

Strategic Options for IP Modernization

Noel Javier C. Sales
Software Engineer zCICS
Development



IBM CICS® User Conference 2009

Agenda

- **Integration requirements, technologies and patterns**
- **Strategic integration technologies**
- **Typical customer deployments**
- **Conclusion and comparison tables**

- **Abstract**
CICS has expanded and optimised the TCP/IP protocols supported over many releases, providing robust, scalable, and high performance connectivity. This session details new features in CICS TS V4.1 and best practices for implementing solutions that use TCP/IP as the backbone for critical business applications including IP Interconnectivity, JCA, Web Services, CICS sockets, WebSphere MQ.

- **Additional material**
 - ▶ Phil Wakelin - CICS Transaction Gateway Technical Planner
 - ▶ Dermot Flaherty - Lead Architect, Messaging Products and Technologies

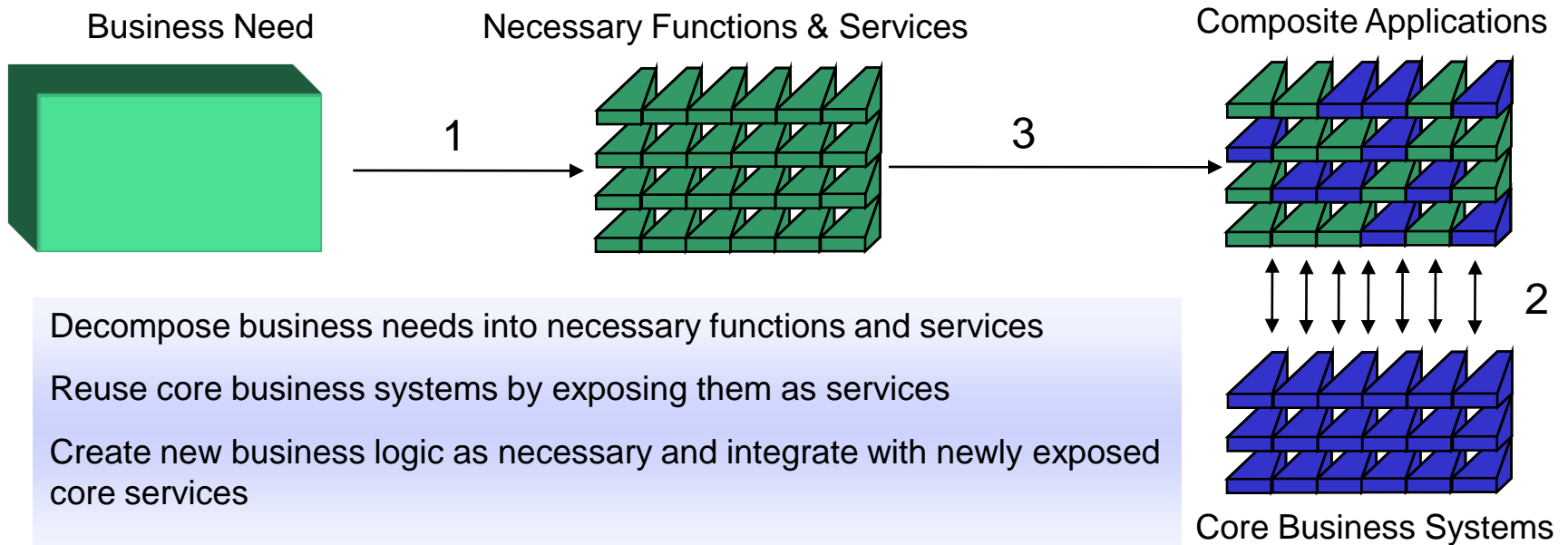
Customers integration requirements

- **Business considerations for your organization**
 - ▶ Industry and company standards and reference frameworks
 - ▶ Preferred application development environment and tools
 - ▶ Time, cost and service level agreements
 - ▶ Availability of skills
- **Solution technical requirements and constraints**
 - ▶ Security and transactionality
 - ▶ Boundaries / where to expose services
 - ▶ Service granularity, Tight vs loose coupling
 - ▶ Performance, reliability, availability and scalability
 - ▶ Synchronous or asynchronous invocation
 - ▶ Inbound and outbound capability
 - ▶ Data sizes and conversion
 - ▶ State management
- **Application reuse - today are typically delivered across multiple channels**

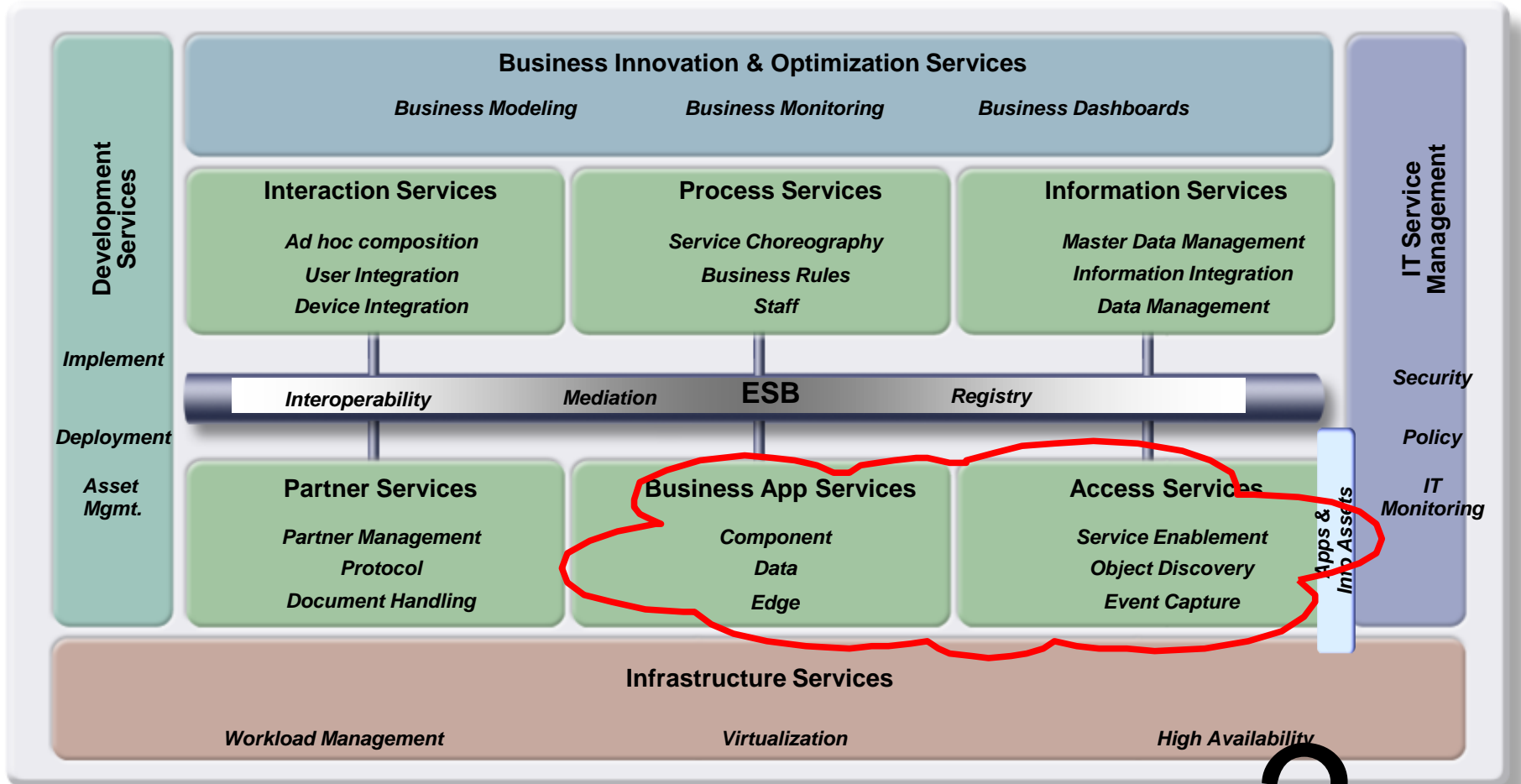
Technologies & patterns driving integration between SOA services and CICS

- **Service Oriented Architecture (SOA)**
 - ▶ A flexible approach to building multi-tiered systems that delivers application functionality as services to end-user applications or to other services
- **Business Events**
 - ▶ Business decisions emitted to event-driven processes and business dashboards
- **Interaction services**
 - ▶ Modern user application, typically simple Web, portals, Wikis, REST, mashups, devices, ...
- **Process services**
 - ▶ Modelling, running, monitoring, analyzing and adapting a business process on a continuous base. A business process is preferably executed using workflow concepts and technology.
- **Enterprise Service Bus (ESB)**
 - ▶ Intermediaries between service providers and requester providing eg. network location independence, mediation between service interfaces + qualities of service, logging ...
- **Infrastructure services**
 - ▶ Industry standardising on TCP/IP as the backbone for physical connectivity and network layering
 - ▶ Qualities of service, such as workload management, security, and high availability provided by combination of devices (eg. Routers) + specialist devices (eg. Sysplex Distributor) and higher level protocols built on standards (eg. SOAP)

Why expose CICS applications as services in an SOA?



