#SHAREorg

SHARE
Technology · Connections · Results

CICS New News

Ian J Mitchell,
IBM Distinguished Engineer, CICS Portfolio Architect
IBM Hursley

Monday 12th August 2013
Session Number : 13822

SHARE in Boston

#SHAREorg

SHARE
Technology · Connections · Results

CICS New News

Ian J Mitchell,
IBM Distinguished Engineer, AIM System z CTO and
CICS Portfolio Architect

IBM Hursley

Monday 12th August 2013
Session Number : 13822

SHARE in Boston

Abstract

The first half of 2013 has been a busy time for the CICS development team, delivering two new feature packs for mobile and modern batch, the new z/OS Explorer, a new version of CICS with an alternative pricing model, not to mention statements of direction for PHP support and distributed security tokens, and of course a refreshed version of the CICS Developer Trial. With so much going on we wanted to take this opportunity to share more detail so that you can get the most out of these recent announcements, and understand how you might exploit these new capabilities in your business.

Please Note

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM's sole discretion.

Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

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.

Agenda

- New ways to get your CICS
  - CICS Value Unit Edition (VUE)
  - CICS Developer Trial
- FeaturePacks for V4.2 and v5.1
  - Modern Batch
  - SAML Security Tokens
  - Dynamic Scripting for v5.1
  - Mobile Extensions
- z/OS Explorer
- Performance and Capacity Questionnaire

CICS TS Value Unit Edition – *For new Java workload and service enablement*

Responding to customer demand for an alternative pricing structure

CICS One-time-charge

Alternative pricing model for new applications using a one-time-charge price metric*

zNALC pricing

*Reduced price for the z/OS operating system on LPARs that run a qualified application***

TS V5.1 avoiding SVC

CICS TS VUE is a separately licensed program and does not initiate Single Version Charging

* zNALC approval is required for each application

CICS TS Value Unit Edition – For new Java workload and service enablement

Responding to customer demand for an alternative pricing structure

New Java Applications*

Using the latest 64-bit, Java 7, multithreaded JVMSERVER architecture for maximum scalability

Service enablement*

Rapidly augment existing CICS applications using the latest CICS Java support and Java service enablement capabilities

Non-disruptive

Exploit the features of CICS TS V5.1 without having to upgrade existing back-end systems

* zNALC approval is required for each application

CICS TS Developer Trial. No cost. No fuss.

- Try before you buy the latest edition of CICS Transaction Server
- No charge license, no single version charging period, no reason to wait
- Easy ordering via IBM ShopzSeries, with internet or physical delivery options

Evaluate the Value *Explore the Capability* *Create the business case*

CICS TS Developer Trial

Try before you buy

Submit and track your CICS requirements online

- Transparent process with no middle-man between you and the lab
- Track the latest status of your CICS requirements online
- Private fields for information you only want to share with IBM

Discuss your requirements directly with the CICS developers

Email notifications to let you know when your CICS requirement is updated

Search, discuss and vote on other submitted CICS requirements

IBM RFE Community

Enhancing our future

CICS TS Early Programs

- Basic Disclosure and Discussions
- Beta programs (varying levels of commitment and involvement available)
- Customer Architecture Forum - CAF - (remote throughout the year, and on-site in Hursley once or twice a year)
- Design Partnerships - a close relationship on a specific area of technology where there is mutual interest

Feedback to the Hursley Lab

Work directly with the CICS developers

Get a time to market advantage

CICS TS Early Programs

Be a part of the future

Managing the Batch Window with Modern Batch

Agenda

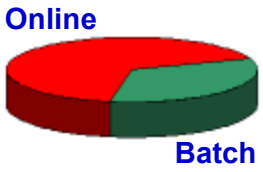
- Business Pressures on Traditional Batch
- IBM WebSphere Java Batch Overview
- IBM CICS TS Modern Batch Feature Pack
- Wrap-Up Summary

Business Pressures on Traditional Batch

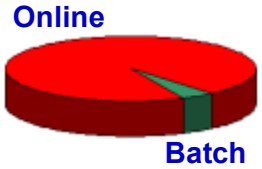
Concept of "Dedicated Batch" Window Going Away

Windows of time which used to be dedicated to batch processing are shrinking.
The demands of online processing require more and more ...

In the past ...



Today ...



24 x 7 x 365 Access

Users of your online systems expect availability at all hours

Users from other parts of the world means availability is expected around the clock

Mobile Users

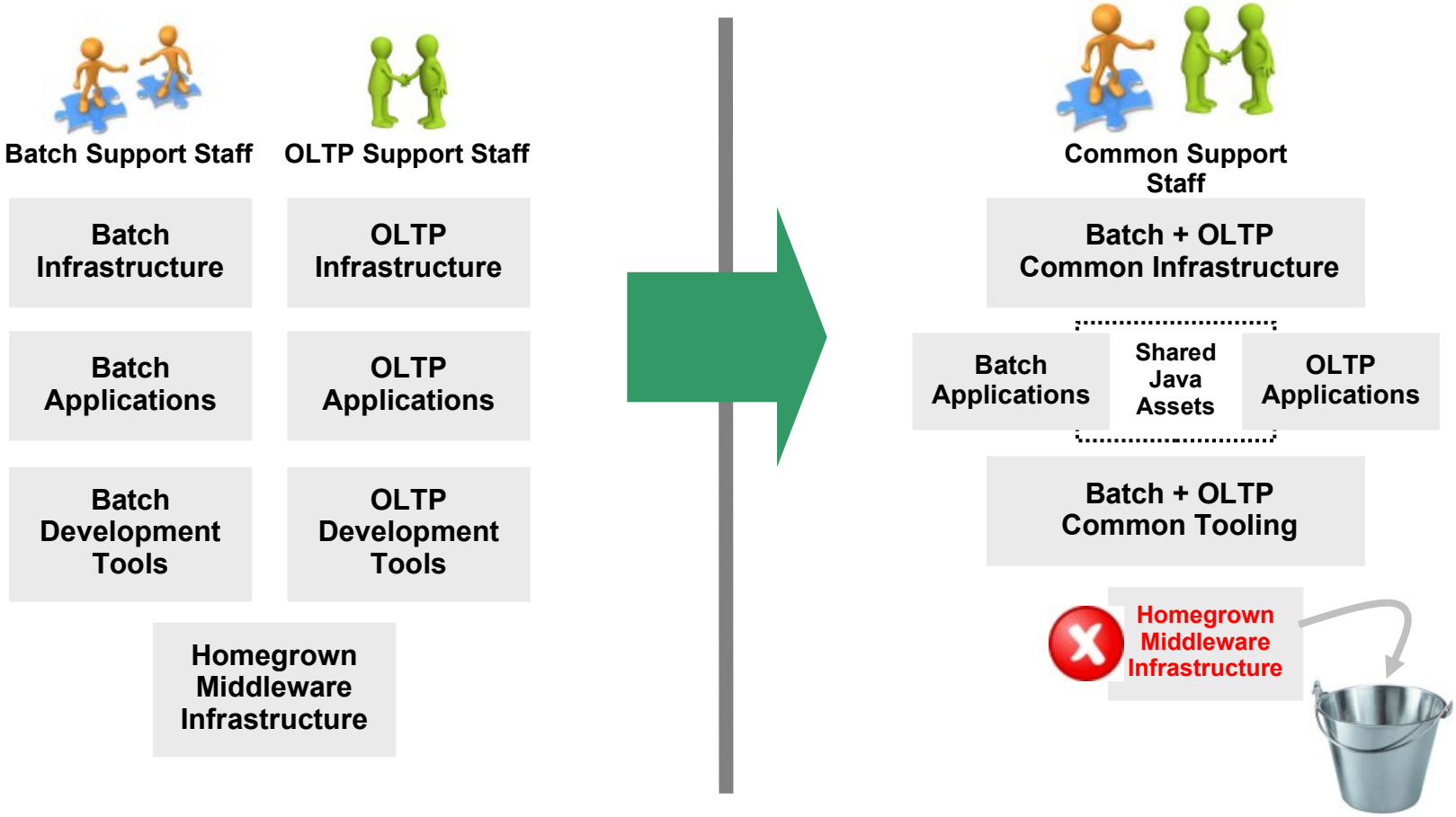
Users are no longer tied to a desk and a computer. Today users have access to mobile computing devices that are with the user wherever they may be. Day or night, home or office.

The need to process batch work has *not* gone away.

The need to perform the work concurrent with OLTP has emerged.

The Value of Shared Services

It's not *just* that the window is shrinking ... it's also the cost pressures on maintaining the batch and OLTP environments:



Efficiencies through consolidation around common assets

Java for Batch Processing?

Yes ... for many very good reasons:

Availability of Skills

Java is a programming language with wide adoption in the industry. Skills for Java programming are common and affordable.

Tooling Support

Development tooling for Java has advanced to the point where some tools (IBM Rational Application Developer) are very powerful and sophisticated.

This also provides an opportunity to consolidate to a common tooling environment for both OLTP and batch development.

z/OS Specialty Engines

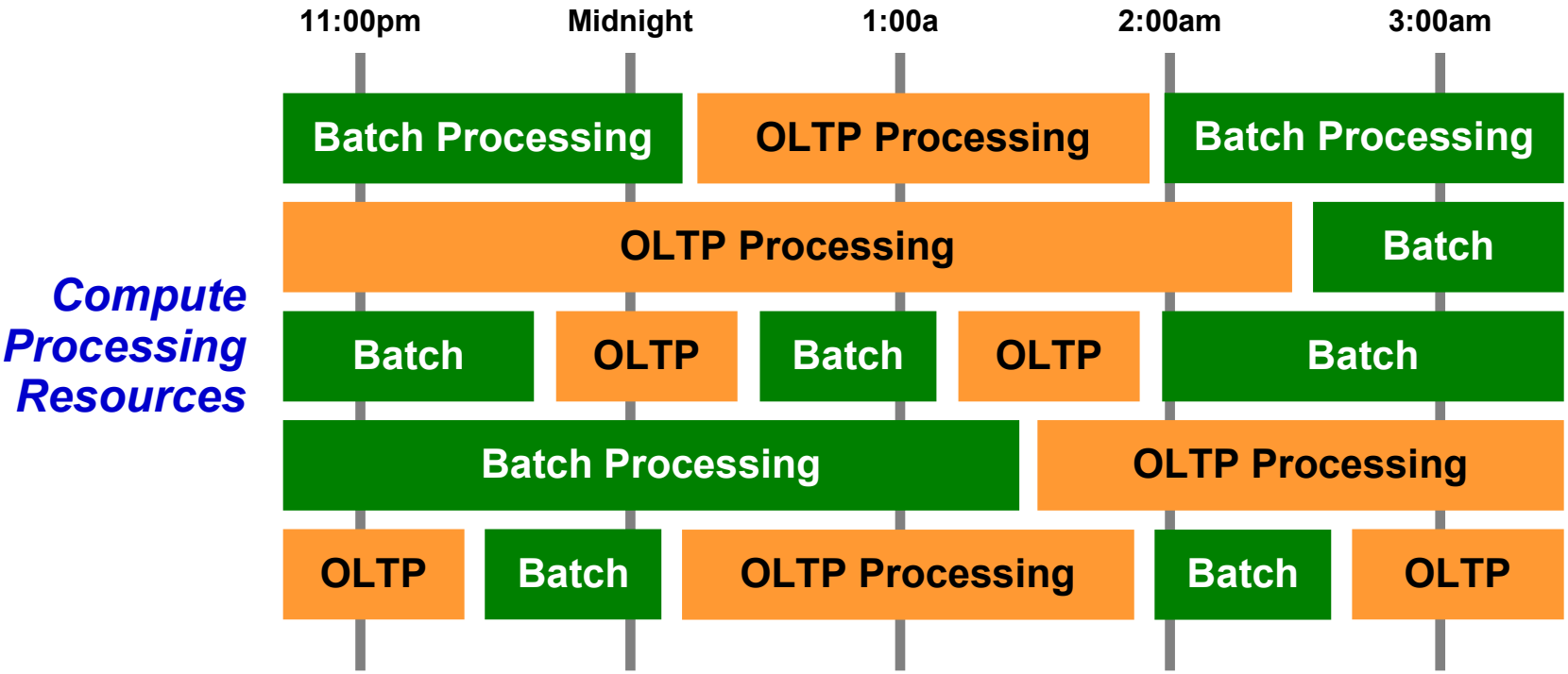
Pressures on cost containment often dictate greater use of z/OS specialty engines. Java offloads to zAAP. Java batch does as well.

Processing in OLTP Runtime

Running Java batch in the same execution runtime as Java OLTP provides an opportunity to mix and manage the two processing types together under the same management model.

The Objective -- OLTP and Batch Mixed and Managed:

OLTP and Batch do not need to be "either / or" ... it can be "both":



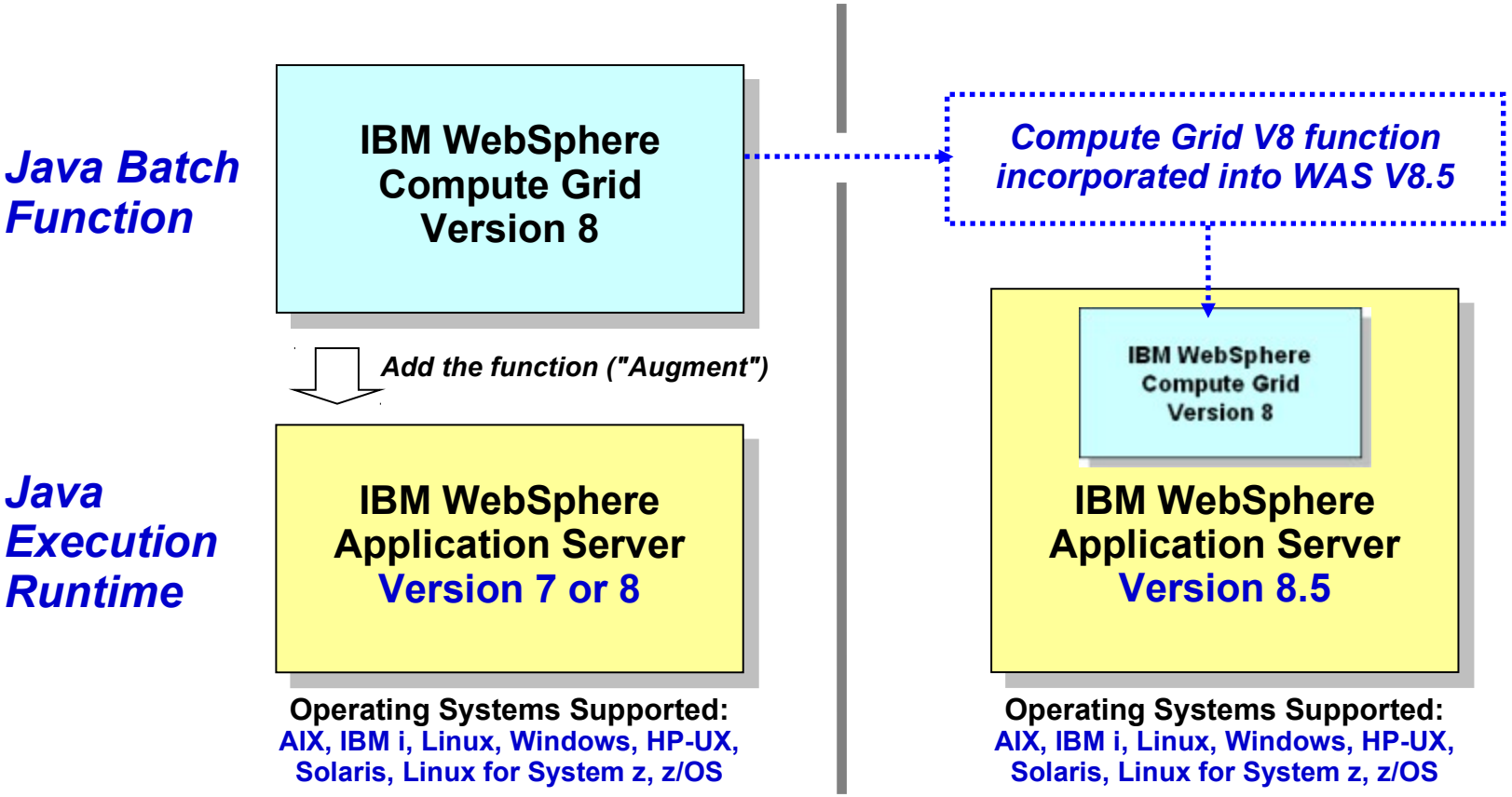
With IBM WebSphere Batch this is possible. OLTP and Batch processing within a common execution runtime (WebSphere Application Server) allows the WAS platform to mix and manage the two workload types.

Overview

A high-level look at the IBM WebSphere Java Batch model

IBM Compute Grid V8 and IBM WAS V8.5

The IBM WebSphere Java Batch function is provided in two ways today:

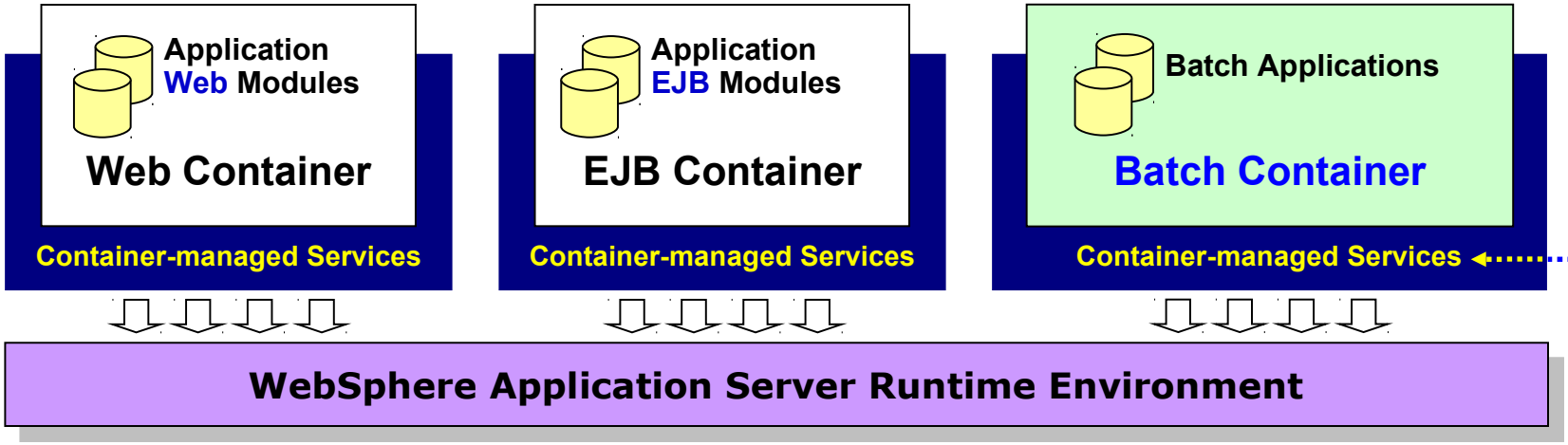


Function is identical between the two environments

Compute Grid V8 available for those who have not yet migrated their execution runtimes to WAS V8.5

Batch Container Added to the WAS Runtime

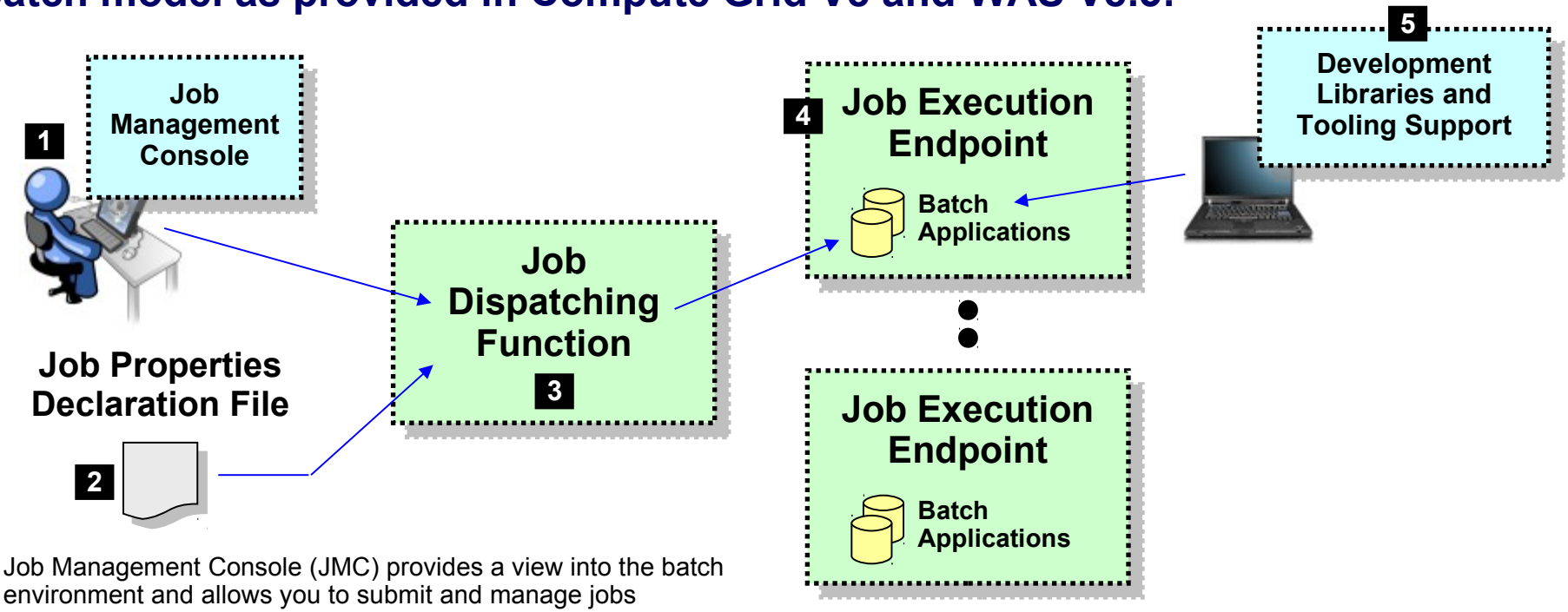
At a very high-level, you may think the IBM WebSphere Java Batch function as a "batch container" operating alongside the other containers of WAS itself:



