BP06: DB2 UDB for iSeries
Improved Transaction Processing

Jarek Miszczyk
PartnerWorld for Developers
**Database Technology Team**

Special thanks extended to
John Orbeck
IBM Rochester Development

F03BP06 - Improved Transaction Processing.PRZ

Agenda

- Adaptive e-Transaction Services
  - ► Historical view of transactions on iSeries servers
  - ► Industry expectations
  - ► Adaptive e-Transaction services objectives
  - ► Architecture
- Using Microsoft Transaction Server with DB2 UDB for iSeries
  - ► MTS overview
  - ► Client Access ODBC driver configuration
  - ► MTS component configuration
  - ► Installing the component under MTS
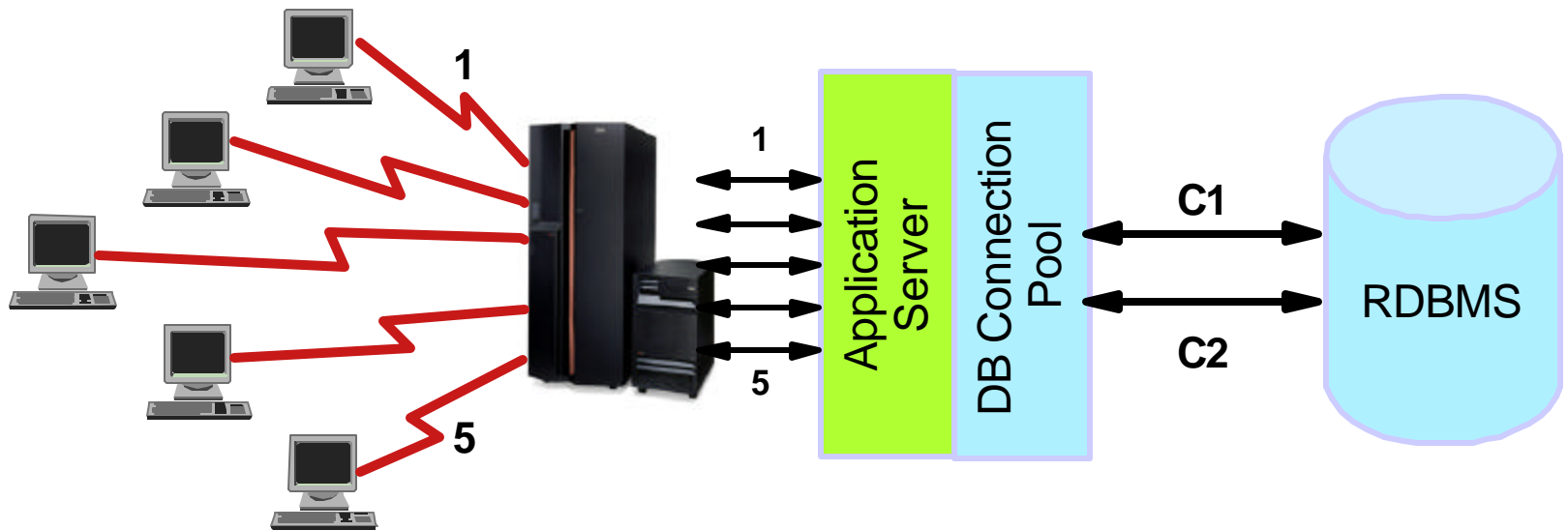 - ► Troubleshooting Tips

e-business Application on iSeries Today
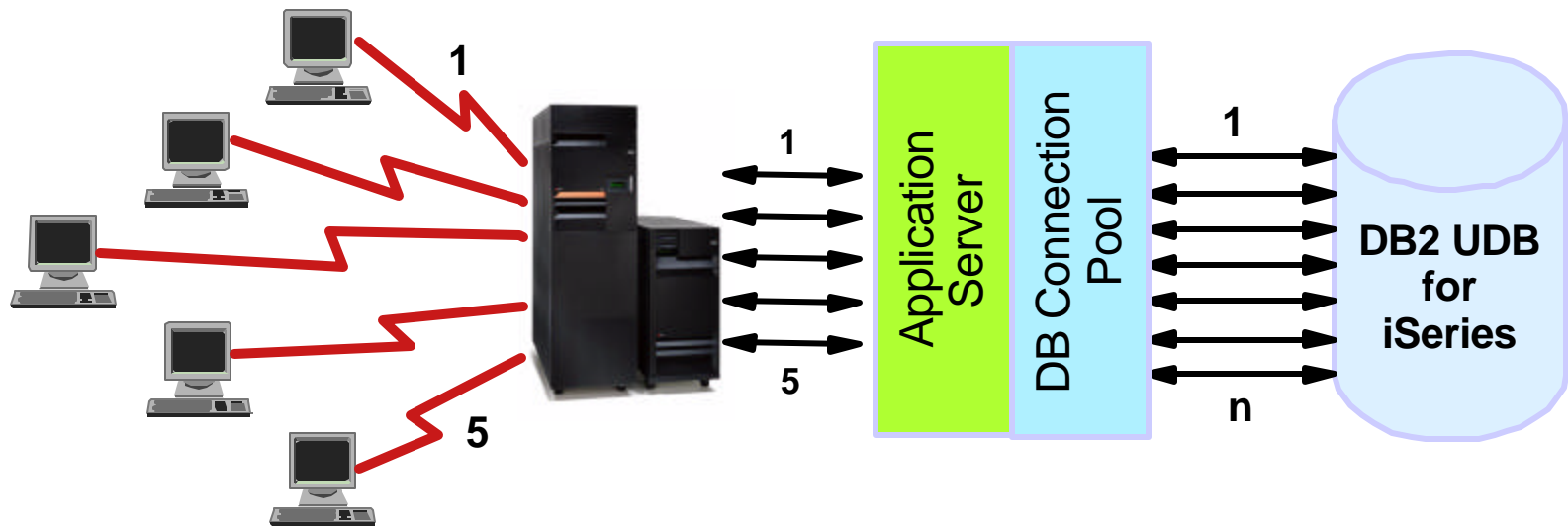
1000s of clients

