)) {
       fprintf(stderr, "Error occurred addressing comm area, rc = %d\n", respCode);
       EXEC CICS ABEND ABCODE("AEER");
   }
   EXEC CICS ADDRESS COMMAREA(pCommArea) RESP(respCode);
   if (respCode != DFHRESP(NORMAL)) {
       fprintf(stderr, "Error occurred addressing comm area, rc = %d\n", respCode);
       EXEC CICS ABEND ABCODE("ACER");
   }

   fprintf(stderr,"commArea from Client is [%s]\n", pCommArea);

   sprintf(pCommArea, "Hello from Server.\n");

   EXEC CICS RETURN;
}
```

附录四：Hello makefile程序(四)

```
all: client server

client: client.c
    xlc_r4 -c -DCICS_AIX -I/usr/lpp/cicscli/include client.c
    xlc_r4 -o client client.o -L/usr/lpp/cicscli/lib -lpthreads -lc_r -lcclaix

server: server.ccs
    cicstcl -lC server
    mv server /var/cics_regions/CICSRG01/bin/server
    cicsadd -c pd -r CICSRG01 -B SERVER RSLKey=public PathName=server
```

附录五:数据库访问client.c(一)

```
#include <stdio.h>
#include <string.h>

#include <cics_eci.h>

/* Global Variables */
ECI_PARMS           EciParms;
char            Server[9] = "CICSRG01";     /* FILL IN YOUR SERVER HERE */
char            UserID[9] = "CICSUSER";     /* FILL IN YOUR USER ID HERE */
char            Passwd[9] = "";             /* FILL IN YOUR PASSWORD HERE */

void                EciSync     (void);

int main(void)
{
  EciSync ();

  return 0;
} /* main */
```

附录五:数据库访问client.c(二)

```
void EciSync (void)       /* Issue a CICS_External call for an ECI_SYNC */
{
  short       Rc;
  char        CommArea[256];
  char        Name[256] = "Hello From Client";

  memset(CommArea, '0', 256);
  memset(&EciParms, 0, sizeof(ECI_PARMS));

  EciParms.eci_version                = ECI_VERSION_1A;
  EciParms.eci_call_type              = ECI_SYNC;
  memcpy(&EciParms.eci_program_name, "SERVER", 6);
  memcpy(&EciParms.eci_userid,   UserID,  8);
  memcpy(&EciParms.eci_password,  PassWd,  8);
  memcpy(&EciParms.eci_system_name, Server,  8);
  EciParms.eci_commarea               = CommArea;
  EciParms.eci_commarea_length          = 256;
  EciParms.eci_extend_mode            = ECI_NO_EXTEND;
  EciParms.eci_luw_token              = ECI_LUW_NEW;
  EciParms.eci_timeout                = 0;

  Rc = CICS_ExternalCall(&EciParms);
   if(Rc == ECI_NO_ERROR)
   {
     CommArea[(256-1)] = '0';
     printf("CommArea Returned:%s",CommArea);
   } /* endif */
} /* EciSync */
```

CICS Application Server

附录五:数据库访问server.sqc(三)

```
#include <stdio.h>
#include <stdlib.h>
EXEC SQL INCLUDE sqlca;
void main(void)
{
  EXEC SQL BEGIN DECLARE SECTION;
    short count;
  EXEC SQL END DECLARE SECTION;
  unsigned long respCode;
  char * pCommArea;

  EXEC CICS ADDRESS EIB(dfheiptr) RESP(respCode);
  if (respCode != DFHRESP(NORMAL)) {
     fprintf(stderr, "Error occurred addressing comm area, rc = %d\n", respCode);
     EXEC CICS ABEND ABCODE("AEER");
  }
  EXEC CICS ADDRESS COMMAREA(pCommArea) RESP(respCode);
  if (respCode != DFHRESP(NORMAL)) {
     fprintf(stderr, "Error occurred addressing comm area, rc = %d\n", respCode);
     EXEC CICS ABEND ABCODE("ACER");
  }
  EXEC SQL SELECT COUNT(*) INTO :count FROM <TableName>;
  if (SQLCODE < 0) {
    fprintf(stderr,"SQL ERROR in SELECT sqlcode [%d]\n",SQLCODE);
    EXEC CICS ABEND ABCODE("DSER");
  }
  sprintf(pCommArea, "Count of sales is [%d]\n",count);
  EXEC CICS RETURN;
}
```

CICS Application Server

附录五:数据库访问makefile(四)