Batch job dispatching and management system	Job resiliency services (skip record, step retry)
Data record read and write support services	Parallel job management and execution services
Checkpoint and job restart services	COBOL module call services

Overview of the Management and Execution Model

This picture illustrates some of the key components of the WebSphere Java Batch model as provided in Compute Grid V8 and WAS V8.5:



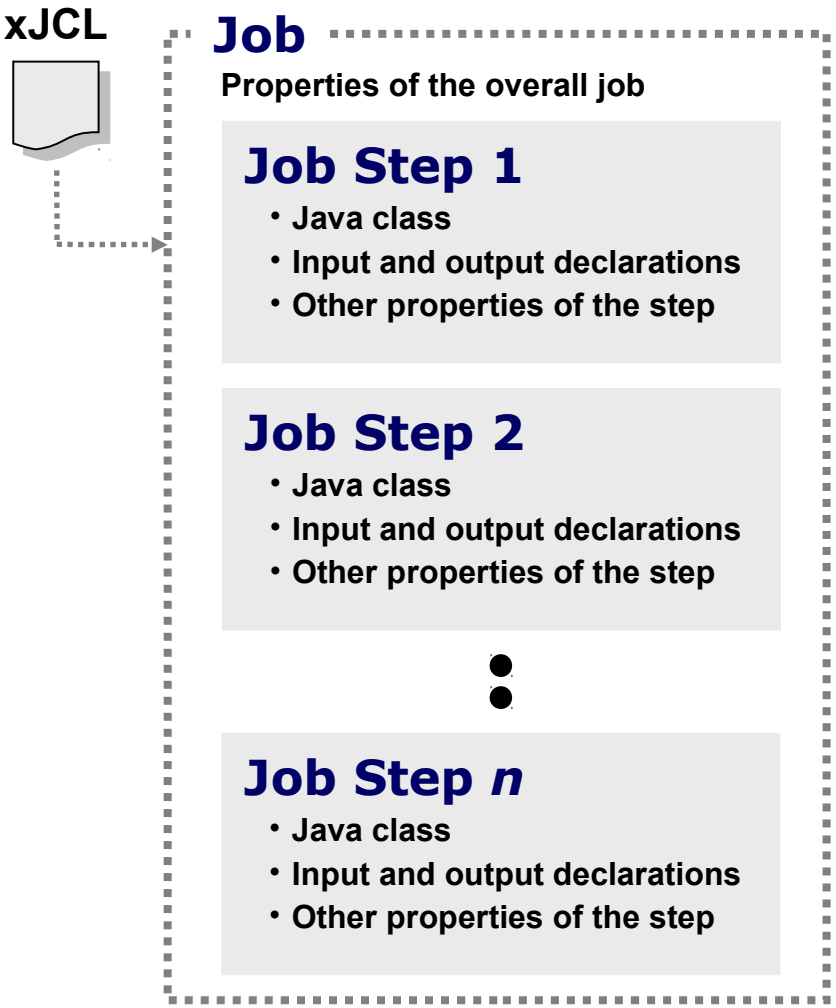
1. Job Management Console (JMC) provides a view into the batch environment and allows you to submit and manage jobs
2. Job declaration file (xJCL) provides information about the job to be run, such as the steps, the data input and output streams and the batch class files to invoke
3. The Job Dispatching function interprets the xJCL, dispatches the job to the endpoint where the batch application resides, and provides ability to stop and restart jobs
4. The Execution Endpoint is a WAS server in which the deployed batch applications run
5. The development libraries and tooling assist in the creation of the batch applications

A comprehensive Java batch execution platform

Built on the proven Java runtime environment of WebSphere Application Server

Batch Job and Batch Job Steps

A batch job consists of one or more steps executed in order specified in xJCL:



The xJCL is submitted through the Job Management Console

Interfaces provided: HTTP browser, command Line, Web Services, RMI

The Job Dispatching function interprets xJCL and determines which endpoint has batch application class files deployed

Dispatching Function invokes job and passes to the endpoint an object containing all the properties in xJCL

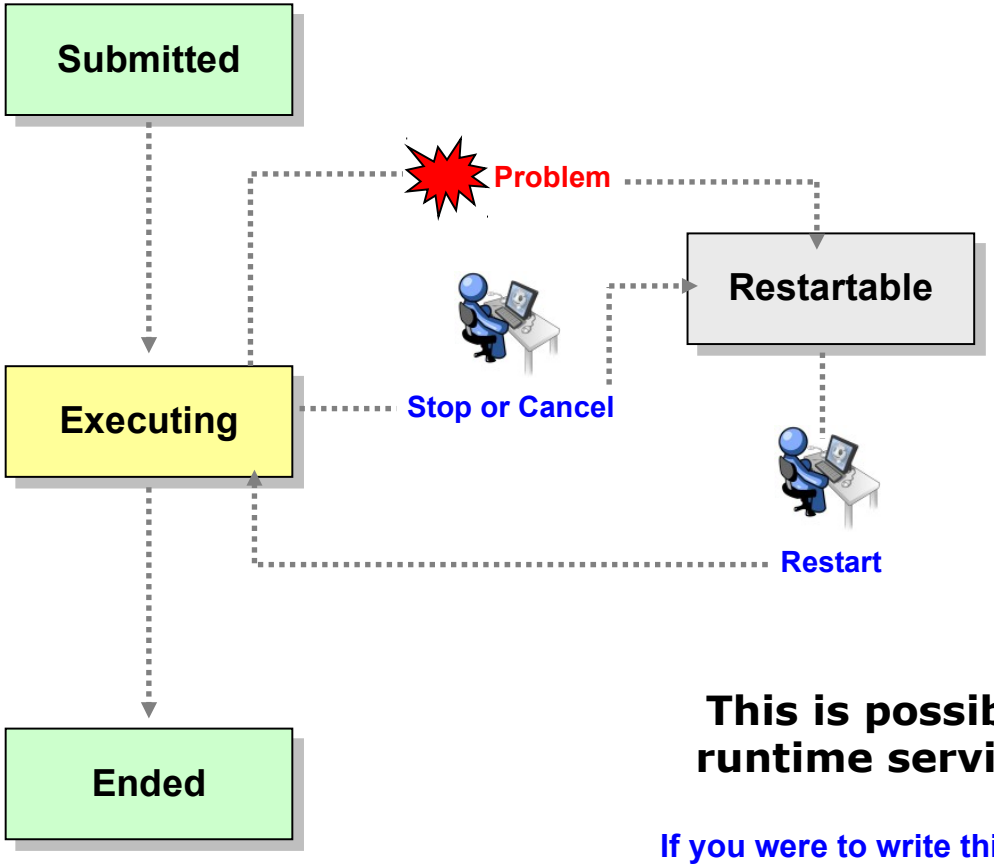
Steps are executed in order, with conditional step processing if declared

Dispatching Function maintains awareness of job state

When job ends, job output file accessible through Job Management Console

Job Execution "State"

The following picture illustrates a simplified view of the job states ...
it helps illustrate a key point: *executing jobs can be acted upon; failed jobs restarted.*



The Job Management Console provides you ability to act upon an executing job

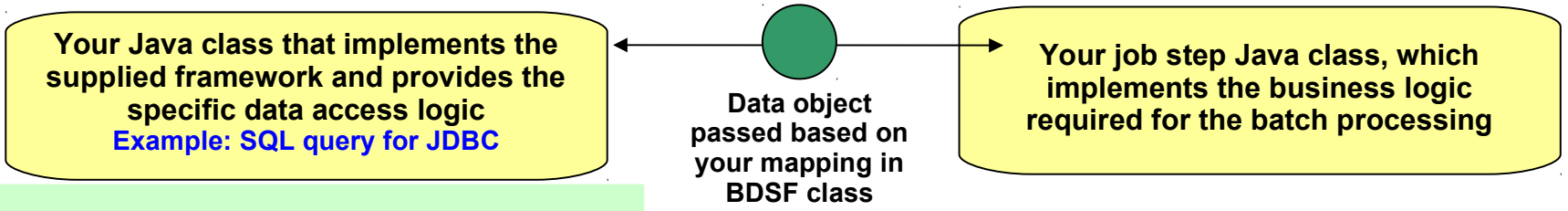
The Batch Container is maintaining checkpoint status and will restart at the last checkpoint interval

This is possible because of the Java batch runtime services that are part of the batch container model

If you were to write this yourself then just what's shown here would require a significant amount of custom batch middleware code. IBM WebSphere Java Batch provides that as part of the product.

Batch Data Stream Framework (BDSF)

This is a key function service provided by the batch container -
it abstracts data read and write operations so your code may focus on the business logic:



Batch Data Stream Framework

Supplied "patterns" for data access:

- JDBC read or write operations
- JPA read or write operations
- File read or write operations
- z/OS Data Set read or write operations

Batch Data Stream retrieves result set from data persistence store (DB, file, etc.)

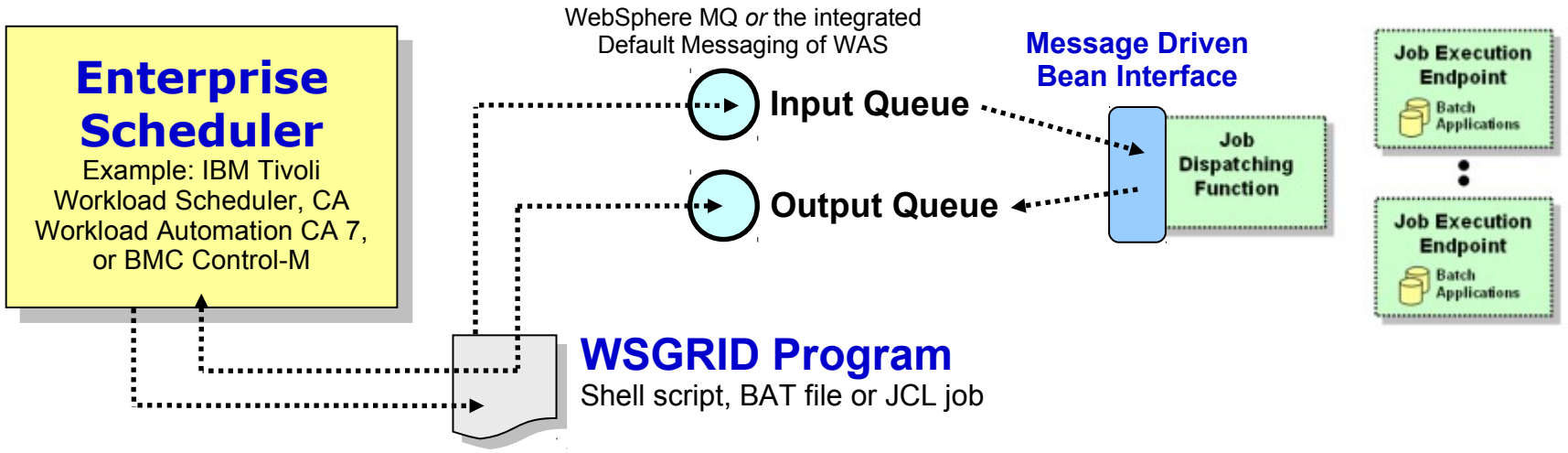
Batch Data Stream maps data fields to data object

For each record in result set, BDSF invokes your job step, passing a data object mapped to your specifications

Your job step code stays focused on business logic, not Java stream handling and data object formatting

Integration with Enterprise Scheduler Functions

The Job Dispatching Function has a Message Driven Bean (MDB) interface.
IBM supplies a program that integrates schedulers with WebSphere Java Batch:



WSGRID is seen by Scheduler as any other batch job it starts and monitors

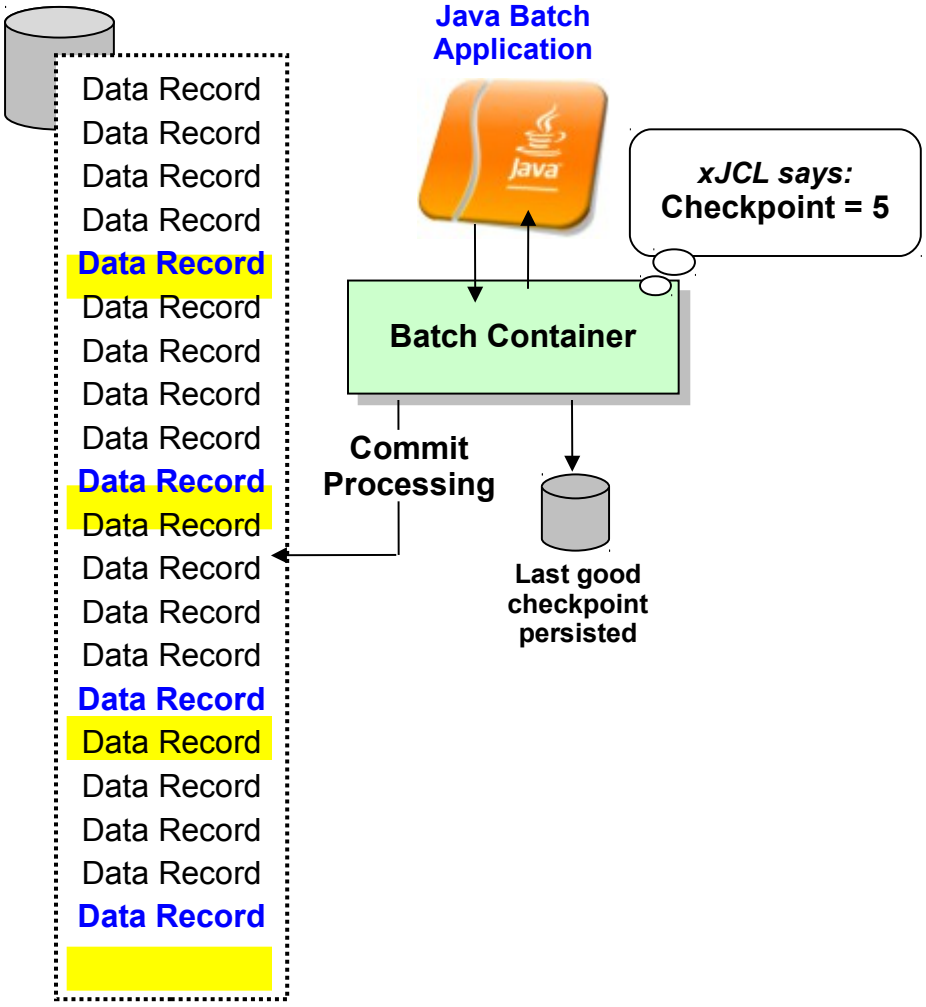
WSGRID interacts with Job Dispatching, submitting the job and processing Java batch job output back to STDOUT or JES Spool if z/OS

WSGRID program stays up for life of job in WebSphere Java Batch

To the Scheduler, WGRID *is* the Java Batch job ... but behind WSGRID is all the WebSphere Java Batch function we'll discuss

Transactional Checkpoint Processing

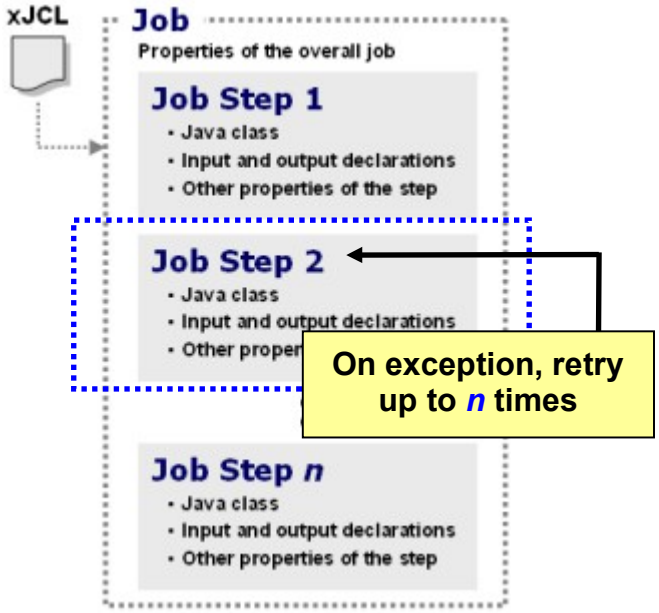
The batch container provides the ability to checkpoint at intervals based on either record count or time. The container keeps track of last checkpoint.



- Checkpoint interval (record or time) specified in the xJCL
- This is a function of the batch container, *not* your application code
- As checkpoint intervals are reached, container commits and records the checkpoint attained
- In the event of a failure, job may be restarted at the last good checkpoint
- Set the checkpoint interval based on your knowledge of balance between recoverability and efficiency

Retry-Step Processing

Provides a means of retrying a job step in the event of an exception thrown.
If successful on retry then the job continues and your processing completes.



xJCL tells Container:

- How many step retries may be attempted
- What exceptions to consider for retry-step processing
- Alternatively, what exceptions to *exclude* from retry-step processing
- Whether to process a delay before attempting a retry of the step

Objective: retry step in attempt to allow overall job to continue and complete when an unanticipated exception is thrown

This is at level higher than skip-record ... this is if an unhandled exception is thrown when the job step function is called

Batch container falls back to last good checkpoint and restarts from there

A "retry-step listener" may be called so you can perform custom action upon retry-step processing
More on "batch listeners" coming up

xJCL properties allow you to specify how many retry attempts will be performed and what exceptions to include or exclude from consideration

When retry limit is reached, job will go into restartable state
Normal restart-at-checkpoint would occur

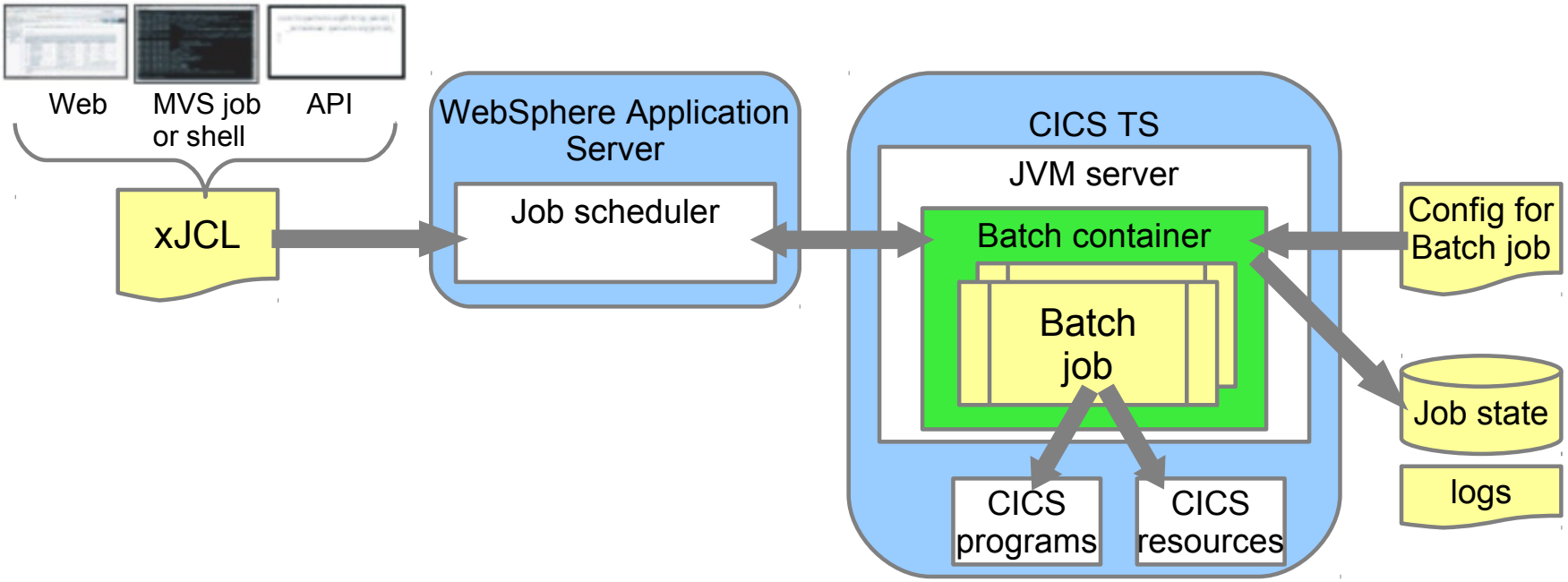
CICS TS Modern Batch Feature Pack

High-level overview

- What it is
  - A Java Batch Container for CICS 4.2 / 5.1 providing
    - Checkpointing, logging, recovery etc
  - Jobs scheduled and managed from WebSphere Application Server

- Delivered as a fully supported Feature Pack
  - See Announcement ENUS213-177
  - Order via Shop Z

Architecture



What we control via Plugpoints

- Communications
  - CICS http
- Persistent Store for Checkpoint Information
  - DB2 via JDBC
- Transaction Management
  - CICS syncpoints
- Executor Service (Threading)
  - Starts CICS thread to process work
- Job Log management
  - Stored on zFS

Input Batch Data Streams

- Doesn't need to be transactional (browse only)
  - CICS
    - KSDS Input Data Stream
  - WAS provided
    - JDBC Reader
    - RecordOrientedDatasetReader
    - File Reader (zFS)

Output Batch Data Streams

- Transactional
  - CICS
    - KSDS Output Data Stream
  - WAS provided
    - JDBC Writer
- Non-Transactional
  - WAS provided
    - RecordOrientedDatasetWriter
    - File Writer (zFS)

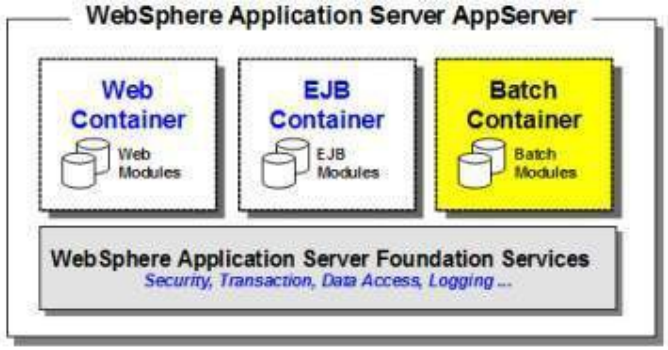
Wrap up and Summary

WebSphere Java Batch

WebSphere Application Server v8.5 integrates capabilities from WebSphere Compute Grid and delivers a complete **enterprise level Java batch** processing solution

Key Features:

- Java Batch programming model
- Java Batch container built on WAS QoS
- Development and deployment tooling
- Batch execution environment
- Concurrent OLTP and batch workloads
- Enterprise scheduler integration
- Parallel processing of batch jobs
- Container based checkpoint and restart
- Mixed batch workloads
- COBOL support on z/OS



Compute Grid capabilities integrated into WAS 8.5

CICS Modern Batch - Value Proposition

Move batch into the CICS environment and **integrate with OLTP** to gain the benefits of **concurrent processing**, **shared business logic**, and **cost efficiencies**

- **Reliable batch infrastructure** – Built on the proven Qualities of Service delivered by CICS Transaction Server.
- **Incremental modernization** – Move at your pace to reduce risk.
- **Resource efficiencies** – Focus resources on business logic and leave the infrastructure to the middleware
- **Enterprise integration** – Integrate with existing enterprise schedulers to help deliver a robust end-to-end solution.
- **Enables new execution patterns** – Dynamic OLTP and Batch runtime environment.
- **Supports a SOA strategy of reuse** – Enable the cost effective sharing of business logic across both the OLTP and Batch paradigms.
- **Reduce batch windows** – Transition from traditional batch windows to running batch 24x7 concurrent with OLTP.

