Reuse the z/VSE applications and focus on the integration for extended business needs

Wilhelm Mild
IT Architect
mildw@de.ibm.com

Trademarks

Trademarks

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries. For a complete list of IBM Trademarks, see www.ibm.com/legal/copytrade.shtml: AS/400, DBE, e-business logo, ESCO, eServer, FICON, IBM, IBM Logo, iSeries, MVS, OS/390, pSeries, RS/6000, S/30, VM/ESA, VSE/ESA, z/VSE, Websphere, xSeries, z/OS, zSeries, z/VM, System z

The following are trademarks or registered trademarks of other companies

Lotus, Notes, and Domino are trademarks or registered trademarks of Lotus Development Corporation
Java and all Java-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries
LINUX is a registered trademark of Linux Torvalds
UNIX is a registered trademark of The Open Group in the United States and other countries.
Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation.
SET and Secure Electronic Transaction are trademarks owned by SET Secure Electronic Transaction LLC.
Intel is a registered trademark of Intel Corporation
* All other products may be trademarks or registered trademarks of their respective companies.

NOTES:

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon 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 throughput improvements equivalent to the performance ratios stated here.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

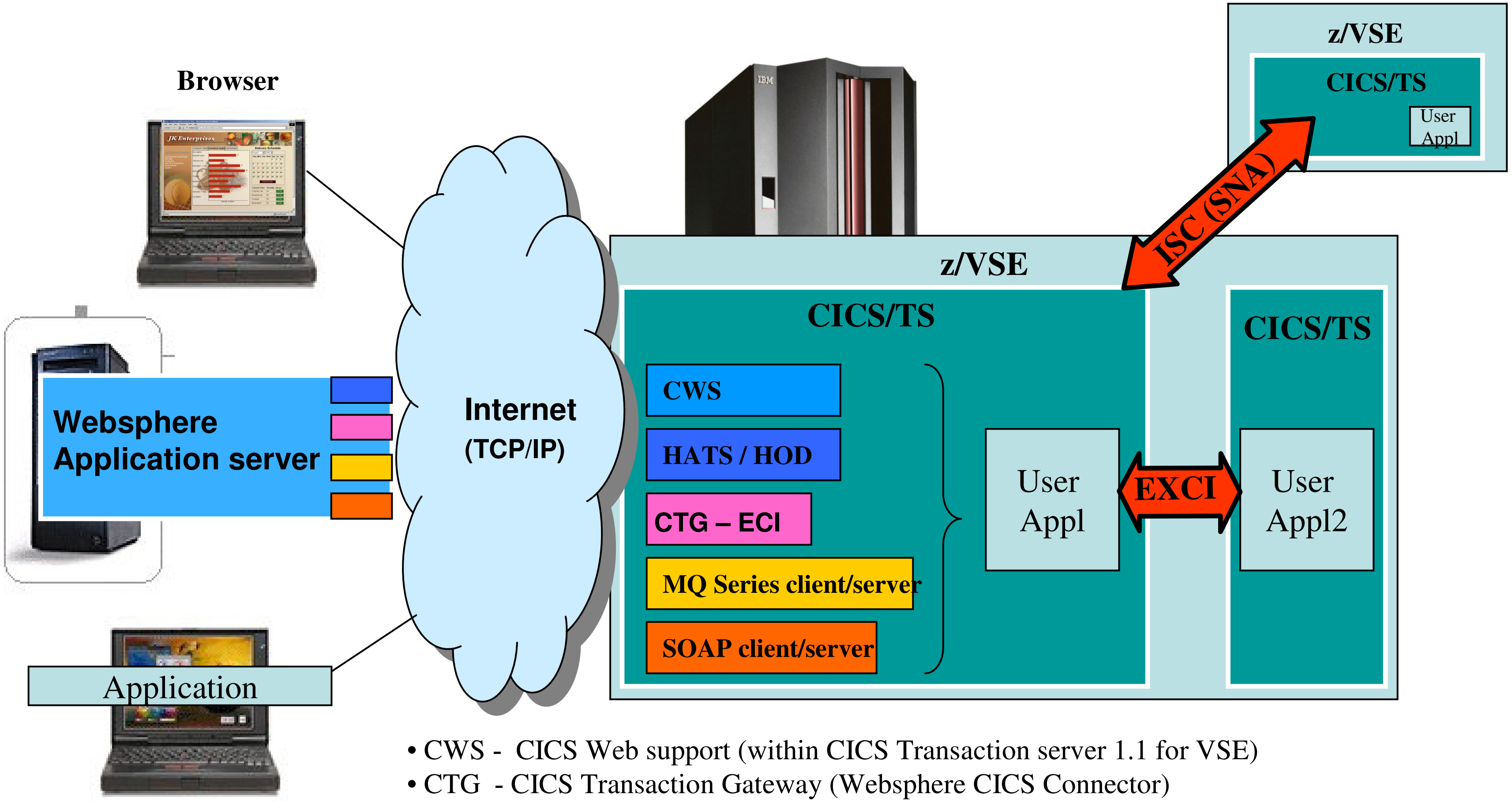
References in this document to IBM products or services do not imply that IBM intends to make them available in every country.

Any proposed use of claims in this presentation outside of the United States must be reviewed by local IBM country counsel prior to such use.

The information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

Inter-Communication with z/VSE Transactions



- CWS - CICS Web support (within CICS Transaction server 1.1 for VSE)
- CTG - CICS Transaction Gateway (Websphere CICS Connector)
- HATS – Host Access Transformation Server (no VSE software component required)
- HOD - Host OnDemand (Websphere Host Integration software)
- SOAP - Simple Object Access Protocol (Web Services based with XML data)

Connectivity to CICS transactions

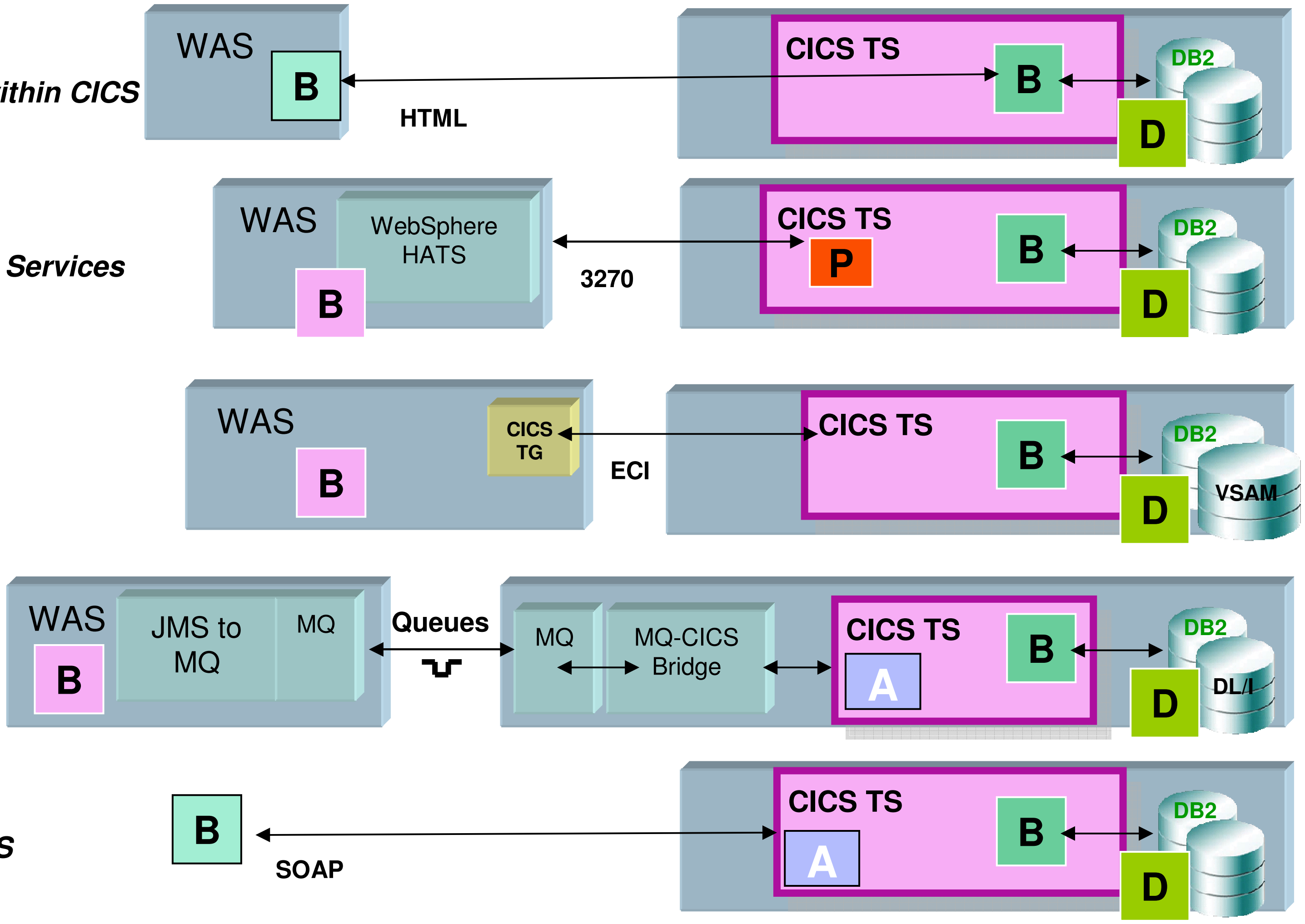
HTML Access:
CICS Web Services (CWS) within CICS

WebSphere
Host Access Transformation Services (HATS)

J2C Connector:
CICS Transaction Gateway

JMS Connector:
MQ to CICS Bridge

SOAP Access:
Web Services access to CICS



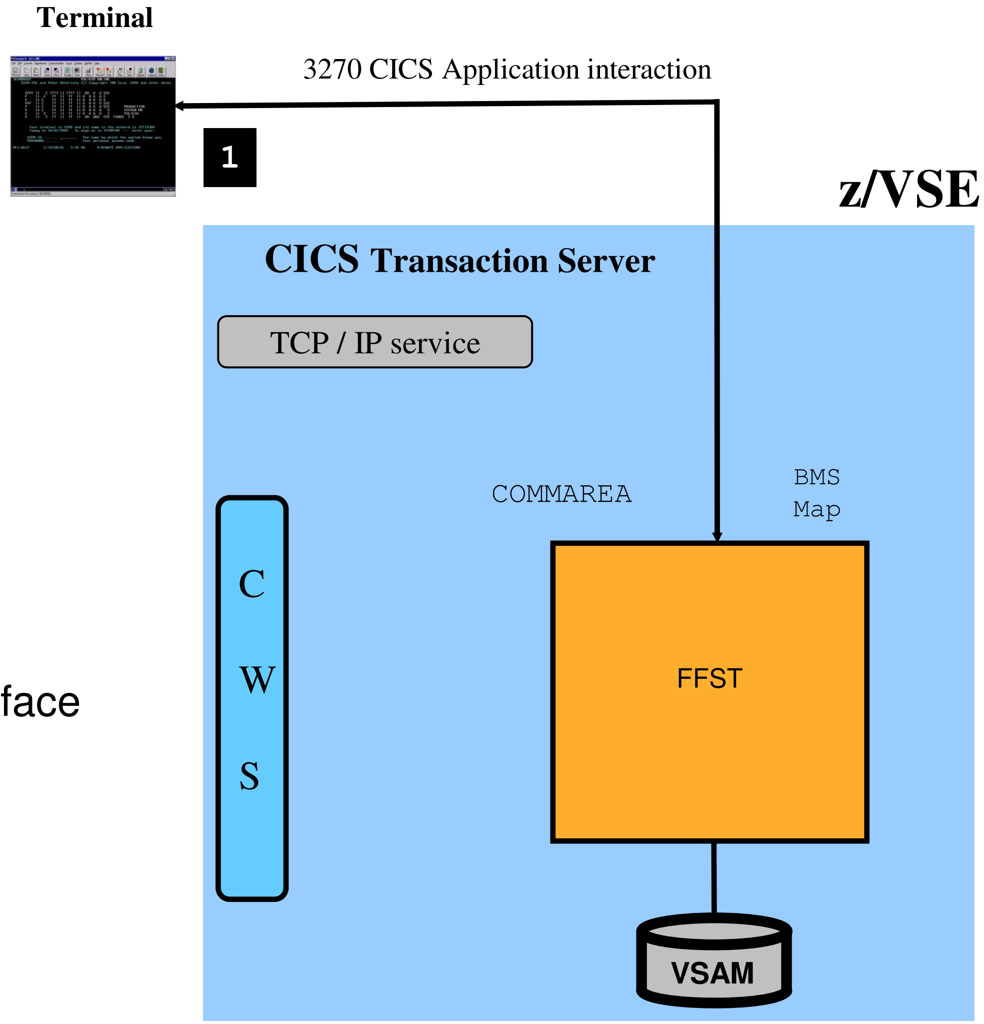
WAS can be on Linux on z or on another distributed platform.
Qualities of Services will vary.

CICS application Interfaces

CICS Application interfaces:

- interaction via 3270 screen
- interaction via 3270 and BMS Maps
- interaction via Commarea
- interaction via TS Queues

Note: For CICS Application integration
we need a callable Application interface



Terminal

3270 CICS Application interaction

1

z/VSE

CICS Transaction Server

TCP / IP service

C
W
S

COMMAREA

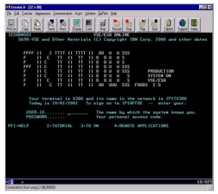
BMS
Map

FFST

VSAM

From 3270 screens to browser interaction

3270 Terminal



Session A - [32 x 80]
File Edit Transfer Appearance Communication Assist Window ZipPrint Help
PrtScrn Copy Paste Send Recv Display Color Map Record Stop Play Quit Clipbrd Support Index

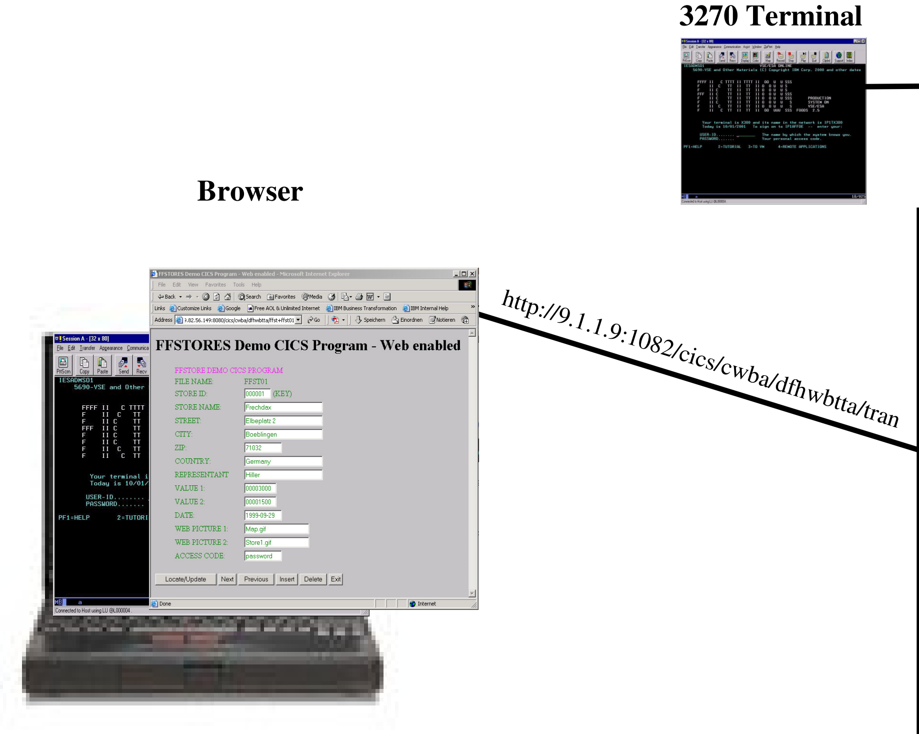
```
IESADMSO1                         VSE/ESA ONLINE
   5690-VSE and Other Materials (C) Copyright IBM Corp. 2000 and other dates

     FFFF II   C TTTT II TTTT II  OO  U  U SSS
     F    II  C   TT  II  TT  II O  O U  U S
     F    II C    TT  II  TT  II O  O U  U S
     FFF  II C    TT  II  TT  II O  O U  U SSS
     F    II C    TT  II  TT  II O  O U  U SSS     PRODUCTION
     F    II C    TT  II  TT  II O  O U  U   S     SYSTEM ON
     F    II  C   TT  II  TT  II O  O U  U   S     VSE/ESA
     F    II   C  TT  II  TT  II  OO  UUU  SSS  FOODS  2.5

       Your terminal is X380 and its name in the network is IP1TX380
       Today is 10/01/2001   To sign on to IP1AFFDE  --  enter your:

      USER-ID........ ________   The name by which the system knows you.
      PASSWORD.......            Your personal access code.

PF1=HELP       2=TUTORIAL   3=TO VM      4=REMOTE APPLICATIONS

MB     a                                                        18/025
Connected to Host using LU IBL000004 .
```

Browser

CICS Web Support (CWS)

From 3270 screens to Browser interfaces for CICS transactions

Note: - CWS is not a 'Web Service', it is a guification based on BMS maps only
- It is a screen based interface - not callable

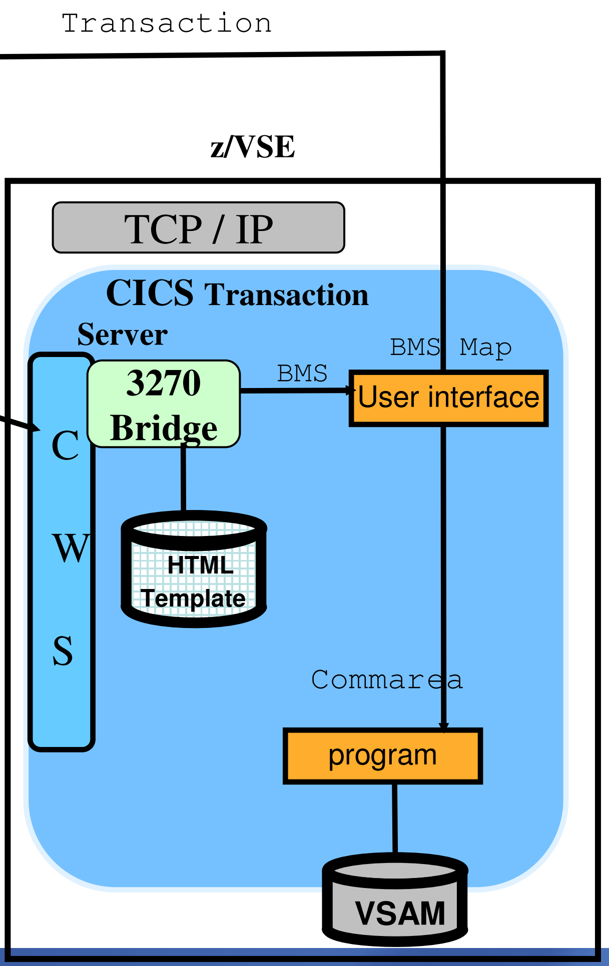

3270 Terminal
Browser
http://9.1.1.9:1082/cics/cwba/dfhwbtta/tran
FFSTORES Demo CICS Program - Web enabled - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Links Customize Links Google Free AOL & Unlimited Internet IBM Business Transformation IBM Internal Help
Address 9.82.56.149:8080/cics/cwba/dfhwbtta/ffst+ffst01
Go Speichern Einordnen Notieren
Done Internet