- 1 to 1 relationship between client transaction and system resources
  - one connection associated with each commit definition
- Resources not reusable until client completes transaction

e-business Industry Application Model

1000s of clients

- 1 to many relationship between client transaction and system resources
- Implemented using industry standard transaction specifications (JTA, XA, CORBA, JTS)
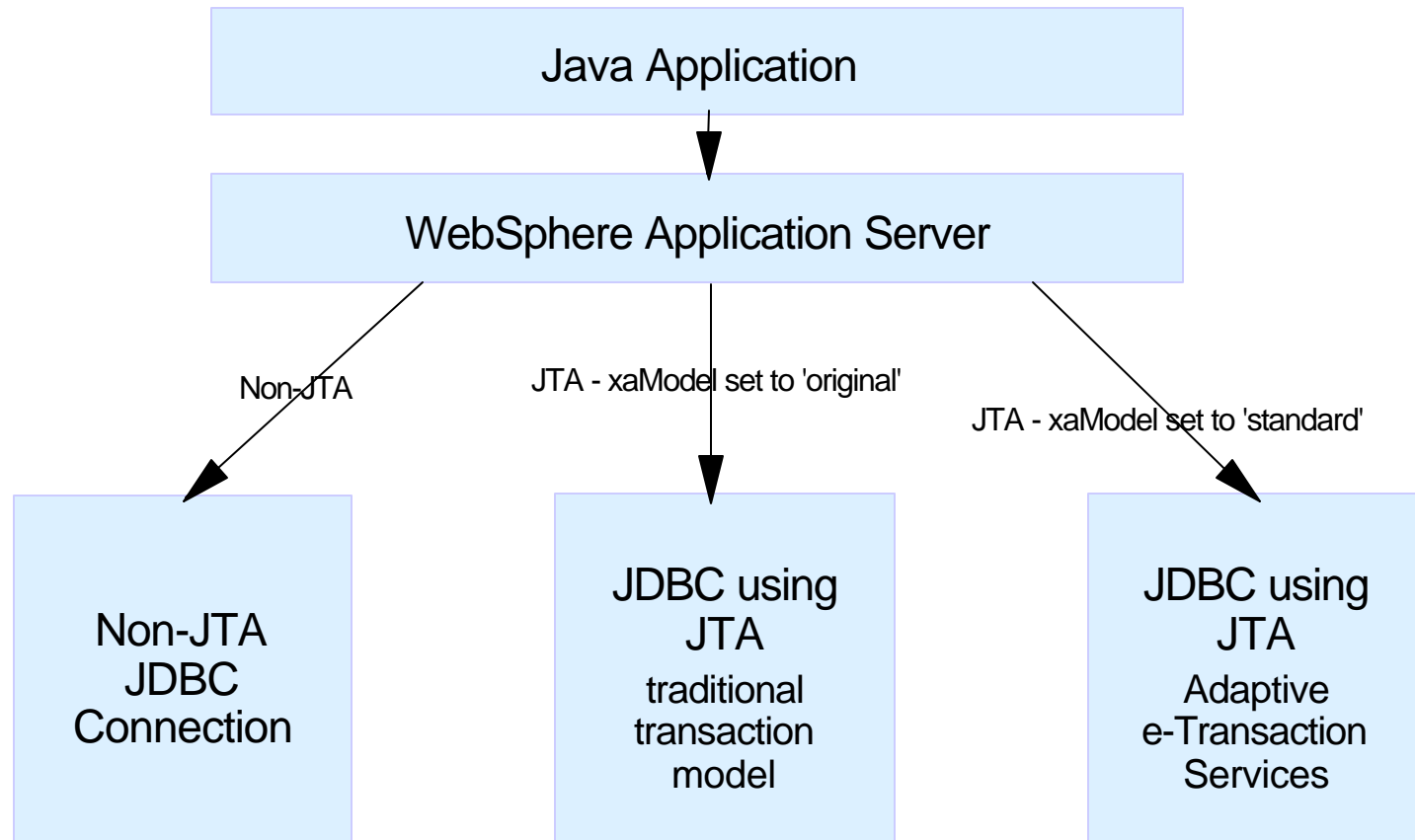- Websphere, Tuxedo provide transaction management