Improving the Integration between Distributed Security and CICS with SAML

Agenda

In this session...

Security... who needs it?

Recap elements of security to set the scene

Introduction to SAML

Walk through SAML as a distributed security system – what you can say, how it gets used, roles of system components, ….

Integrating distributed security realms with CICS

What you can do today

What customers want to do in the future

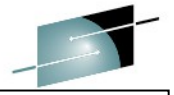
CICS and SAML

What we're working on for SAML – this piece will discuss future capabilities

Summary and Questions

Security... who needs it?

Transaction processing trends



RedGuide: Transaction Processing: Past, Present and Future

Published October 2012

Business

- New business services to attract customers and maintain their loyalty
- Business agility and optimization
- Control of risks and ability to respond to regulatory scrutiny
- Requirement to build partner relationships,and manage acquisitions and mergers
- Pressure to reduce costs

Technical

- Continued evolution of SOA
- Mobile
- Web 2.0
- Business events and rules
- BPM

"We try to provide a friendly and pleasant online experience to our customers and that also rewards them for their loyalty." ***(Misha Kravchenko, Marriott International)***

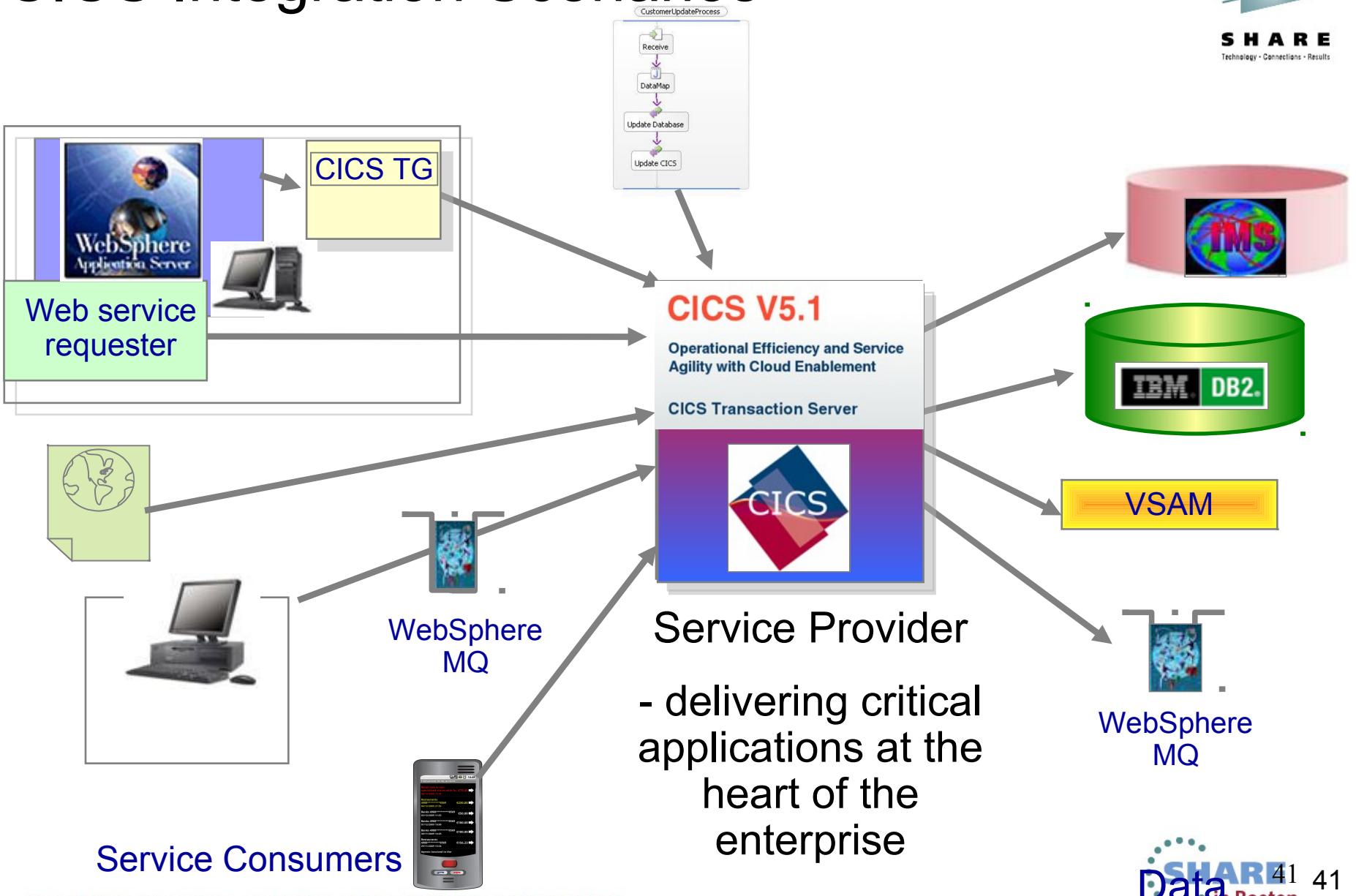
"The major business trends impacting our TP systems are increasing customer expectation, the need for quicker delivery of applications and more partner integration" ***(China Merchants Bank)***

"The overall cost of the service layer is greater than the process layer, which in turn is greater than the media access layer. This means that the best ROI is achieved through service reuse."

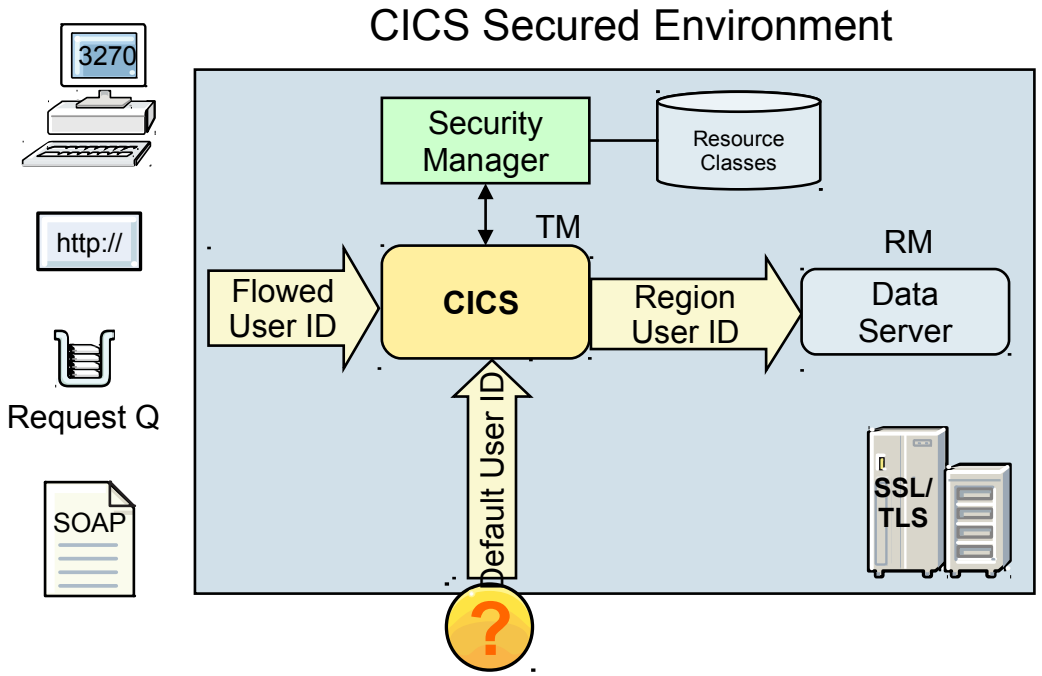
"The use of web services is strategic for the bank." ***(Marcel Däppen, UBS WM&SB)***

"We expect more growth coming from the mobile channel and we also foresee a workload increase from new self-service applications." ***(ABN AMRO Bank)***

CICS Integration Scenarios



CICS secure integration



Flowed User ID - authentication token for external user

Default User ID – used when no credentials have been established

Region User ID – used for checking CICS region access to system resources

Authentication - CICS requires a password/pass phrase, digital certificate or identity assertion

Identification - CICS requires an 8-character userid for use with its external security manager

Authorization - CICS uses ESM to authorize the userid to a specified resource class

Confidentiality/Integrity - CICS uses TLS/SSL or WS-Security

Common challenges

- **End-to-end security** is often hampered by the issue of how to provide secure access between middleware components that use disparate security technologies, such as user registries and security token formats

- Often security is at odds with performance, because the most secure techniques require the most processing overhead

- The range of options is vast and the required

Introduction to SAML

What is SAML?

Security Assertion Markup Language (SAML)

OASIS XML-based standard

Used to exchange authentication and authorization data between parties

Identity Provider – handles authentication and the creation and verification of SAML tokens.

Service Provider – accepts a SAML Token as an identity assertion

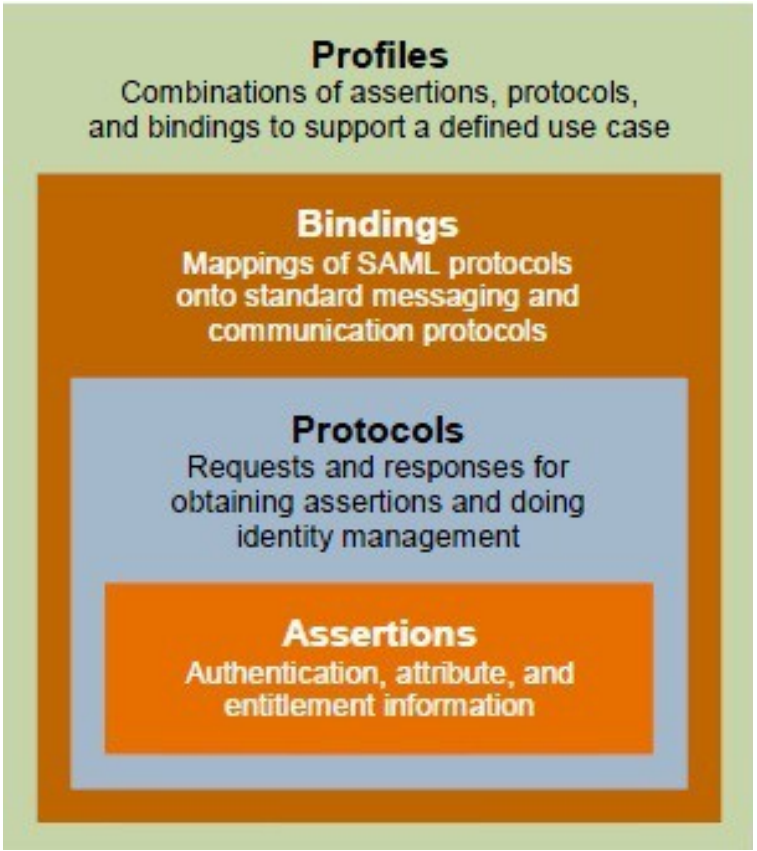
SAML comprised of many (20+) “profiles” describing very specific uses cases on how to use SAML

SAML – Security Assertion Markup Language

"..XML based framework for describing and exchanging security information between on-line business partners."

Web Single Sign-On

Dynamic creation of Identity Federations



What is a Security Token?

A security token is a **collection of claims**

A token almost always contains information about an **identity**

- User identity or system identity

In Identity Federations very often aliases are used

- Accounts can be linked out-of-band

It can also contain **additional identity information**, such as:

- How the identity was first authenticated
- Group membership
- Customer status information (“Gold” customer)
- Any type of custom information

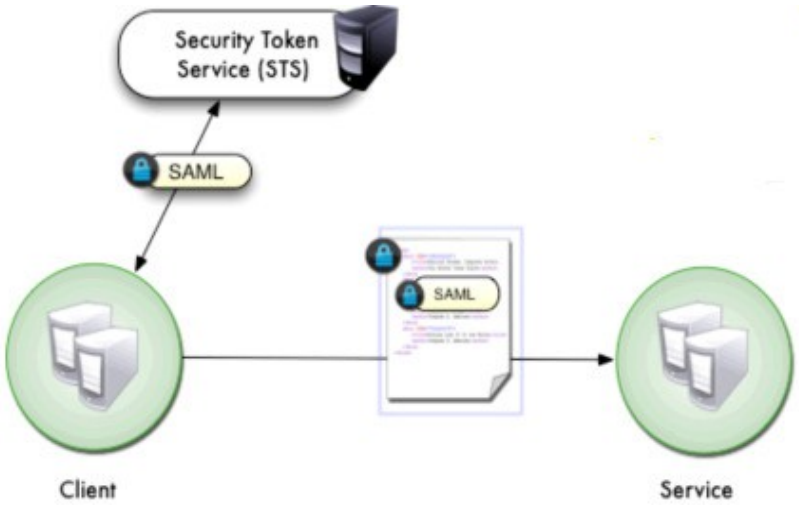
Types of Tokens include

- X.509 Public Key Certificates
- Kerberos shared-secret tickets
- SAML

How to get a token?

A token may be "**Self Issued**"

- Easy to set up
- No Authority to validate authenticity
- May require transfer "out-of-band"



Security Token Service (STS) Issued

- STS defined in WS-Trust specification
- WS-Trust specifies how to request creation, mapping or validation of Security Tokens (including SAML) from an STS
- Harder to configure
- Centralised management of token creation and identity mapping
- IBM Tivoli Federated Identity Manager can act as an STS

48

Example SAML token – part 1

```
<saml2:Assertion ID="_285BFE4D057C7CB11513589335678 48"
    IssueInstant="2013-01-23T09:32:30.808Z" Version="2.0"
    xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">
    <saml2:Issuer>Name-of-STS</saml2:Issuer>
    <Signature>
      <SignedInfo>
        <CanonicalizationMethod.../>
        <SignatureMethod.../>
        <Reference URI="#_285BFE4D057C7CB11513589335678 48"/>
      </SignedInfo>
      <SignatureValue>Signature-of-Token</SignatureValue>
      <KeyInfo>
        <X509Data>
          <X509Certificate>Public-Key-of-Certificate</X509Certificate>
        </X509Data>
      </KeyInfo>
    </Signature>
```

When token was issued

Name of STS

Signature of token

Certificate used to verify signature

Example SAML token – part 2

```
<saml2:Subject>
   <saml2:NameID>MyName</saml2:NameID>
   <saml2:SubjectConfirmation Method="urn:oasis:names:tc:SAML:2.0:cm:bearer" />
</saml2:Subject>
<saml2:Conditions NotBefore="2013-01-23T09:32:30.808Z"
   NotOnOrAfter="2013-01-23T10:32:30.808Z">
   <saml2:AudienceRestriction>
      <saml2:Audience>http://TheRelyingParty.com</saml2:Audience>
   </saml2:AudienceRestriction>
</saml2:Conditions>
 <saml2:AttributeStatement>
    <saml2:Attribute Name="PersonAffiliation">
      <saml2:AttributeValue>Manager</saml2:AttributeValue>
    </saml2:Attribute>
    <saml2:Attribute Name="CreditLimit">
      <saml2:AttributeValue>500.00</saml2:AttributeValue>
    </saml2:Attribute>
</saml2:AttributeStatement>
</saml2:Assertion>
```

Name of Subject

Timeframe token is valid for

Intended receiver of token

Some attributes that specify things like what role the user has, maximum values etc.

Integrating distributed security realms with CICS

Integrating distributed security realms with CICS

What you can do with SAML and CICS today

- *CICS support for message security*
- *CICS and TFIM*
- *CICS and DataPower*
- *z/OS Identity Propagation*

Why customers want to use SAML directly with CICS

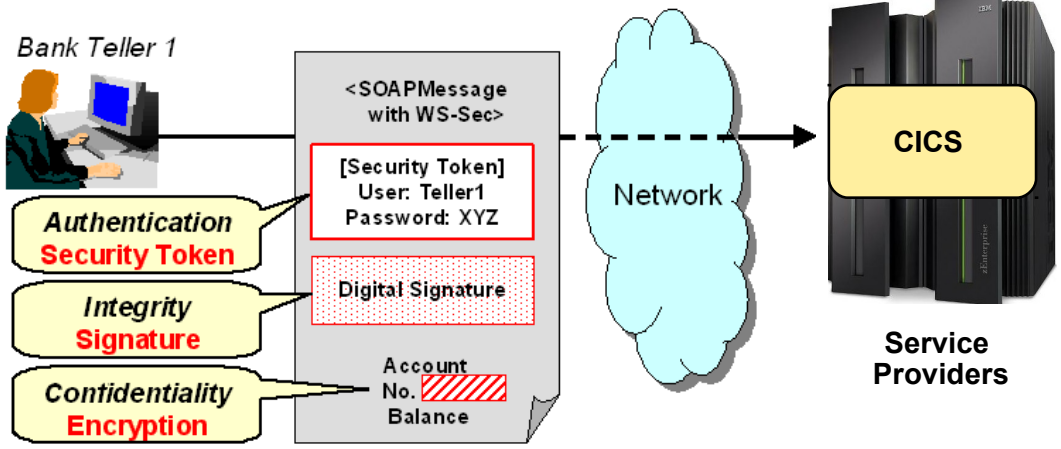
- *Current RFEs*
- *Customer comments*

Future scenario – consume a SAML assertion in CICS

- *Signed SAML token*

CICS support for message security

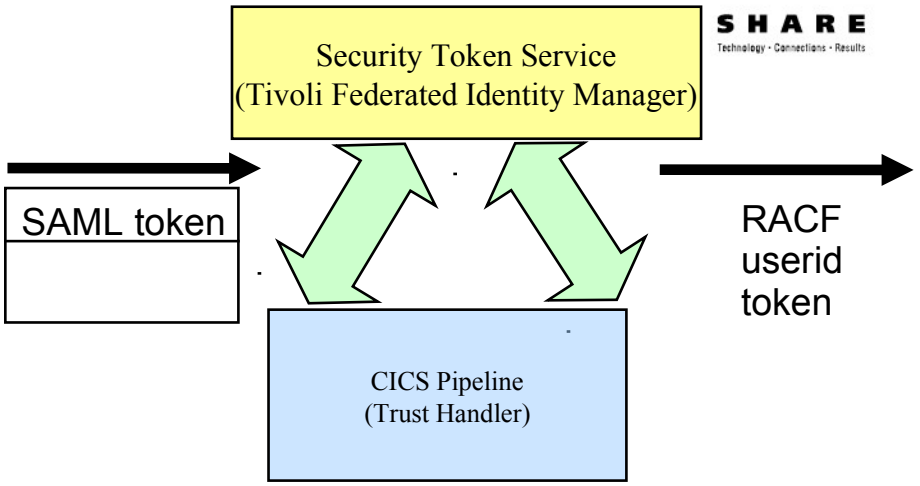
- Various mechanisms for deriving a user ID from an inbound message, including:
  - Basic authentication
  - X.509 certificate
  - Identity assertion
  - Interoperation with a trusted third party



- Various mechanisms for attaching a security token to outbound message, including:
  - X.509 certificate
  - Identity assertion
  - Interoperation with a trusted third party
- Signature **validation** of inbound message signatures and signature **generation** for the SOAP body on outbound messages
- **Decryption** of encrypted data in inbound messages and **encryption** of the SOAP body content on outbound messages
- Enabled by including the <wsse-handler> element in the pipeline configuration file

CICS and TFIM

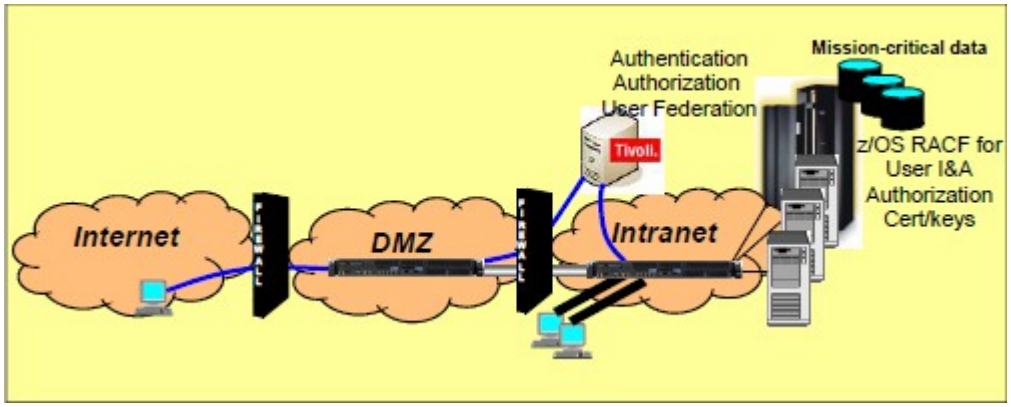
- WS-Trust provides a framework for building trust relationships
  - Sender and Receiver in different security domains
  - Security tokens must be vouched for by trusted third party
  - Trusted third party called a Security Token Service (STS)