FFSTORES Demo CICS Program - Web enabled

FFSTORE DEMO CICS PROGRAM

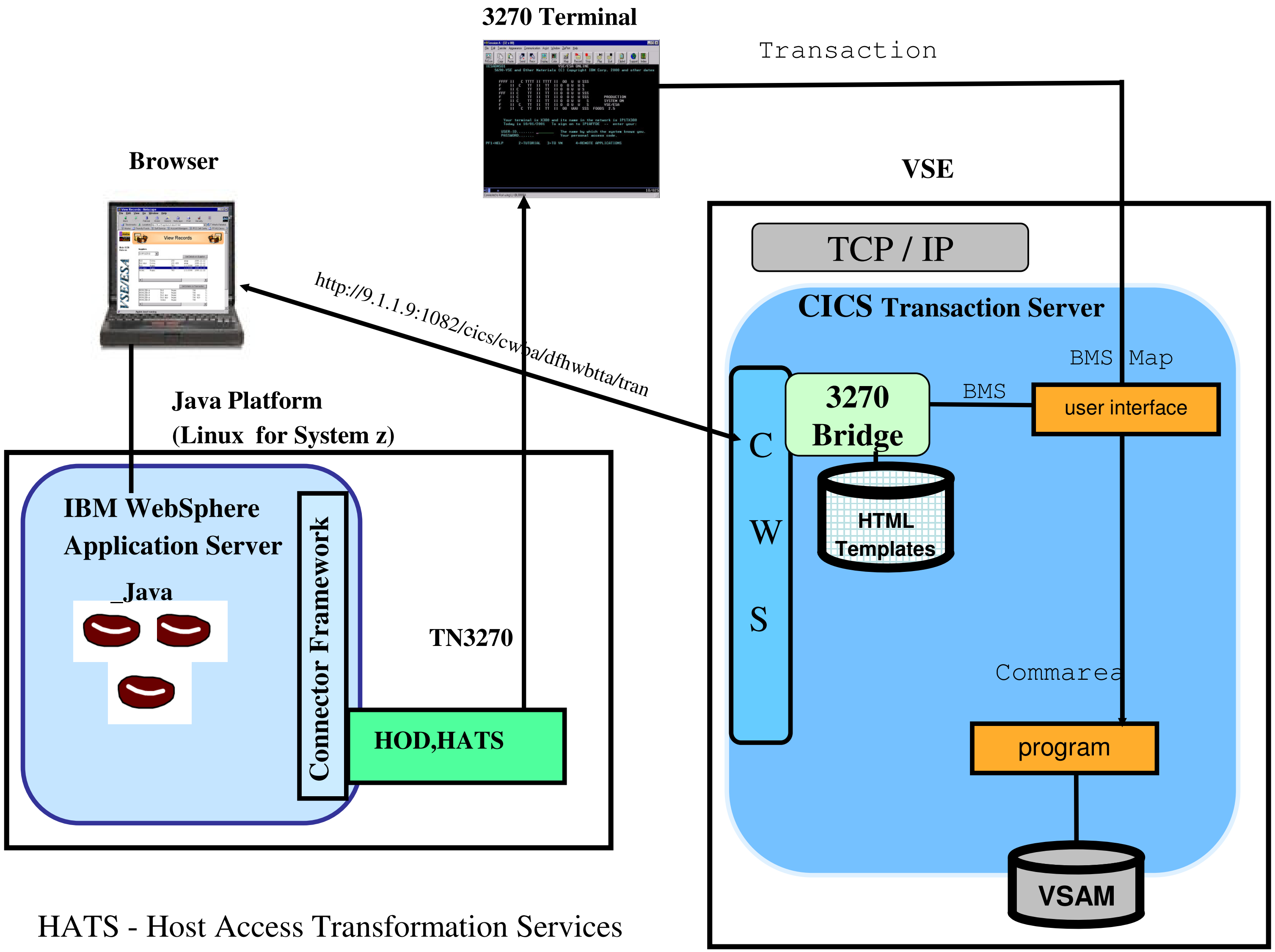
FILE NAME:	FFST01
STORE ID:	000001 (KEY)
STORE NAME:	Frechdax
STREET:	Elbeplatz 2
CITY:	Boeblingen
ZIP:	71032
COUNTRY:	Germany
REPRESENTANT	Hiller
VALUE 1:	00003000
VALUE 2:	00001500
DATE:	1999-09-29
WEB PICTURE 1:	Map.gif
WEB PICTURE 2:	Store1.gif
ACCESS CODE:	password

Locate/Update | Next | Previous | Insert | Delete | Exit



- ►direct access to z/VSE transactions via web Browser
- ►Without the need of a web server on z/VSE
- ►Build-in Function in CICIS TS

From 3270 screens to Browser interaction with HATS



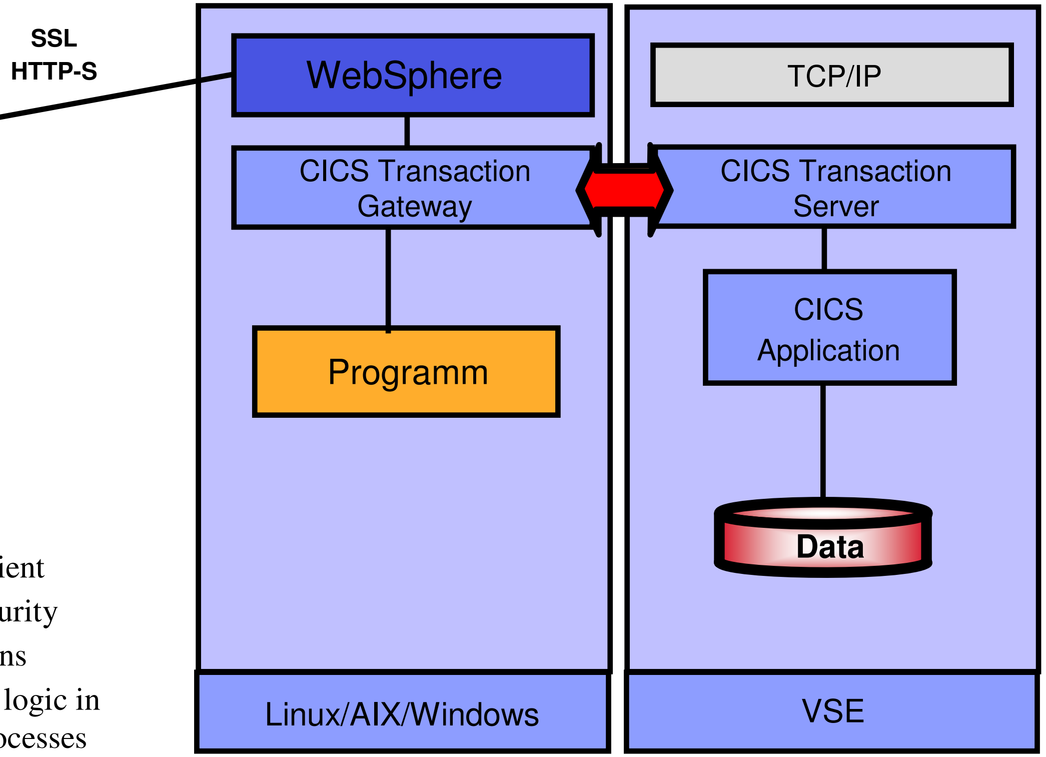
HATS - Host Access Transformation Services

Access to VSE transactions from remote

CTG (CICS Transaction Gateway)

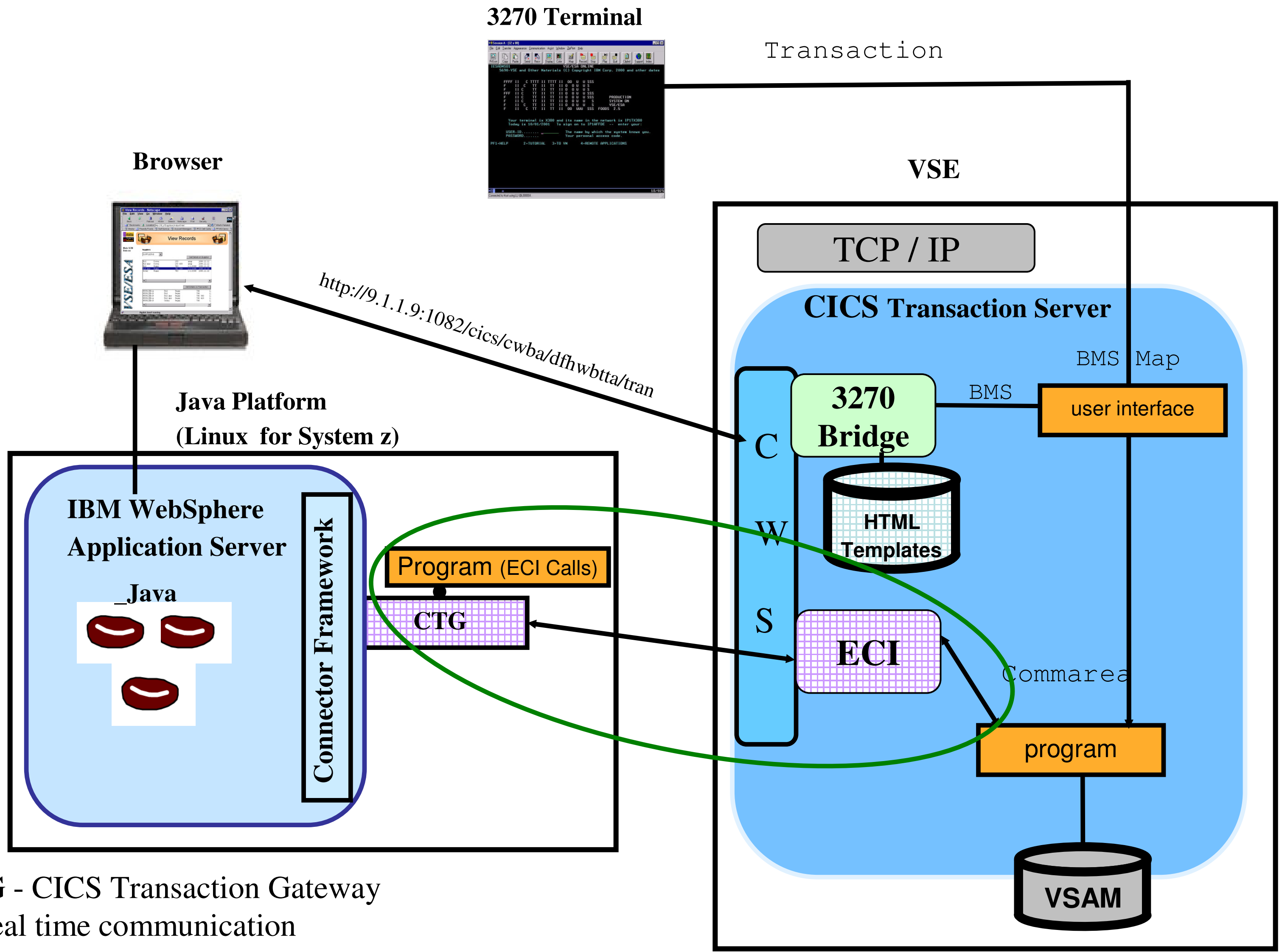
Integration of z/VSE transaction processes

CICS Transaction Gateway - Implementation



- ►Contains CICS Universal Client
- ►Synchronous transaction security
- ►Allows secure communications
- ►Integration of CICS business logic in heterogeneous transaction processes

From 3270 screens to Browser interfaces for CICS transactions



CTG - CICS Transaction Gateway

- Real time communication

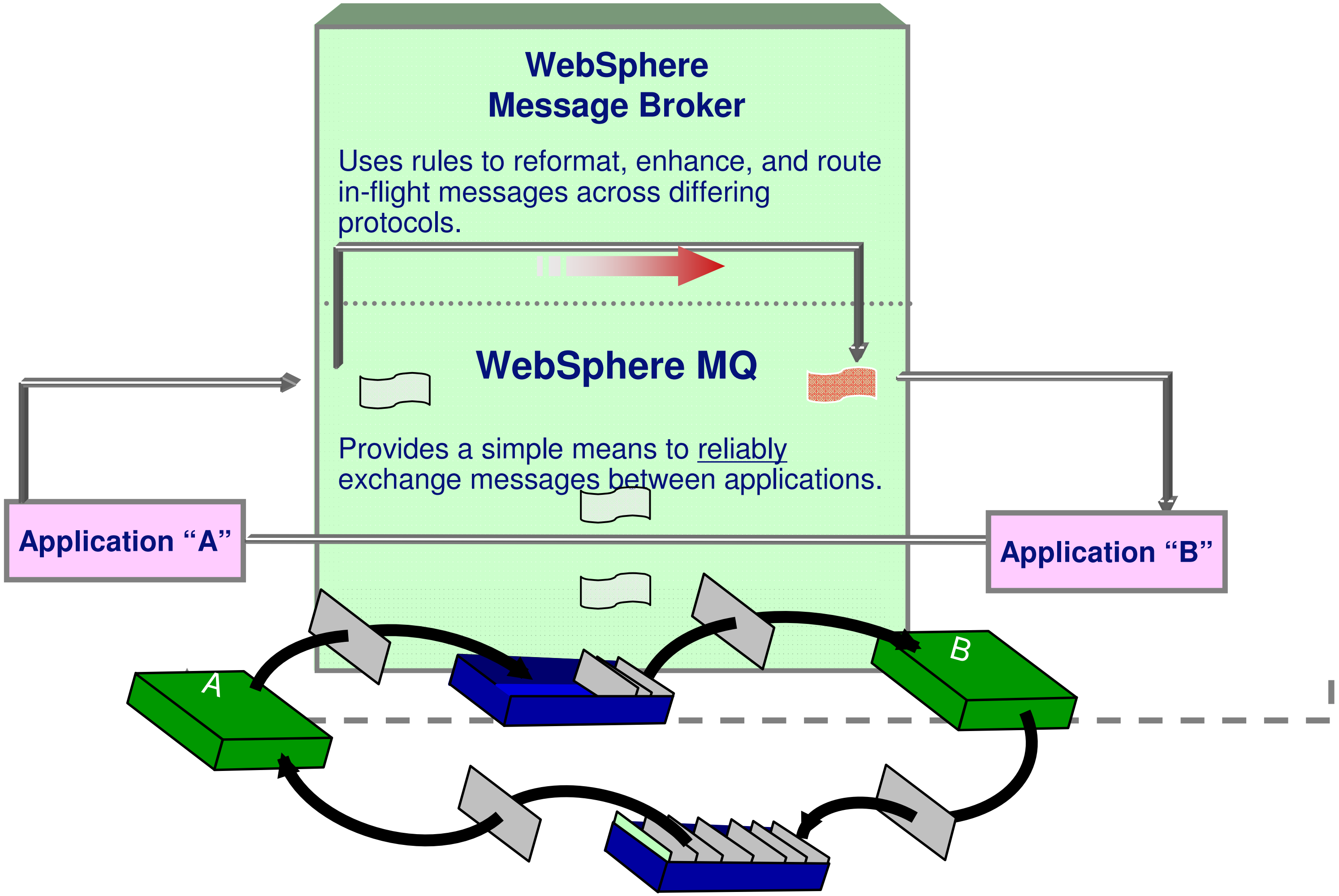
- Reuse z/VSE logic as a callable application from remote Systems

MQ Series

Synchronous/Asynchronous application communication

Messaging Overview

Event Notification (1 way communication), Request / Response (2 way communication)



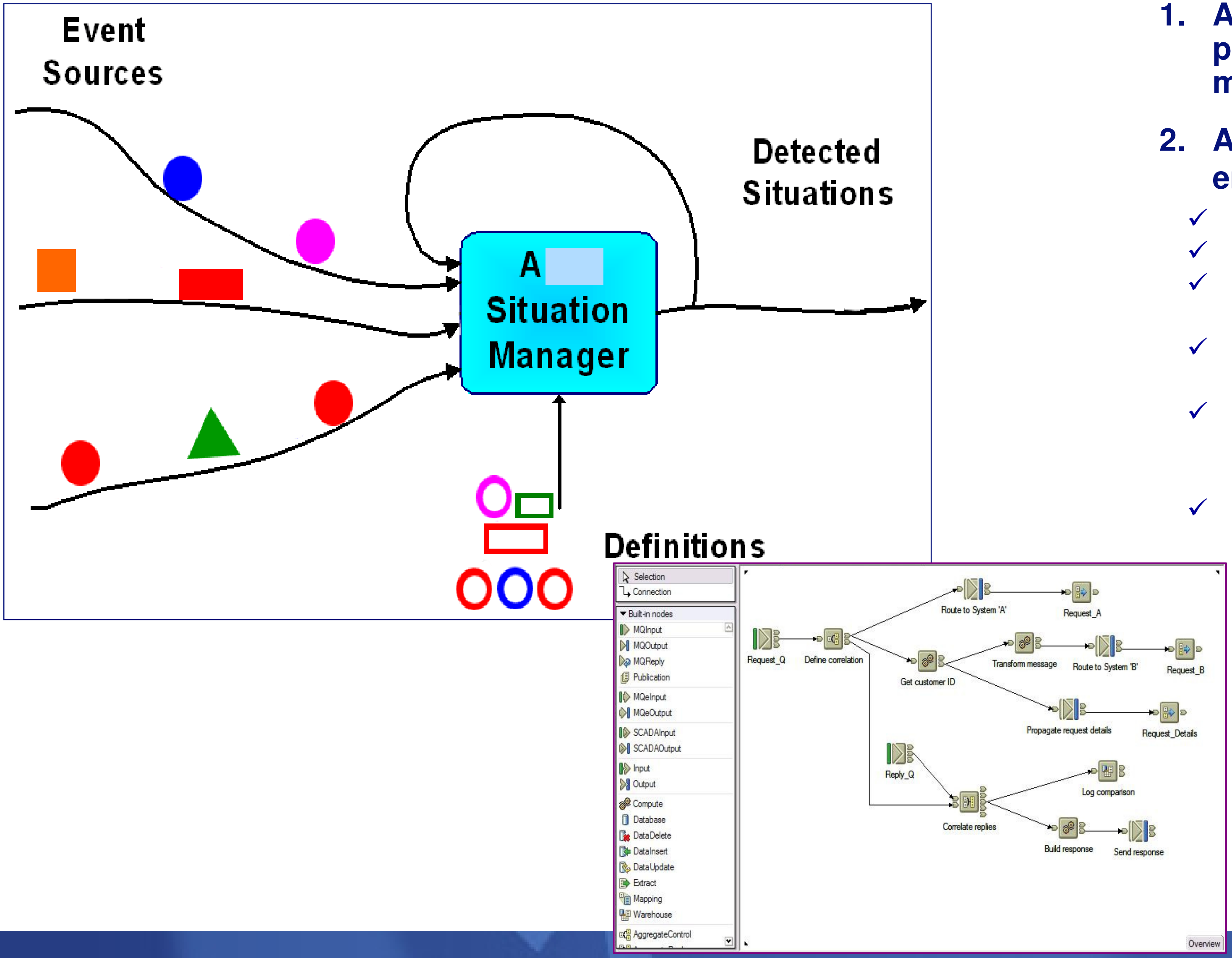
WebSphere Message Broker

Uses rules to reformat, enhance, and route in-flight messages across differing protocols.

WebSphere MQ

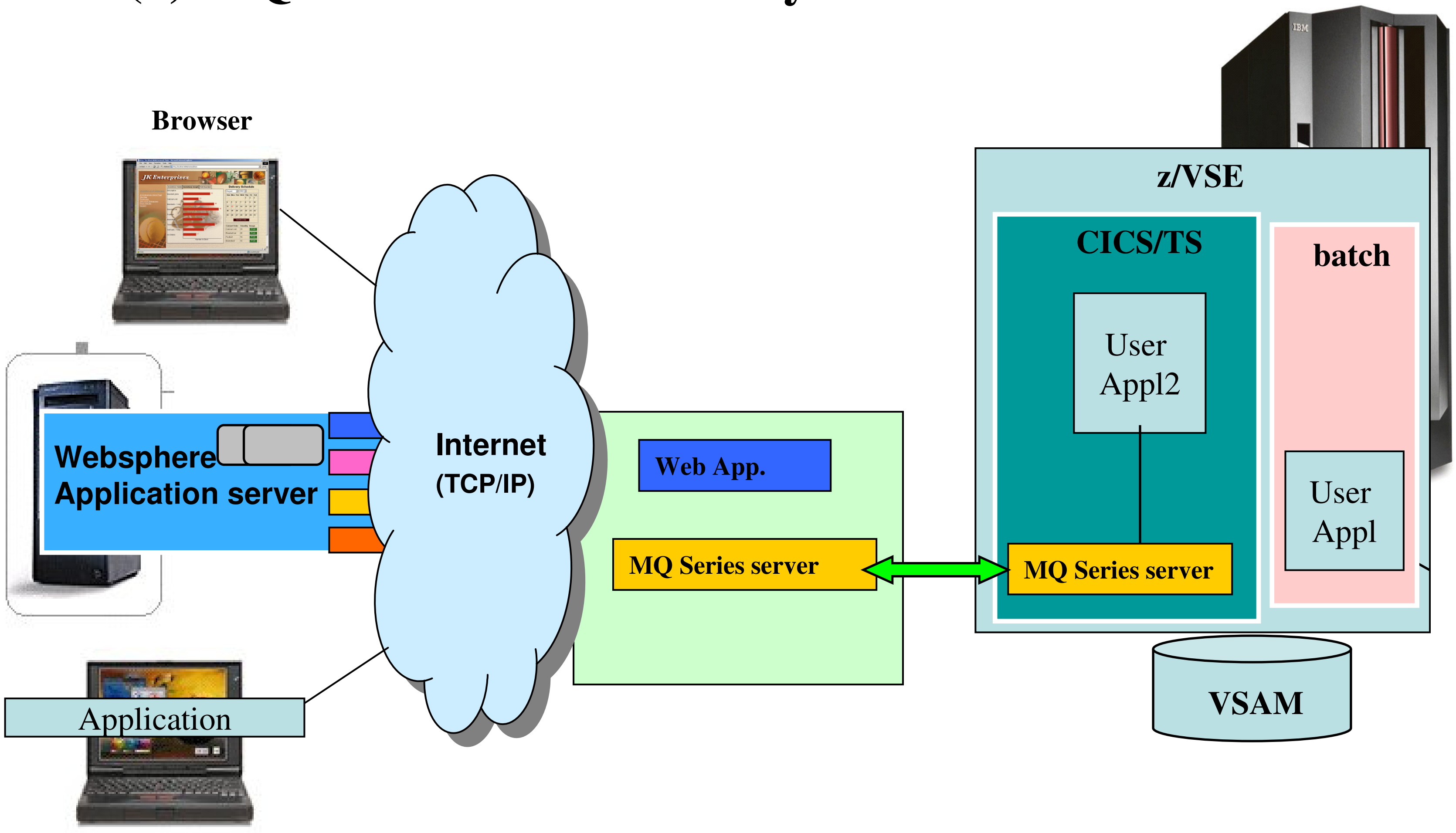
Provides a simple means to reliably exchange messages between applications.