Application Server support for JTA
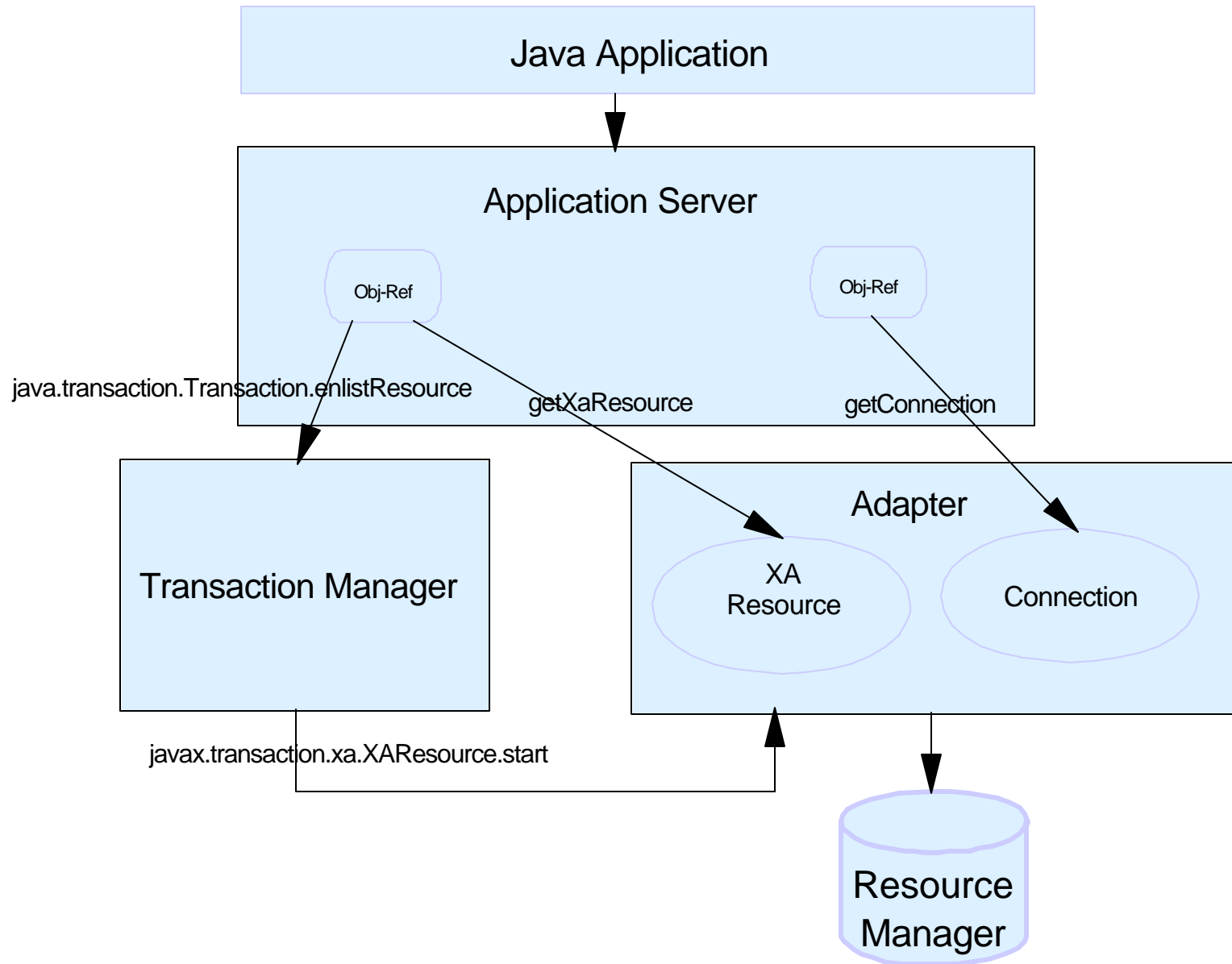
# iSeries Adaptive e-Transactions Services Objectives

- Compatible with other DB2 UDB offerings
  - ► Equivalent XA and JTA support
  - ► Savepoint transactions
- Compliant with XA standard and JTA
  - ► Multiple transactions per DB connection
  - ► Multiple DB connections per transaction
- Eliminate iSeries specific code to manage transactions and DB connections
- Attract new applications and ISVs
  - ► V5R2 support enables porting of applications to iSeries servers
- Increased performance
- Improved scalability
  - ► Resources shared between clients
  - ► Efficient use of system resources

Enabling Adaptive e-Transaction Services

```
                    ┌─────────────────────────────────────────┐
                    │           Java Application               │
                    └─────────────────────────────────────────┘
                                      │
                                      ▼
                    ┌─────────────────────────────────────────┐
                    │       WebSphere Application Server       │
                    └─────────────────────────────────────────┘
```