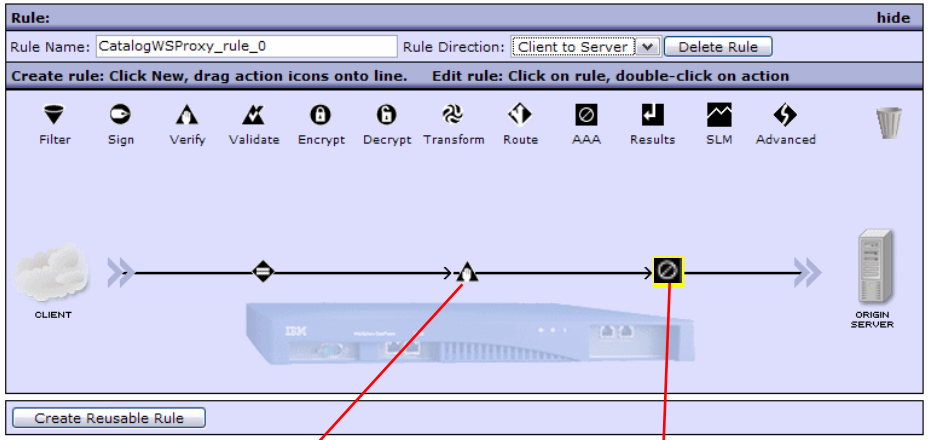


- STS can be used to transform one form of user identification into another form
  - e.g from SAML token to RACF user ID
- Tivoli Federated Identity Manager (TFIM) can act as an STS
  - Provides framework to support standards-based, federated identity management between enterprises that have established a trust relationship
- TFIM supports a wide range of security tokens, including SAML, UsernameTokens, Kerberos, LTPA, Passticket and X.509 tokens
- CICS has supported WS-Trust since CICS TS V3.2

CICS and DataPower

- To offload expensive security operations such as XML digital signature validation, encryption and validation
- As an XML firewall to protect CICS system against threats such as XML denial of service attacks
- Authentication and identity propagation
  - DataPower supports a wide range of security tokens including SAML
  - DataPower can interoperate with LDAP or Tivoli Federated Identity Manager (TFIM)
- DataPower supports z/OS Identity Propagation
- For transformation between XML and COMMAREA data formats

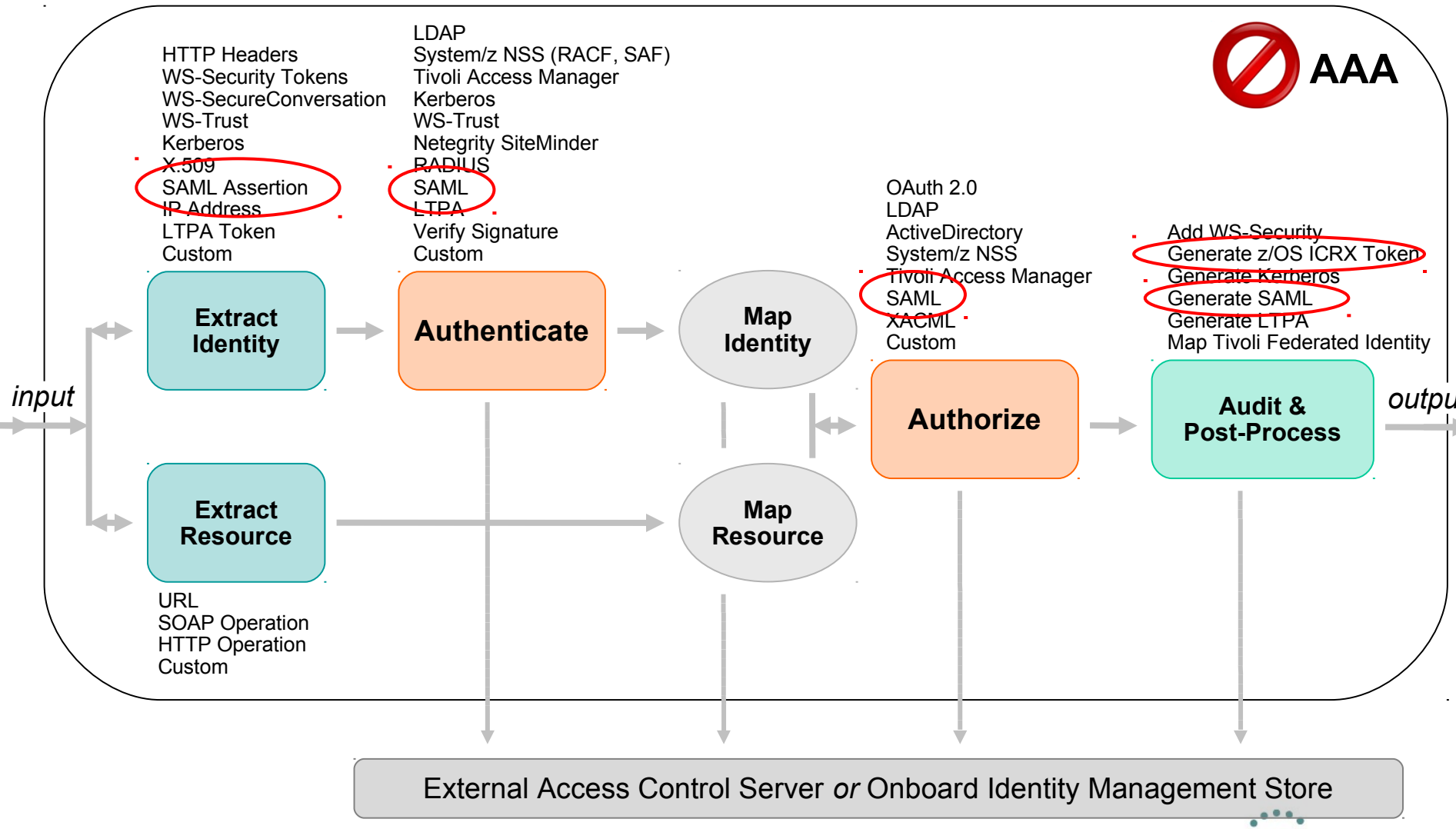




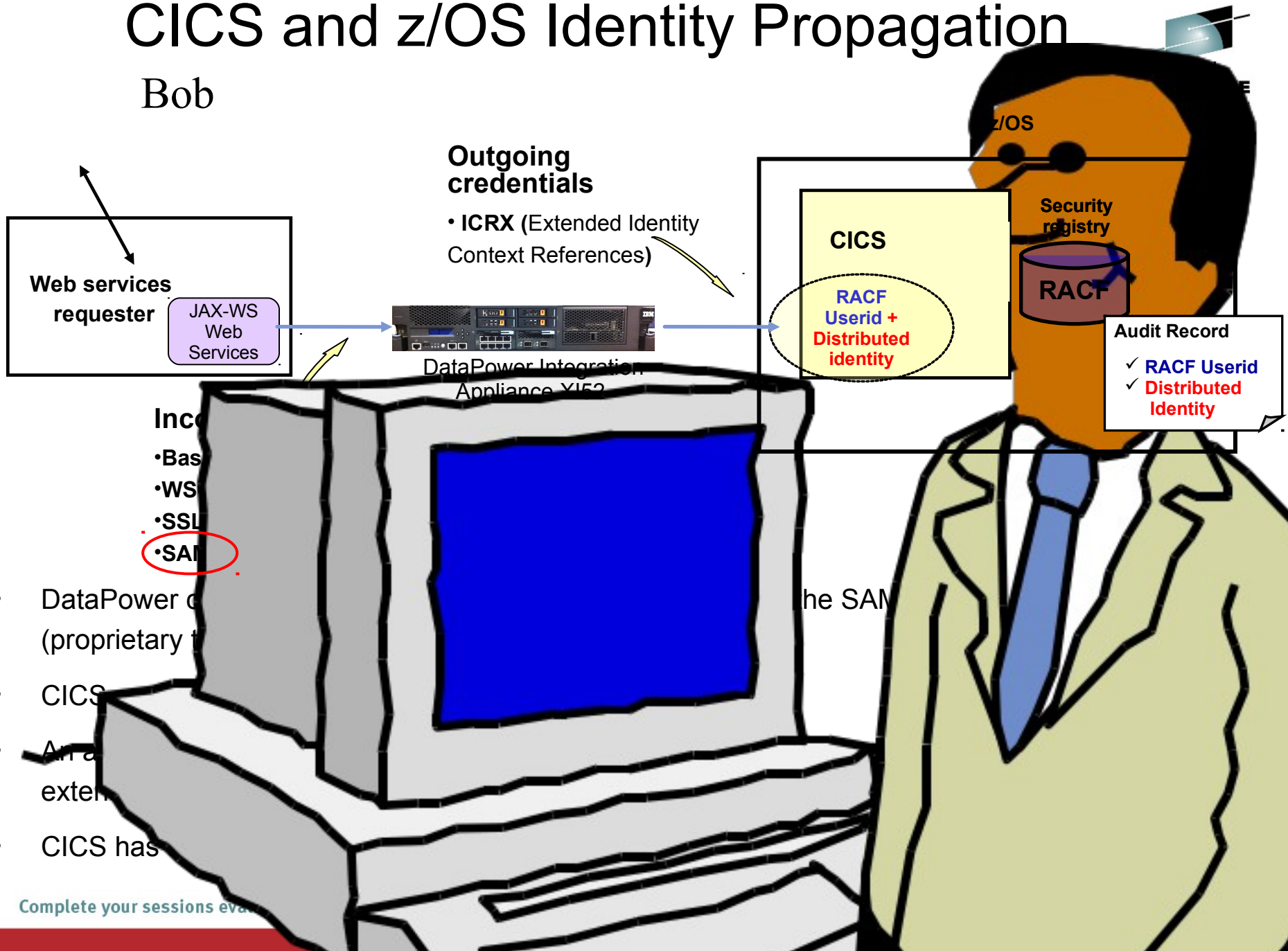
Verify Rule

AAA – Authenticate, Authorise, Audit

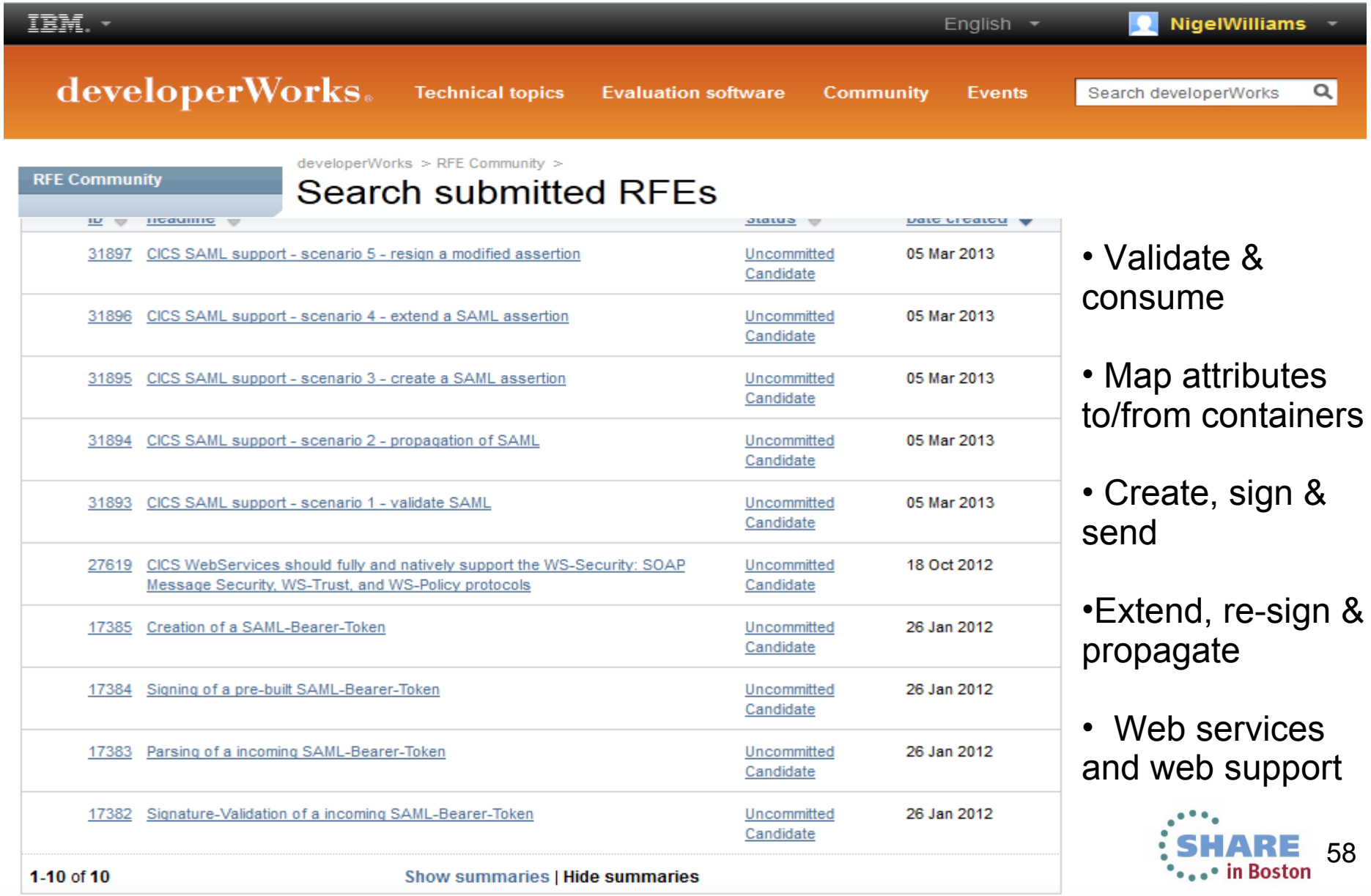
DataPower AAA provides full SAML support



CICS and z/OS Identity Propagation



Current RFEs

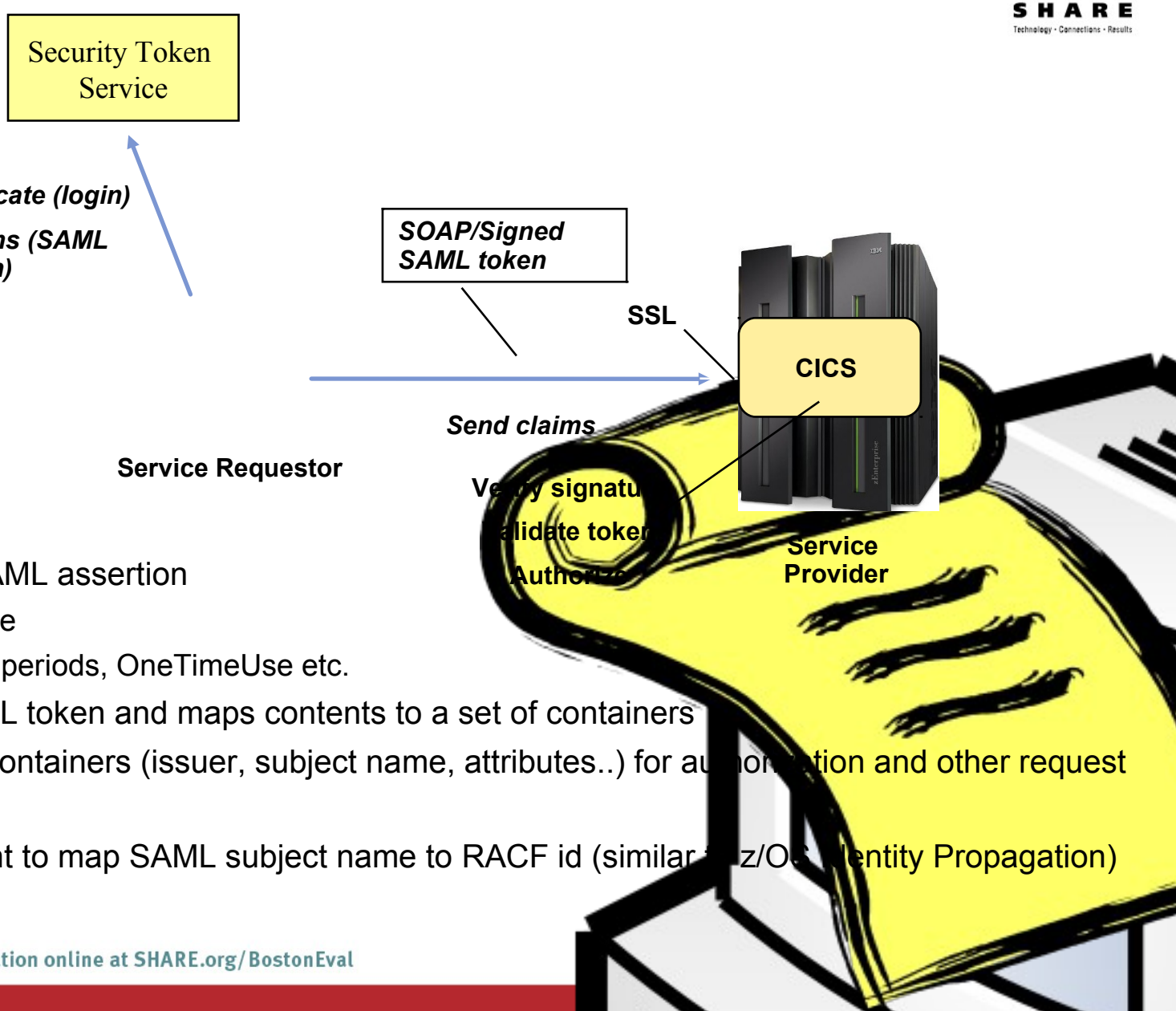


- Validate & consume
- Map attributes to/from containers
- Create, sign & send
- Extend, re-sign & propagate
- Web services and web support

Customer comments

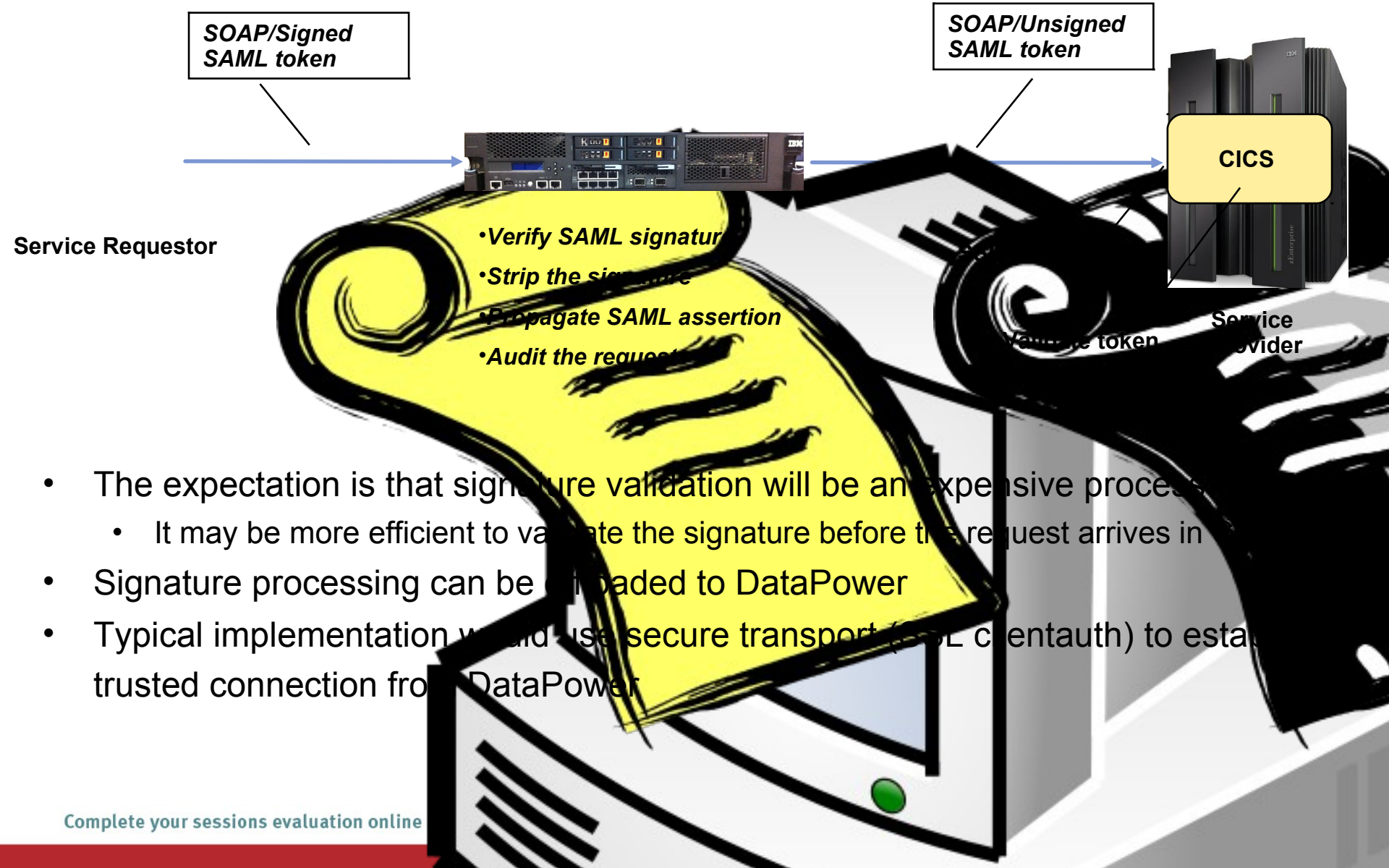
- *"SAML is an agreed industry-standard for the propagation of identities."*
- *"Currently, CICS offers very limited support for SAML … CICS relies on an external STS to process the SAML token which has negative impact on performance and stability."*
- *"Give a possibility within CICS to parse an incoming SAML Bearer Token and extract the various information in the token like "Issuer" and "Subject" and attribute statements (Name/Value-Pairs) into containers."*
- *"Support is required for signed and unsigned SAML tokens."*
- *"A RACF keystore is the most likely implementation for storing certificates."*
- *"Propagation of the SAML token between CICS regions is required for DPL."*
- *"Optionally CICS should execute the transaction under the identity as entered in the name claim."*