What is WebSphere Message Broker?

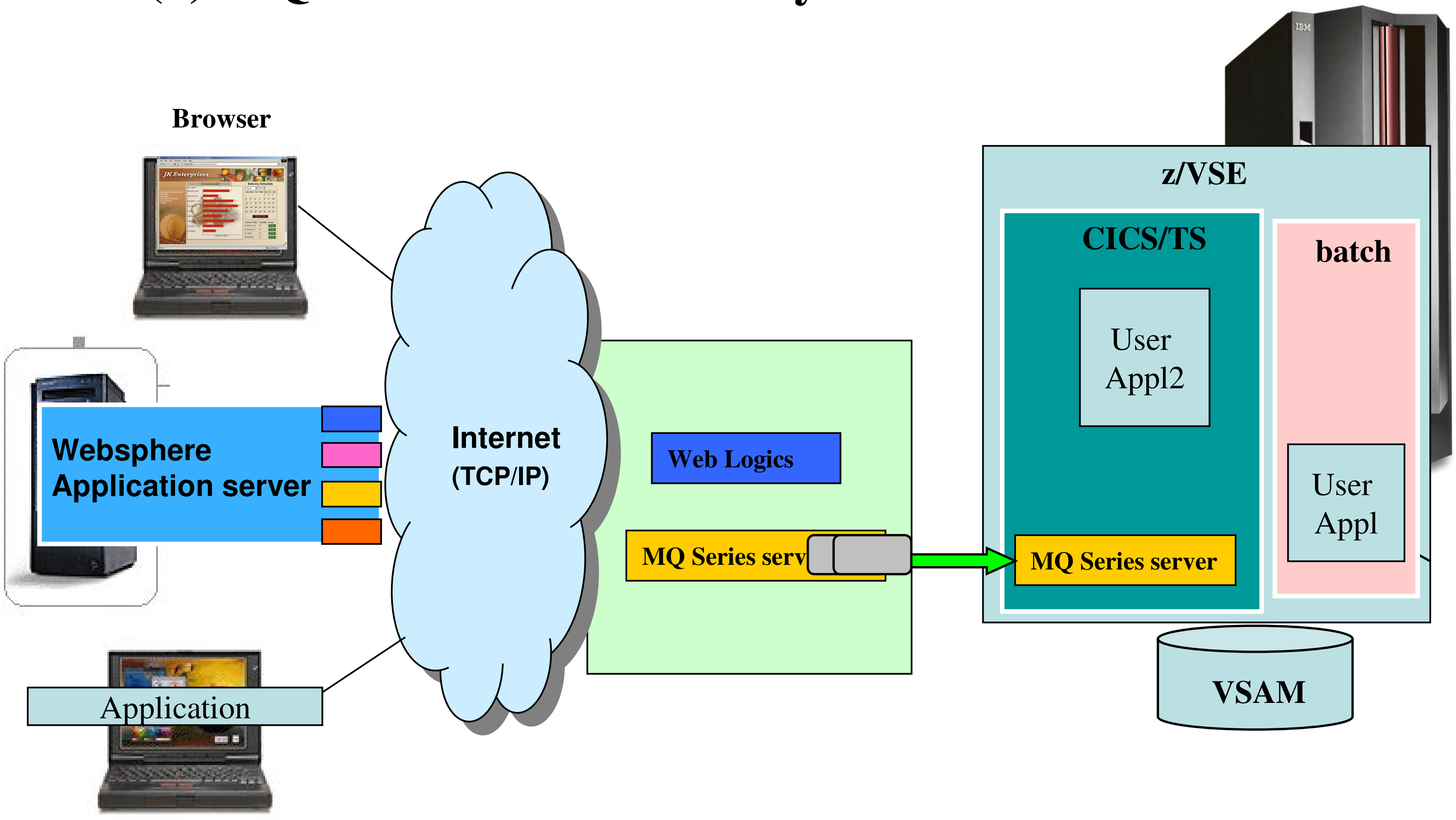


1. **A framework for processing MQ messages**
2. **A robust hosting environment for:**
   - ✓ Transforming data
   - ✓ Enriching data
   - ✓ Interacting with databases
   - ✓ Routing messages based on content
   - ✓ Detecting complex combinations of messages
   - ✓ Interacting existing applications with Web Services

(4) MQ Environment with asynchronous work



(4) MQ Environment with asynchronous work



SOA
Web Services

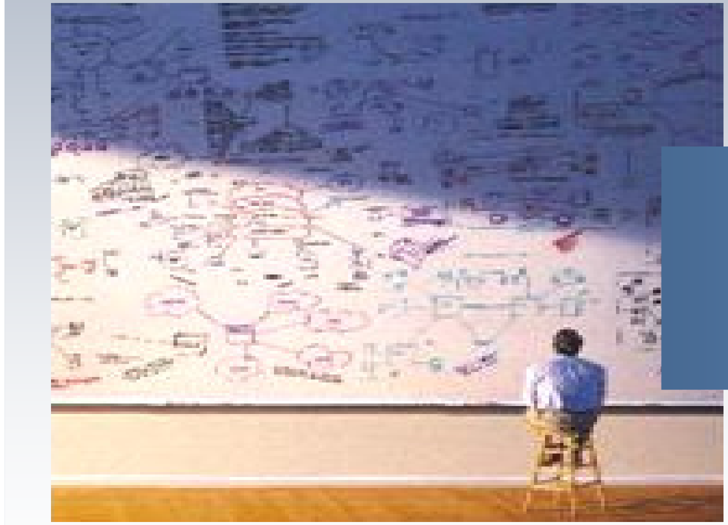
Modern architecture of program communication using XML data and the SOAP protocol

Think different with SOA:

With SOA to a Business Process driven IT

Traditional Thinking

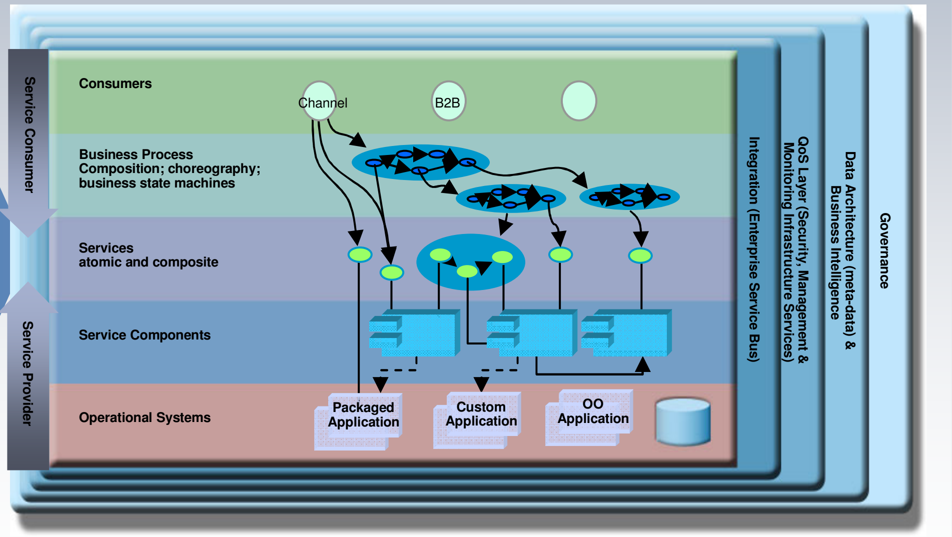
IT manages IT **assets** that support the business – implementation thinking



Silos, static

Business Thinking

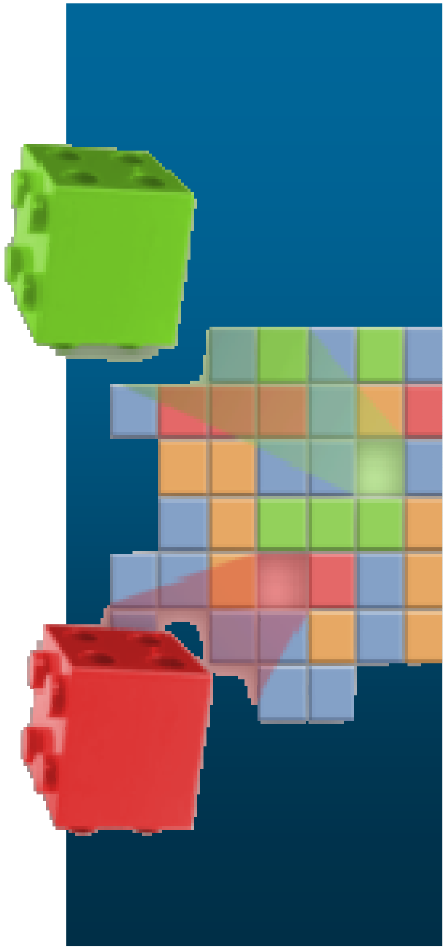
IT manages **Services** and **Components** which **reflect** the Business processes – functional thinking


Service Consumer
Service Provider
Consumers
Channel
B2B
Business Process Composition; choreography; business state machines
Services atomic and composite
Service Components
Operational Systems
Packaged Application
Custom Application
OO Application
Integration (Enterprise Service Bus)
QoS Layer (Security, Management & Monitoring Infrastructure Services)
Data Architecture (meta-data) & Business Intelligence
Governance

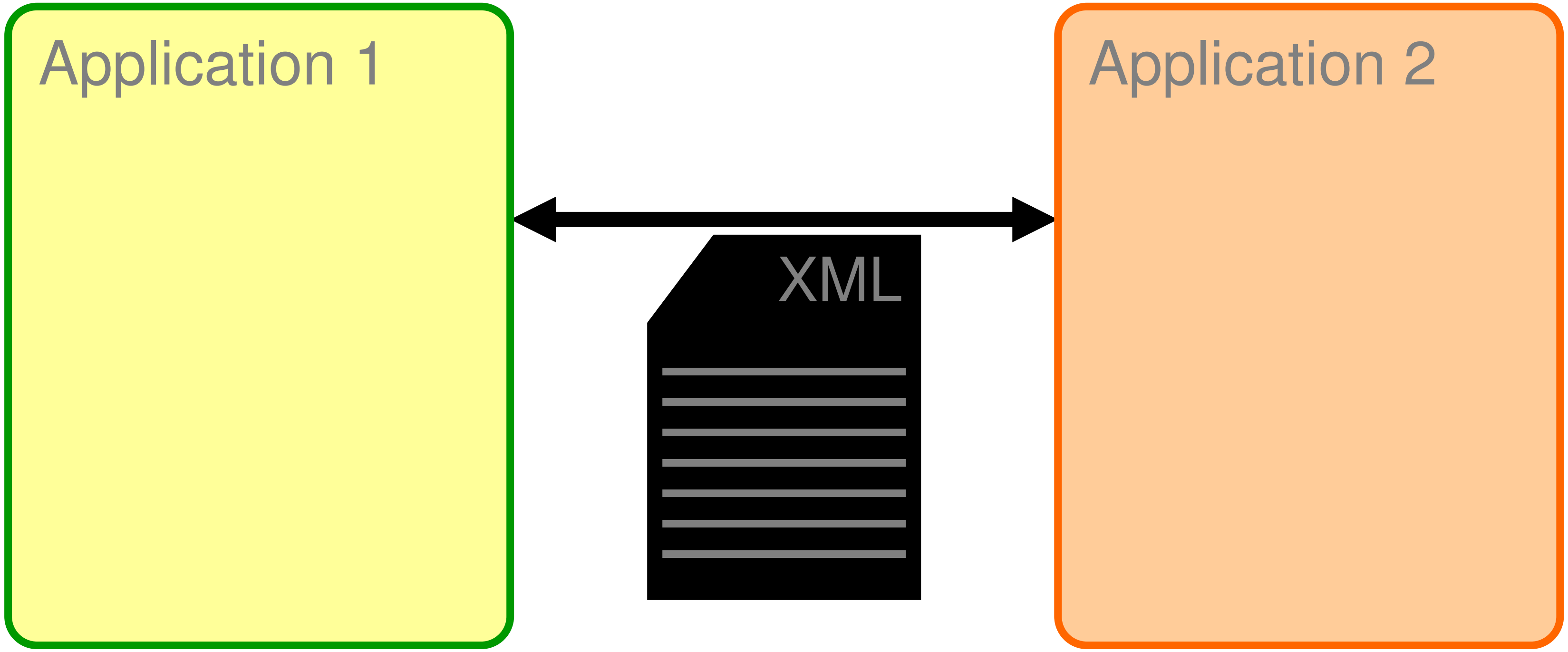
Flexibility, dynamic, virtualised

Why Reuse for Service Oriented Architectures?

- **Existing applications** are among the most valuable assets a company owns
- It is **5X less expensive** to reuse existing applications than to write new applications from scratch*
- Reusing proven, time-tested applications results in significantly **lower risks** and faster time to market
- **Maintenance** overhead **shrinks** with greater use of proven and tested code for common functions
- **Best practices** provide key framework for business policies and reuse as a design point for SOA

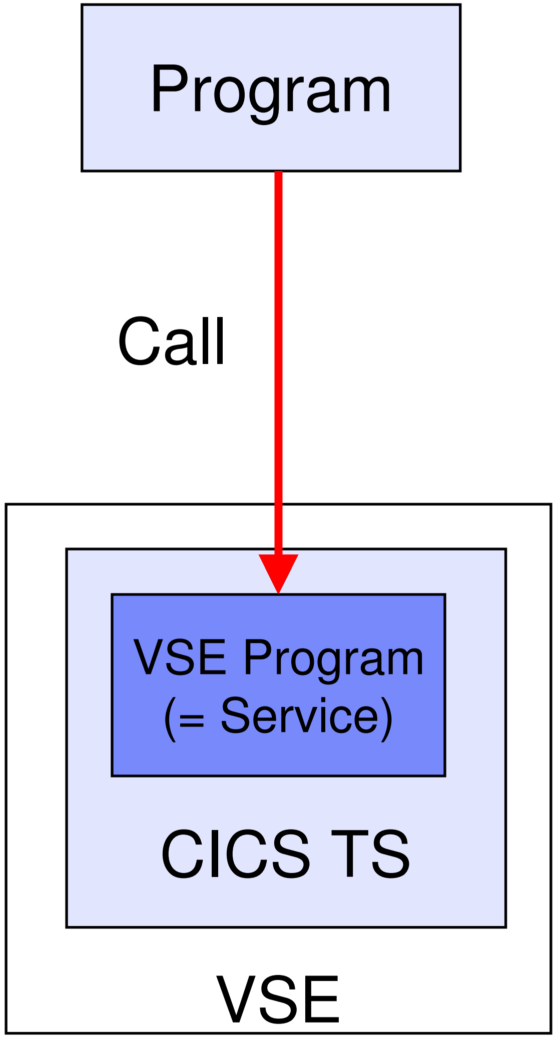


What are Web Services? Applications !



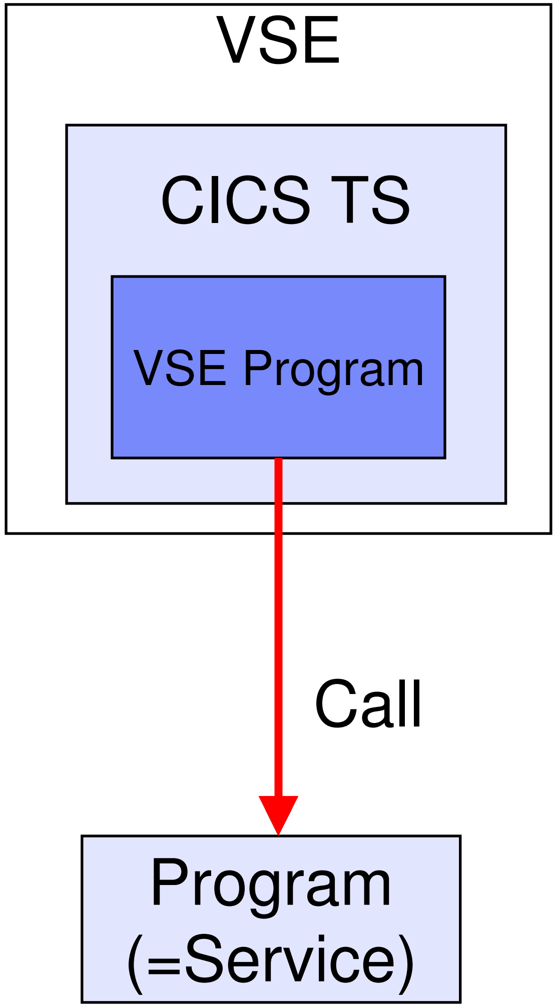
What is a Web Service?

- Assume you have a VSE program that implements some kind of important business logic
- Someone else (outside VSE) wants to use this program
  - 1. Possibility: Rewrite the same logic
    - May need access to VSE data
    - Changes/Fixes in VSE code needs to be re-done in new code also
  - 2. Possibility: Call the VSE program from remote
    - VSE program can be treated as a Web Service
    - VSE is the Web Service provider



What is a Web Service?