```
all: client server

client: client.c
    xlc_r4 -c -D CICS_AIX -I/usr/lpp/cicscli/include client.c
    xlc_r4 -o client client.o -L/usr/lpp/cicscli/lib -lpthreads -lc_r -lcclaix

server.ccs: server.sqc
    db2 connect to sample
    db2 prep server.sqc
    db2 grant execute on package server to public
    mv server.c server.ccs

server: server.ccs
    CCFLAGS="-I/usr/lpp/db2_05_00/include -L/usr/lpp/db2_05_00/lib /usr/lib/db2.o";\
    export CCFLAGS;\
    cicstcl -lC server
    mv server /var/cics_regions/CICSRG01/bin/server
```

附录六：MQSeries访问程序

- 增加一个CICS Region XA资源：

```
cicsadd -c xad -r CICSRG01 SwitchLoadFile="/usr/lpp/mqm/lib/amqzsc"
XAOpen=<queue_manager_name>
```

- 参考例子程序：

```
/usr/lpp/mqm/samp/amqzscin.c
```

- 编译：

```
CCFLAGS=-I/usr/lpp/mqm/inc -L/usr/lpp/mqm/lib -lmqm_r
cicstcl -lC server.ccs
mv server /var/cics_regions/CICSRG01/bin/server
```

附录七:通存通兑程序

- cicsadd -c cd -r CICSRG01 RG01 TcpAddress="190.9.200.1" TcpPort="1436"
- cicsterm

CRTE SYSID=RG02

CESN ...

附录八:单笔多次查询程序

附录九:禁止CICS交易程序

附录十：ADDRESS

- EXEC CICS ADDRESS
  - [COMMAREA(ptr-ref)]
  - [CWA(ptr-ref)]
  - [EIB(ptr-ref)]
  - [TCTUA(ptr-ref)]
  - [TWA(ptr-ref)]

附录十：ABEND

- EXEC CICS ABEND
  - [ABCODE(name)]
  - [CANCEL]

附录十：HANDLE ABEND

- EXEC CICS HANDLE ABEND
  - [PROGRAM (name)] | [LABEL(label)] |
  - CANCEL | RESET

附录十:IGNORE CONDITION

- EXEC CICS IGNORE CONDITION
  - condition

附录十：SYNCPOINT

- EXEC CICS SYNCPOINT
  - [ROLLBACK]

附录十 :RETURN

- EXEC CICS RETURN
  - [TRANSID (name) [COMMAREA (data-area)
  - [LENGTH (data-value)]]]

附录十：ASKTIME

- EXEC CICS ASKTIME
  - [ABSTIME(data-area)]

附录十:FORMATTIME

- EXEC CICS FORMATTIME
  - ABSTIME(data-area) [YYDDD(data-area)]
  - [YYMMDD(data-area)] [DDMMYY(data-area)]
  - [MMDDYY(data-area)] [DATE(data-area)]
  - [DATEFORM (data-area)] [DATESEP(data-area)]
  - [DAYCOUNT(data-area)] [DAYOFWEEK(data-area)]
  - [DAYOFMONTH(data-area)] [MONTHOFYEAR(data-area)]
  - [YEAR(data-area)] [TIME(data-area) [TIMESEP [(data-area)]]]

附录十：GETMAIN

- EXEC CICS GETMAIN
  - SET(ptr-ref)
  - {LENGTH(data-value)|FLENGT(data-value)}
  - [INITIMG(data-value)]
  - [NOSUSPEND]
  - [SHARED]

附录十：FREEMAIN

- EXEC CICS FREEMAIN
  - DATA(data-area)

附录十：ENQ

- EXEC CICS ENQ
  - RESOURCE(data-area)
  - [LENGTH(data-value)]
  - [NOSUSPEND]

附录十：DEQ

- EXEC CICS DEQ
  - RESOURCE(data-area)
  - [LENGTH(data-value)]

附录十:LOAD

- EXEC CICS LOAD
  - PROGRAM (name)
  - [SET(ptr-ref)]
  - [LENGTH(data-area) |FLENGTH(data-area)]
  - [ENTRY(ptr-ref)]
  - [HOLD]

附录十：RELEASE

- EXEC CICS RELEASE
  - PROGRAM(name)

附录十:LINK

- EXEC CICS LINK
  - PROGRAM (name)
  - [COMMAREA (data-area) [LENGTH (data-value)]]
  - [SYSID (name)]
  - [SYNCONRETURN]
  - [TRANSID (data-value)]