Future – consume a SAML assertion in CICS



- CICS validates SAML assertion
  - Verify signature
  - Check validity periods, OneTimeUse etc.
- CICS parses SAML token and maps contents to a set of containers
- Application uses containers (issuer, subject name, attributes..) for au tion and other request processing
- Future requirement to map SAML subject name to RACF id (similar z/O entity Propagation)

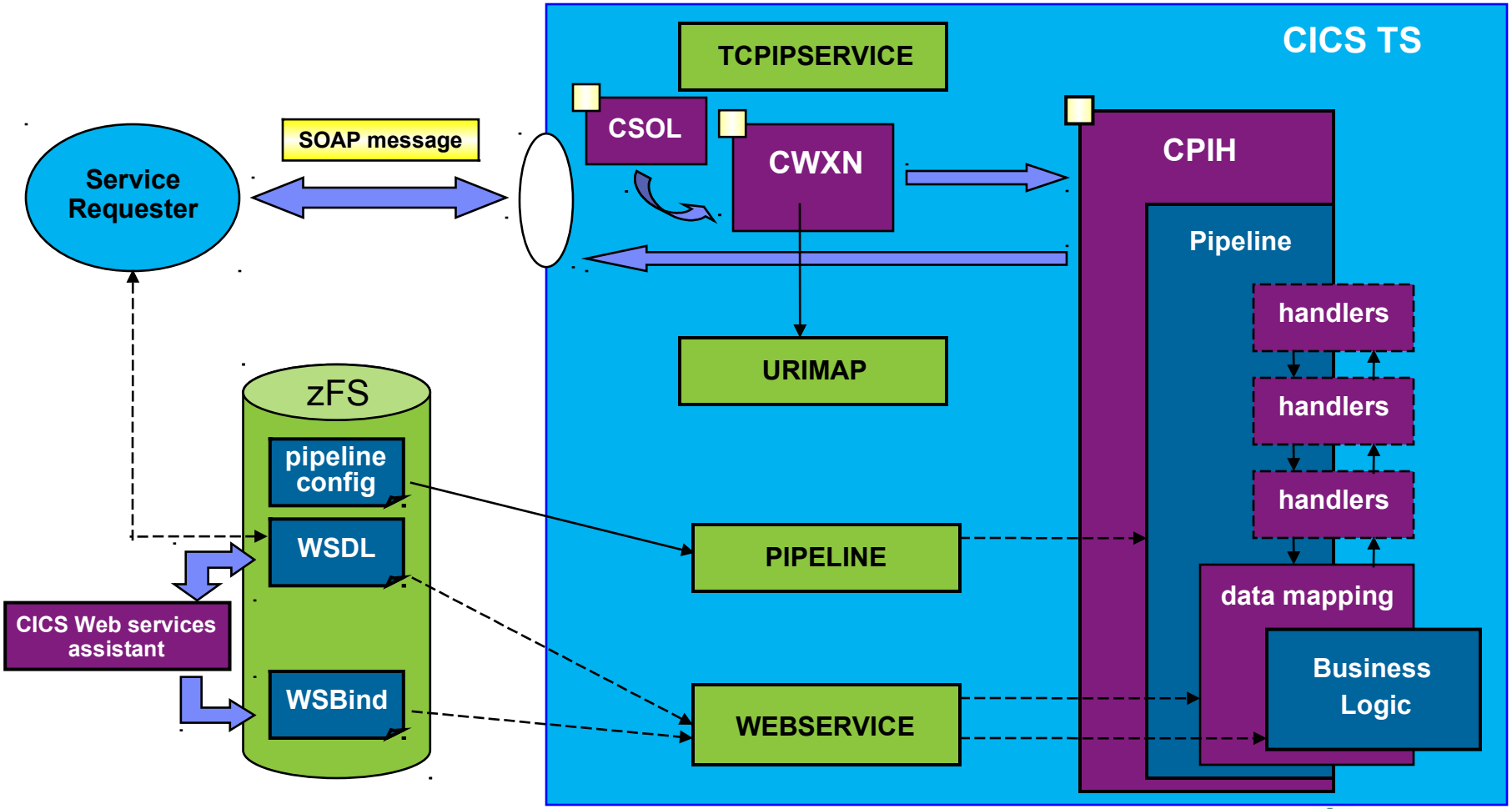
Balancing security with performance



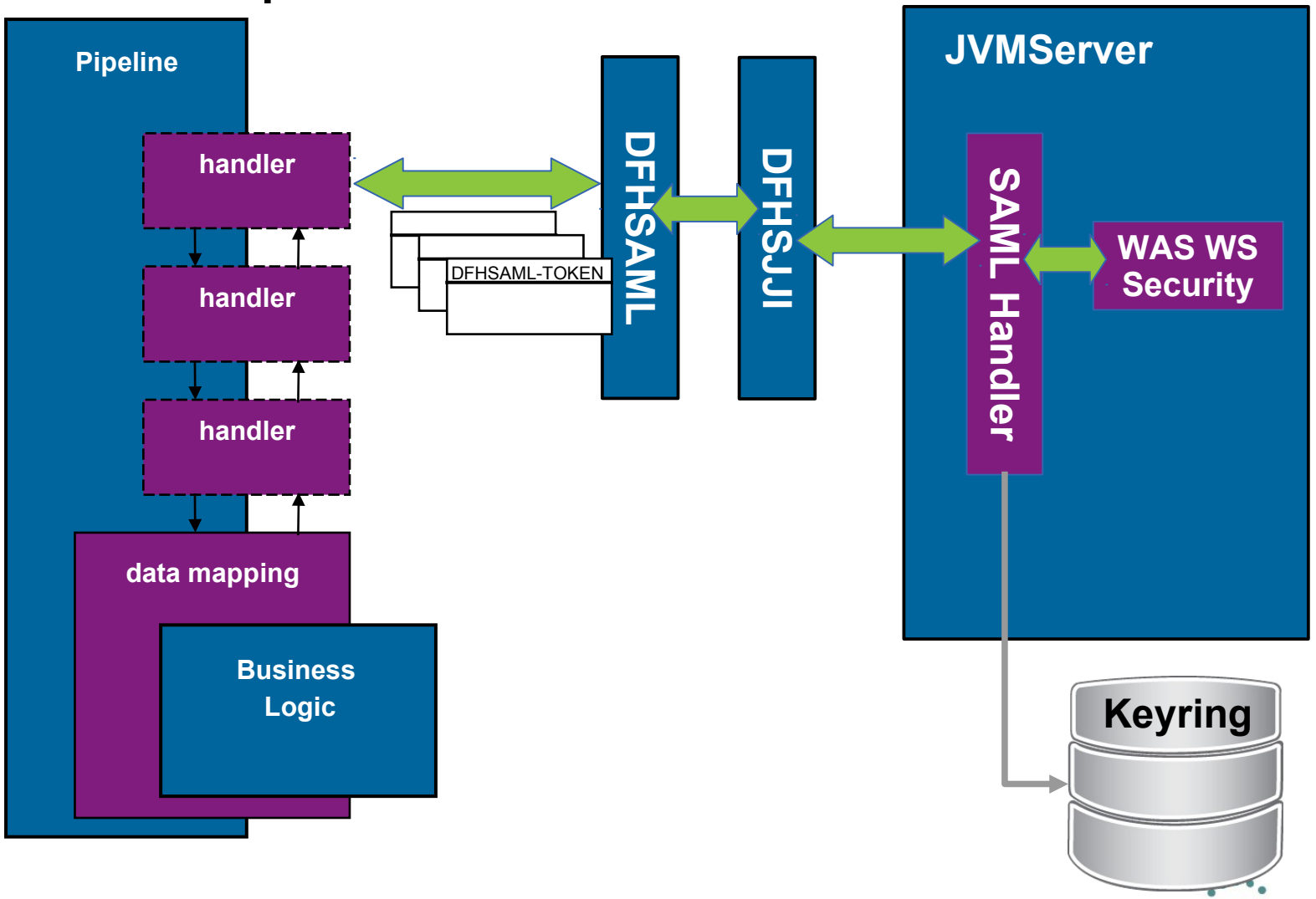
- The expectation is that signature validation will be an expensive proces
  - It may be more efficient to va ate the signature before the request arrives in
- Signature processing can be offloaded to DataPower
- Typical implementation would use secure transport (SSL clientauth) to estab trusted connection fro DataPower

CICS and SAML

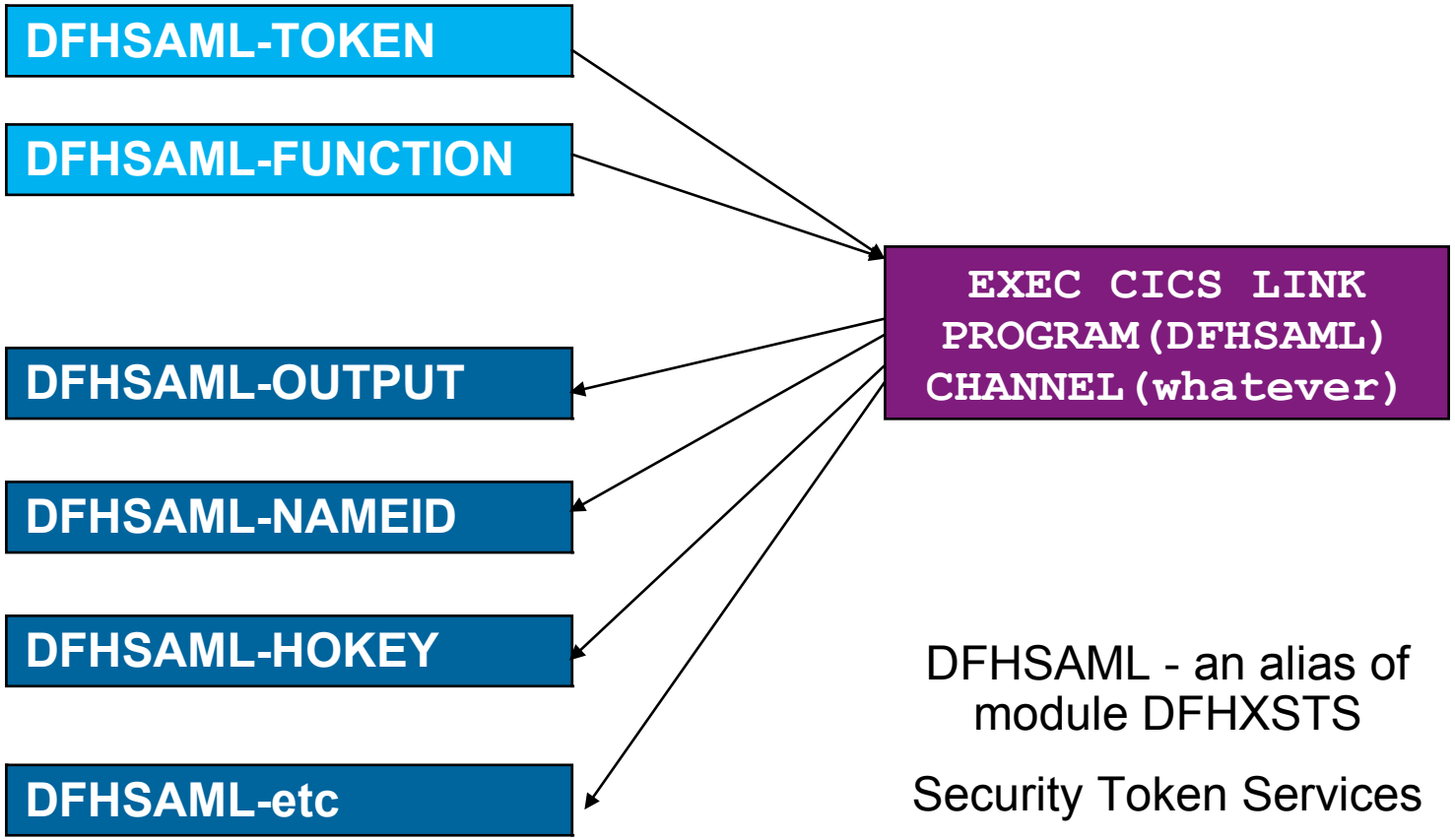
Security for Web services – Pipeline Architecture Review



Security for Web services – Pipeline Architecture Review



LINKable Interface – SAML



DFHSAML - an alias of module DFHXSTS

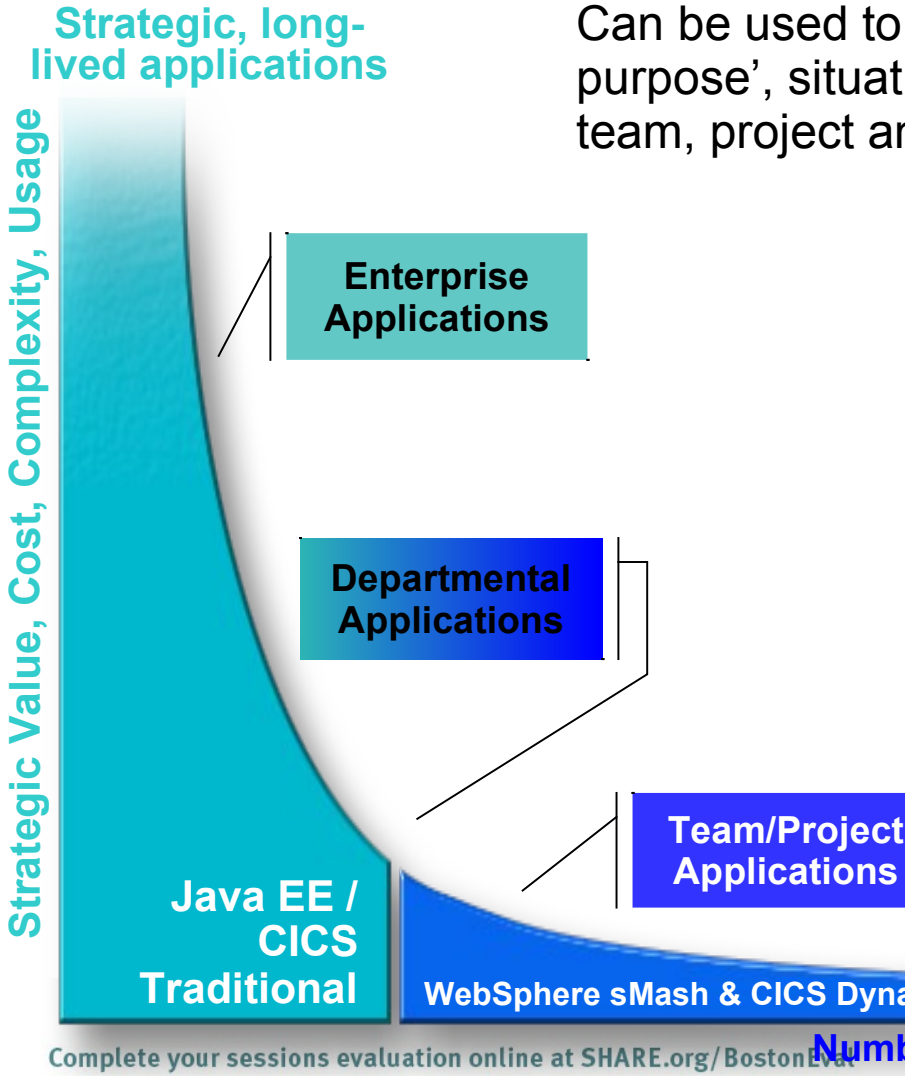
Security Token Services

Using Scripting to Dynamically Extend CICS

Agenda

- What is CICS Dynamic Scripting
- Dynamic Scripting Feature Pack 1.0, 1.1
- Dynamic Scripting Feature Pack 2.0
  - What's new
  - How it works
  - How to use it
  - Debug
  - Migration

CICS Dynamic Scripting



Can be used to develop and deploy lightweight, 'fit for purpose', situational applications that meet departmental, team, project and personal requirements, e.g.:

- **Creating reports, dashboards and widgets**
- **Quickly front ending existing applications**
- **Exposing CICS assets in mash-ups**
- **Creating productivity applications**
- **Quickly trying out new business ideas**
- **Introducing new IT staff to CICS via PHP**
- **Developing without a dedicated budget**
- **Porting existing unmanaged PHP into CICS**

CICS Dynamic Scripting Feature Pack 1.0 , 1.1

Technology from **Project Zero**, WebSphere sMash v1.1.1.3 (projectzero.org)

Robust environment for **situational** reports, dashboards, and Web feeds

Provides **PHP and Groovy** support in CICS – agile, productive environment

Zero Resource Model (**ZRM**) with data managed by DB2 for z/OS

Uses CICS TS V4.1 **JVMServer** Technology

Manageability, Scalability, and Security

Situational applications - Quickly try business ideas

Introduce **new staff** to CICS via PHP

Run unmanaged PHP and WebSphere sMash applications in CICS

Easily expose CICS assets with **REST**ful interfaces

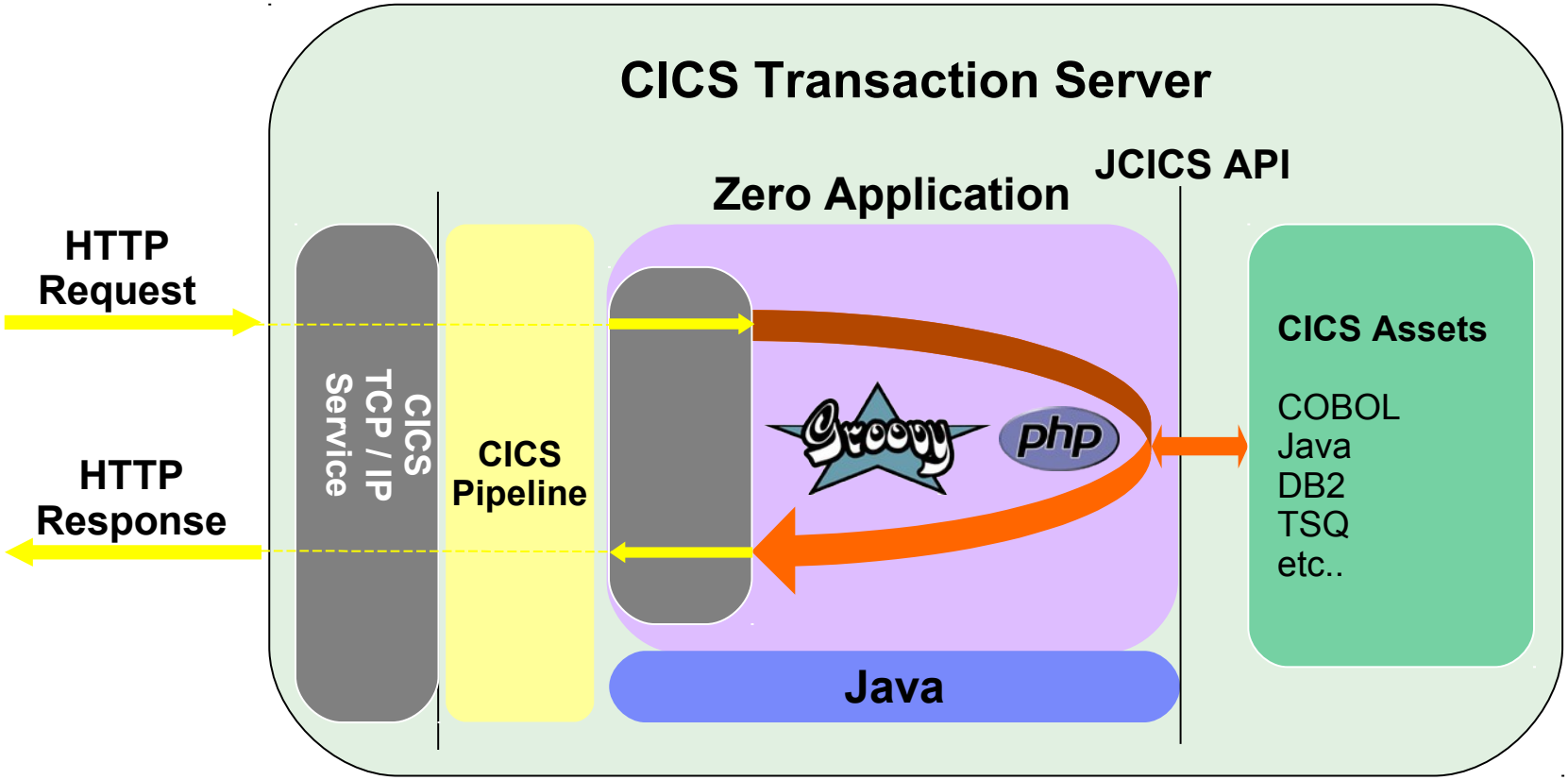
Feature Pack 1.0

Optional **no charge** product extension to CICS TS V4.1

Feature Pack 1.1

Optional **no charge** product extension to CICS TS V4.2

Dynamic Scripting Environment in CICS



CICS Dynamic Scripting Feature Pack 2.0

Each version of Dynamic Scripting is specific to a version of CICS. The Dynamic Scripting Feature Pack version 2 is developed for CICS Transaction Server for z/OS®, Version 5 Release 1 only; earlier releases are not supported.