Non-JTA   JTA - xaModel set to 'original'   JTA - xaModel set to 'standard'

| Non-JTA JDBC Connection	JDBC using JTA traditional transaction model	JDBC using JTA Adaptive e-Transaction Services

- Enabling Adaptive e-Transaction Services for WebSphere (V5R2)

  - Add a value to a new custom property xaModel and set its value to 'standard'

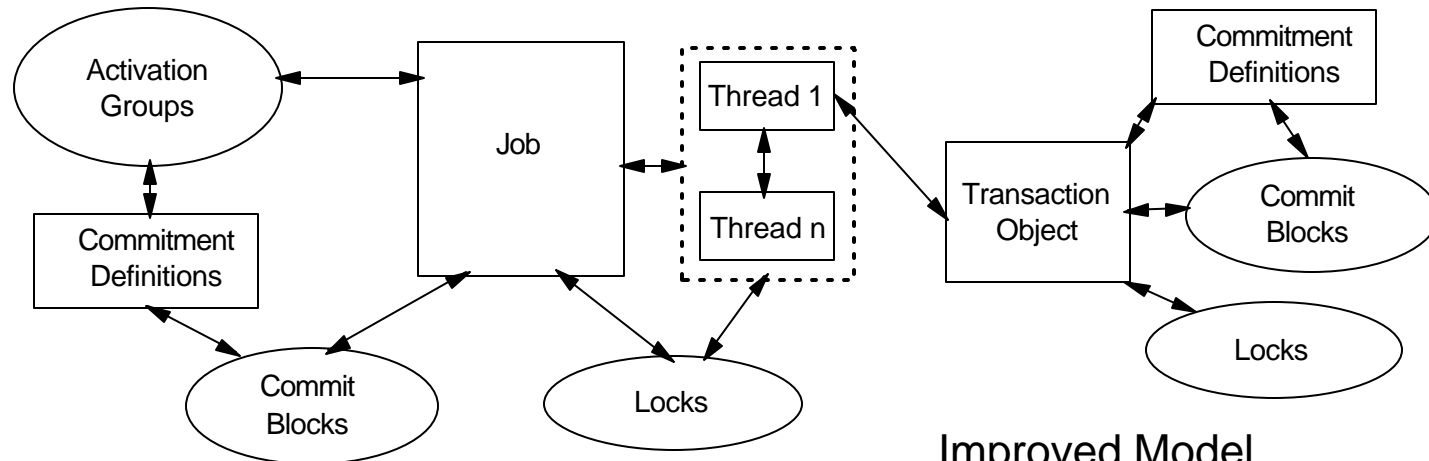 - If not specified, the current code path is used

V5R2 - Transaction Object

Transaction Object
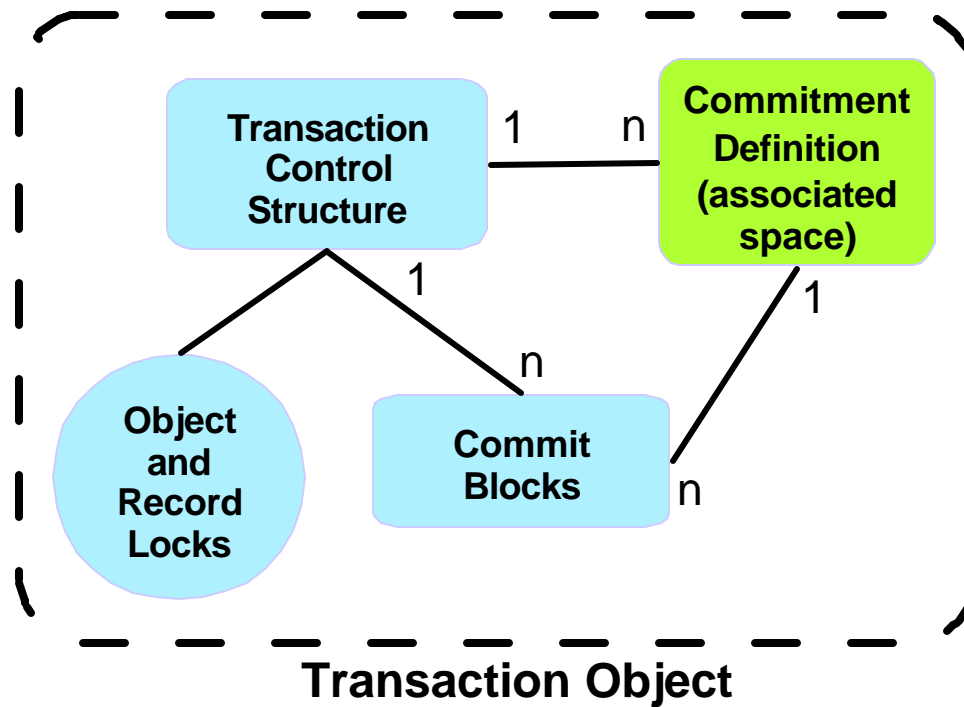
- Breakthrough technology that is a natural extension of the iSeries architecture
- Enables resources to be available where needed