Typical SOA architecture, services and CICS



CICS is designed for mixed integration styles and interaction patterns

Over many years CICS has made available a large choice of protocols, interfaces and APIs to connect to and from CICS applications

Between TXSeries and CICS

Offers high performance and QoS, DPL & TR

Web services requester & provider

Offers high Qualities of service, portability of service, standard and comprehensive interfaces, tooling, registries, extensibility, and simplicity

JCA via CICS Transaction Gateway

Offers high performance and Quality of Service, standards, good tooling

Messaging via WebSphere MQ

Offers assured delivery, loose coupling, highly performant asynchronous, efficiently handles very large data

HTTP / Atom / RESTful interfaces

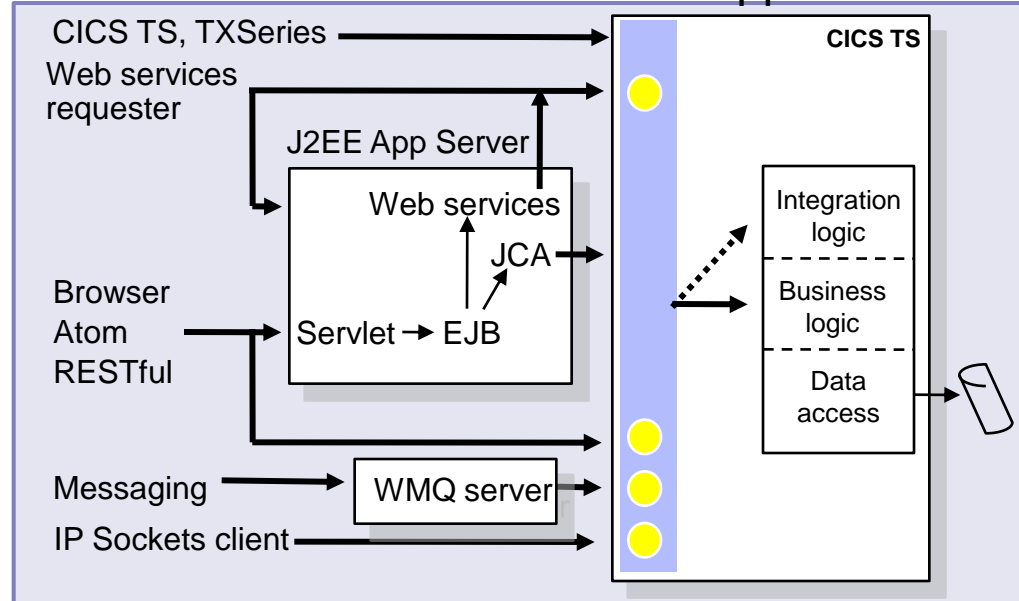
Offers simple & standard client interfaces, good-enough qualities of service, some self-discovery

IP Sockets via CS CICS Sockets feature

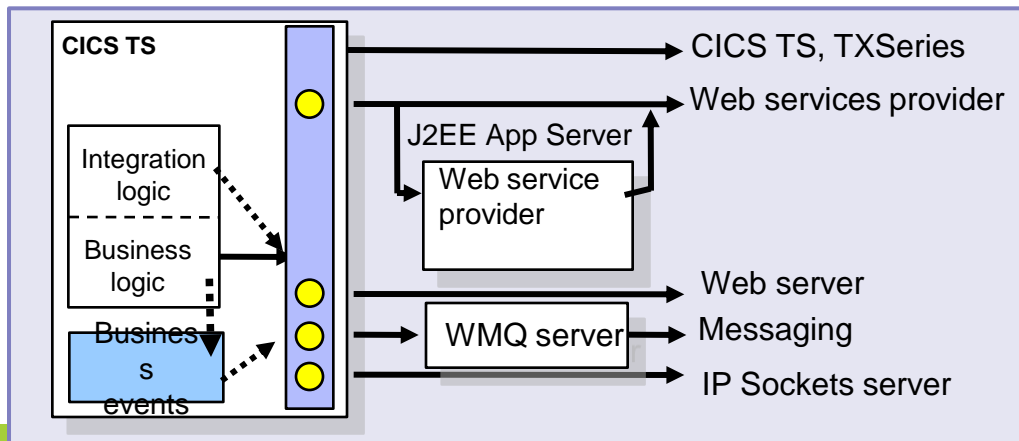
Offers complete customised protocol handling

NOTE: IIOP, ONC RPC, DFHWBCLI, CWI COMMAREA and the Web server plugin have been stabilised

Inbound from IP clients to CICS applications



Outbound from CICS to IP servers



CICS IP connectivity options

- **Open standard architectures provide comprehensive development tools and runtime support**
 - ▶ Web services
 - ▶ JCA part of Java Enterprise Edition Connector Architecture

- **Open standard transports are suitable for use by applications that require greater control of the protocol and do not need the development tools or qualities of service provided by the standard architectures. These applications will assume more responsibility for systems management, state management, security, and recovery**
 - ▶ HTTP (HyperText Transfer Protocol)
 - ▶ Atom
 - ▶ RESTful

- **Non open standard IP transports**
 - ▶ WebSphere MQ supports several IP protocols incl. Web services, RESTful, and integrates well with IBM WebSphere business event products
 - ▶ CICS IP interconnectivity (IPIC)
 - ▶ IP Sockets

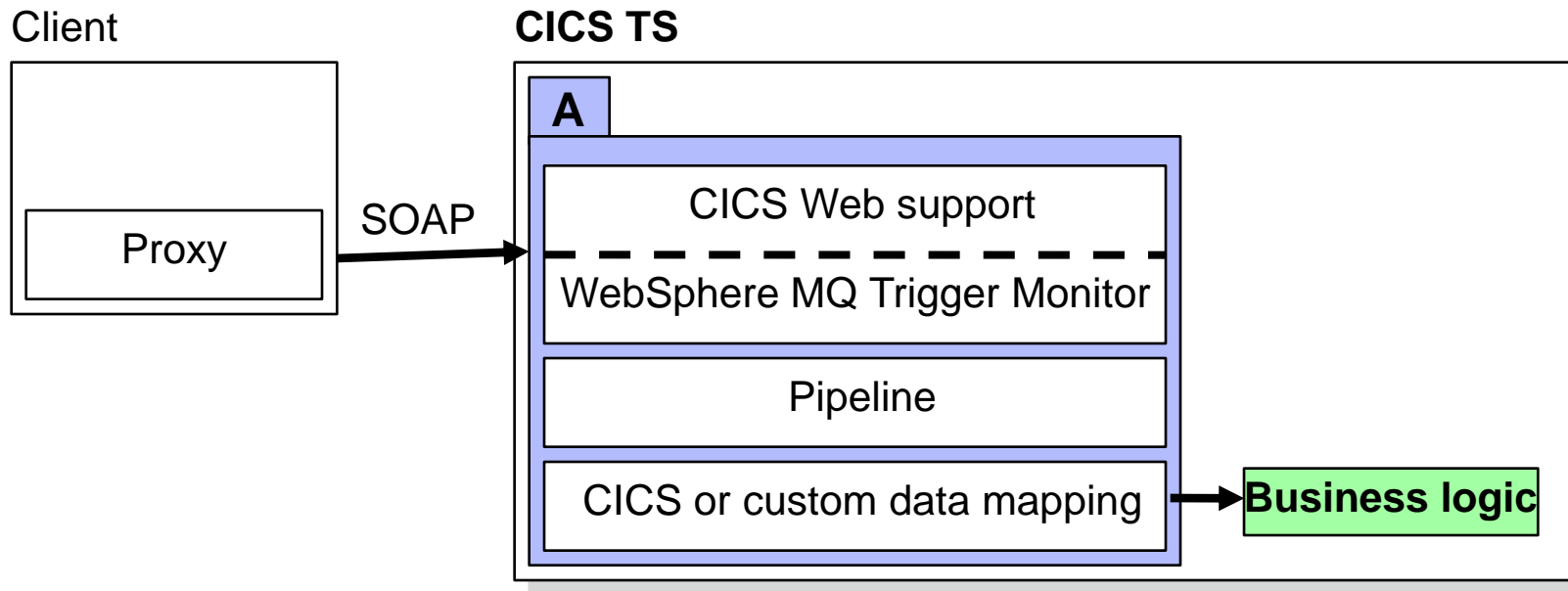
What CICS applications can be exposed as a service?