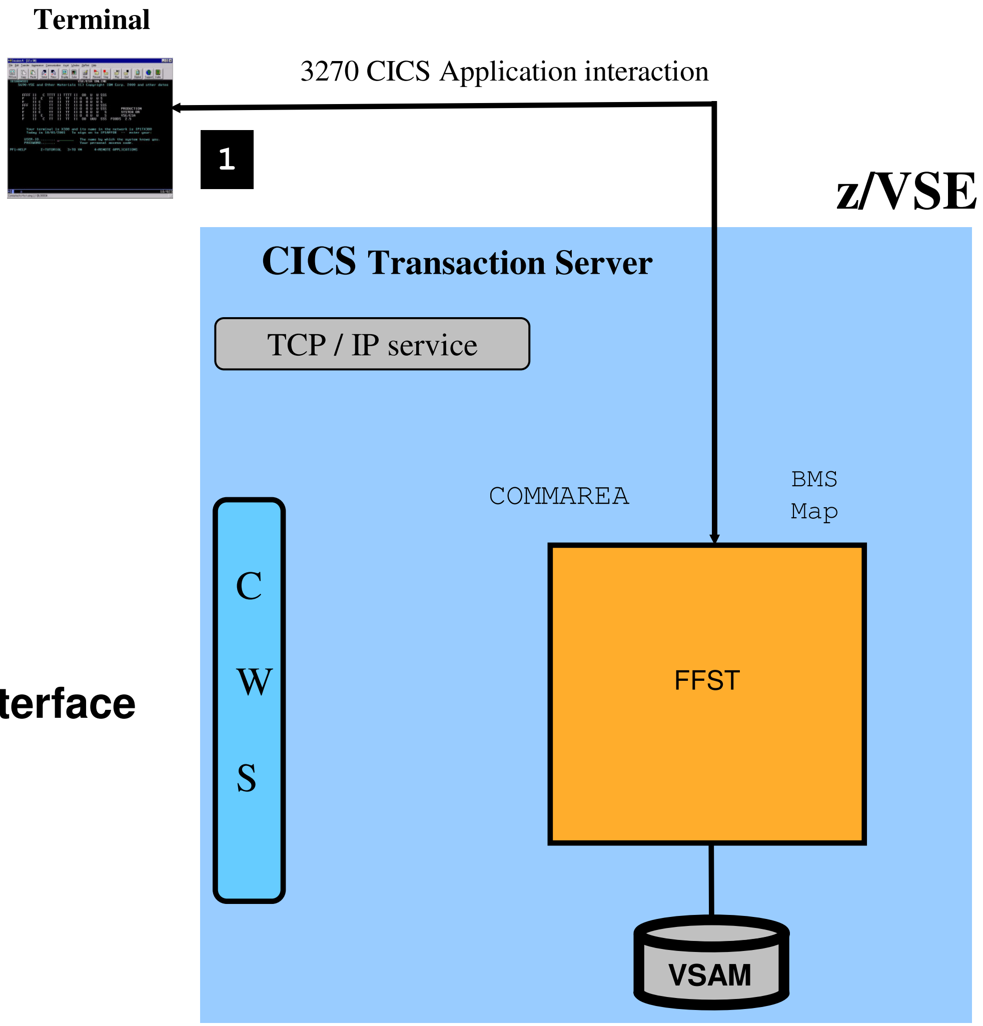
- Assume someone has a program that implements some kind of important business logic
- You want to use this program inside a VSE application
  - 1. Possibility: Rewrite the same logic
    - May need access to the remote data
    - Changes/Fixes in code needs to be re-done in VSE code also
  - 2. Possibility: Call the external program from VSE
    - External program can be treated as a Web Service
    - VSE is the Web Service Requestor



CICS application Interfaces

CICS Application interfaces:
- interaction via 3270 screen
- interaction via 3270 and BMS Maps
- interaction via Commarea
- interaction via TS Queues

Note: For CICS Application integration
we need a callable Application interface



Terminal

3270 CICS Application interaction

1

z/VSE

CICS Transaction Server

TCP / IP service

COMMAREA

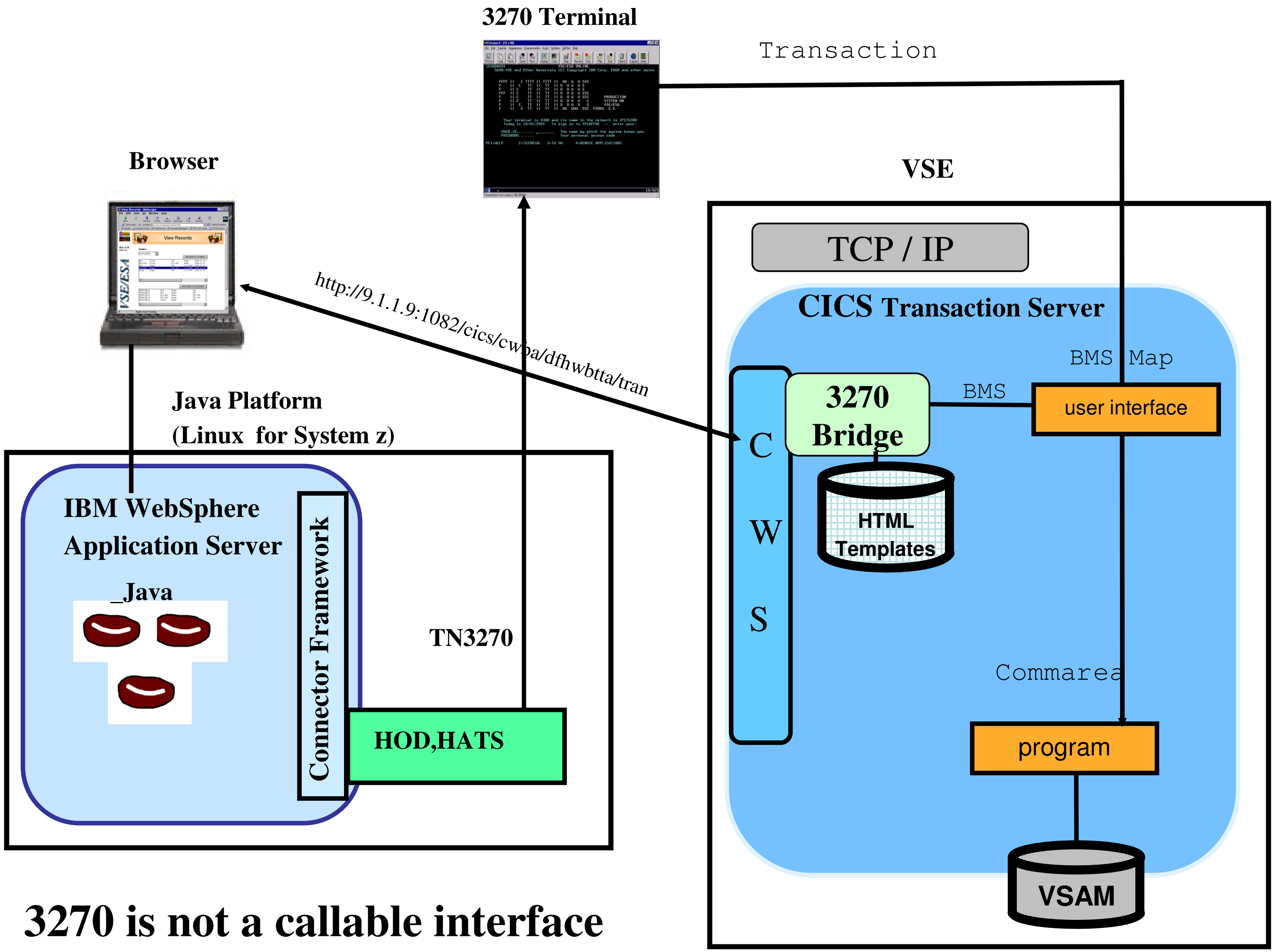
BMS
Map

C
W
S

FFST

VSAM

From 3270 screens to Browser interfaces for CICS transactions

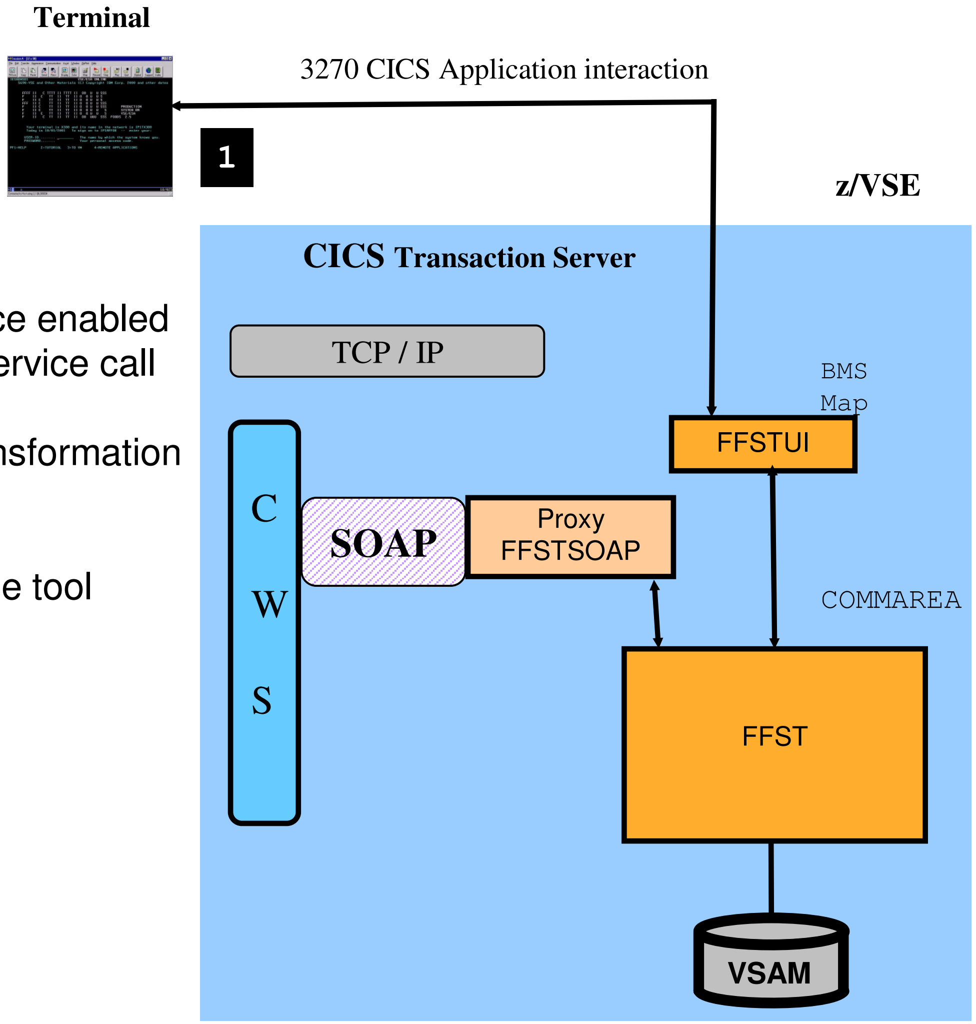


3270 is not a callable interface

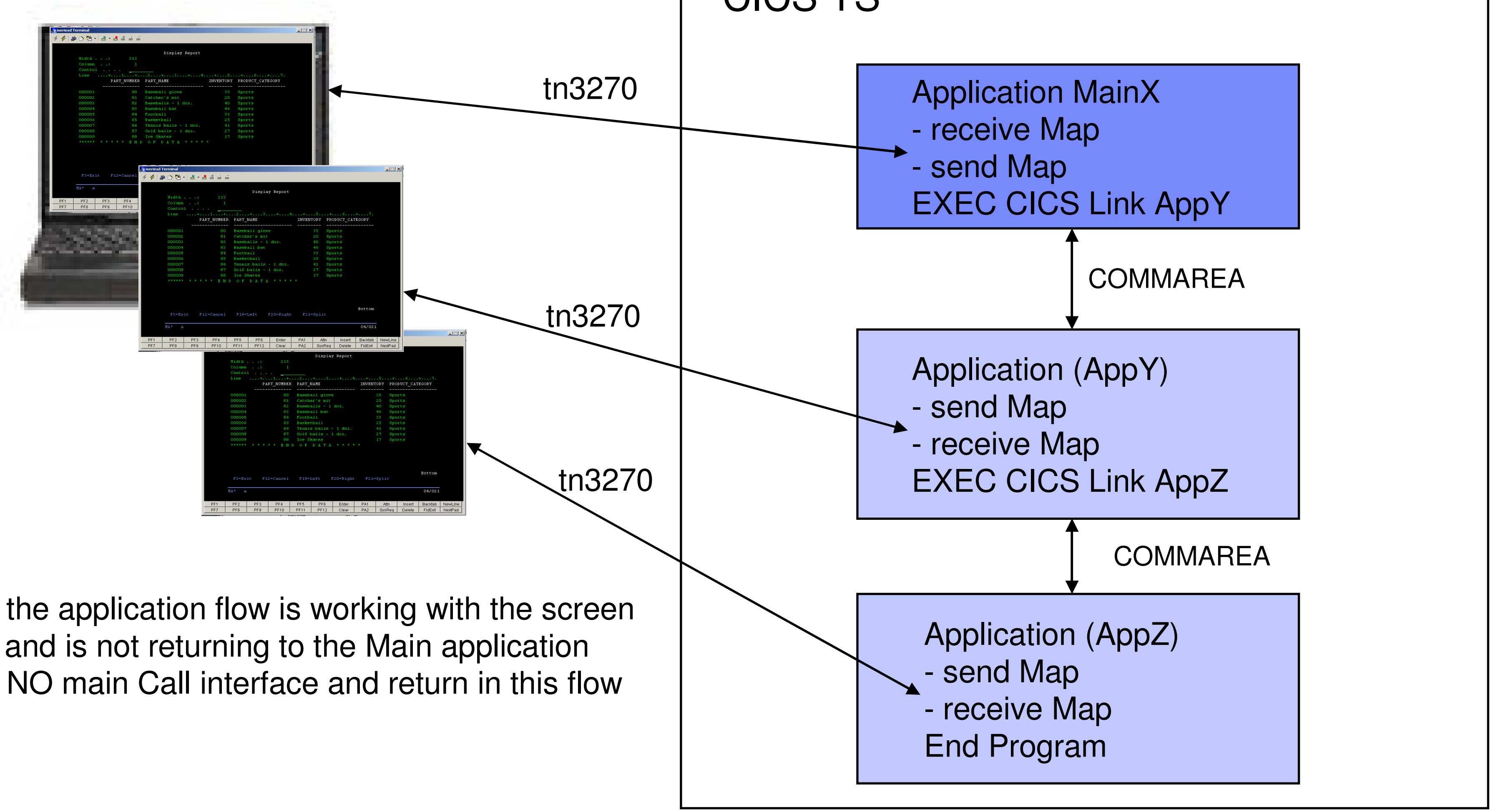
CICS application structure for Web Services - with Proxy Code

- the application presentation logic, BMS maps remain unchanged if application is Web Service enabled
- CWS - is used as Listener in CICS for Web Service call
- the Proxy FFSTSOAP is the proxy code used for a Web Service to CICS interface transformation (i.e. XML - COMMAREA)

Note: This proxy code can be generated with the tool CICS2WS.



CICS 3270 applications and their behavior



iseriesd Terminal

```
                         Display Report
 Width . . .:       213
 Column  . .:         1
 Control  . . . .
 Line   ....+....1....+....2....+....3....+....4....+....5....+....6....+....7.
           PART_NUMBER  PART_NAME               INVENTORY  PRODUCT_CATEGORY
         -------------  ----------------------  ---------  -----------------
 000001             80  Baseball glove                 35  Sports
 000002             81  Catcher's mit                  20  Sports
 000003             82  Baseballs - 1 doz.             40  Sports
 000004             83  Baseball bat                   46  Sports
 000005             84  Football                       33  Sports
 000006             85  Basketball                     25  Sports
 000007             86  Tennis balls - 1 doz.          41  Sports
 000008             87  Golf balls - 1 doz.            27  Sports
 000009             88  Ice Skates                     17  Sports
 ****** * * * * *  E N D  O F  D A T A  * * * * *

                                                                    Bottom
  F3=Exit    F12=Cancel    F19=Left    F20=Right    F21=Split

MA*   a                                                             04/021
```

CICS TS

Application MainX
- receive Map
- send Map
EXEC CICS Link AppY

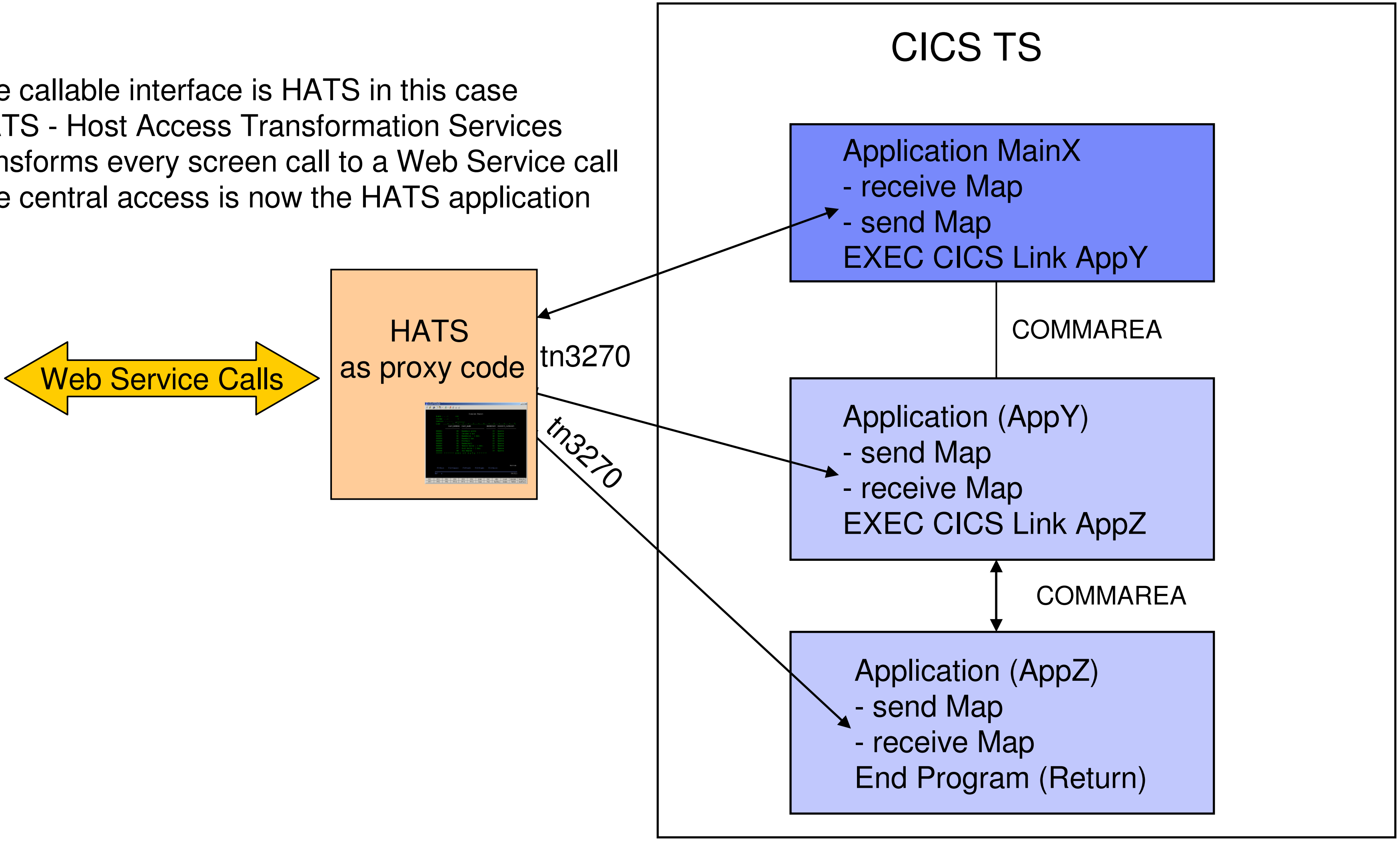
Application (AppY)
- send Map
- receive Map
EXEC CICS Link AppZ

Application (AppZ)
- send Map
- receive Map
End Program

- the application flow is working with the screen and is not returning to the Main application
- NO main Call interface and return in this flow

CICS 3270 applications as **Web Services**

- The callable interface is HATS in this case
- HATS - Host Access Transformation Services transforms every screen call to a Web Service call
- The central access is now the HATS application