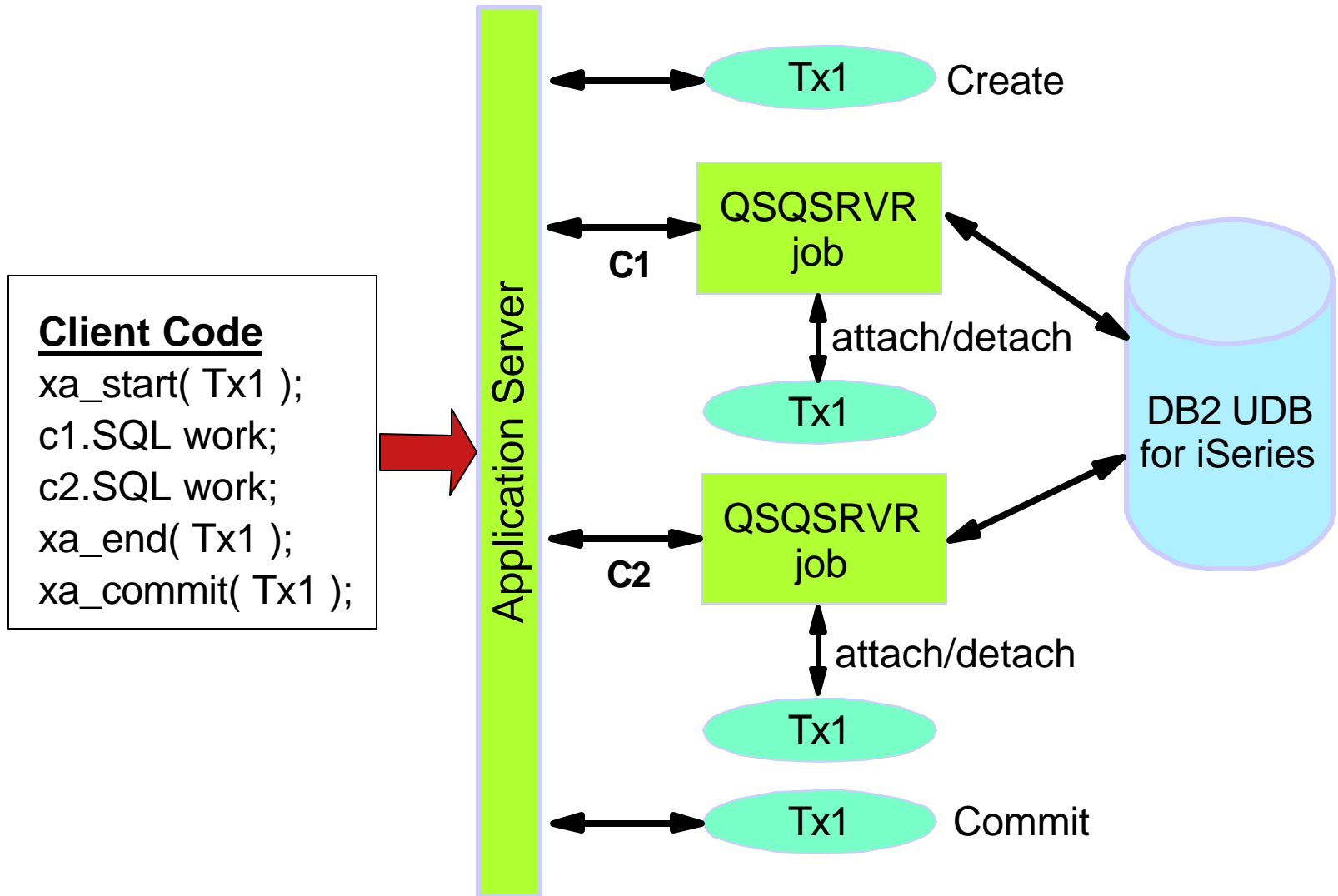# Traditional versus Improved (Ae-TS) Model
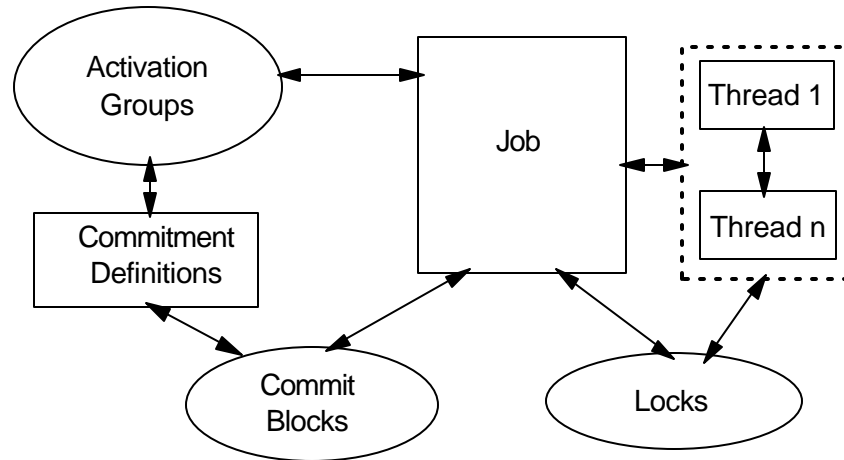
Traditional Model
- Resources and locks associated with job