- CICS application design – maintain separation of concerns
 - ▶ Interfaces between presentation, integration, business, data access
 - ▶ Provides framework for easier reuse, ownership, optimisation, maintenance, operations,...

- CICS programs with COMMAREA or CHANNEL interfaces
 - ▶ Request and response, similar to remote procedure call
 - ▶ Typically stateless
 - ▶ Typically CICS manages transactional scope + security
 - ▶ COMMAREA typically optimised binary as limited to 32KB size
 - ▶ CHANNEL more flexible - eg. mixture of binary, text, XML - as not limited in size

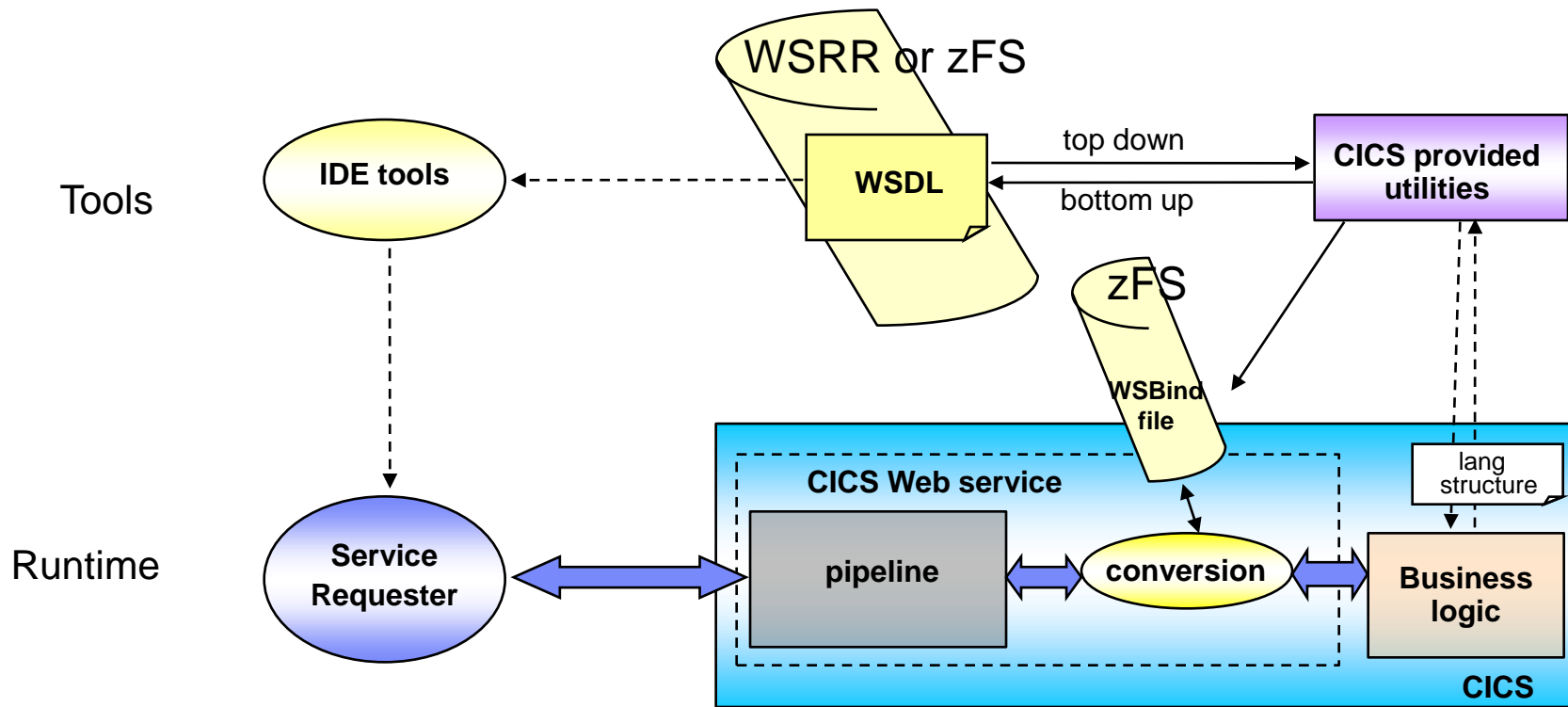
- CICS programs with a terminal-oriented 3270 interface
 - ▶ Single interaction or pseudo-conversational
 - ▶ Basic Mapping Support (BMS) used to abstract static + dynamic data
 - ▶ Transform terminal-oriented application into CHANNEL or COMMAREA
 - CICS Service Flow runtime + Rational Developer for System z Service Flow Modeler

Web services support



CICS provides direct access from Web service requesters and to Web service providers
 WSDL (Web Services Definition Language) is used to describe the service
 Tools are typically used to generate the client proxy and message adapter

Web services tooling and runtime



Web services support in CICS TS V3 capabilities

▪ Capabilities

- ▶ Synchronous when over HTTP
- ▶ Asynchronous when over WebSphere MQ
- ▶ WS-Addressing in CICS TS V4.1
- ▶ Inbound and outbound
- ▶ Flexible configuration using enhanced System Management
- ▶ Multiple protocol configurations per server
- ▶ Ease of use for header handlers

▪ Security to System z

- ▶ Web services security (WS-Security)
- ▶ Transport level security
- ▶ Basic Authentication and SSL when over HTTP
- ▶ User ID and password when over WebSphere MQ

▪ Transactional scope

- ▶ Web services distributed transactions (WS-AtomicTransaction)
- ▶ CICS local transaction

Interface

Language structure (COBOL, C, C++, PL/I) in a COMMAREA
or XML body provided in CONTAINER

Coupling

Loose coupling and interoperability are inherent in a service-oriented architecture (SOA) and make it a natural choice for many enterprise applications

Tools

CICS Web services assistant
Rational Developer for System z
WebSphere Service Registry and Repository (WSRR)

Messaging via WebSphere MQ backbone

Extends the reach of your Enterprise to deliver Business data with a range of QoS for processing by CICS

Support for virtually every commercial IT System

80+ platform configurations

AIX

Linux x86, x86-64

Linux POWER

Linux for System z

Windows x86, x64

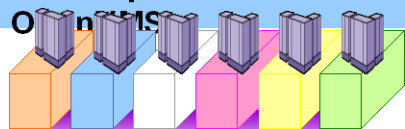
Solaris x86 / SPARC

iSeries i5/OS

HP-UX Itanium/PA-RISC

HP NonStopServer

HP OpenView



80+ platforms

Exploit System z for messaging powerhouse

Unique code-base designed natively for tight z/OS integration

Runs as formal MVS sub-system

Leverage Parallel Sysplex to provide Shared Queues for continuous availability, scalability and capacity

Support for event-driven publish-subscribe delivery



z/OS

Enable popular applications and environments

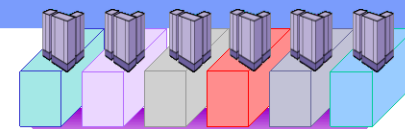
Built-in bridge for CICS, IMS, Batch/TSO