  - [DATALENGTH (data-value)]

附录十：XCTL

- EXEC CICS XCTL
  - PROGRAM (name)
  - [COMMAREA (data-area) [LENGTH (data-value)]]

附录十：START

- EXEC CICS START
  - [INTERVAL(hhmmss) |
  - AFTER [HOURS(data-value) [MINUTES(data-value)] |
    [SECONDS(data-value)] |
  - AT [HOURS(data-value) [MINUTES(data-value)] |
    [SECONDS(data-value)]]
  - TRANSID(name) [REQID(name)] [FROM(data-area)
    LENGTH(data-value)] [TERMID(name)] [SYSID(name)]
    [RTRANSID(name)] [RTREMID(name)] [QUEUE(name)]
    [NOCHECK] [PROTECT]

附录十：RETRIEVE

- EXEC CICS RETRIEVE
  - [INTO (data-area) |SET (ptr-ref)]
  - [LENGTH (data-area)] [RTRANSID (data-area)]
  - [RTERM ID (data-area)] [QUEUE (data-area)]

附录十：DELAY

- EXEC CICS DELAY
  - [INTERVAL(hhmmss)|TIME(hhmmss)|
  - FOR [HOURS(data-value)]
    - [MINUTES(data-value)]
    - [SECONDS(data-value)]|
  - UNTIL [HOURS(data-value)]
    - [MINUTES(data-value)]
    - [SECONDS(data-value)]]
  - [REQID(name)]

附录十：CANCEL

- EXEC CICS CANCEL
  - REQID(name)
  - [TRANSID(name)]
  - [SYSID(name)]

附录十：DELETE

- EXEC CICS DELETE
  - FILE(name)
  - [SYSID(name)
  - [RIDFLD(data-area)
  - [KEYLENGTH(date-value)
  - [GENERIC [NUMREC(data-area)]]]]
  - [RRN]

附录十：READQ TD

- EXEC CICS READQ TD QUEUE(name)
  - [INTO (data-area) |SET (ptr-ref)] [LENGTH (data-area)]
  - [SYSID (name)] [NOSUSPEND]

附录十：WRITEQ TD

- EXEC CICS WRITEQ TD QUEUE(name)
- FROM (data-area) [LENGTH (data-value)]
- [SYSID (name)]

注释：加一条记录到TDQ中。

附录十：DELETEQ TD

- EXEC CICS DELETEQ TD
  - QUEUE(name)
  - [SYSID(name)]

注释：删除Intrapartion TDQ的内容。

附录十：READQ TS

- EXEC CICS READQ TS QUEUE(name)
- {INTO(data-area)|SET(ptf-ref)} [LENGTH(data-area)]
- [NUMITEMS(data-area)] [ITEM(data-value)|NEXT]
- [SYSID(name)]

注释：从temporary storage queue中读一条记录；
所有交易共享这TSQ的一个记录指针；
[NUMITEMS]返回总的记录条数；
[ITEM]指定读哪条记录。

附录十：WRITEQ TS

- EXEC CICS WRITEQ TS QUEUE(name)
- FROM (data-area) [LENGTH (data-area)]
- [ITEM (data-value) [REWRITE]] [SYSID (name)]
- [MAIN |AUXILIARY] [SUSPEND]

注释：加一条记录到temporary storage queue中；
若TSQ不存在，自动创建；
[ITEM]返回当前记录，或与[REWRITE]一起
修改某条记录。

附录十：DELETEQ TS

- EXEC CICS DELETEQ TS
  - QUEUE(name)
  - [SYSID(name)]

注释：删除 main 或 auxiliary Temporary storage queue；

CICS会根据RD中定义的TSQAgeLimit来自动删除一段时间未用得TSQ.

附录十：DUMP

- EXEC CICS DUMP
  - DUMPCODE(name)
  - FROM (data-area) {LENGTH(data-value)|FLENGTH(data-value)]
  - [COMPLETE][TASK][STORAGE][PROGRAM][TERMINAL]
  - [TABLES][DCT][FCT][PCT][PPT][SIT][TCT]

注释：获取CICS内部存储区的DUMP.

附录十：ENTER

- EXEC CICS ENTER
  - TRACEID(data-value)
  - [FROM(data-area)]
  - [RESOURCE(name)]
  - [ENTRYNAME(name)]
  - [ACCOUNT]
  - [MONITOR]
  - [PERFORM]

注释：写一条TRACE记录。

附录十：ASSIGN

- EXEC CICS ASSIGN
  - [ABCODE]
  - [SYSID]
  - [TWALENG]
  - [APPLID]
  - [CWALENG]

注释：获取当前CICS REGION的属性.