Improved Model
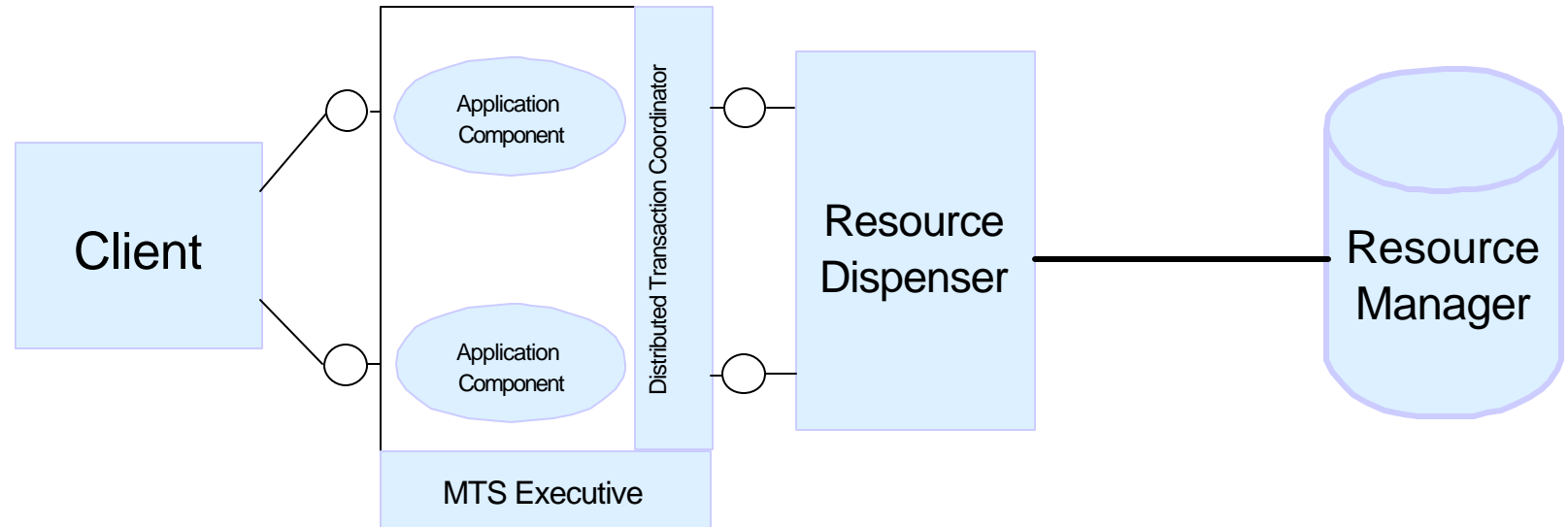- Resources and locks associated independent of job

V5R2 Application Model

Client Code
xa_start(Tx1);
c1.SQL work;
c2.SQL work;
xa_end(Tx1);
xa_commit(Tx1);