JCA support to provide JMS services for JEE App Servers

Support for Databases DB2, Oracle etc

Support for External Transaction Coordinators

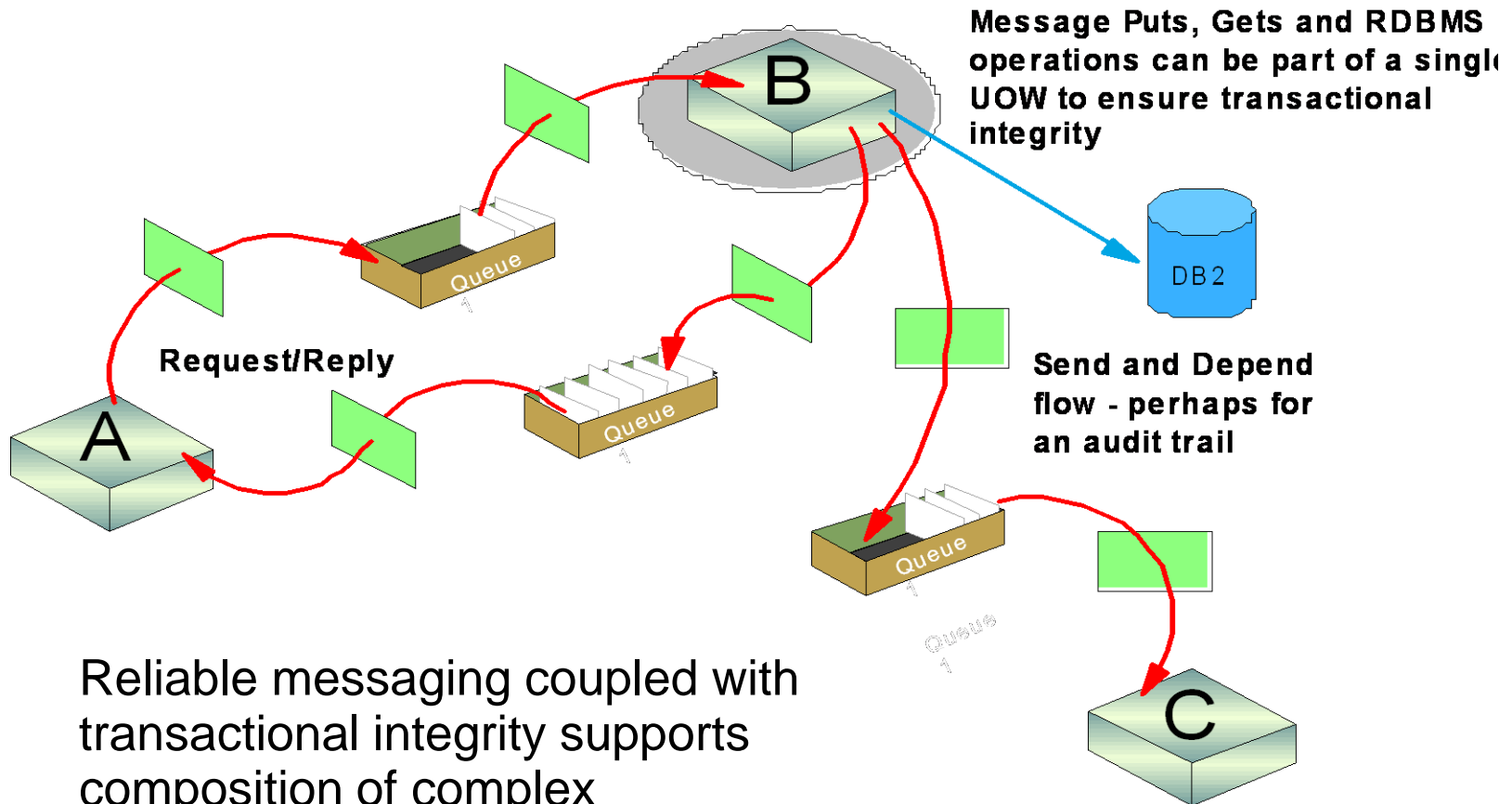
Over 950 ISVs with enabled Packaged Applications



Applications

UNIVERSAL MESSAGING BACKBONE

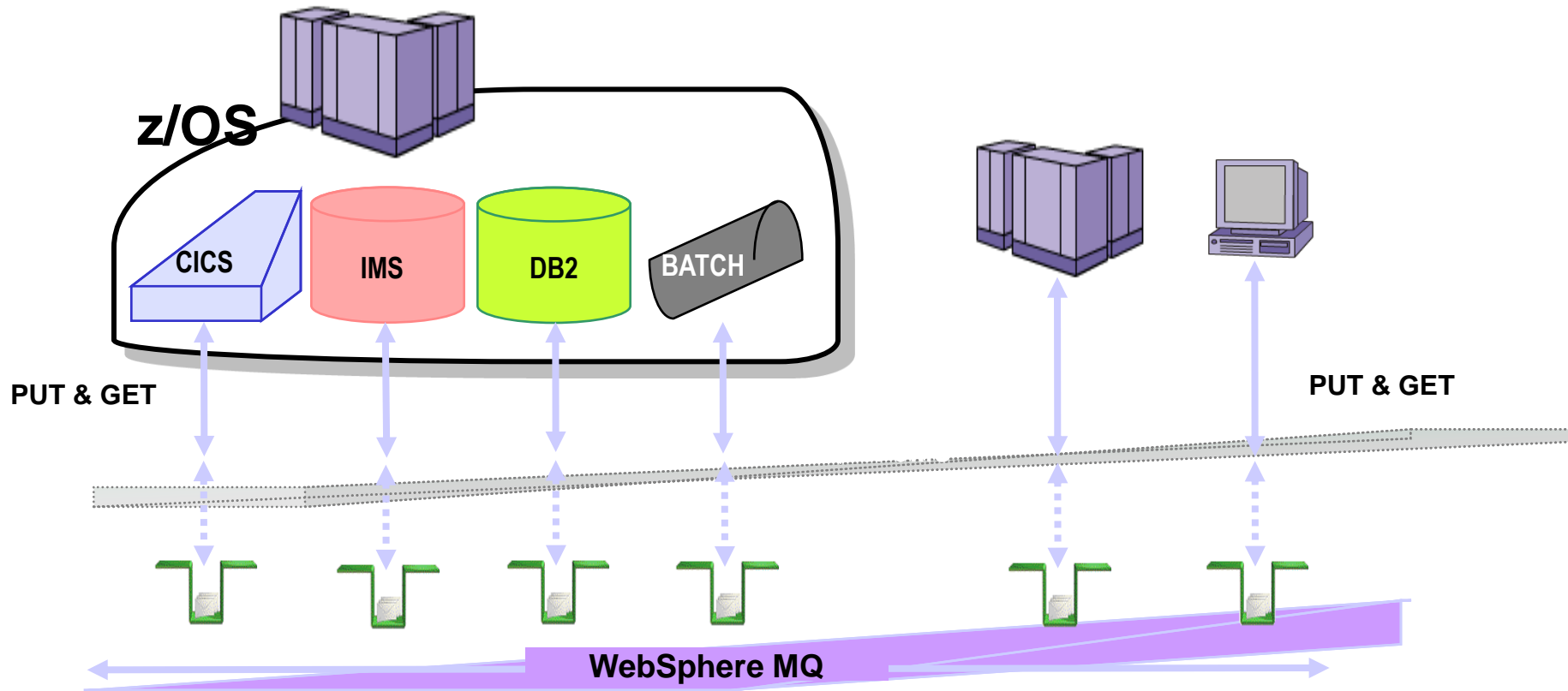
WebSphere MQ in a nutshell



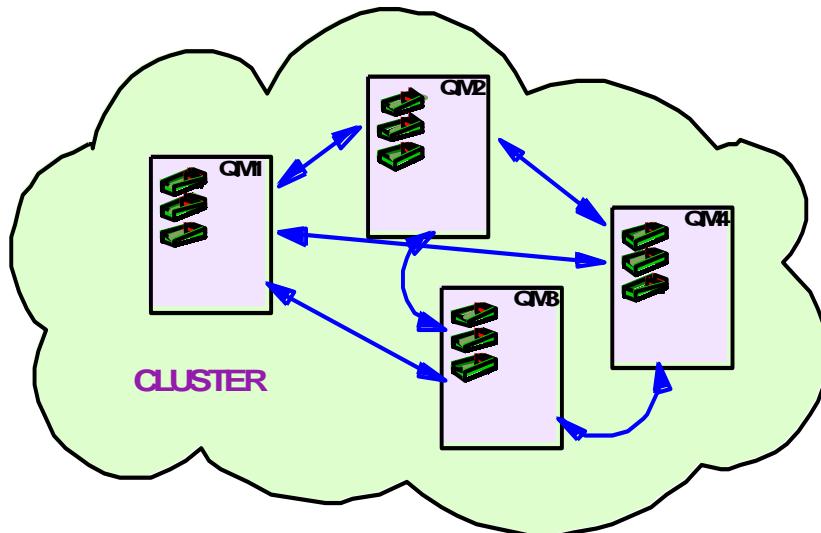
Reliable messaging coupled with transactional integrity supports composition of complex applications to fully support parallel processing.