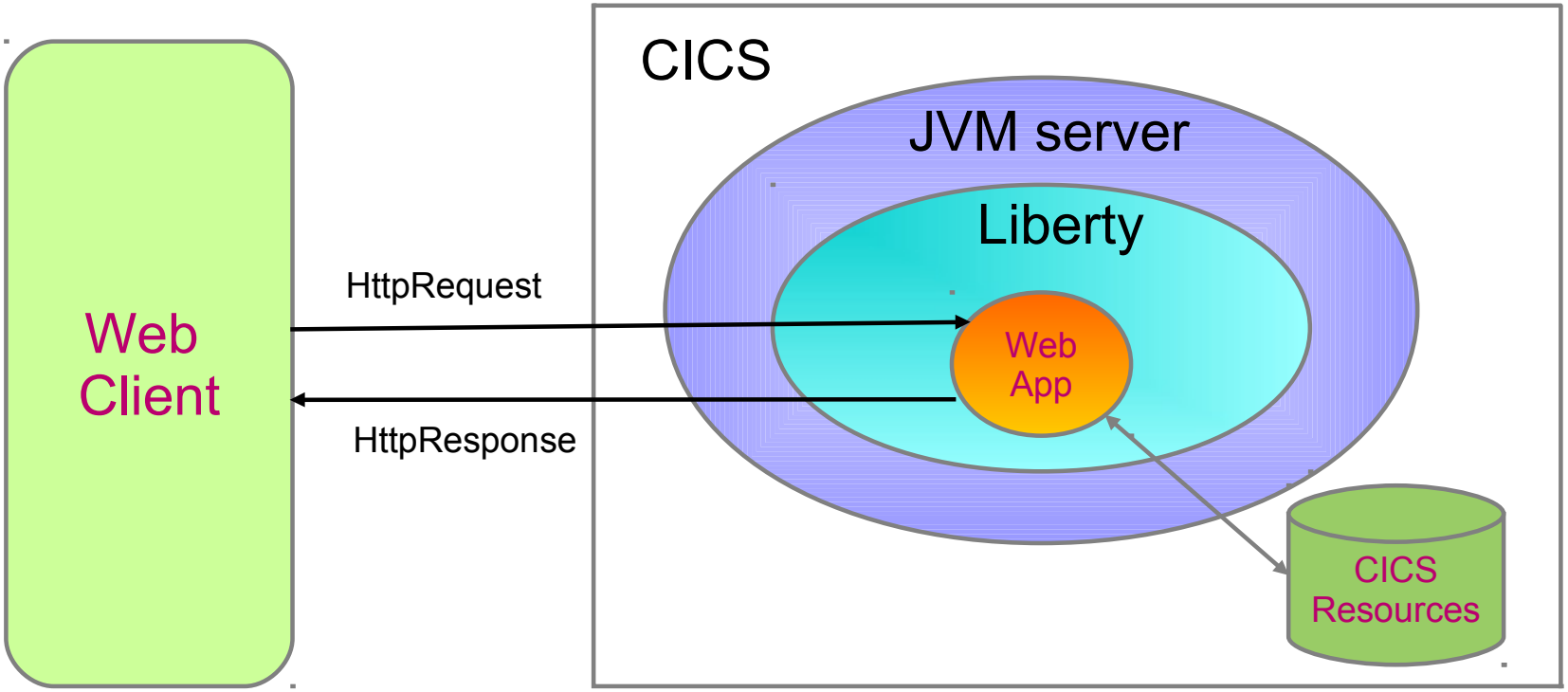
You can use the feature pack to take advantage of the following web technologies:

A dynamic scripting CICS based run time for PHP applications

Java as a system programming language

The CICS Dynamic Scripting Feature Pack 2.0 is developed using technology Liberty and P8 (PHP Engine).

Liberty Profile In CICS

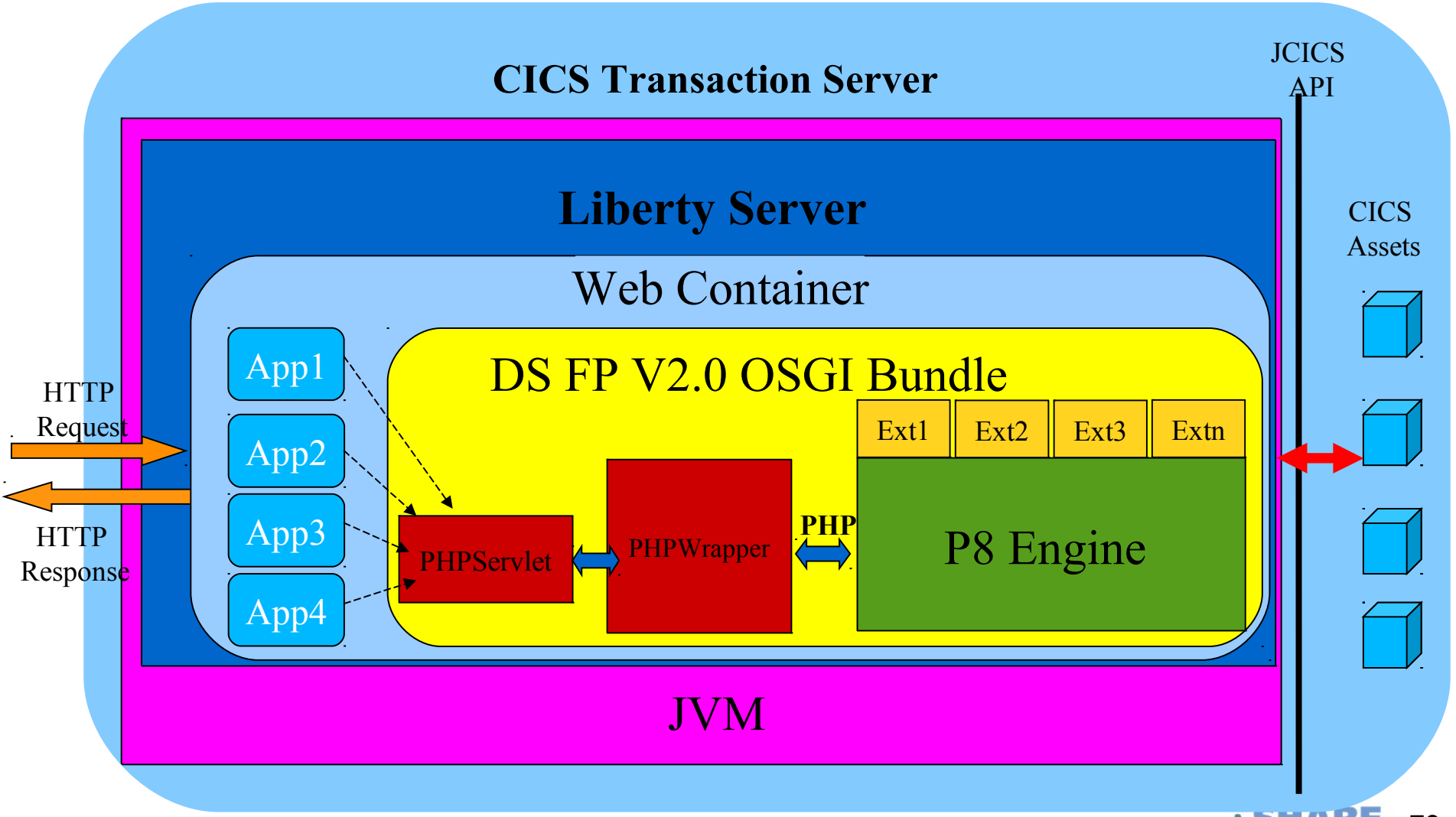


OSGi Framework

Runs in a JVMSERVER

Web App developed and deployed using Eclipse IDE & CICS Explorer SDK

Structure of CICS Dynamic Scripting 2.0



Benefit of running PHP in Liberty

Run all the PHP application in one JVMSERVER which has **multi-thread capability**

Manage PHP application in BUNDLE. CICS controls the whole **life-cycle** of application.

Packaged as OSGi in **EBA**, easy to be managed.

Mixed with other JEE technology, eg **JSP,servlet**, user can embed PHP into JSP

Failures in one PHP application will not affect other PHP applications or other Java applications on the server.

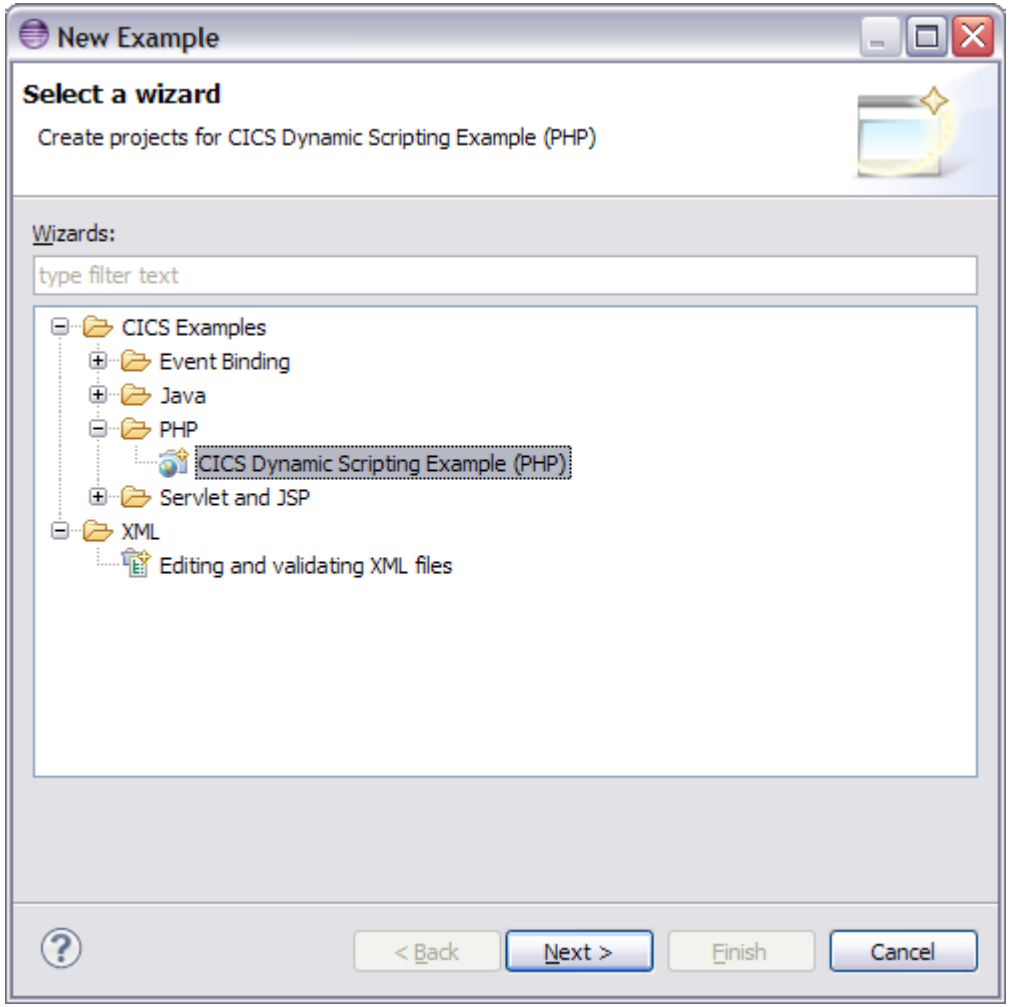
Liberty Web container provides lots of capability like **Security, CICS thread, Transaction**.

Installation and Pre-requirement

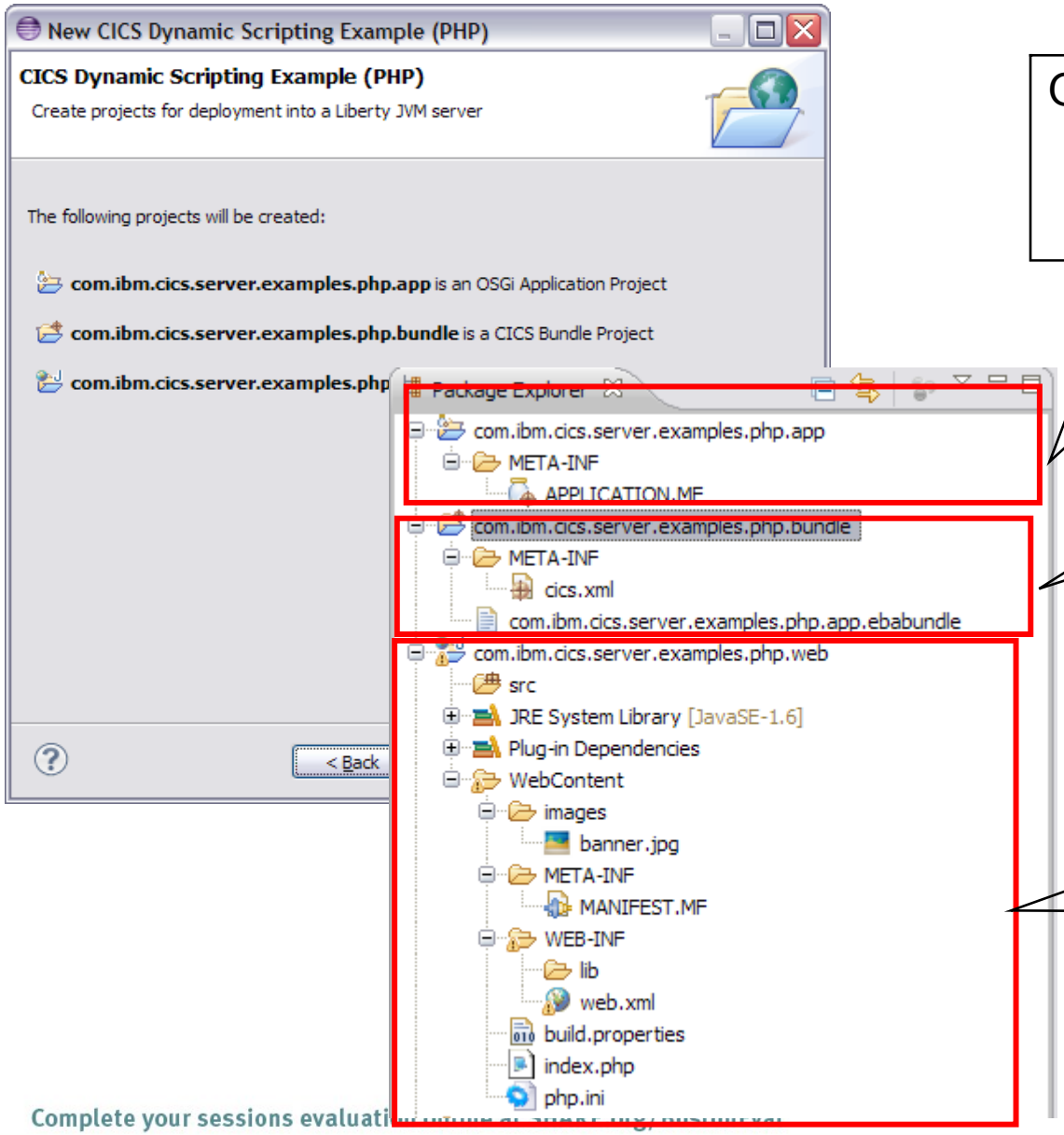
- CICS TS 5.1 with APAR PM80214 applied
- Install DS Feature Pack 2.0 in z/OS
  - Use FTP to copy the Dynamic Scripting Feature Pack 2.0 SMP/E installation package to a suitable directory in z/OS UNIX. You must have write access to this directory.
  - Use SMP/E to install Dynamic Scripting Feature Pack 2.0
  - Setup Dynamic Scripting Feature Pack 2.0 environment, such as extension lib, set liberty bundle repository and etc.
- Development environment

Start from sample

New Example in CICS Explorer SDK



Sample created

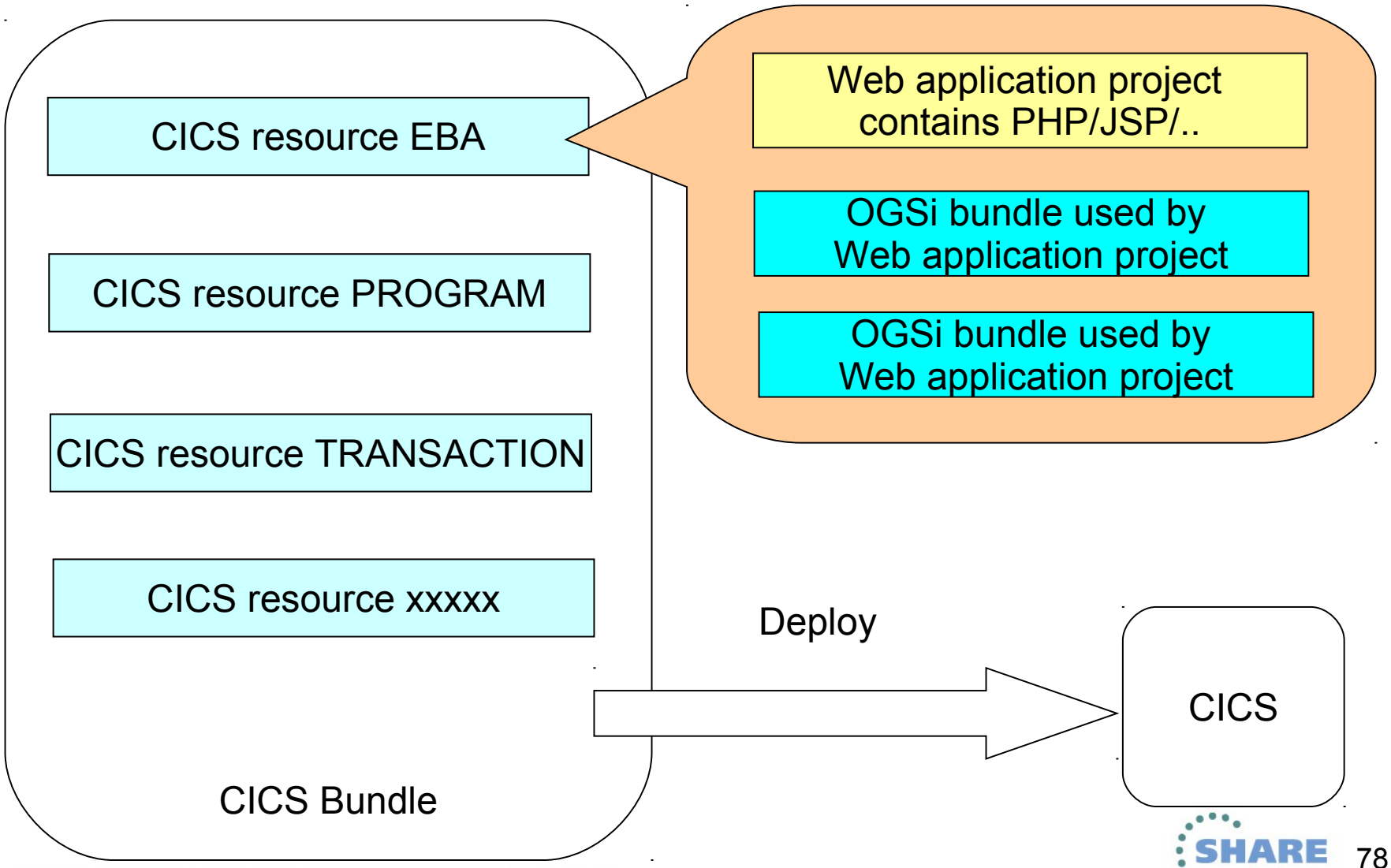


OSGi Application project to create EBA

CICS bundle project used to package EBA

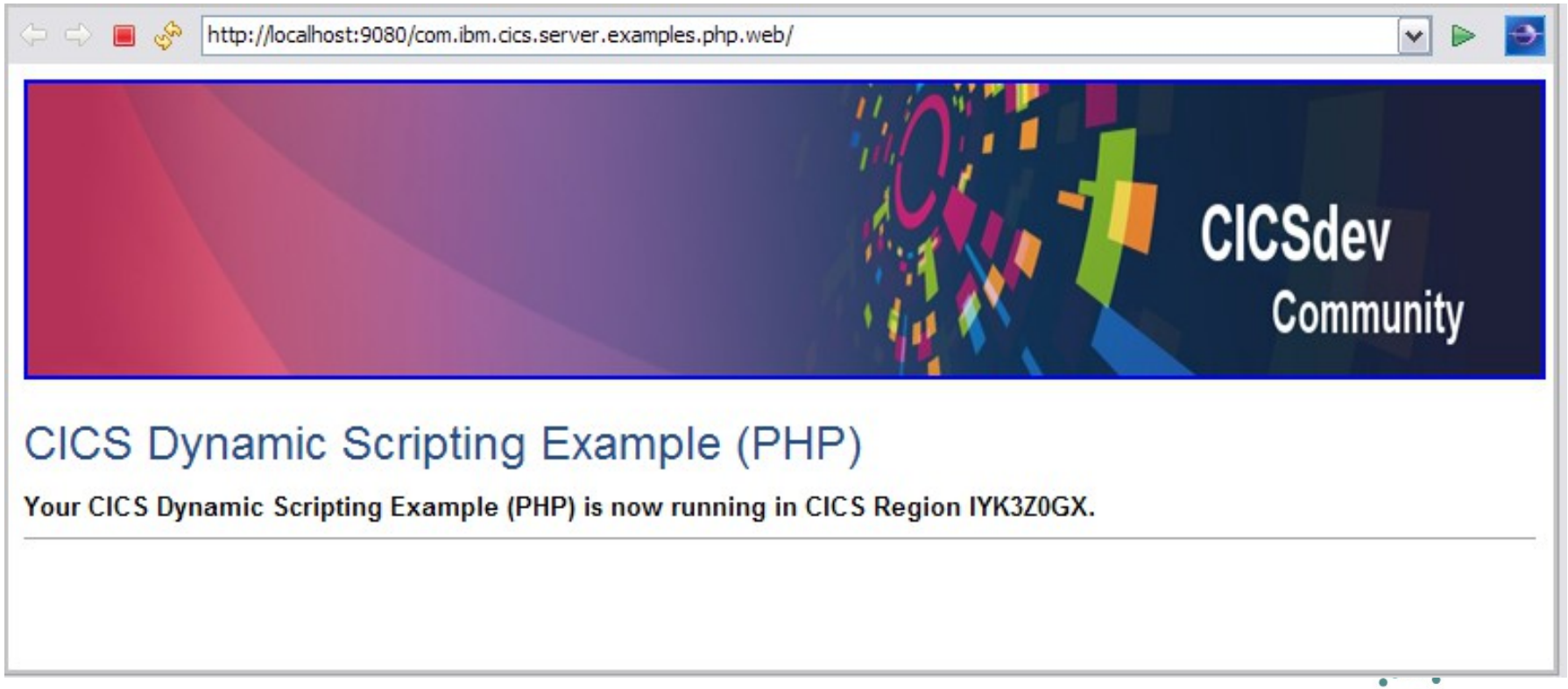
OSGi Bundle Project with Web support which contains all the PHP code