Application Server

Tx1 Create

QSQSRVR job

C1

attach/detach

Tx1

QSQSRVR job

C2

attach/detach

Tx1

Tx1 Commit

DB2 UDB for iSeries
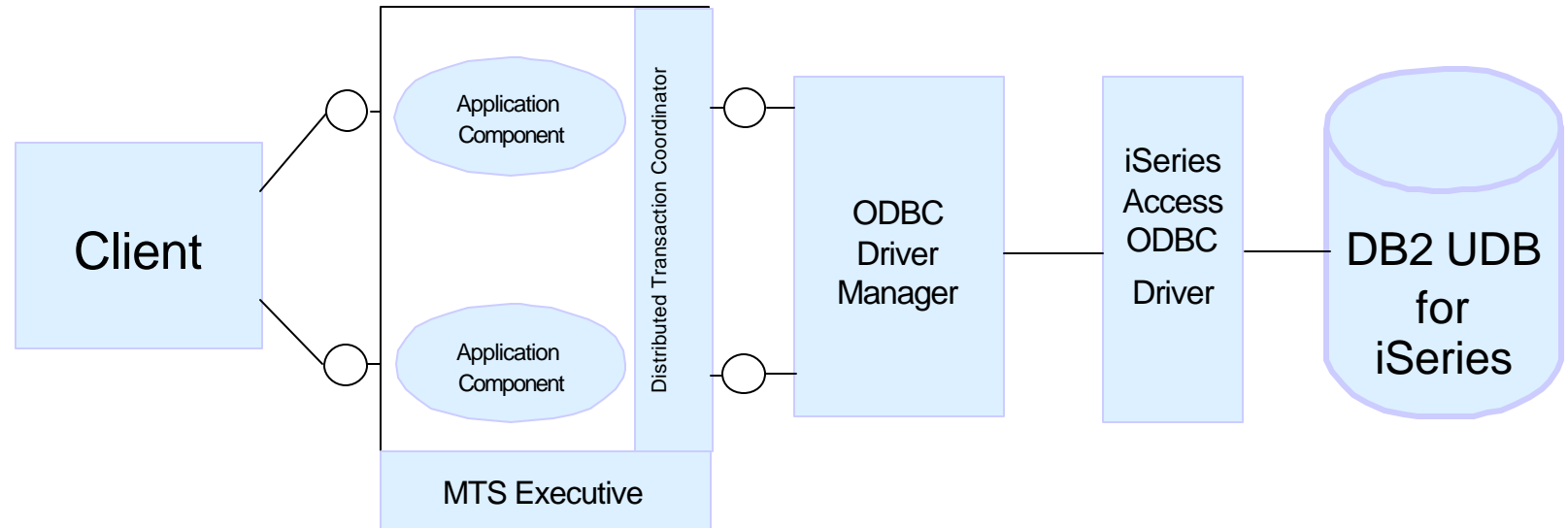
MTS Support

MTS Architecture

- Application Components implement the business rules
  - ▶ Components run in the MTS run-time environment
- MTS Executive is a dynamic-link library (DLL) that provides run-time services for MTS components
  - ▶ MTS handles the concurrency, resource pooling, security, context management
- ODBC resource dispenser manages pools of database connections for MTS components
  - ▶ ODBC 3.x Driver Manager is the ODBC resource dispenser

MTS Architecture cont.

- Microsoft Distributed Transaction Coordinator (MS DTC) is a system service that coordinates transactions
  - ► Work can be committed as an atomic transaction that can span multiple resource managers
  - ► Implements two phase commit protocol
- Resource Manager is a system service that manages persistent data
  - ► Works in cooperation with the Microsoft Distributed Transaction Coordinator to guarantee atomicity and isolation to an application
  - ► MTS supports the resource managers that implement OLE Transactions protocol (OLE TX)
    - – Also provides an OLE TX to XA mapper

Using DB2 UDB for iSeries with MTS

- MTS communicates with DB2 UDB for iSeries through the iSeries Access ODBC driver
- DB2 UDB for iSeries MTS support was successfully tested in the following environment
  - ► iSeries servers
    - – OS/400 V5R1, database group PTF SF99501-06, V5R1 Client Access ODBC driver with Service Pak 3
  - ► Microsoft Windows
    - – Windows 4.0 with SP6, MTS 2.0 SP1, and MDAC 2.6 (for development)
    - – Windows 2000 Professional with SP1, COM+, and MDAC 2.6

Client Access ODBC Driver Setup

- **Turn off the OLE DB Session Pooling**
  - ▶ Use Registry Editor to locate the following key :
    HKEY_CLASSES_ROOT\CLSID\{c8b522cb-5cf3-11ce-ade5-00aa0044773d}\
  - ▶ Change the key valueOLEDB_SERVICES to 0xfffffffc

- **Turn on the ODBC Connection Pooling**
  - ▶ Use Connection Pooling tab on the ODBC Administration dialog
  - ▶ Set the timout value (for example to 600 seconds)

- **Make sure the Isolation Level is set to a level higher than *NONE**

- **The default AUTOCOMMIT setting (ON) MUST not be changed**
  - ▶ Client Access ODBC internally sets this parameter to OFF

Sample Visual Basic Component Configuration

- **The project needs to reference the Microsoft Transaction Server Type Library (COM+ Services Type Library on Windows 2000)**

- **Typically the ActiveX Data Object (ADO) interface is used to access database**

  - ▶ Microsoft's OLEDB-ODBC bridge (MSDASQL) is used to map OLEDB calls to ODBC APIs

```
strConnect = "Provider=MSDASQL;DSN=" + _
DataSource.Text + ";UID=" + UserID.Text + _
";PWD=" + Password.Text
With adoConn
        .CursorLocation = adUseClient
        .ConnectionString = strConnect
        .Open
End With
```

Sample Visual Basic Component Configuration cont.

- **Component's transaction support can be set directly in the Visual Basic project**
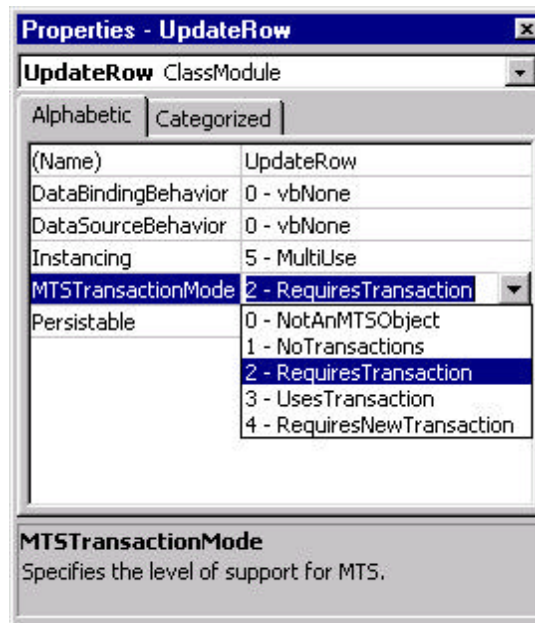
- **Transaction property can also be set through Transaction Server Explorer**