WebSphere MQ access to CICS

Can access business data from CICS from ANYWHERE – either on z/OS or elsewhere in the Enterprise either explicitly via the MQI from CICS or implicitly via the MQ-CICS Bridge



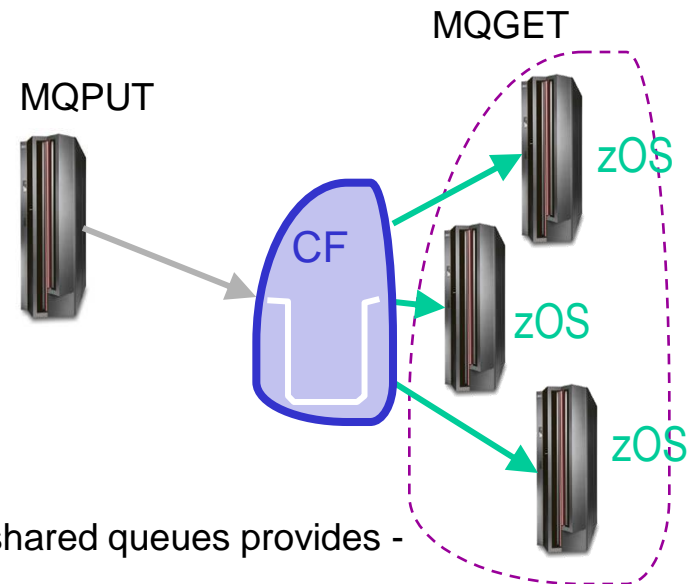
WMQ Clustering and shared queues



MQ clustering provides -

Automatic advertising of queues within a cluster to reduce MQ administration

Workload balancing by spreading messages destined for a single logical queue over multiple queues of the same name on different Queue Managers



MQ shared queues provides -

Queues not owned by Qmgrs
Automatic load-balancing
Scalable throughput
Sysplex-wide access to shared message data
VIPA support

Queue Sharing Group

Failure isolation
Peer Recovery for failing Queue Managers
24x7 Availability

WebSphere MQ and wider Web

Capabilities

Asynchronous

Assured delivery

Inbound and outbound

Flexible configuration and handles very large message sizes

Security to z/OS

Q and message level security, SSL

Transaction scope

Unit-of-work includes GET and PUT

Interface

WMQ DPL bridge provide non-evasive call/return to COMMAREA applications

WMQ triggering and API provides all WMQ facilities, eg. pub/sub

Web services (SOAP/JMS) transport capability

Uses JMS message formats (RFH2 etc.)

Options to control other WMQ details such as queue manager, persistence etc

Coupling

Medium when using binary/text/XML

Low when using Web services

RESTful HTTP-MQ Bridge support

Supports zero-footprint client requirements

Provides simple mapping of MQ to HTTP verbs

Supportpac MA0Y (delivered with WMQ V7) provides servlet for J2EE container

Supportpac MA94 provides native MQ listener on Distributed platforms

MQ Service Definition allows standalone MQ Applications to be described as Services

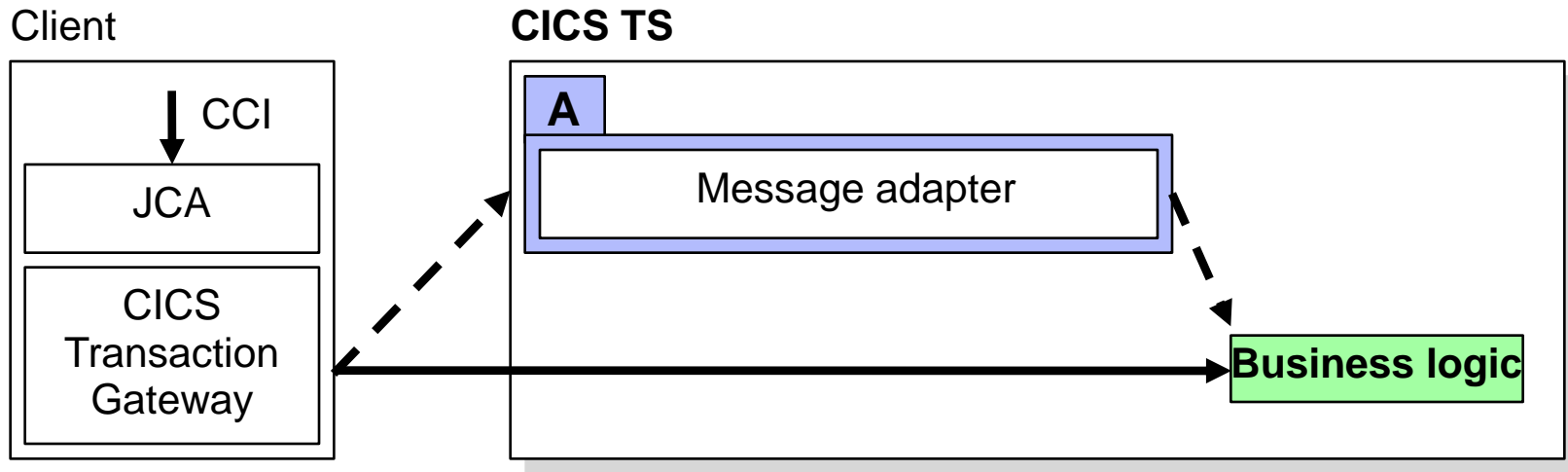
To be inventoried and catalogued in Service Registry