Relationship between projects and deploy



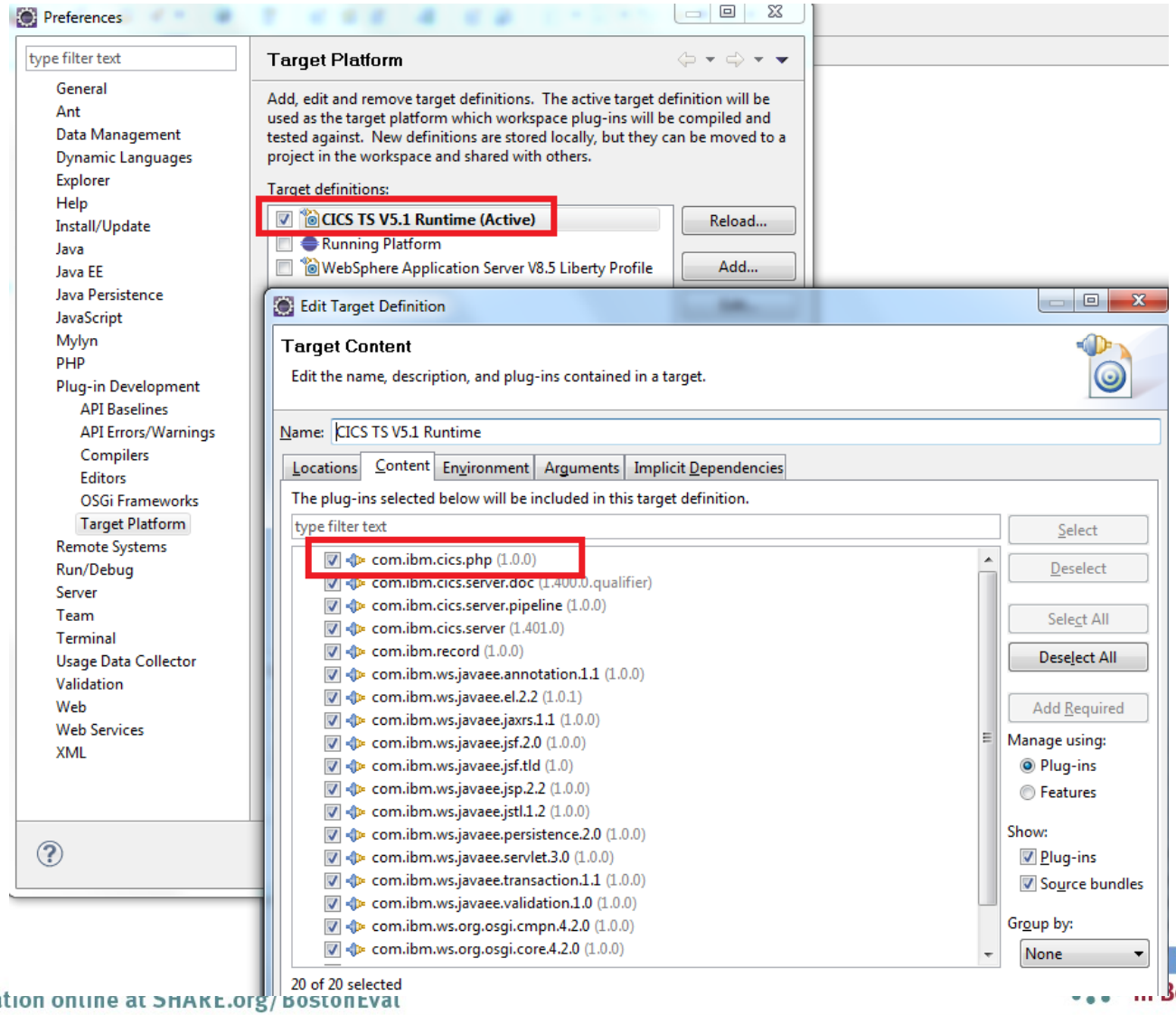
PHP code in Sample

```
<?php
    java_import("com.ibm.cics.server.Region");
    echo "Your CICS Dynamic Scripting Example (PHP) is now running in CICS Region
    " . Region::getAPPLID() . ".";
?>
```

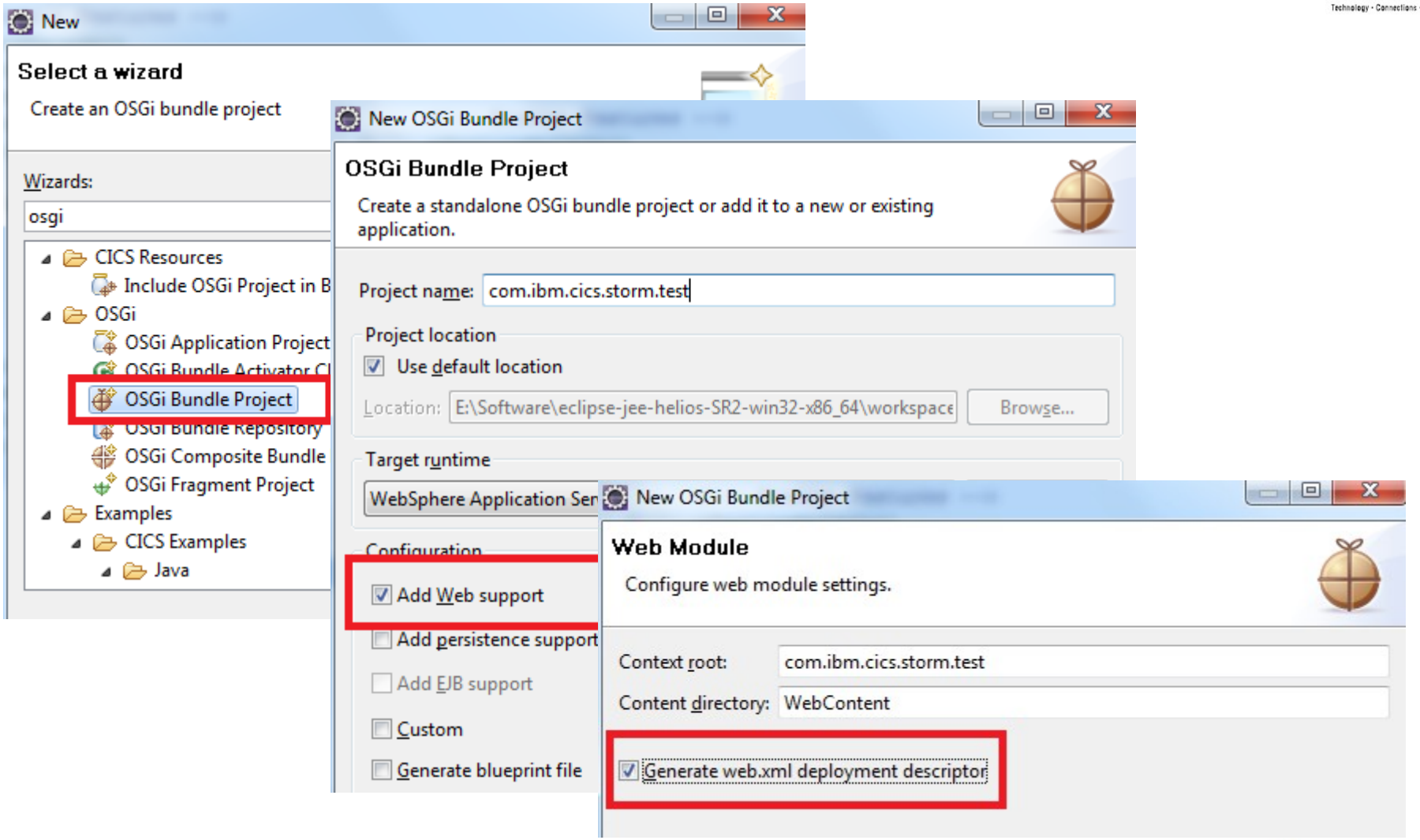


Start CICS explorer SDK

Update target running platform



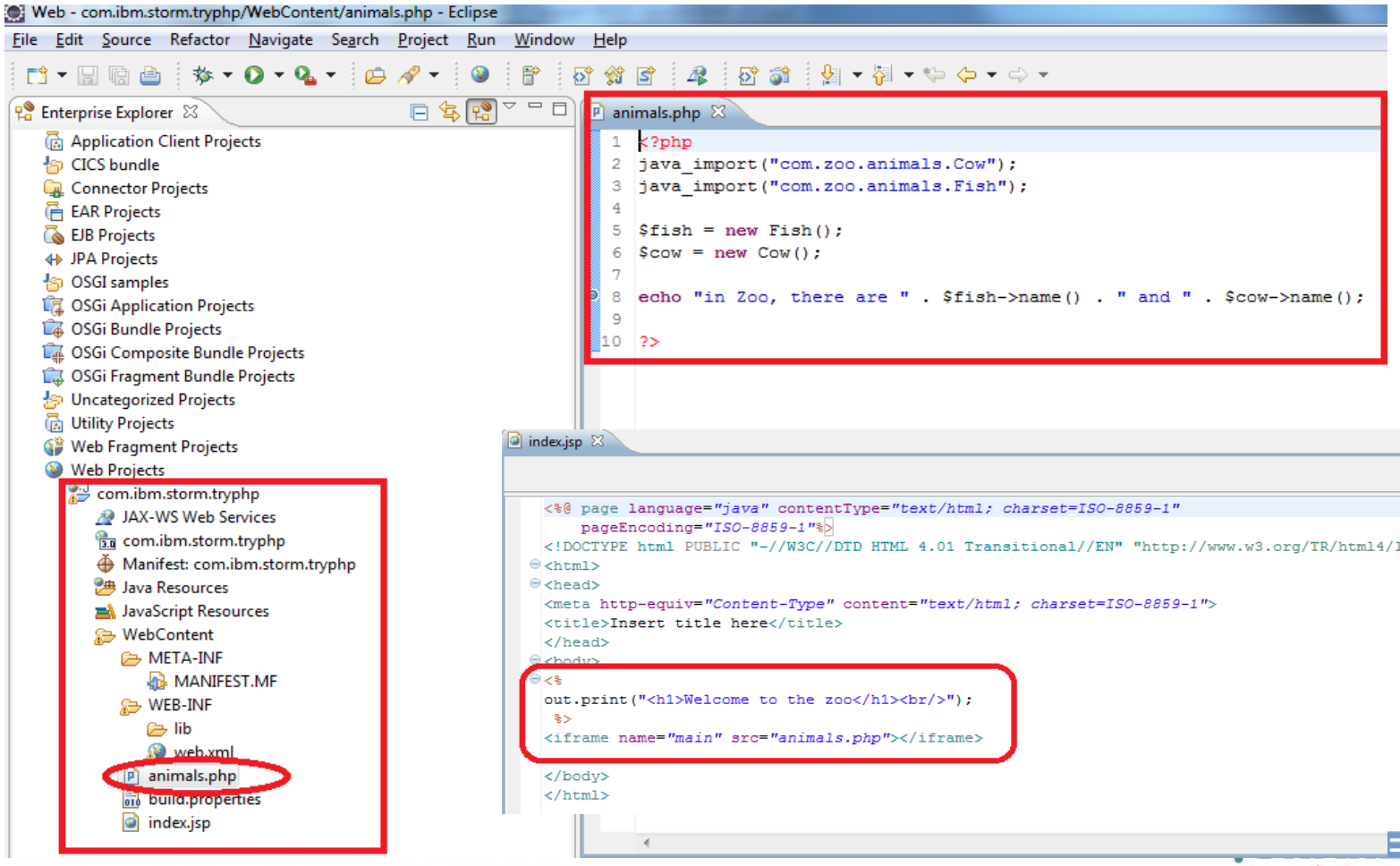
Create OSGi Bundle Project



PHP Eclipse IDE

PDT(PHP development toolkit) in eclipse

any other PHP IDE



Add CICS PHP package into MANIFEST.MF of PHP project

Manifest-Version: 1.0

Bundle-ManifestVersion: 2

Bundle-Name:

Bundle-Symbo

Bundle-Versio

Bundle-ClassP

Bundle-Requir

Web-ContextPa

Import-Packag

javax.el;*versio*

javax.servlet;*ve*

javax.servlet.a

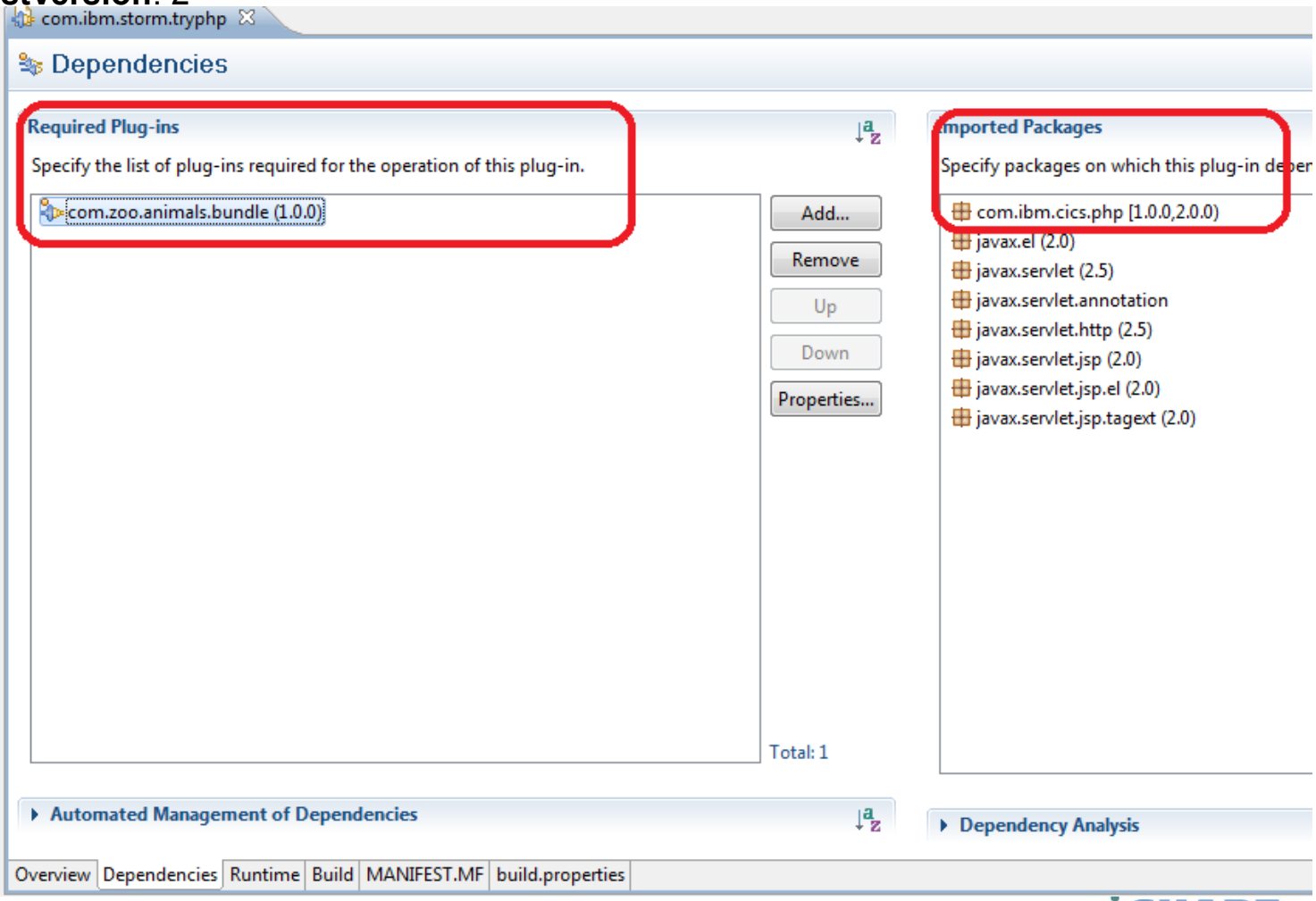
javax.servlet.ht

javax.servlet.js

javax.servlet.js

javax.servlet.js

Require-Bundl



Change Web.xml in PHP project

```
<web-app id="com.ibm.cics.server.examples.php.web" version="3.0" xmlns="http://java.sun.com/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd">
  <display-name>com.ibm.cics.server.examples.php.web</display-name>
  <welcome-file-list>
    <welcome-file>index.php</welcome-file>
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.htm</welcome-file>
    <welcome-file>index.jsp</welcome-file>
</welcome-file-list>
  <servlet-mapping>
      <servlet-name>PHPServlet</servlet-name>
      <url-pattern>*.php</url-pattern>
  </servlet-mapping>

  <servlet>
    <servlet-name>PHPServlet</servlet-name>
    <servlet-class>com.ibm.cics.php.PHPServlet</servlet-class>
    <multipart-config/>
  </servlet>
</web-app>
```

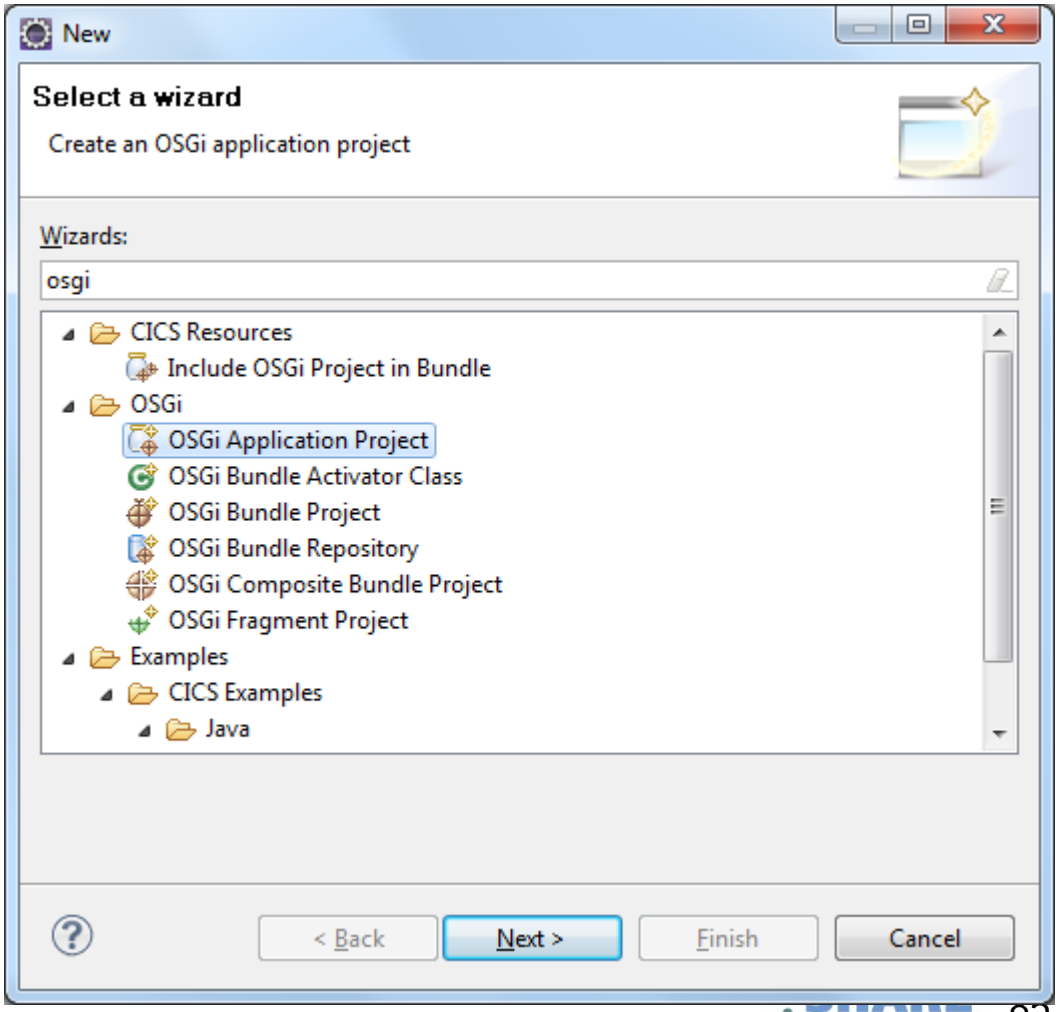
Map all the *.php requests to PHPservlet which is provided by DS 2.0 feature pack, used to handle all the PHP requests

PHPServlet definition in package com.ibm.cics.php

Package as Enterprise Bundle Archive (EBA)

Create ‘OSGi Application Project’ to package your web project and OSGi project

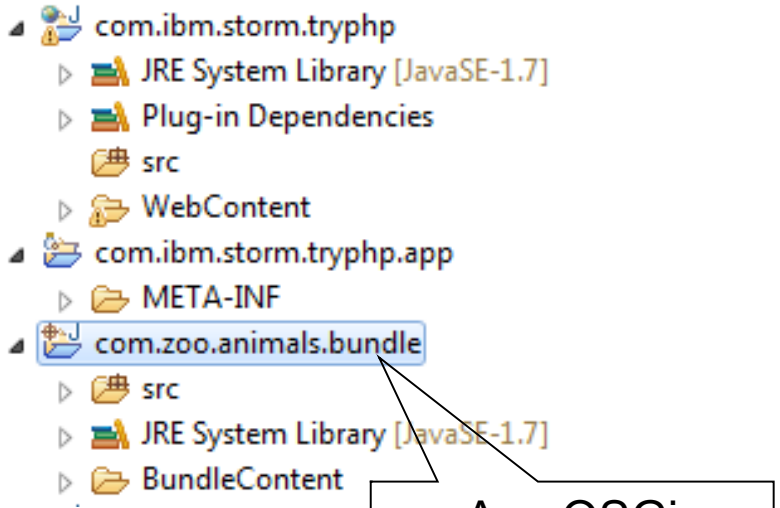
EBA is the only package CICS DS 2.0 supports