  ▸ Component needs to be installed under MTS

Creating and Installing of the MTS component

- **The component is compiled into DLL**
  - ▸ Once the DLL is built, set the binary compatibility
  - ▸ Make sure that the threading model is apartment threaded

- **The Transaction Server Explorer (Component Services on Windows 2000) is used to install the component under MTS**
  - ▸ Create a new package
  - ▸ Import the DLL functions into the newly created package
  - ▸ Start MS DTC service, if not already started

Troubleshooting Tips

- **The most common error is the enlist phase failure**

  [IBM][Client Access Express ODBC Driver (32-bit)]Enlist with DTC phase failed. x

  - ► Make sure that the CWBODBC.DLL (Client Access ODBC driver) is in the global PATH on the Windows system where MTS is running

- **The detail ODBC trace**

  - ► May help pin down the problem area

  - ► Shows the transaction flow between MTS and Client Access ODBC driver

  - ► Usually required by IBM service

Taking the detail ODBC trace

- **The trace needs to be taken in the same context under which MTS component and MS DTC are running**
  - ► Use the Transaction Server Explorer to verify that the Identity property for the component is set to Interactive user
  - ► Configure the MSDTC service's Log On property so that a profile with Administrator authority is used
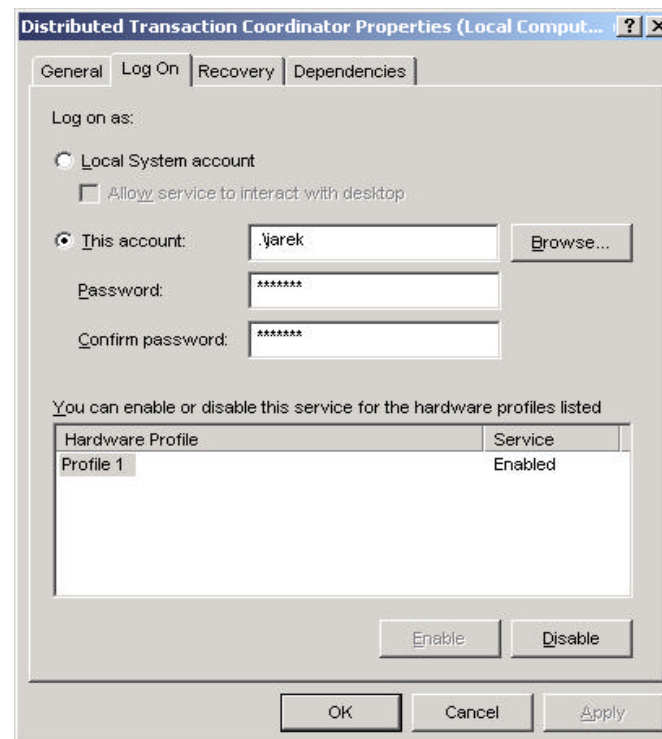
Taking the detail ODBC trace cont.

■ **Enabling the detail ODBC trace**

➤ Start Diagnostic Tools utility from the Client Access —>Service menu

➤ Once the Diagnostic Tools were started find the tool's icon in the right-hand side of the launch pad

  ‒ Looks like a computer with a red dot

➤ Select Detail Trace —>Start from the Diagnostic Tools' context menu

■ **After you recreated the problem, the trace should contain MTS transaction flow entries**

Summary

- **Adaptive e-Transaction Services**

  - ➤ Fully and efficiently supports open standards (X/Open DTA, Java JTA)

  - ➤ Decompiles transactions from jobs and activation to the multiplexed on a single JDBC/CLI connection

  - ➤ Enables multiple database connections to work on the same transaction

- **Support of the Microsoft Transaction Server**

  - ➤ Allows DB2 UDB for iSierse to be used as a resource manager for MTS application components

Additional Material

- **DB2 UDB for iSeries home page**

  ➤ **ibm.com**/eserver/iseries/db2

- **'MTS Considerations for DB2 UDB for iSeries' white paper**

  ➤ **ibm.com**/eserver/iseries/developer/db2/porting.html

- **DB2 UDB for iSeries Publications**

  ➤ Online Manuals: **ibm.com**/eserver/iseries/db2/books.htm

  ➤ DB2 UDB for AS/400 Redbooks (**ibm.com**/redbooks)
    - DB2/400 Advanced Database Functions (SG24-4249-02)

Trademarks

IBM eServer, iSeries, OS/400, DB2, and DB2 UDB for iSeries are trademarks or registered trademarks of International Business Machines Corporation.

Microsoft, Windows, ActiveX, and the Windows logo are trademarks or registered trademarks of Microsoft Corporation.

IBM makes no commitment to make available any products referred to herein.

All other registered trademarks and trademarks are properties of their respective owners.

References in this publication to IBM products or services do not imply that IBM intends to make them available in every country in which IBM operates.