To be re-used as Services in composite SOA applications

To be managed and traced with SOA tools

Service definition and SOAP binding available in supportpac MA93

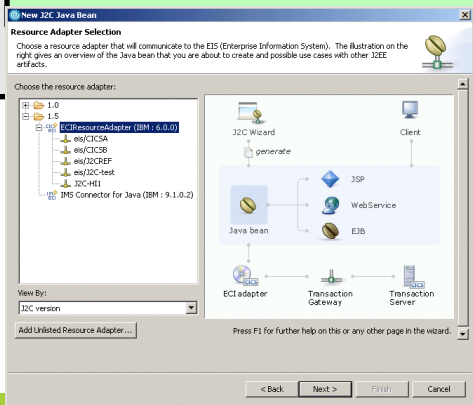
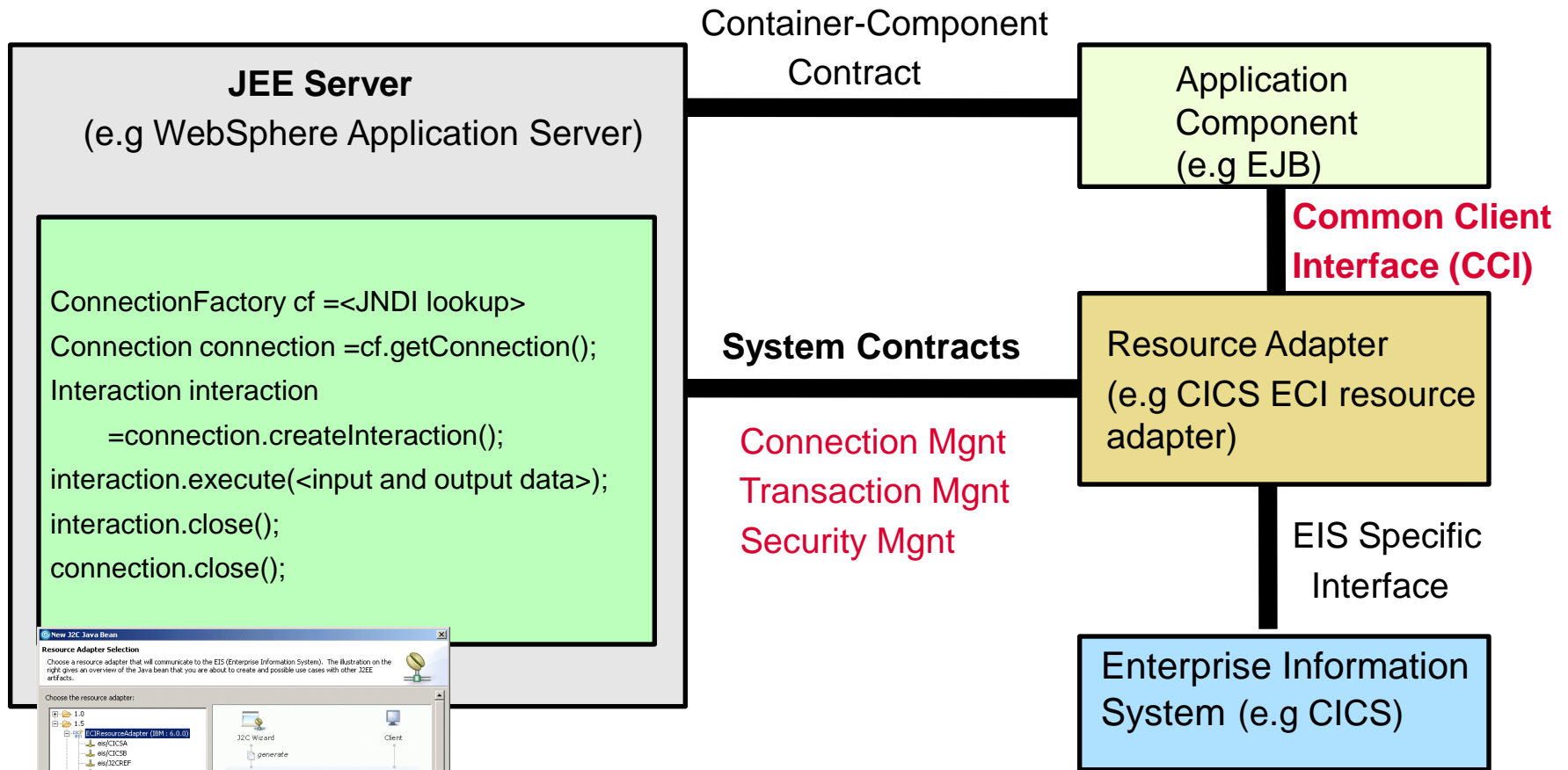
JCA via CICS Transaction Gateway



The J2EE Connector Architecture (JCA) defines the common client interface (CCI) for the client to use to drive interactions with enterprise information systems such as CICS

The CICS Transaction Gateway provides the JCA access to CICS

JEE Connector Architecture (JCA)



Connector code can be hand-written or built using the 'J2C bean' generation capability of Rational Application Developer

CICS Transaction Gateway

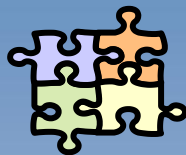


*Primary
connector into
CICS*

Interconnectivity

High performing and scalable
inbound connector to CICS
applications

Provides connectors to
COMMAREA, container and 3270-
based CICS applications



*Java and
non-Java
API's*

Interfaces

Standard JCA interface is
strategic and provides best
Qualities of Service

Base Java, C, C++, COBOL
and COM interfaces are
supported but stabilized

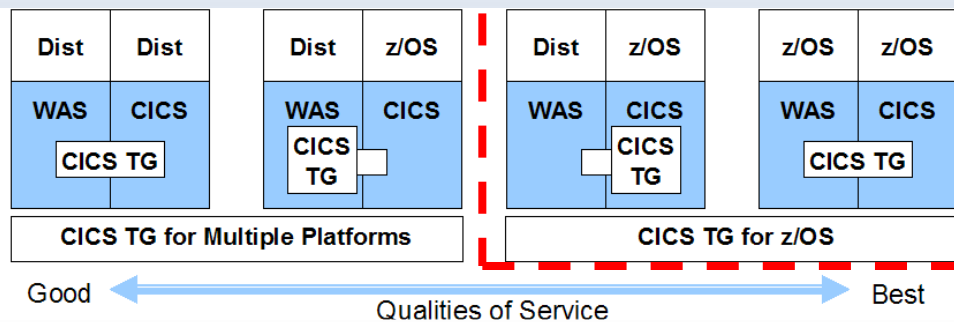


*WebSphere,
CICS
and others*

Integration

Every in support CICS server
on every platform to
WebSphere SOA foundation
servers

5 SNA servers (AIX, Windows,
Linux)

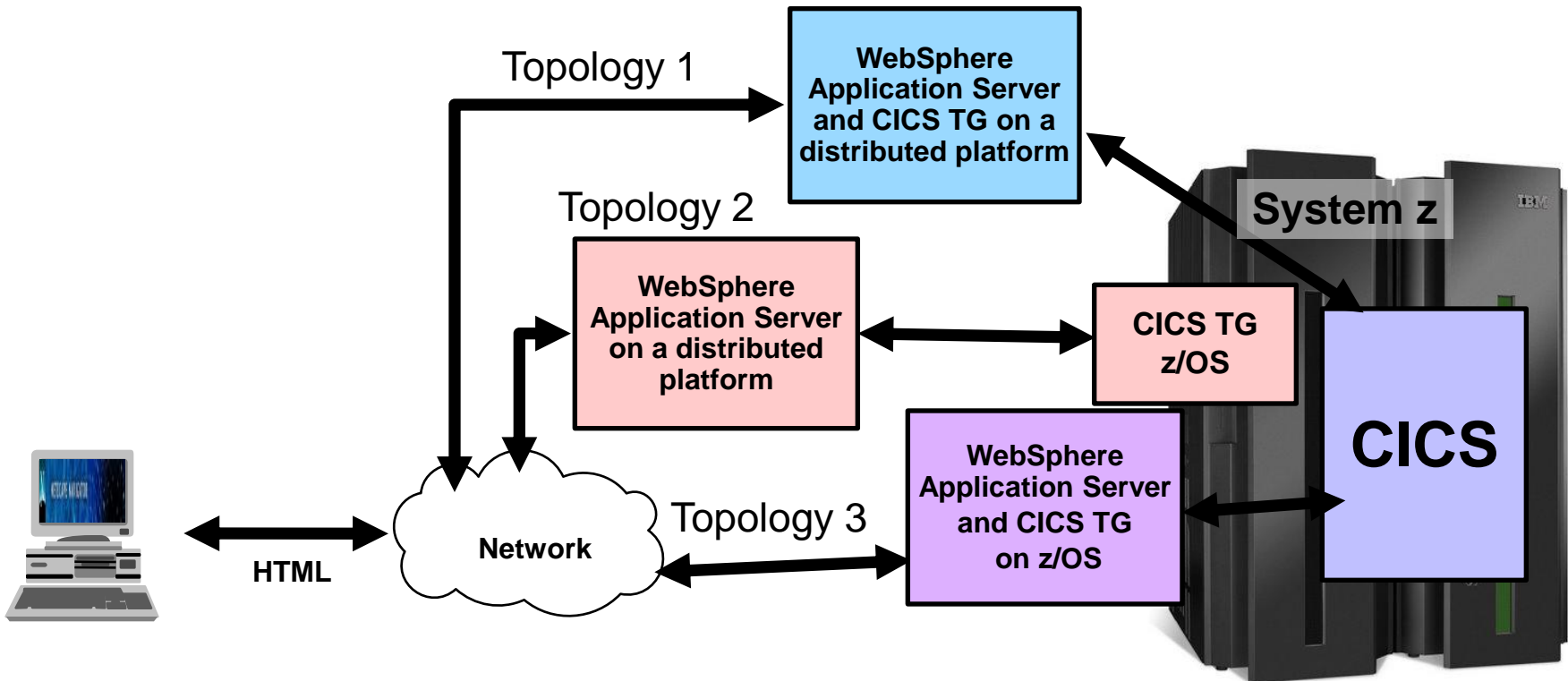


Supported Platforms

IBM's flagship z/OS

Linux on Intel, POWER, & System z
AIX, HP-UX and Solaris UNIX support
Windows: 2003, XP, Vista