CICS TS

Web Service Calls

HATS
as proxy code

tn3270

tn3270

Application MainX
- receive Map
- send Map
EXEC CICS Link AppY

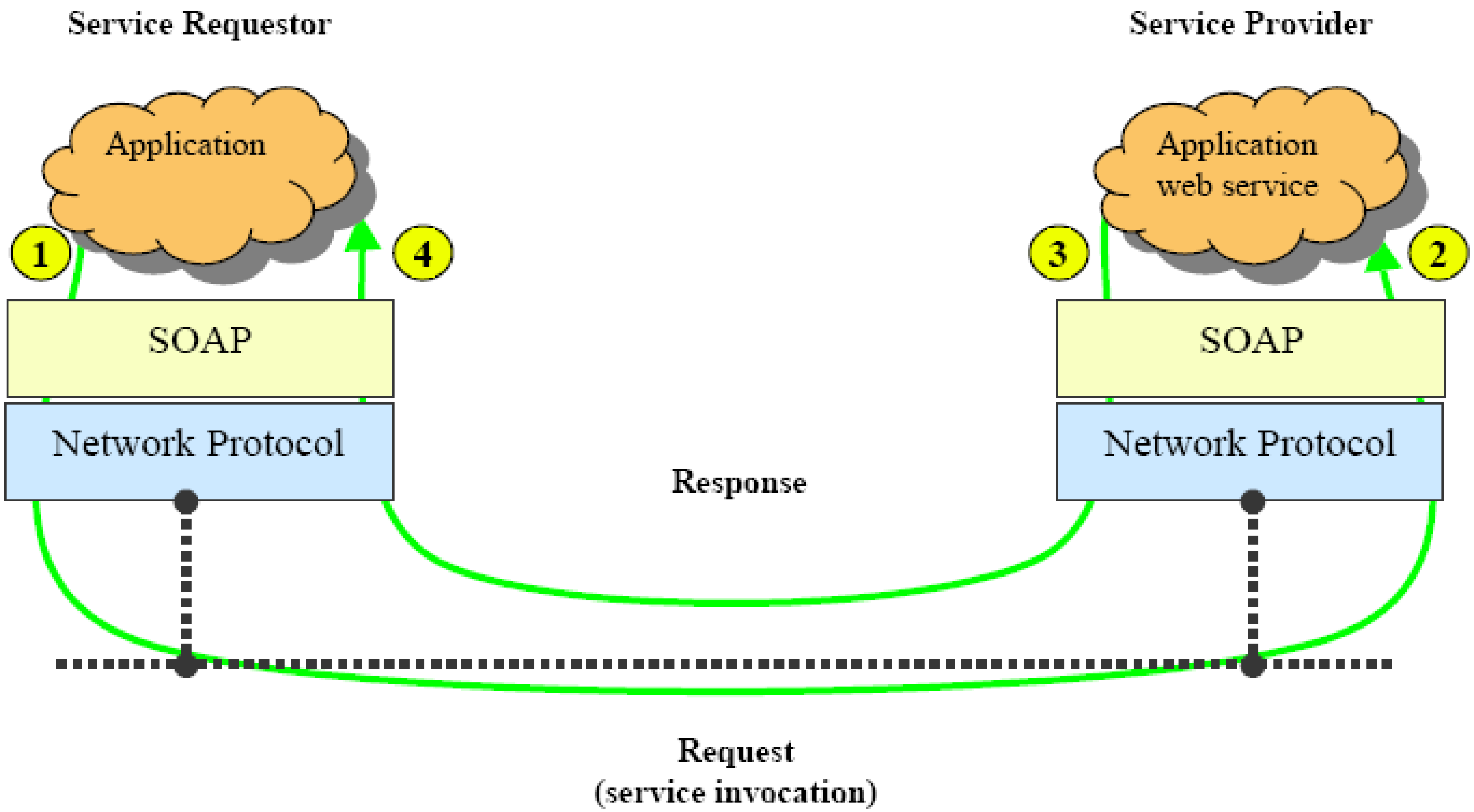
COMMAREA

Application (AppY)
- send Map
- receive Map
EXEC CICS Link AppZ

COMMAREA

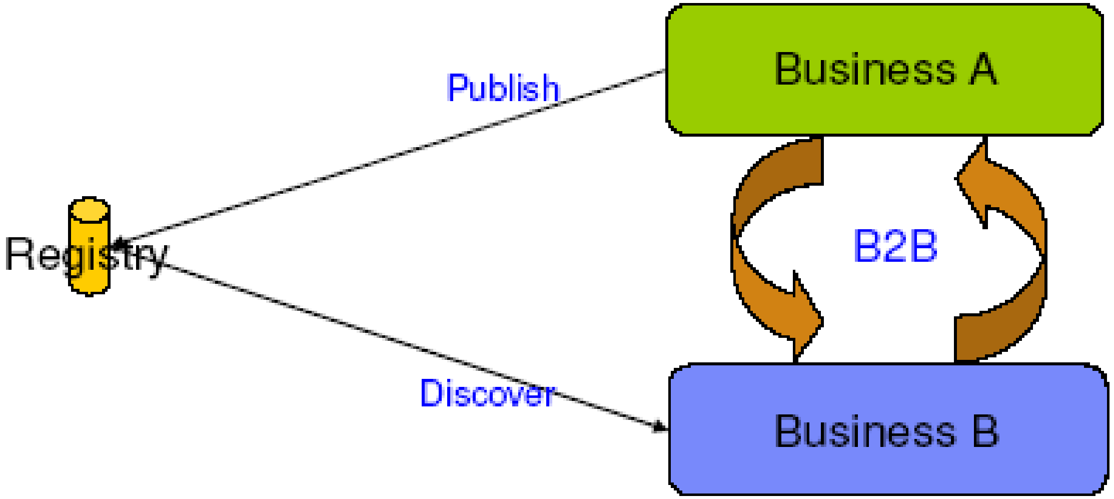
Application (AppZ)
- send Map
- receive Map
End Program (Return)

Web Services communicate via SOAP
What is Simple Object Access Protocol (SOAP)?
Application communication protocol with XML !

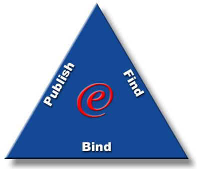


What is a Registry?

- An infrastructure that enables the publishing and discovery of Web Services
- Facilitates business-to-business (B2B) interactions

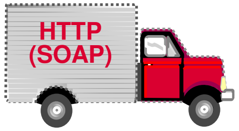


What is WSDL?



- Web Services Description Language

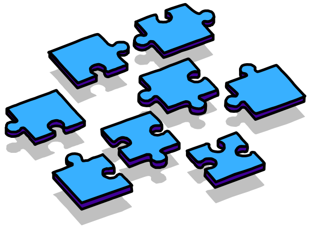
- WSDL is an XML based vocabulary for defining a Web Service:
  - interfaces
    - operation types (i.e. one-way, request-response, notification)
    - messages defining a Web Service interface
    - definition of data types (XML Schema)
  - access protocol (i.e. SOAP over HTTP)
  - contact endpoints (i.e. Web Service URL and URNs[1])



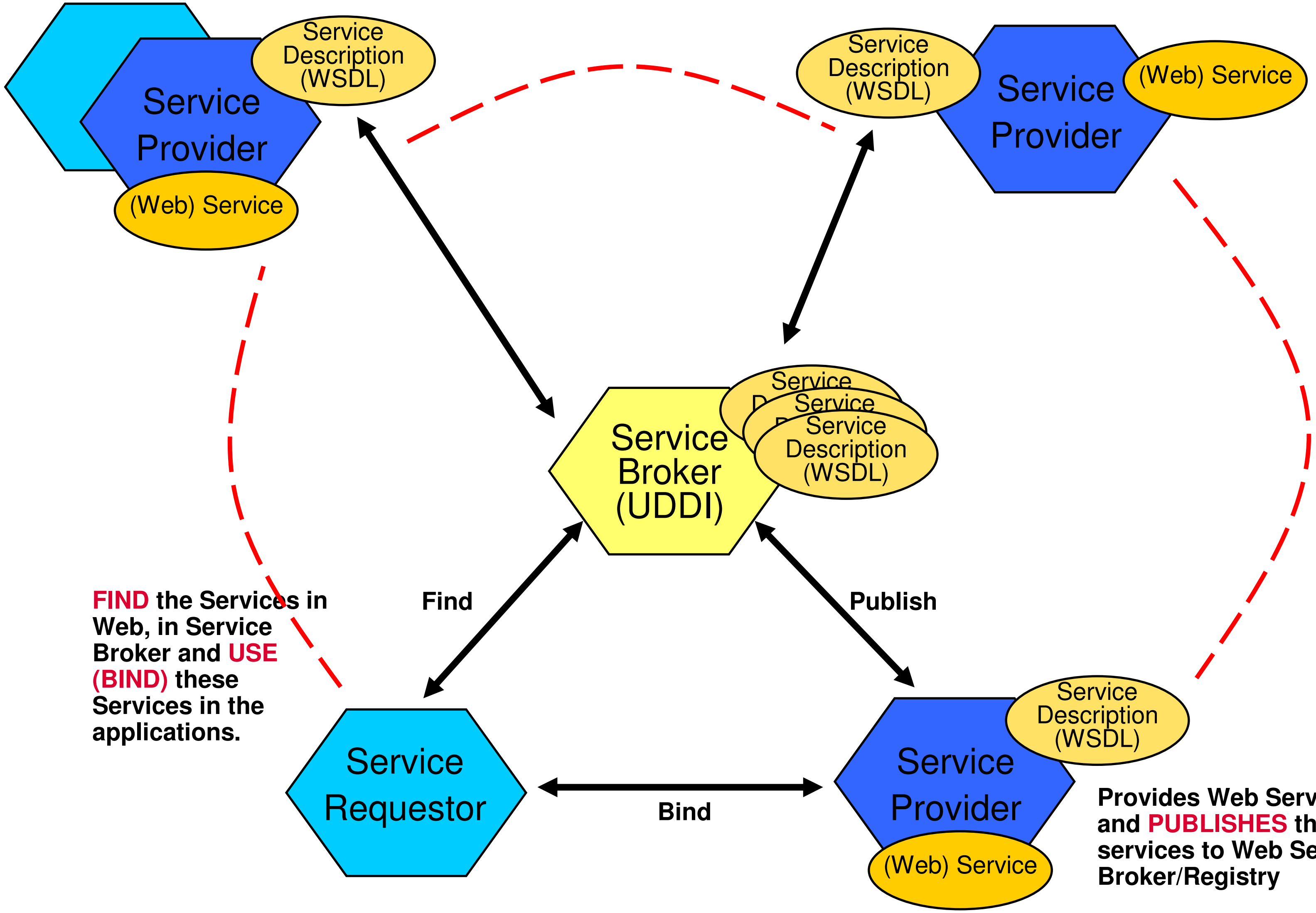
(**1** URNs are location independent pointers to a file, or to different representations of the same content. In most ways they can be used like URLs)

- A Web Service URL returning WSDL makes Web Services self-describing

- Similar in purpose to IDL (Interface Definition Language)
  - From a WSDL file, wizards can generate:
  - proxy classes for calling Web Service
    - skeleton classes to implement a Web Service



How Web Services work



FIND the Services in Web, in Service Broker and USE (BIND) these Services in the applications.

Provides Web Services and PUBLISHES these services to Web Service Broker/Registry

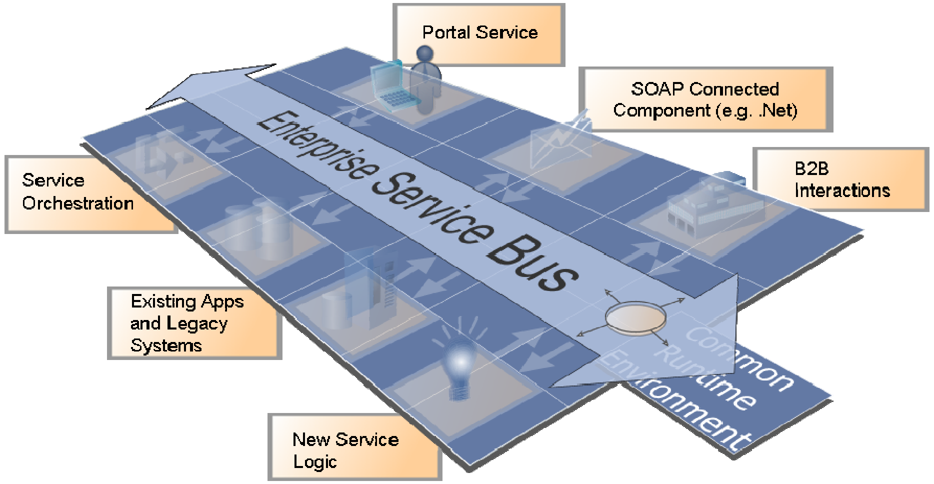
What is an Enterprise Service Bus?

An Enterprise Service Bus (ESB) is a flexible Infrastructure for services and application integration

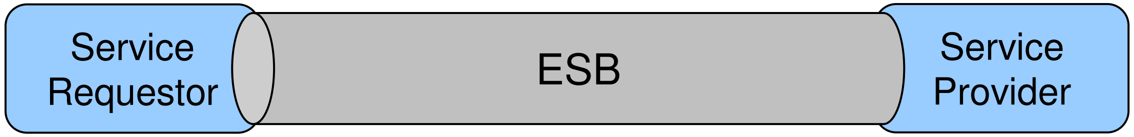
An ESB reduces the number, size and complexity of your interfaces in a SOA solution.

An ESB realizes following tasks between requestor und service

- ROUTING of messages between Services
- CONVERTING the transport protocol between requestor and service
- TRANSFORMING message formats between requestor and service
- HANDLING of business events between different types of services


Portal Service
SOAP Connected Component (e.g. .Net)
B2B Interactions
Service Orchestration
Enterprise Service Bus
Existing Apps and Legacy Systems
New Service Logic
Common Runtime Environment

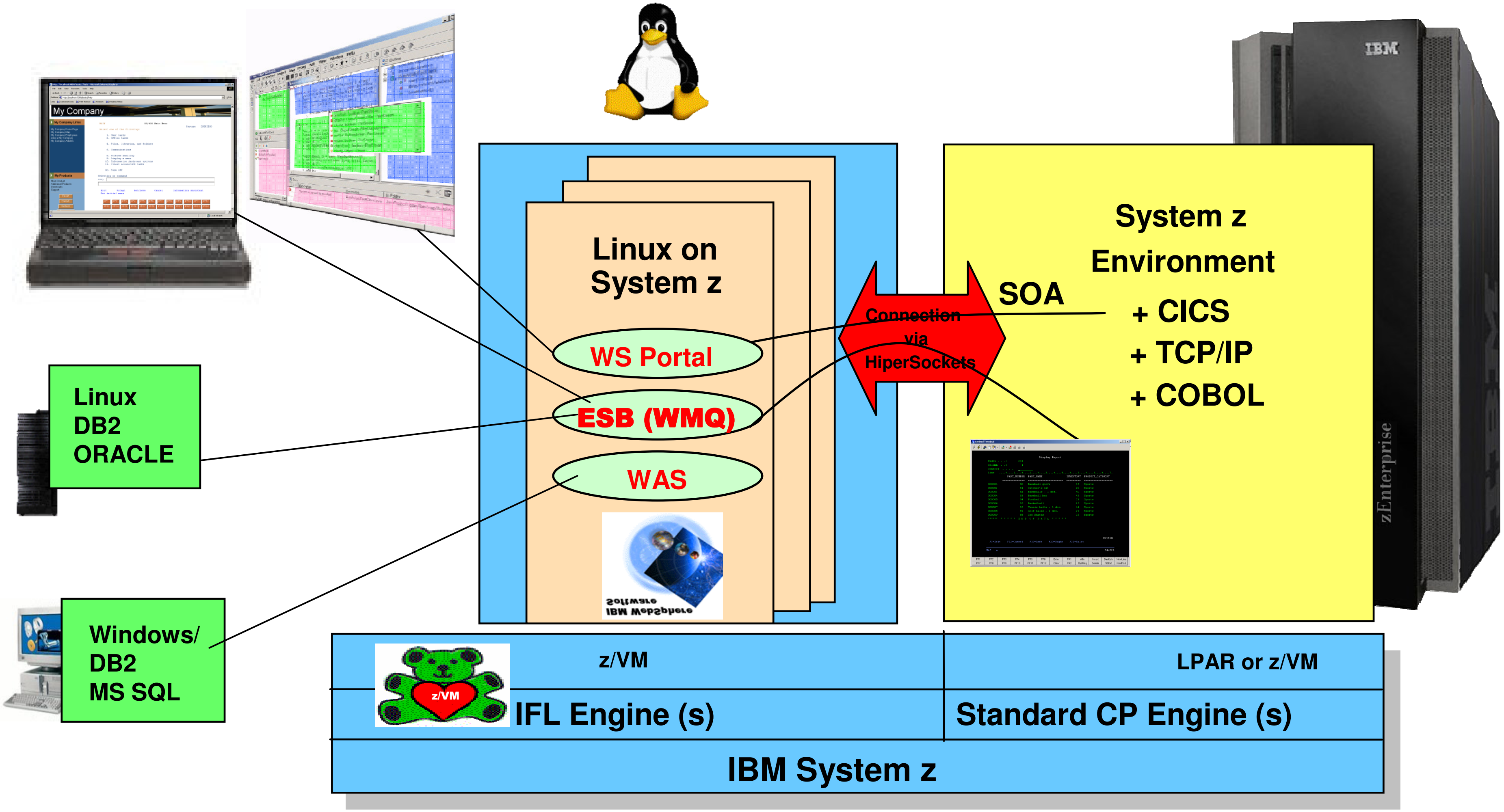
Core Principles of the ESB Architectural Pattern



- ESB inter-connects requestor and provider
  - Interactions are *decoupled*
  - Supports key SOA principle – *separation of concerns*
- ESB provides ***Service Virtualization*** of
  - *Identity* via routing
  - *Protocol* via conversion
  - *Interface* via transformation
- ESB also enables ***Aspect Oriented Connectivity***
  - Security
  - Management
  - Logging
  - Auditing

Linux on System z as Central Integration Point

Enterprise Service Bus (ESB) integrates applications high performant between platforms
- integrates, routes, aggregates and communicates based on rules and events



Why should VSE customers consider SOA?

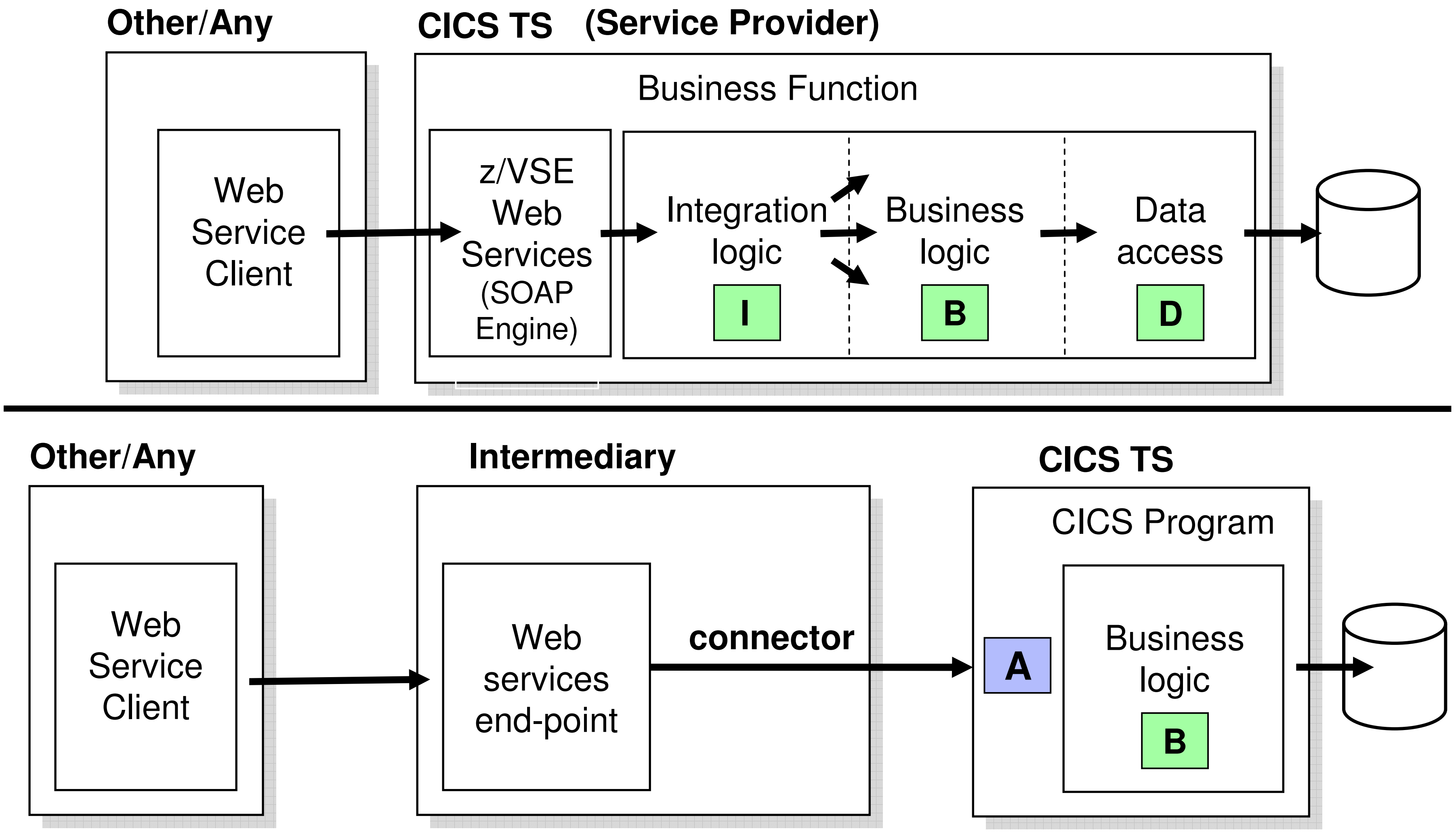
- **Easy integration of existing VSE programs and processes**
  - Reducing the interface complexity
  - Reuse of existing application logic as services
  - Use of standard protocols (XML, SOAP, HTTP)