CICS TG platforms and topologies

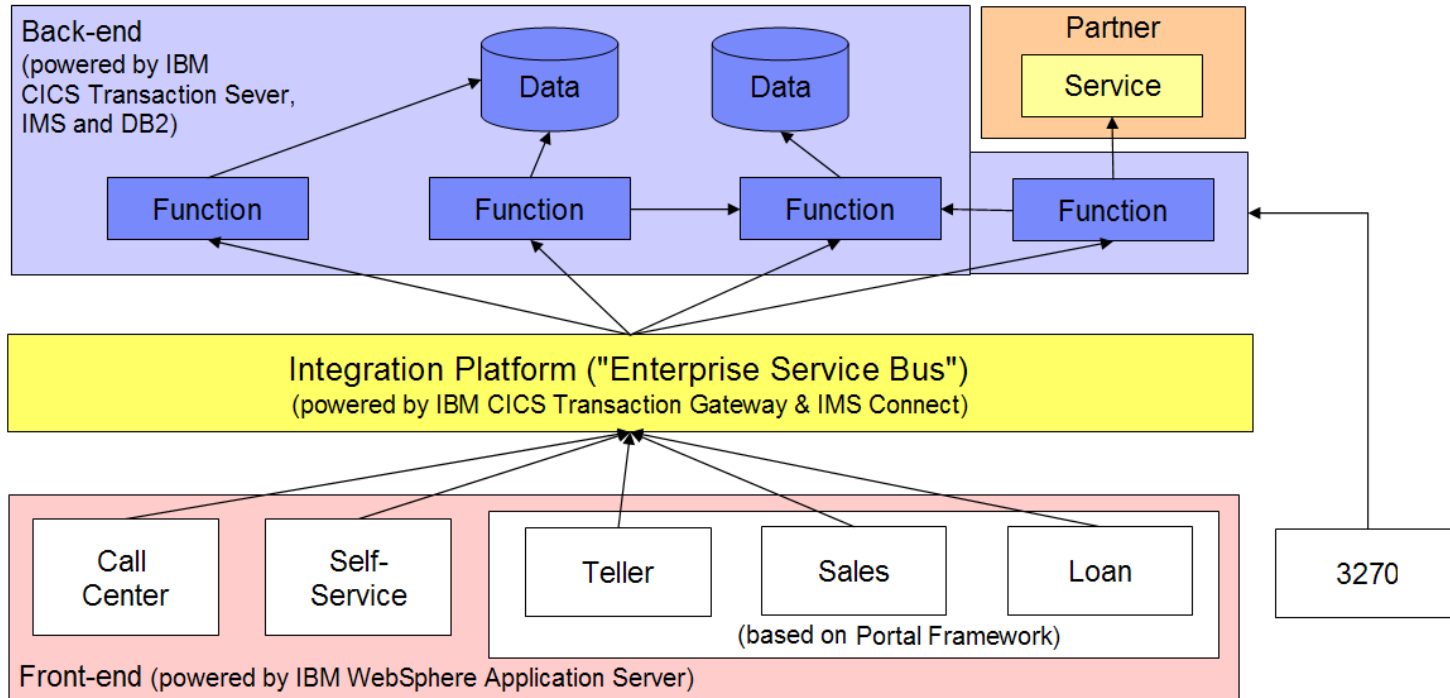


CICS TG capabilities

- **Capabilities**
 - ▶ Inbound only
 - ▶ Synchronous and asynchronous (within a client request!)
- **Security to z/OS**
 - ▶ SSL to the Gateway daemon or CICS TS.
 - ▶ User ID and password
 - ▶ Asserted identity options for z/OS
- **Transactional scope**
 - ▶ Local transactions
 - ▶ XA Global transactions
- **Interface**
 - ▶ Language structure (COBOL, C, C++, PL/I) in a COMMAREA or CHANNEL
- **Coupling**
 - ▶ Medium – typically the COBOL formatted records are exposed to the client
 - ▶ RAD J2C tools can abstract the COMMAREA interface by generating easier to use proxies

A large European Retail Banking Solution

Solution Approach - A Service Oriented Architecture



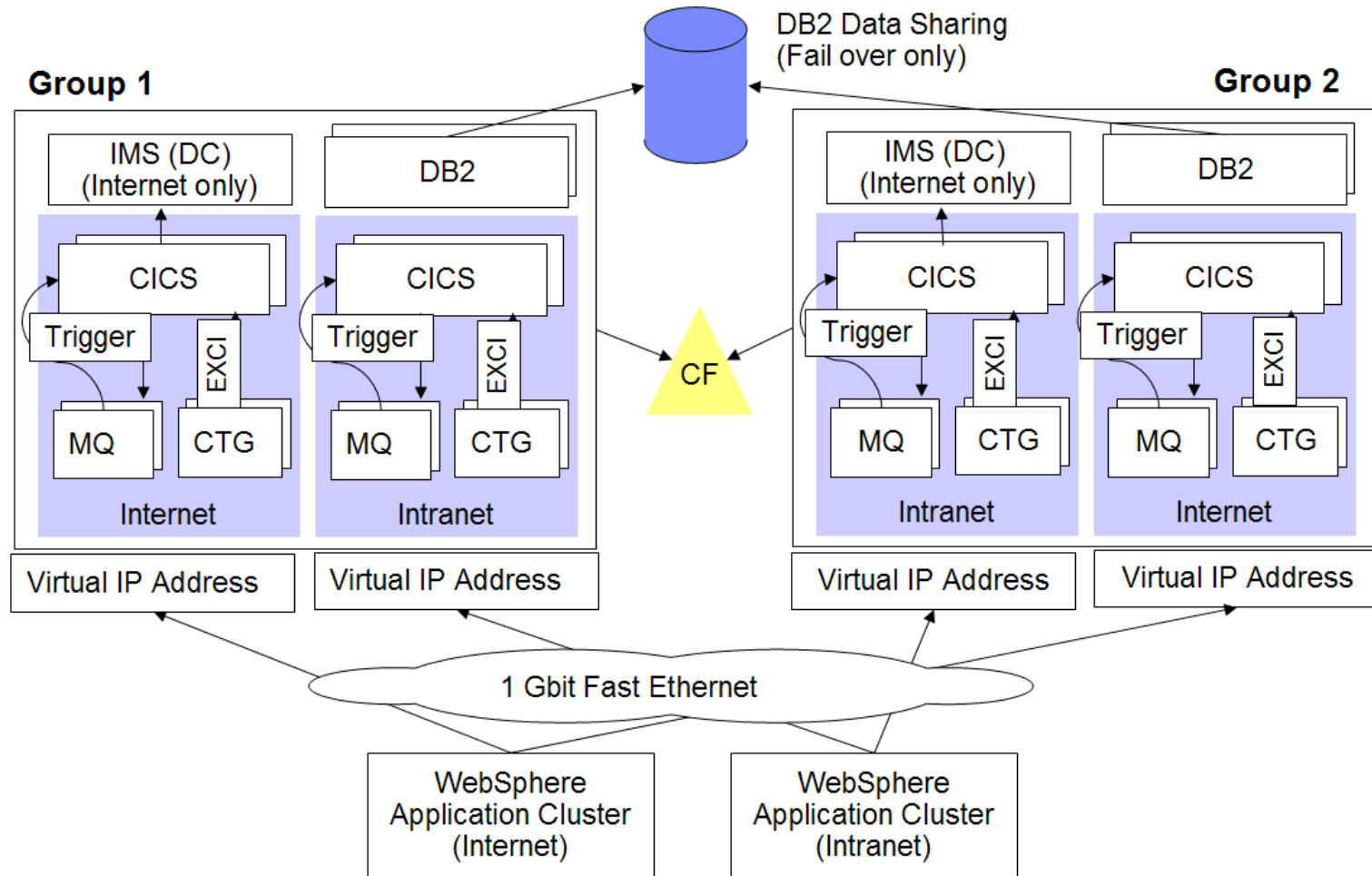
CICS Transaction Server
 Transactions/year=11Bn
 Peak Trans/day=74M

IBM CICS Transaction Gateway
 Transaction/year = 1.7Bn.
 Peak Trans/day = 12.5M
 Peak trans/sec >500