- **Integration is platform independent**
  - independent of application programming language
  - independent of the data involved

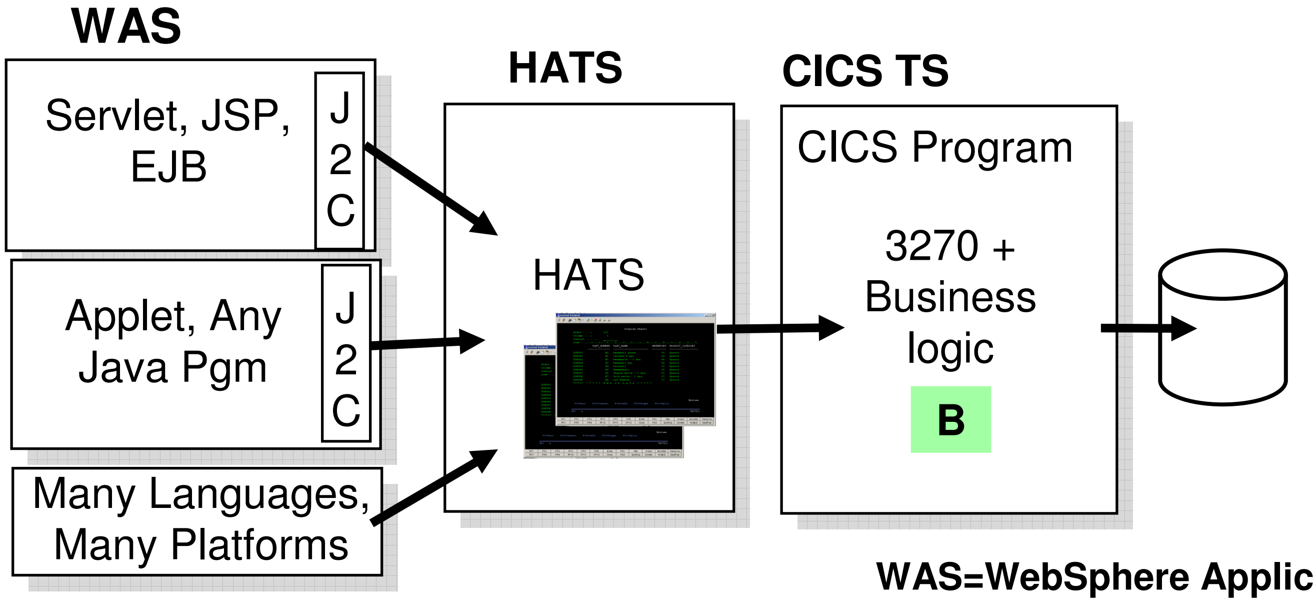
- **Integration of VSE into a Microsoft .Net environment**
  - without the use of Java
  - the most incompatible environments can be integrated

- **SOA enables the extension of VSE applications**
  - to other platforms and architectures
  - to partners and open world

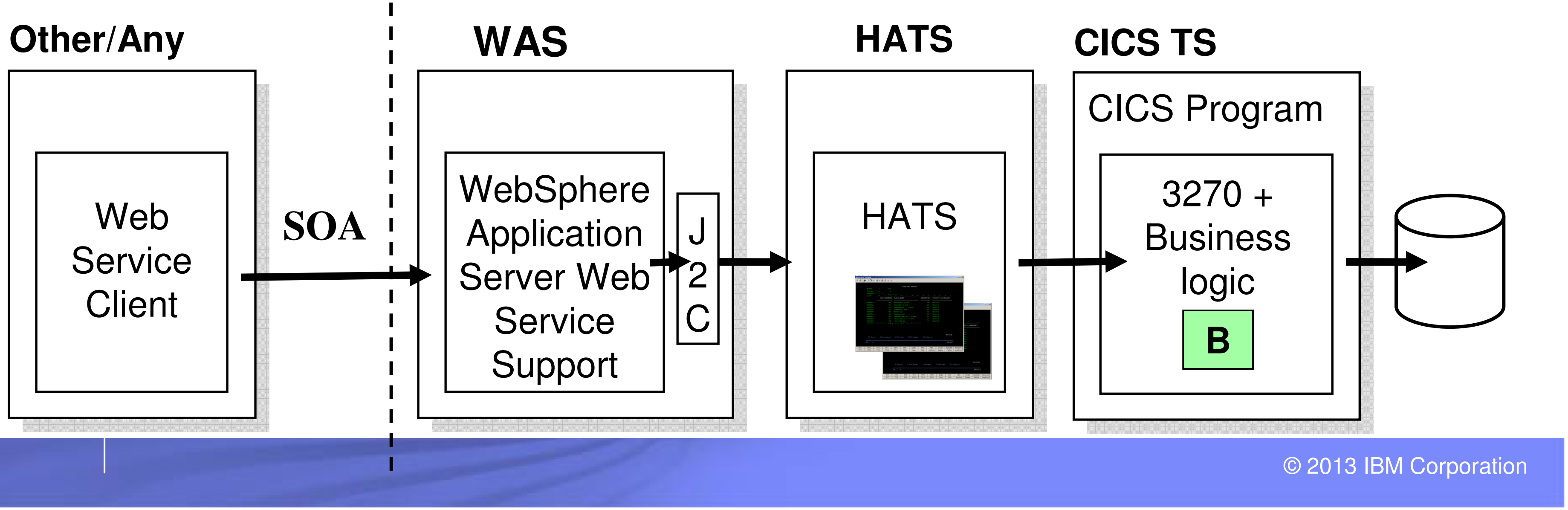
The Two CICS Models of SOA Integration



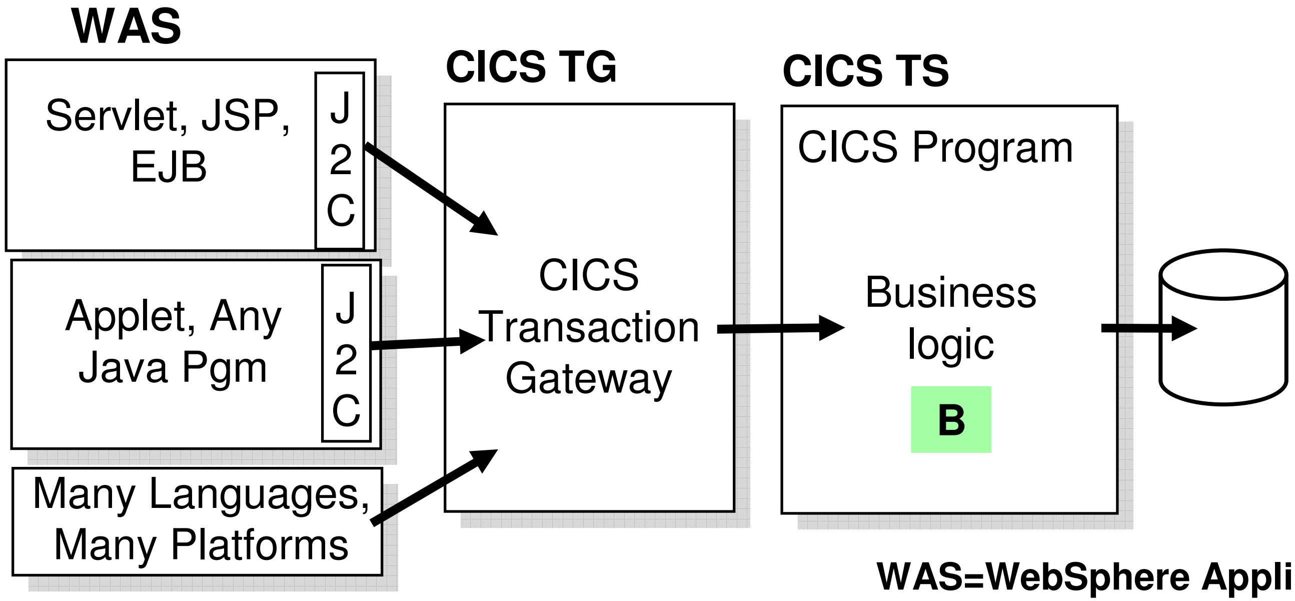
HATS Connector for Web Services Architecture



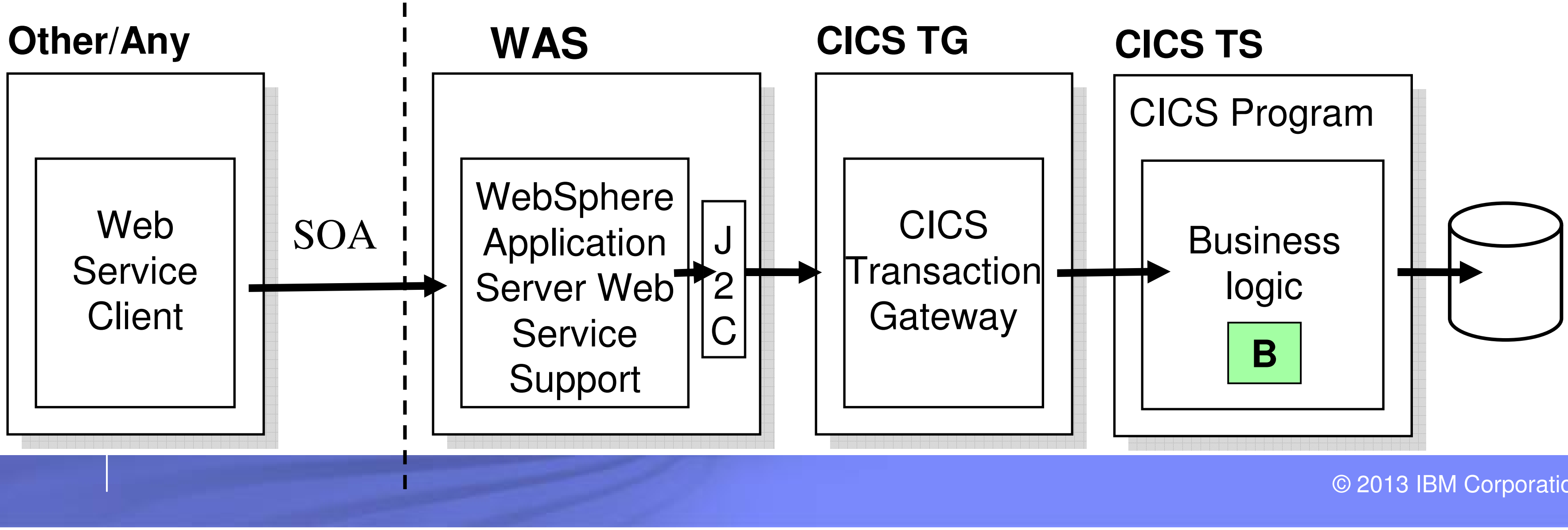
WAS=WebSphere Application Server



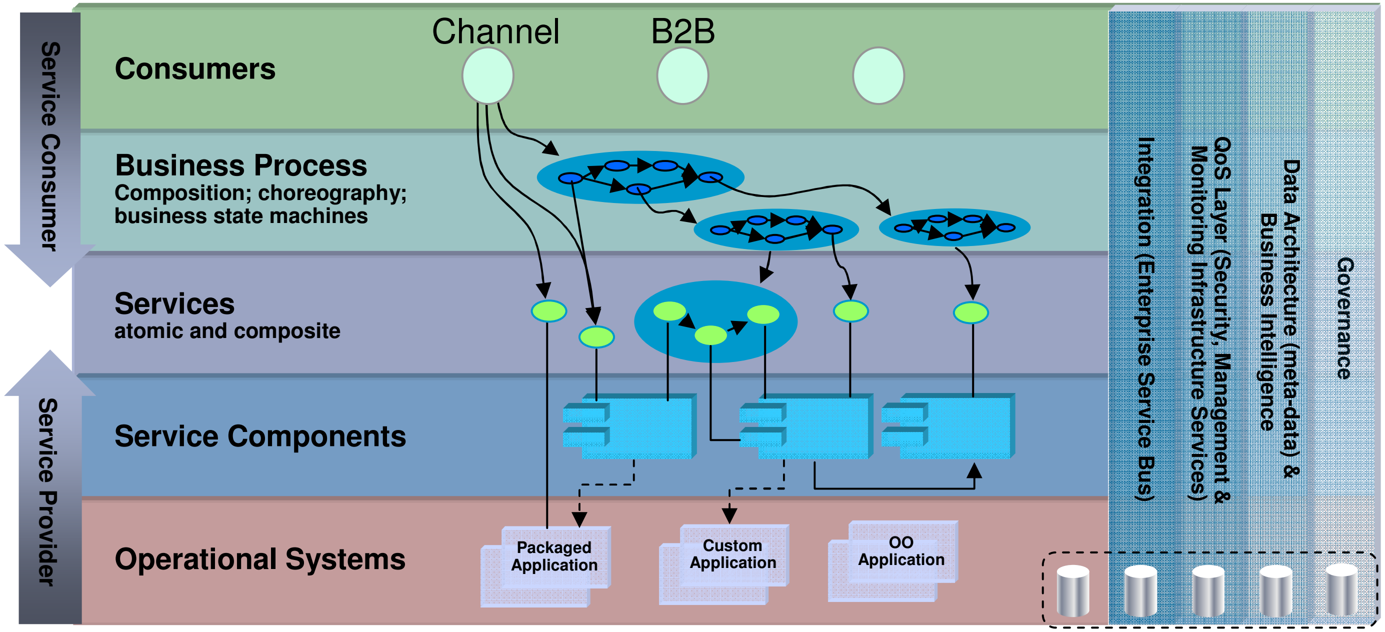
J2EE Connector Architecture via CICS TG



WAS=WebSphere Application Server



SOA Solution Layers



Service Consumer
Service Provider
Consumers
Channel
B2B
Business Process
Composition; choreography; business state machines
Services
atomic and composite
Service Components
Operational Systems
Packaged Application
Custom Application
OO Application
Integration (Enterprise Service Bus)
QoS Layer (Security, Management & Monitoring Infrastructure Services)
Data Architecture (meta-data) & Business Intelligence
Governance

Service Consumers

- ❑ Portal, B2B, Standalone, .Net

Business Process Layer

- ❑ BPEL Processes

Service Definition Layer

- ❑ WSDL, XML Schema, WS-Policy

Service Facade Layer

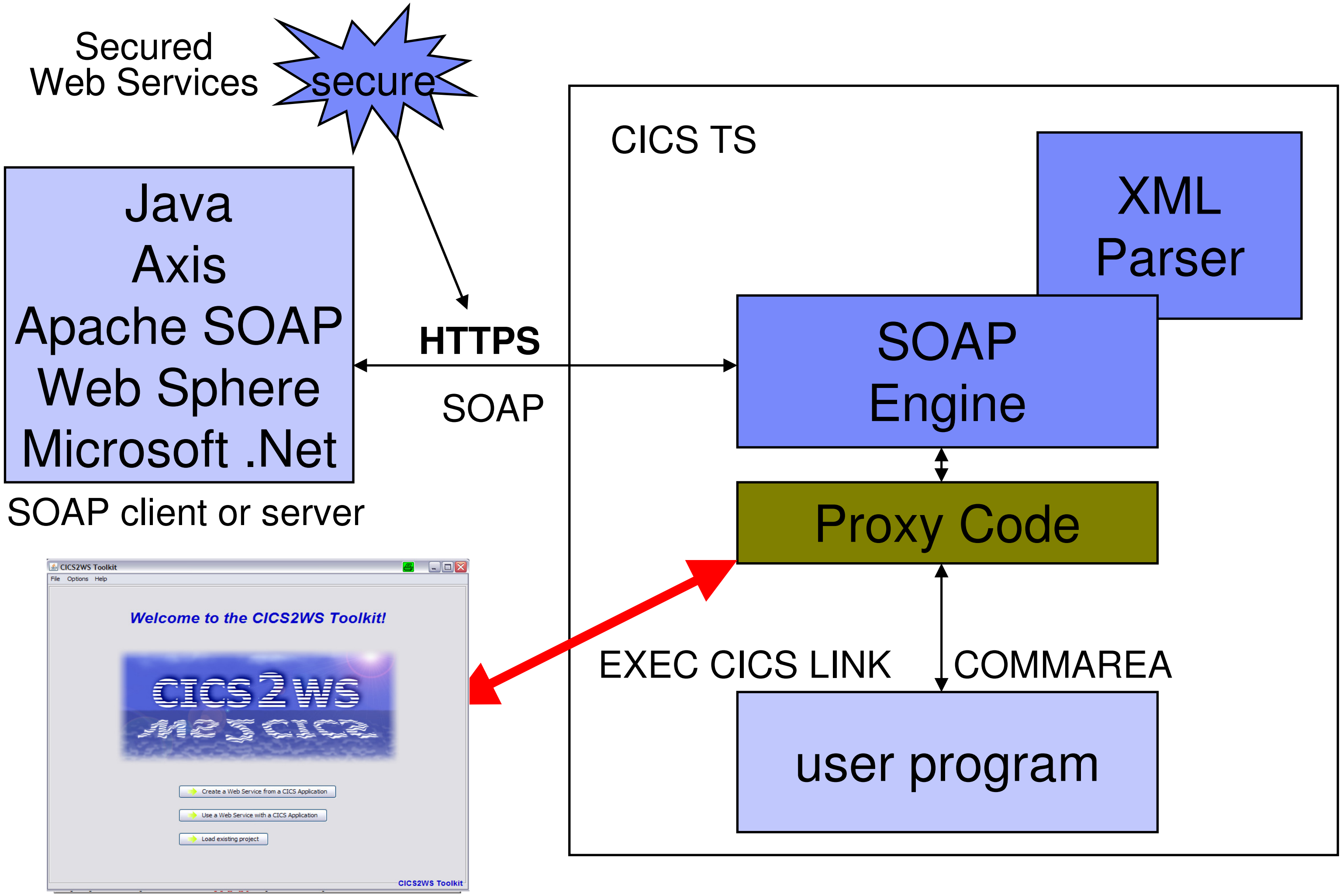
- ❑ Service Platform based service facades: J2EE, .Net, SCA etc.

Operational Systems Layer

- ❑ ISVs, Middleware, Custom Apps, Platforms, Network

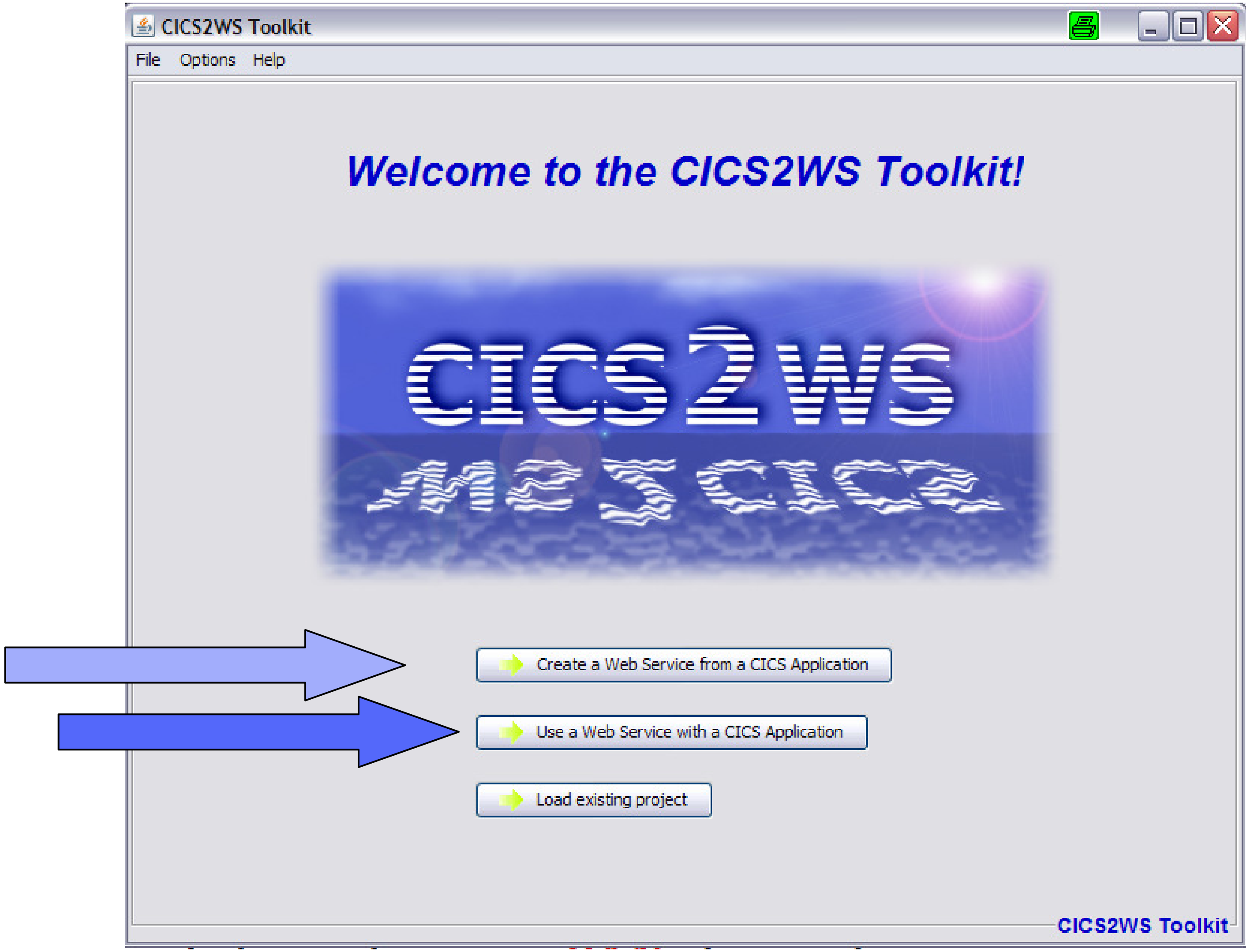
Atomic Service
Composite Service
Registry

Web Services in and with VSE



Tool to generate the Proxy code

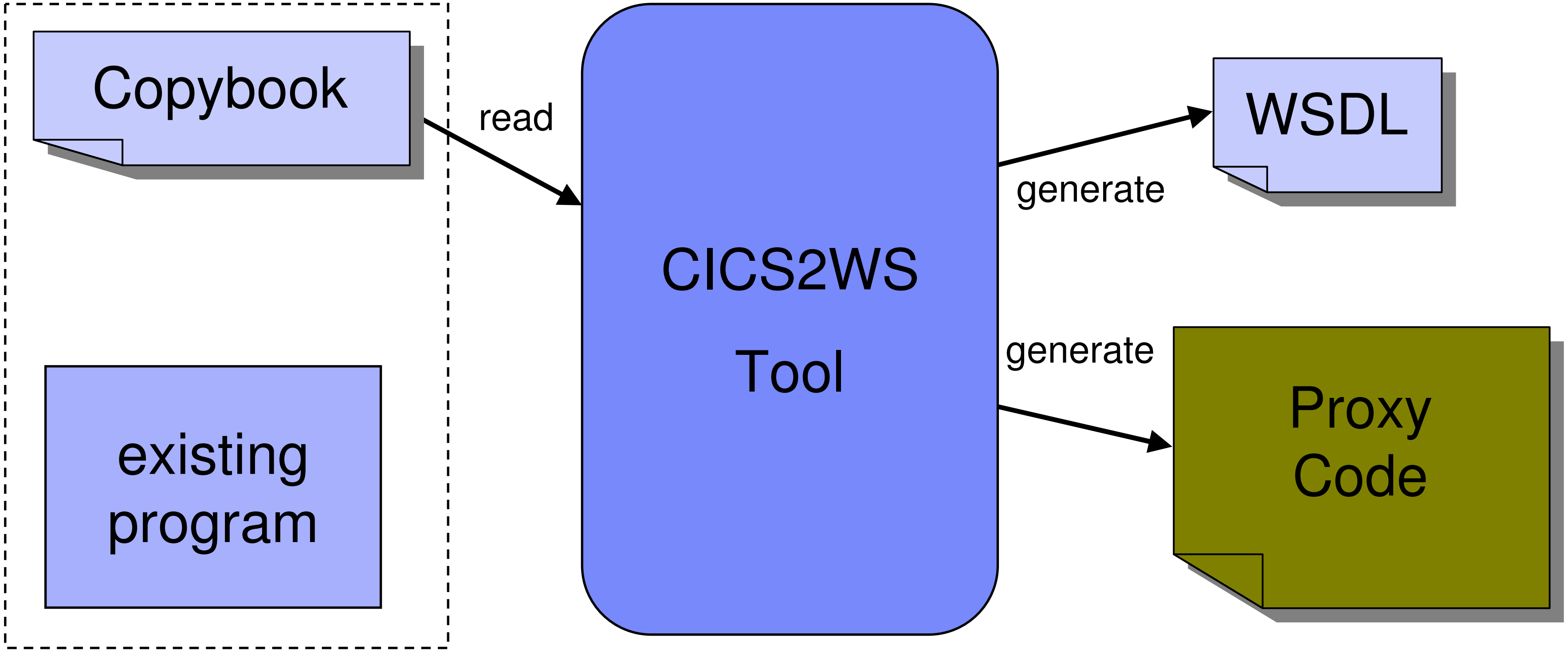
CICS to Web Services Tool



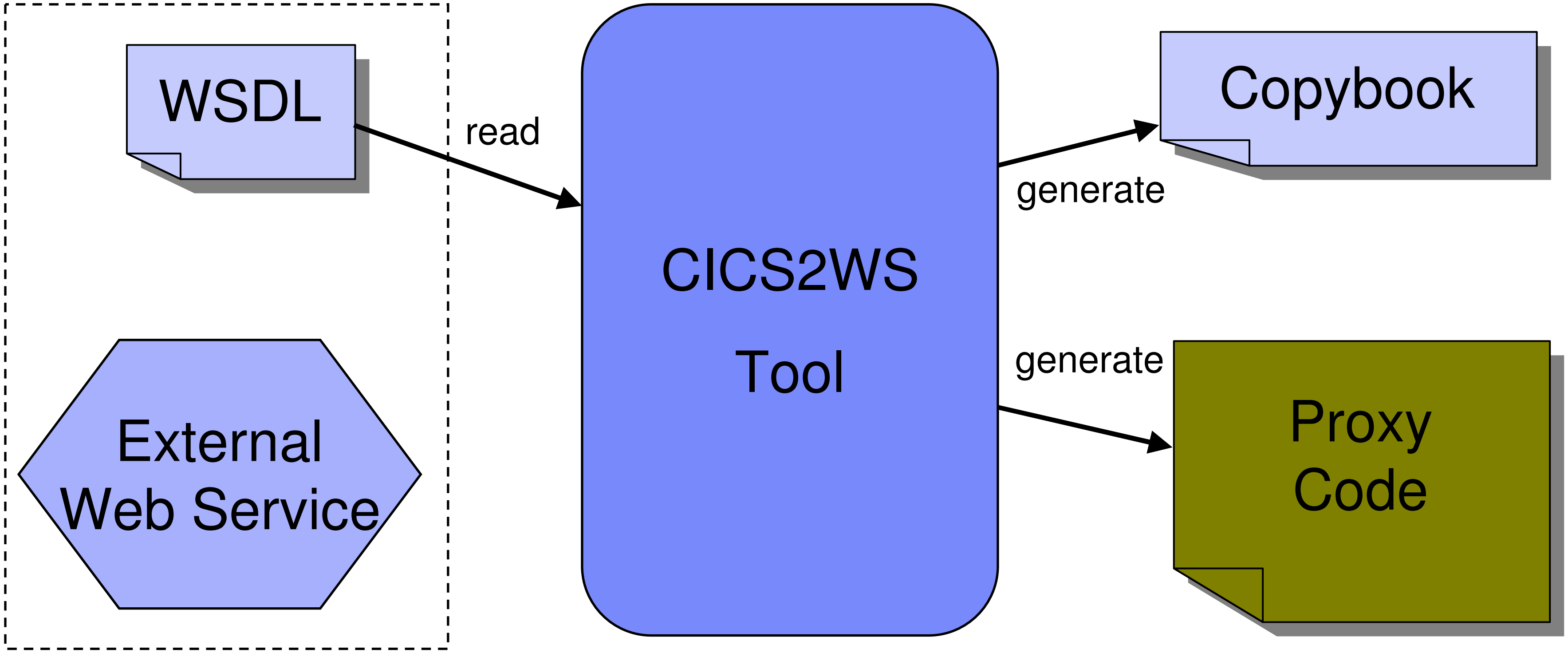
CICS2WS Toolkit
File
Options
Help
Welcome to the CICS2WS Toolkit!
CICS2WS
Create a Web Service from a CICS Application
Use a Web Service with a CICS Application
Load existing project
CICS2WS Toolkit

http://www-03.ibm.com/systems/z/os/zvse/downloads/

VSE as a SOAP server (service provider)



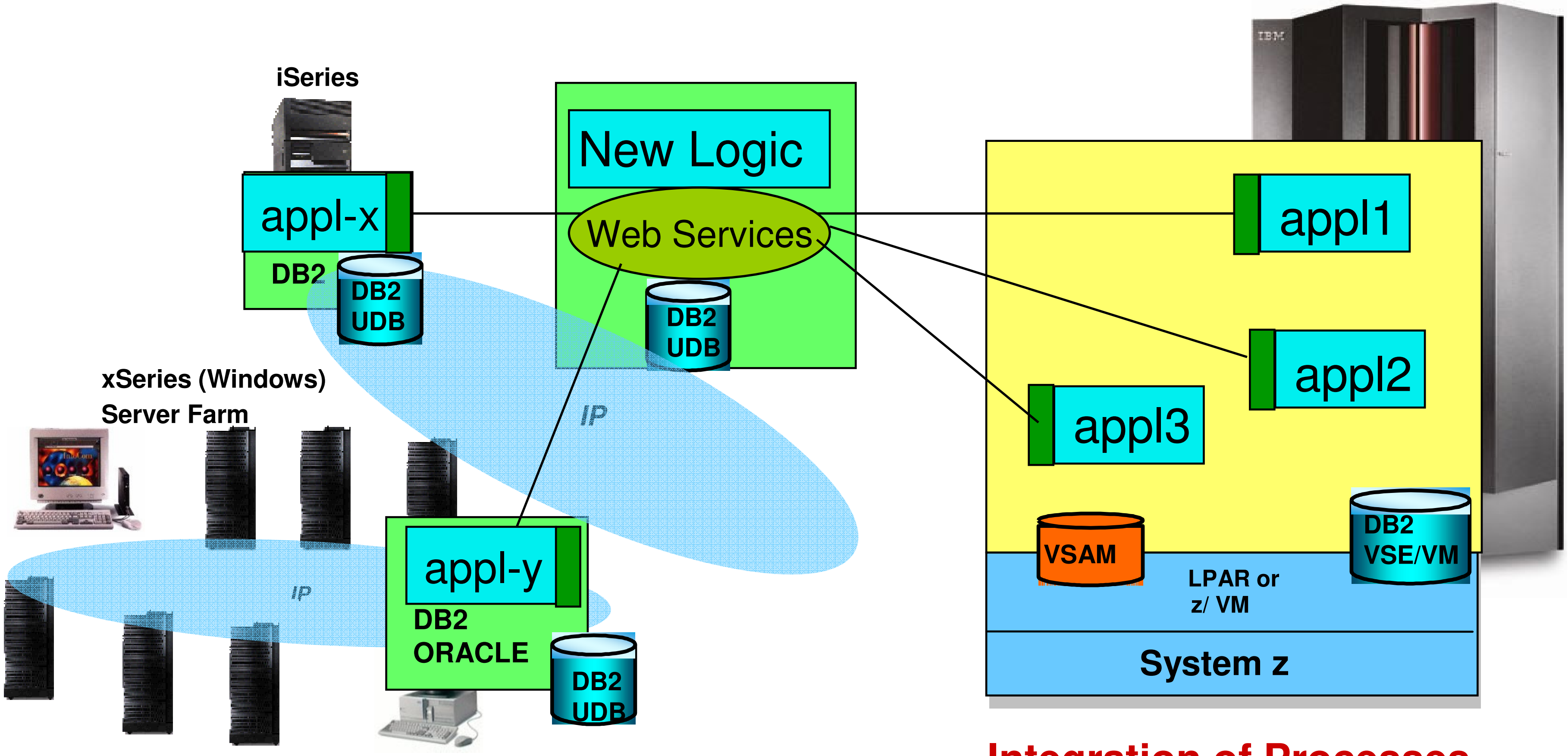
VSE as a SOAP client (service requestor)



SOA – the way to New applications and processes

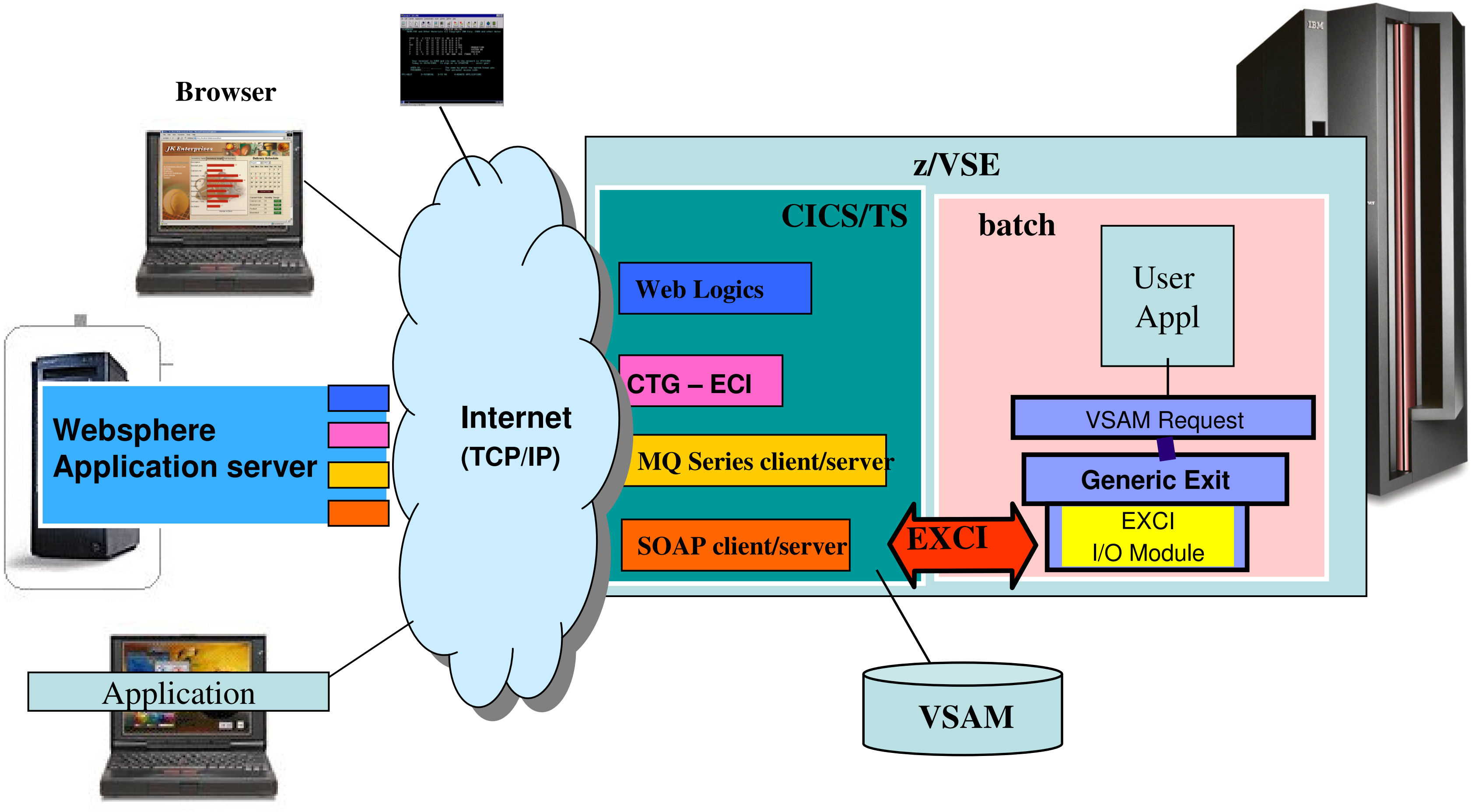
- Applications look the same for all users
- Core applications can be enhanced (independent of their language, COBOL, ASM, PL/I)
- New business logic is built

Increased success for the Company

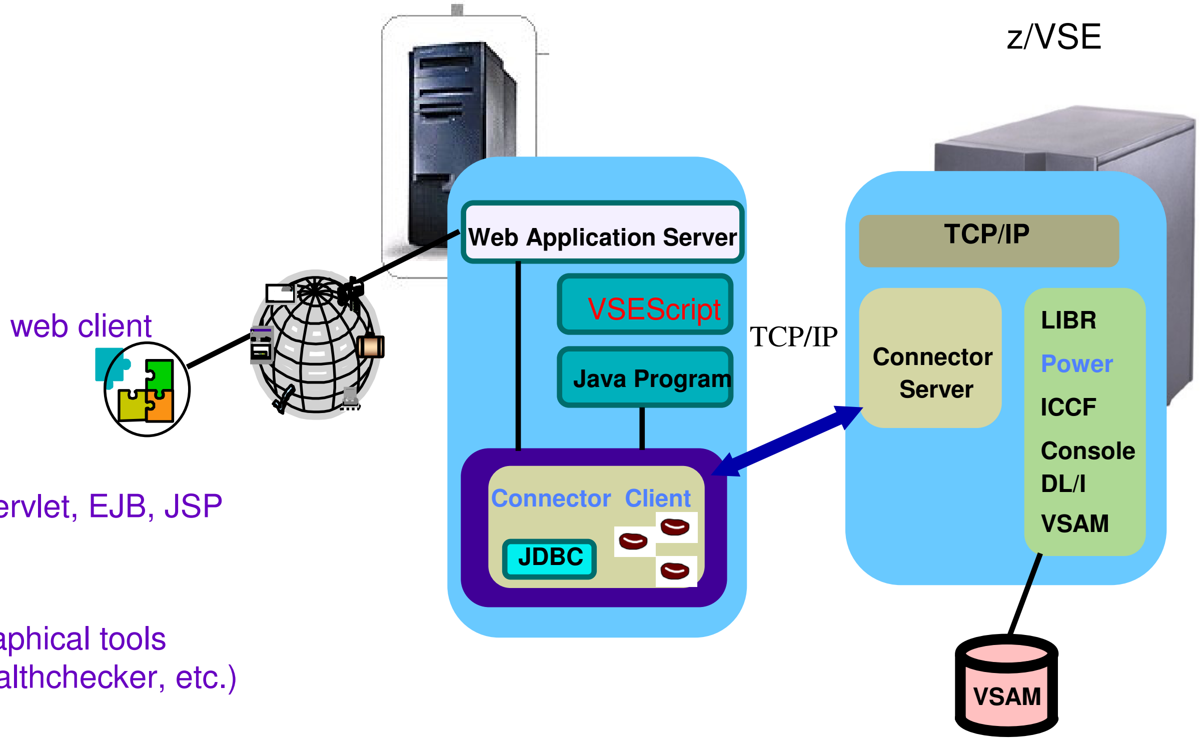


Integration of Processes

Inter-Communication with z/VSE batch Applications



Real Time start of batch workload with Java-Based Connector

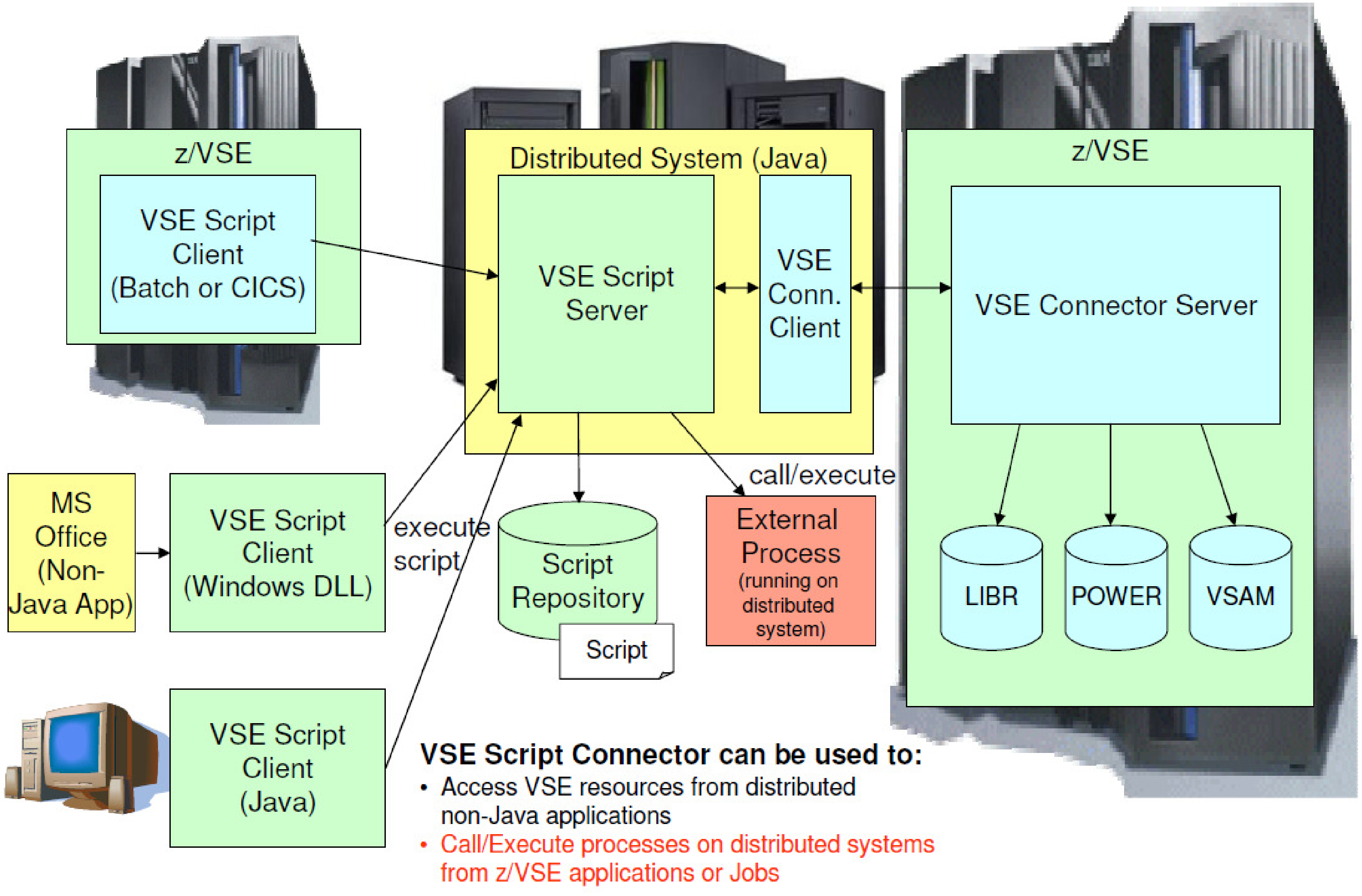


- HTML
- Java applet, servlet, EJB, JSP
- JDBC
- VSEScript
- specialized graphical tools (Navigator, Healthchecker, etc.)