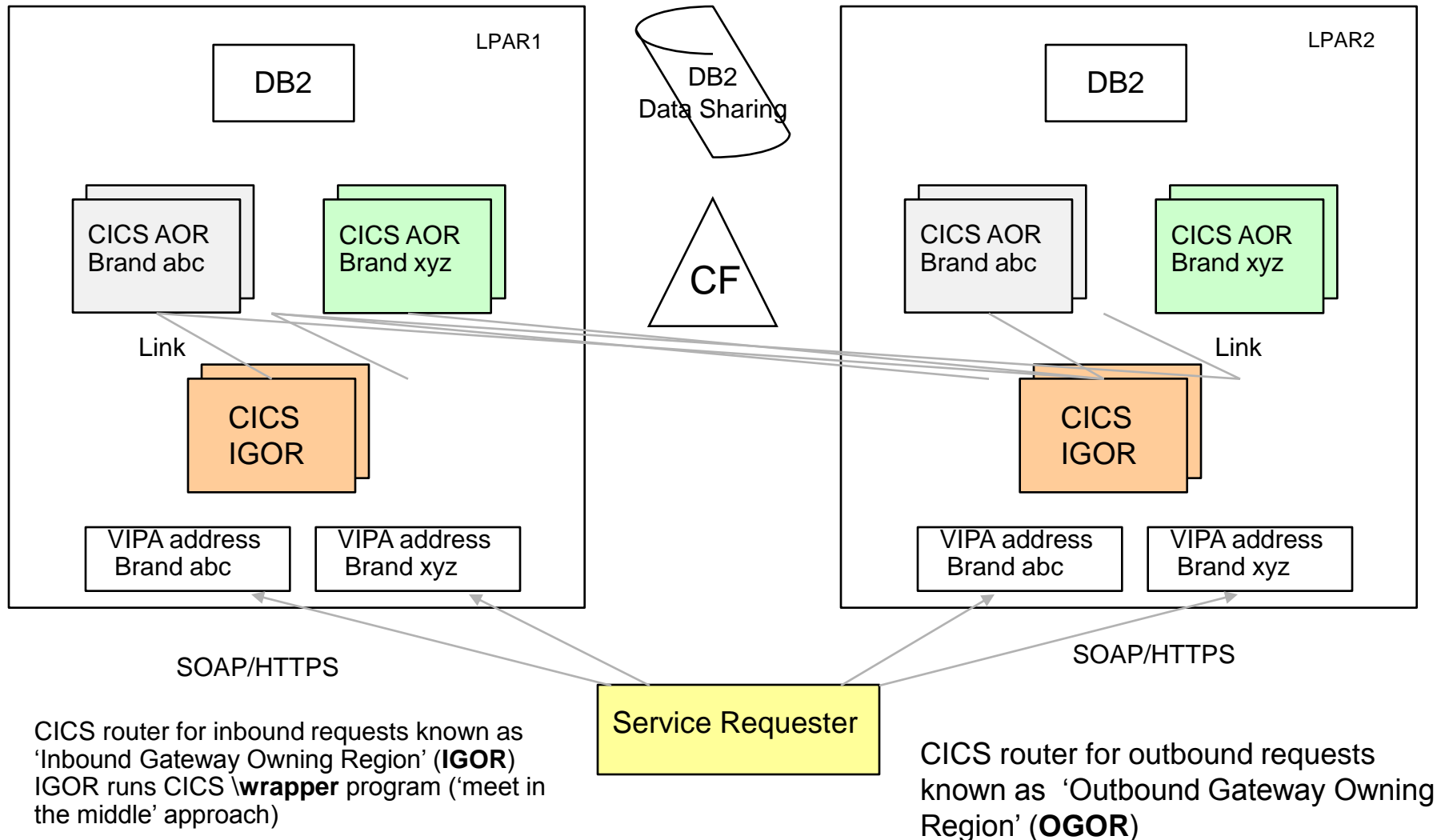
IBM WebSphere Application Server
 Web Applications
 in Production = 90
 Capacity in CPU's = 600

A large European Retail Banking Solution

An SOA that fully exploits the QoS of System z



Typical customer deployment: QoS customer benchmark of an SOA solution with Web services



Summary of strategies for IP modernisation

- CICS applications can be full participants in an SOA environment
- To achieve the best reuse of your CICS programs and to support multiple access options, architect clear and concise business logic interfaces of the right granularity for the solution
- CICS and other tools help transform and aggregate your programs to provide the right level of granularity for your service requesters
- CICS provides a broad choice of access options based on TCP/IP, open standard connectivity architectures, and transport mechanisms
- In an SOA solutions these typically are
 - ▶ **Web services, JCA via CICS TG, Messaging via WebSphere MQ**
- In broader solutions these typically are
 - ▶ **HTTP, Atom, RESTful interfaces, IP Sockets via CS CICS Sockets Feature**
- Business considerations and solution technical requirements should be compared with the capabilities of each access option to select the most appropriate
- Architects, CICS application developers, and CICS system programmers will find it easy take advantage of new access options and make existing applications available for reuse

Comparison and recommendation

| Architecture/transport | Capabilities | Security to zSeries | Transactional scope | Interface | Coupling |
|-------------------------------|--|--|-----------------------------------|------------------------------|--|
| Web services | Inbound and outbound Synchronous (HTTP) Asynchronous (WMQ) | Web services SSL User ID + password | Web service Sync on return | COMMAREA CONTAINER | Low |
| JCA via CICS TG | Inbound only Synchronous Asynchronous | SSL User ID + password Thread identity | Local Global Sync on return | COMMAREA CONTAINER | Medium |
| Messaging via WebSphere MQ | Inbound and outbound Asynchronous Assured delivery | SSL User ID + password | Sync on return | COMMAREA WebSphere MQ API | Low when used with ESB or WS. Medium if WMQ only. |

| Architecture/transport | Description | Positioning | Recommendation |
|-------------------------------|--|---|--|
| 1. Web services | Comprehensive W3C standards for messaging over the Web supporting SoA to and from CICS | Industry-wide open standard integration technology that includes CICS connectivity. Improving QoS, features and performance | <i>Establish plans to transform CICS apps so they can participate in a SOA pattern with Web services</i> |
| JCA via CICS TG | Lightweight J2EE standard for calling CICS and other EIS's | Widely adopted precision CICS connectivity with highest qualities of service today | <i>Continue to exploit JCA and CICS TG and use within an SOA and ESB</i> |
| Messaging via WebSphere MQ | Comprehensive industry standard for assured messaging | Widely adopted B2B integration technology that includes CICS connectivity | <i>Continue to exploit WebSphere MQ for optimised messaging and flowing Web services</i> |

References

- **White papers**

www.ibm.com/software/htp/cics/tserver/v32/library/index.html#wpapers

- ▶ Options for integrating CICS applications in an SOA
- ▶ Deploying CICS Web services to preserve IT investments in the banking industry
- ▶ Increase the value of CICS applications with WebSphere MQ
- ▶ Integrating WebSphere Application Server and CICS using the CICS Transaction Gateway

- **IBM Redbooks**

www.redbooks.ibm.com/cgi-bin/searchsite.cgi?query=cics

- ▶ Architecting Access to CICS within an SOA (SG24-5466-05)
- ▶ Implementing CICS Web Services (SG24-7206-02)
- ▶ CICS Transaction Gateway for z/OS (SG24-7161-00)

- **Product information**

- ▶ IBM CICS - www.ibm.com/software/htp/cics
- ▶ IBM CICS Transaction Gateway - www.ibm.com/software/htp/cics/ctg/
- ▶ IBM WebSphere MQ - www.ibm.com/software/integration/wmq

© IBM Corporation 2009. All Rights Reserved.

The workshops, sessions and materials have been prepared by IBM or the session speakers and reflect their own views. They are provided for informational purposes only, and are neither intended to, nor shall have the effect of being, legal or other guidance or advice to any participant. While efforts were made to verify the completeness and accuracy of the information contained in this presentation, it is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this presentation or any other materials. Nothing contained in this presentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in this presentation may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries: ibm.com/legal/copytrade.shtml AIX, CICS, CICSplex, DataPower, DB2, DB2 Universal Database, i5/OS, IBM, the IBM logo, IMS/ESA, Power Systems, Lotus, OMEGAMON, OS/390, Parallel Sysplex, pureXML, Rational, Redbooks, Sametime, SMART SOA, System z, Tivoli, WebSphere, and z/OS.

A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office

Intel and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.