- Real Time access to z/VSE Resources from remote or Web applications
  - Batch jobs can be started and results retrieved
  - Monitor and analyze using the Console or Power Java API

z/VSE interaction via Script Connector

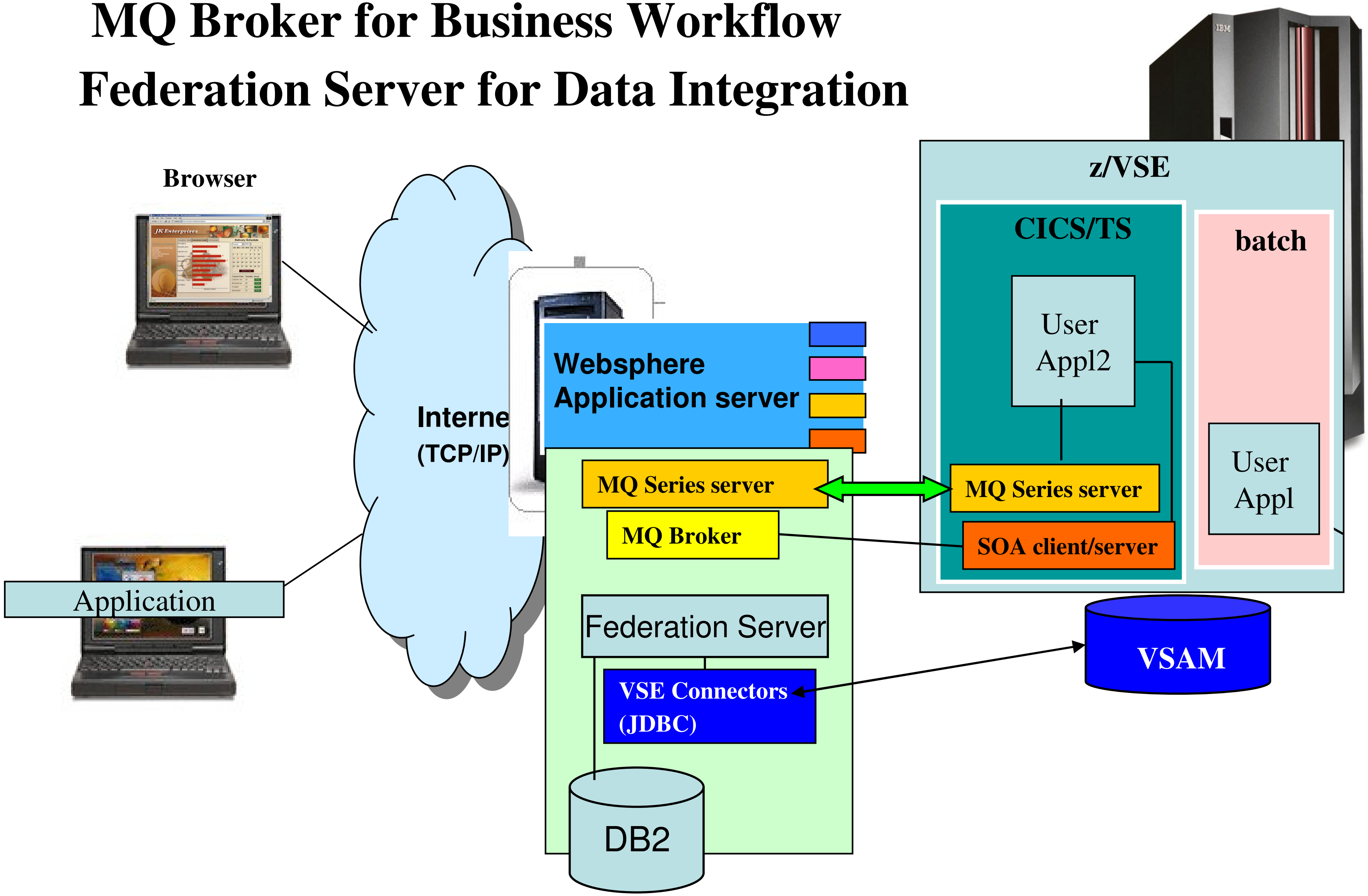
VSE Script Connector



VSE Script Connector can be used to:

- Access VSE resources from distributed non-Java applications
- Call/Execute processes on distributed systems from z/VSE applications or Jobs

Workflow: Connectors for Validation, MQ Broker for Business Workflow Federation Server for Data Integration



Reuse your transactional processing from CICS TS

Solution	Connector to use
Webify	▪CWS – CICS Web Support ▪HATS – Host Access Transformation Services ▪HOD – Host on Demand server
CICS application access from remote	▪CTG – CICS Transaction Gateway ▪HATS – Host Access Transformation Server ▪MQ Series (Client or Server)
SOA - Flexible, platform independent, CICS application integration, the most advance Application-to-application communication Method	▪Web Services – using XML data and SOAP protocol

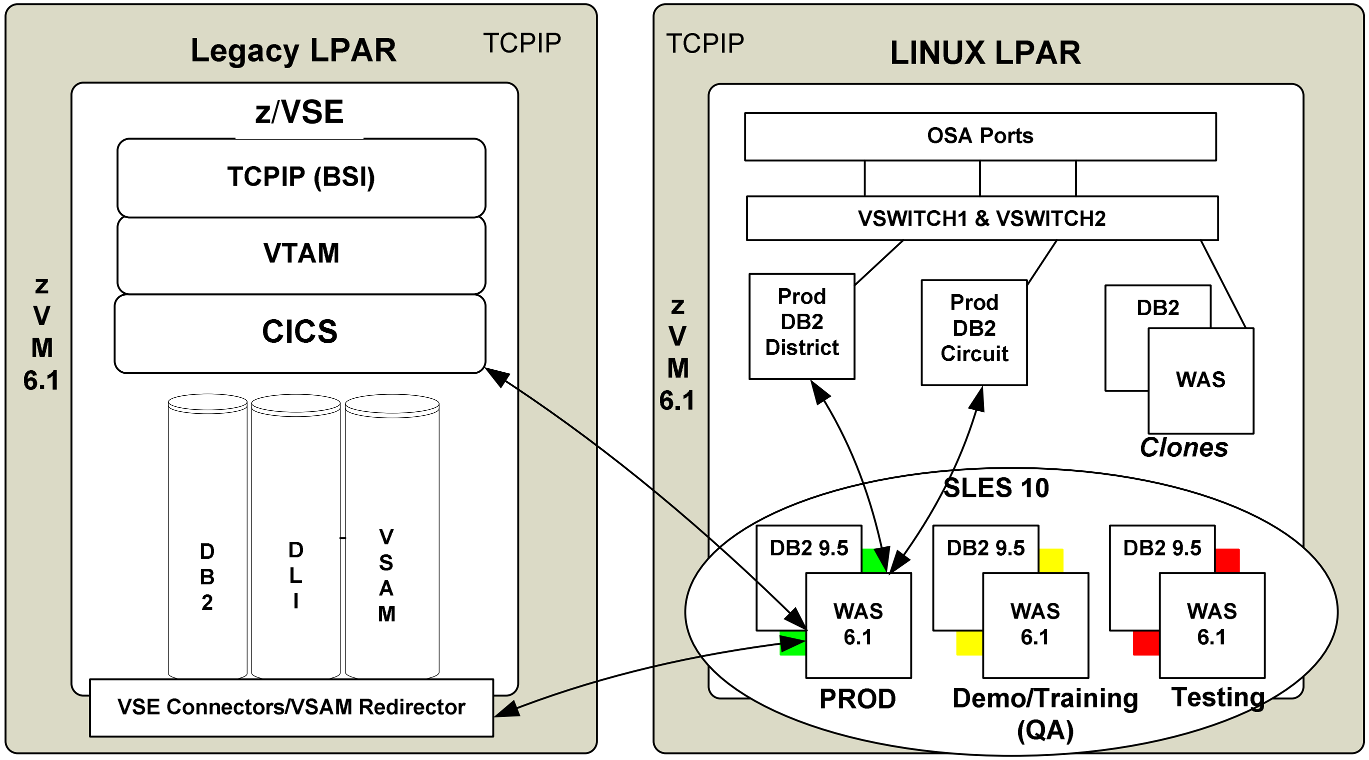
SOA Sample: Ball State University Indiana, USA „Student Address-System Integration“

- Business Need: Reduce Administrative effort
  - “Coordinate 40 name and address systems to streamline administrative processes and ensure information integrity for users”

- **SOA Solution: Building reusable components with SOA**
  - The IBM SOA solution runs in the mainframe environment utilizing **IBM CICS Transaction Server**. This version strengthens the application development capabilities of CICS and extends CICS applications to an SOA. In a scenario in which information is provided as a service and applications are decoupled rather than hard-coded together, CICS makes it possible for transactions to readily connect with multiple interfaces and repositories to provide authoritative data.
  - **IBM WebSphere Enterprise Service Bus**, works in conjunction with IBM WebSphere Application Server, to communicate between decoupled back and front ends, choosing the destination for a message and transforming it into the correct format.
  - **IBM WebSphere Host Access Transformation Server (HATS),** Version 7, generates Web Services Definition Language (WSDL) that is callable by BSU’s .NET front end to publish information to the user interface.
  - **IBM WebSphere Integration Developer** enables developers to assemble complex applications across the ESB by connecting reusable components.
  - **IBM DB2,** is the database for the SOA.

So, the advent of SOA and Web services is not just an evolutionary step, to me it’s a revolutionary step.”
–Fred Nay, IT Director, Ball State University

Customer Example: Supreme Court of Virginia



- 1 + 1 z10 BC L02
- 2 + 2 CPs
- 5 + 5 IFLs
- 112 + 112 GB memory
- 2 + 2 z/VM V6.1 LPARs
- 8 + 4 z/VSE V4.1 guests
- 73 + 24 SLES 10 SP2 guests
- WAS V6.1, DB2 V8.2, DB2 V9

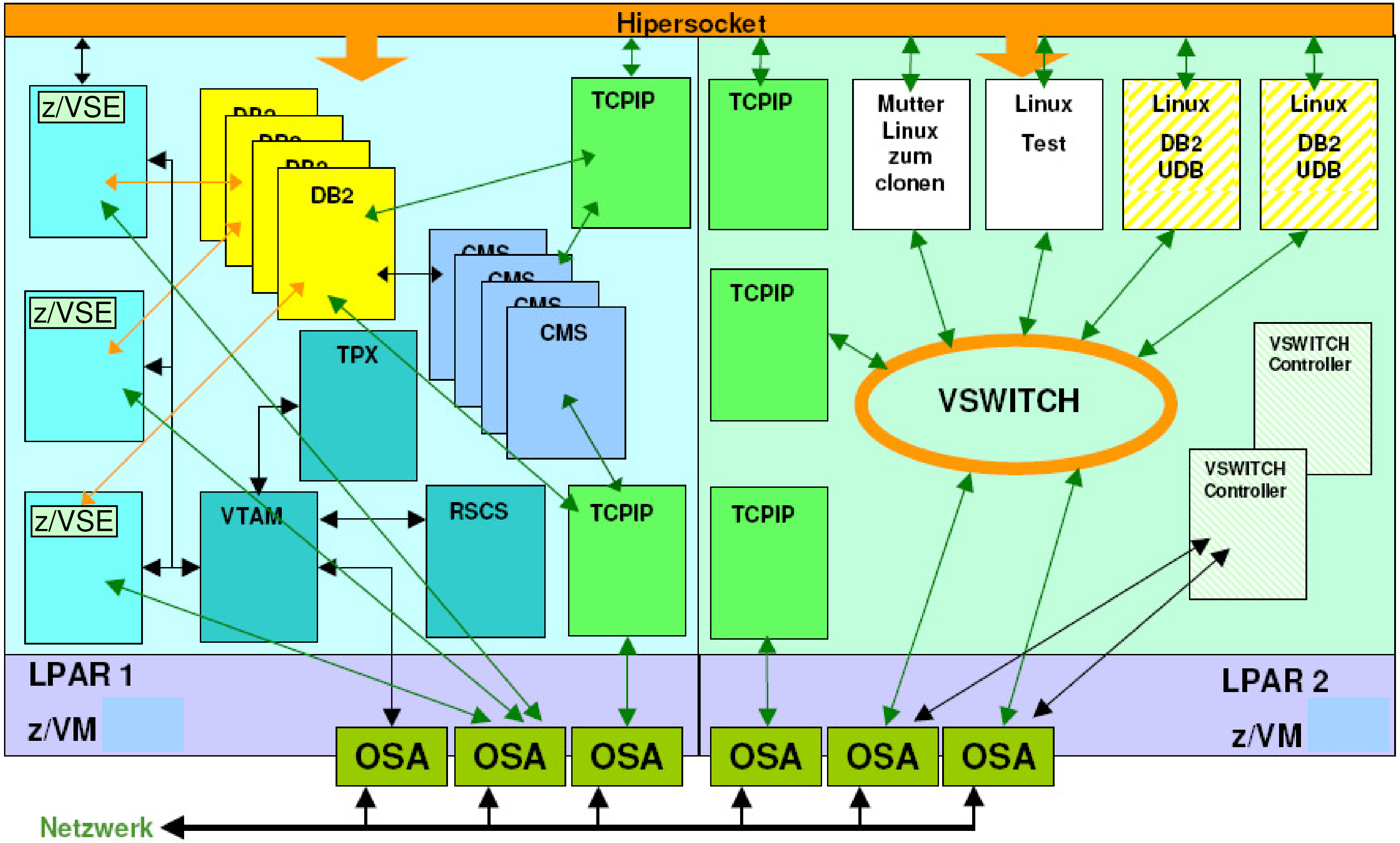
- **z10 BC L02 for Court System (internal)**
  - Serves 325 courts, 5.000+ users, 4.2 million new cases in 2009
  - Integrating z/VSE, DB2/UDB and WebSphere applications
  - eMagistrate* system serves 125 locations, 2.800 trans per day

**2007 ComputerWorld Honors Program Laureate*

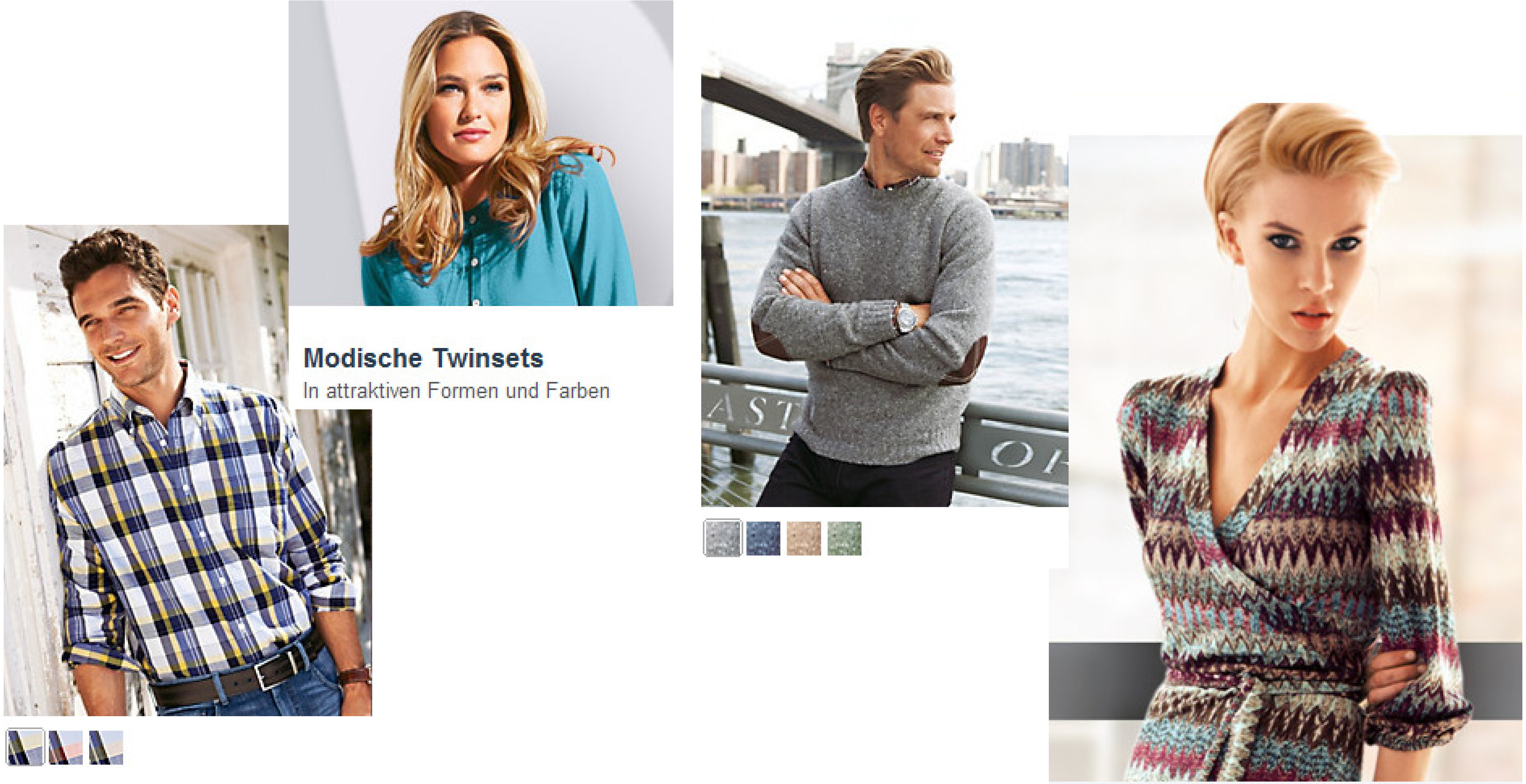
- **z10 BC L02 for Internet**
  - eCommerce application integrating z/VSE and WebSphere appls

Customer Architecture and implementation

EDV-Umgebung



Peter Hahn - Fashion with z/VSE and Linux environment



Questions?

Wilhelm Mild

IBM IT Architect

THE Open GROUP
Master
Certified IT Architect

IBM Deutschland Research
& Development GmbH
Schönaicher Strasse 220
71032 Böblingen, Germany

Office: +49 (0)7031-16-3796
mildw@de.ibm.com