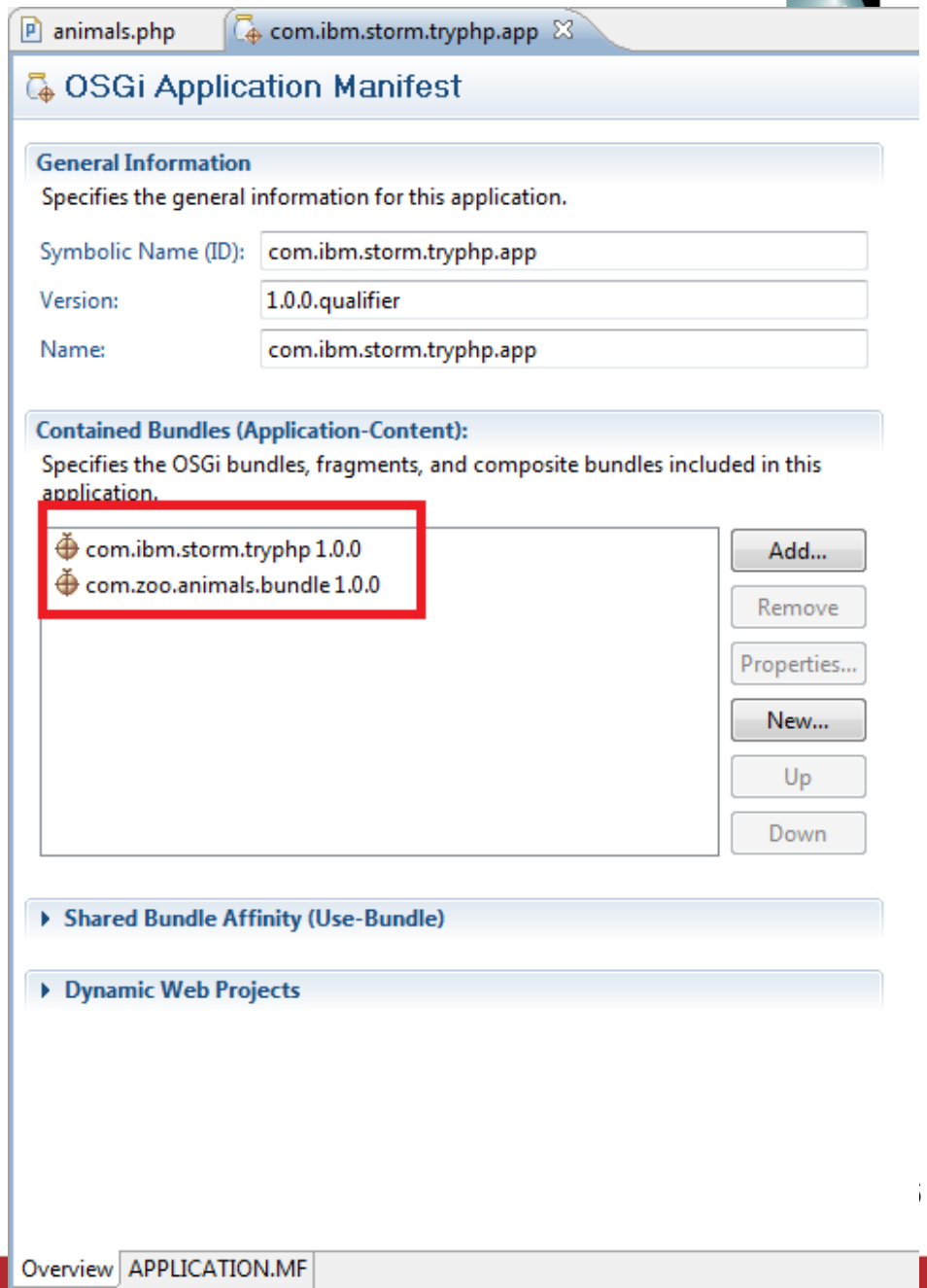
Add bundle into EBA

Put your PHP project into the EBA project.

Put all the OSGi project you want to refer into EBA project.



Any OSGi bundle which is used in the PHP project



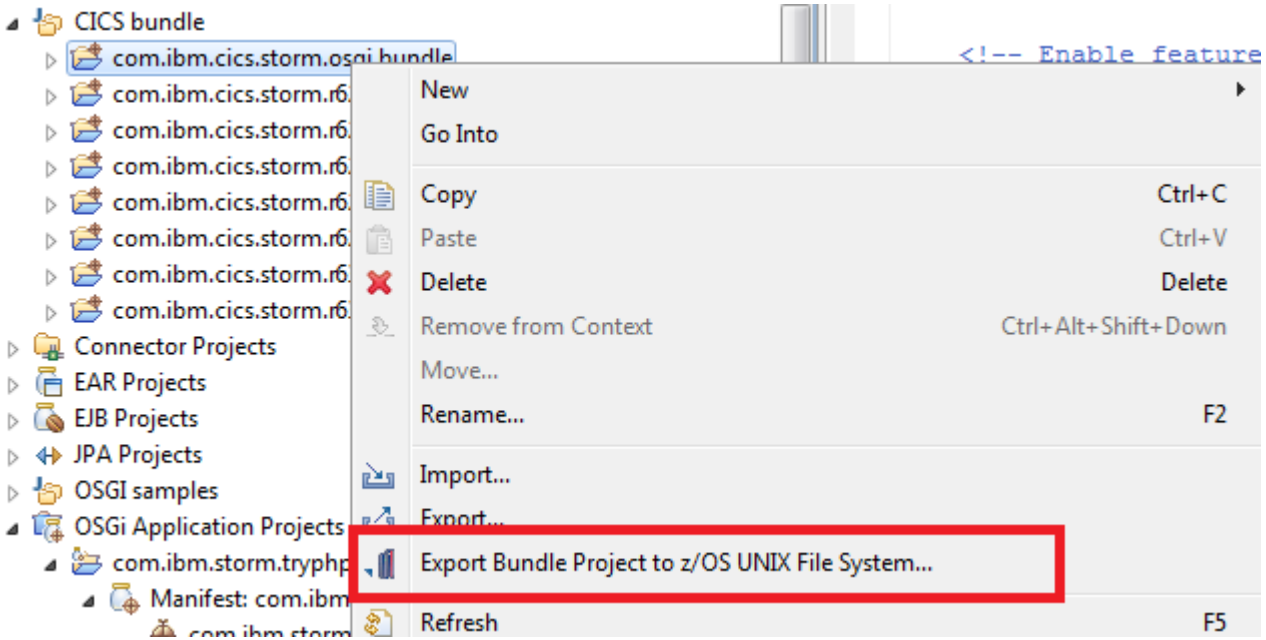
Application.MF

```
Application-Name: com.ibm.storm.tryphp.app
Application-SymbolicName:
 com.ibm.storm.tryphp.app
Application-ManifestVersion: 1.0
Application-Version: 1.0.0.qualifier
Manifest-Version: 1.0
Application-Content:
 com.ibm.storm.tryphp;version="1.0.0",
 com.zoo.animals.bundle;version="1.0.0"
```

Export to z/OS USS

Create CICS Bundle project which contains EBA bundlepart

Export CICS Bundle project to zFS



All the works in CICS explorer done

1. setup target runtime in CICS explorer
2. Create Web application contains PHP
3. Create OSGi application project to package PHP project as EBA
4. Create CICS Bundle project to manage lifecycle of PHP application
5. deploy CICS Bundle to z/OS USS

Next step

configure runtime in CICS if it is the first PHP application in your CICS region

Install bundle to start your PHP application

Configure Liberty to run PHP

update Liberty server.xml
Add CICS PHP jar into bundle repository

```
 <bundleRepository >
        <fileset dir="USS_HOME/wlp/bundlerepository"
includes="com.ibm.cics.php_1.0.0.jar"/>
 </bundleRepository>
```

Update JVMProfile
  update LIBPATH_SUFFIX for php extensions
- LIBPATH_SUFFIX =USS_HOME/lib/php

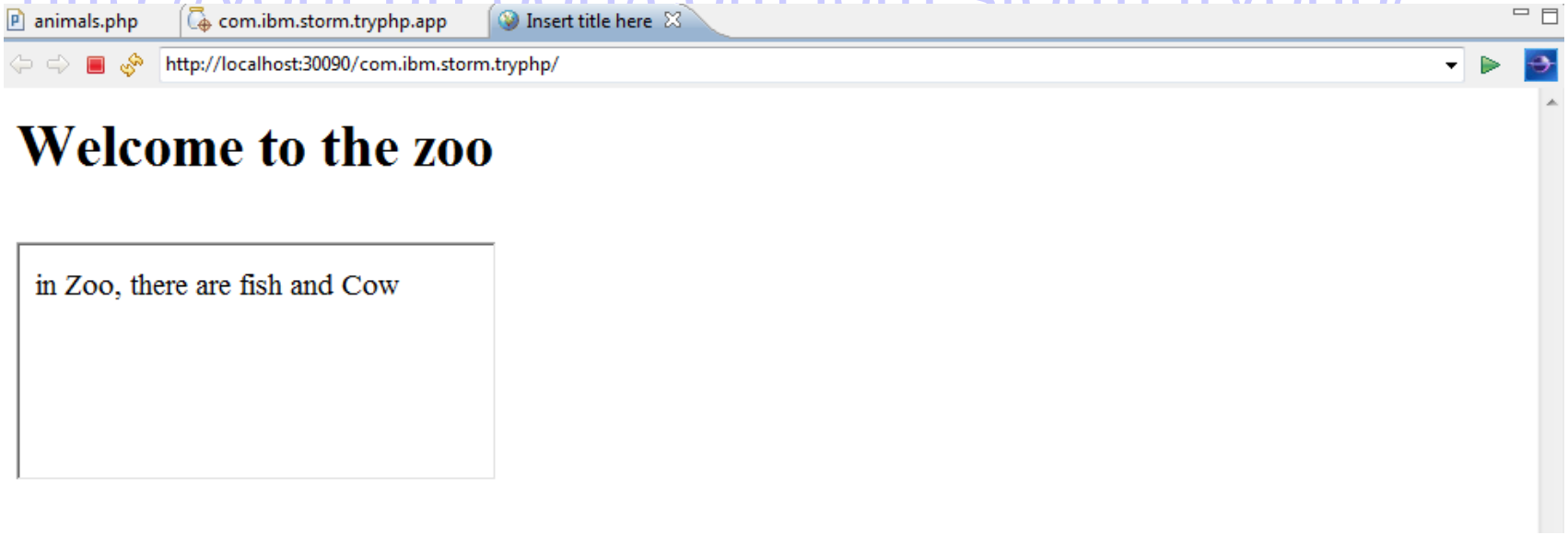
Start Liberty Profile in CICS

Install CICS bundle, test

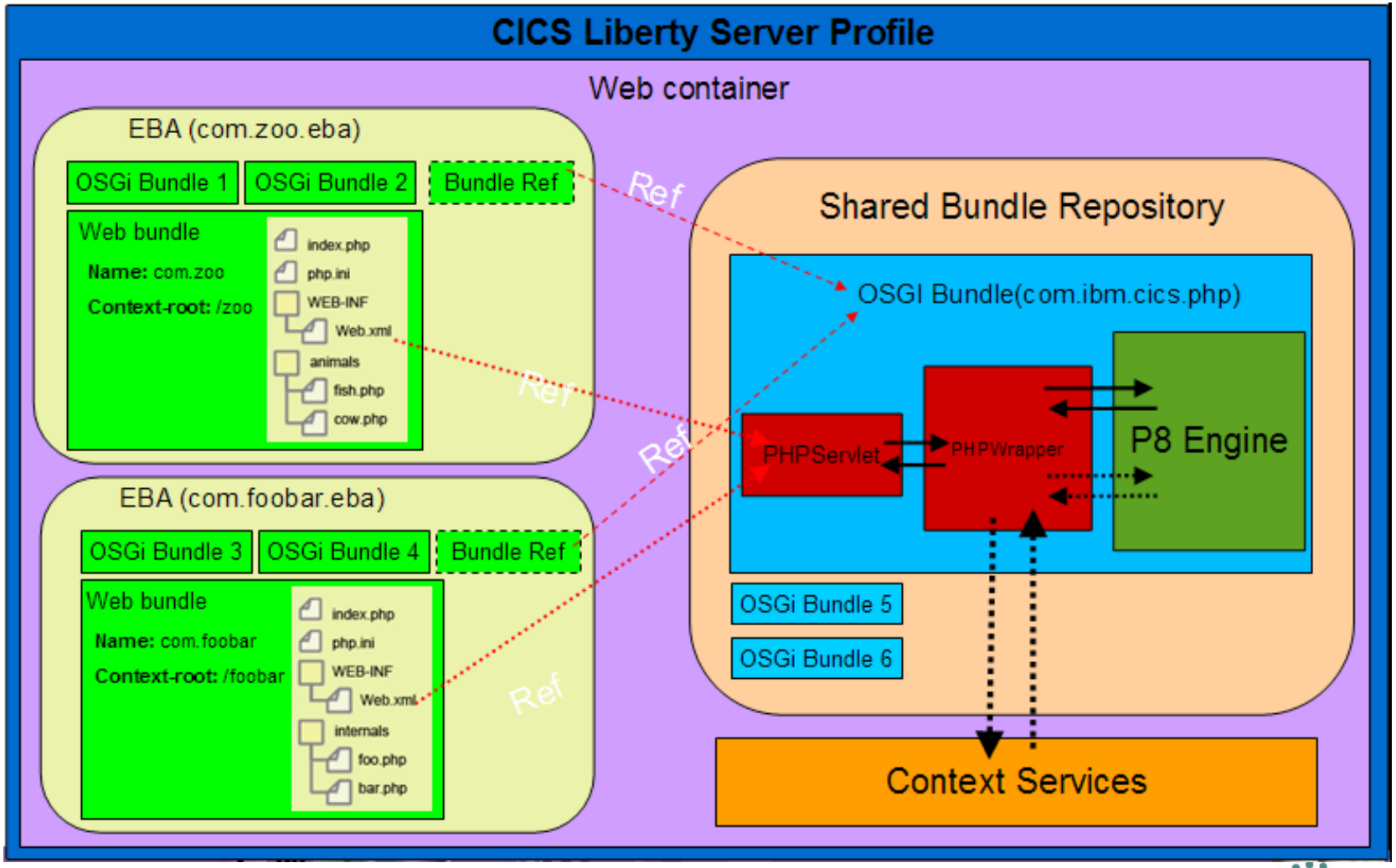
Install bundle to deploy EBA into Liberty

Input URL

http://your.url:port/com.ibm.storm.tryphp/



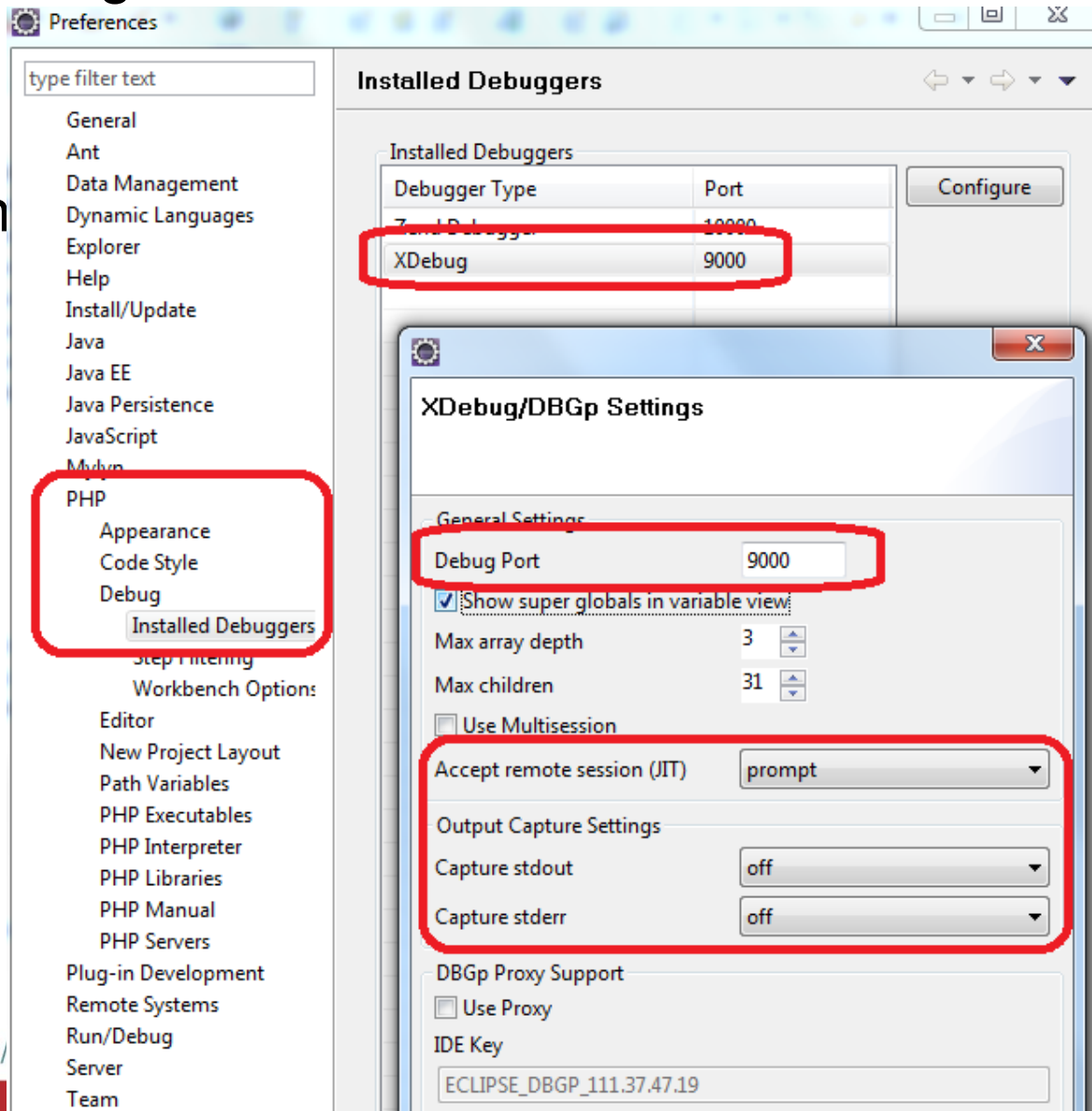
Overview of PHP in CICS



Debug

Debugger in PDT 2.1 or higher

XDebug
Port
Accept remote session

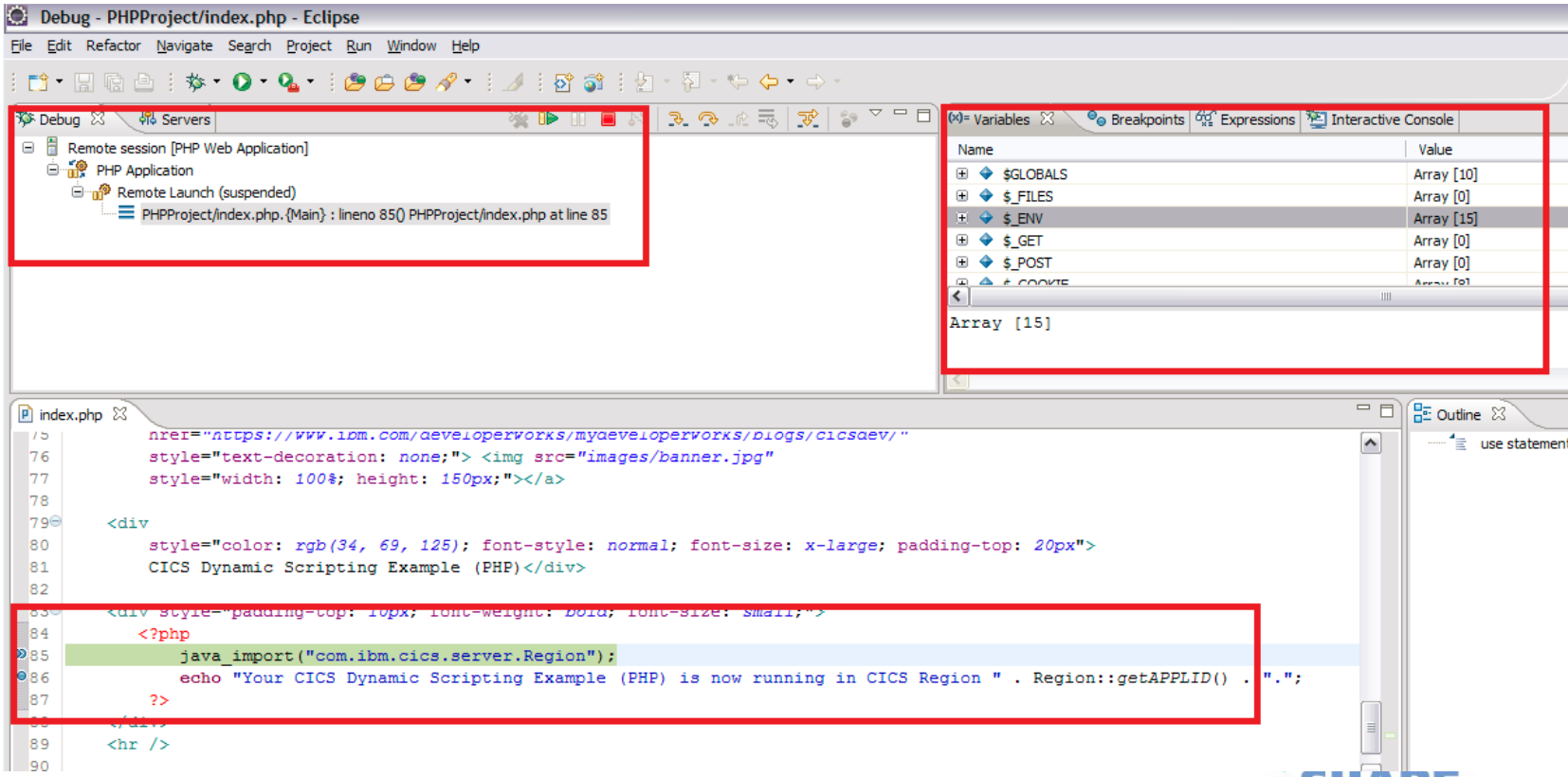


Debug - JVMProfile

Update JVMProfile

-Dp8.debug=idekey=ECLIPSE_DBGP&remotePort=port&

remoteHost=ip_address



Migration from previous versions (CICS dynamic feature pack 1.0,1.1)

Understand OSGi framework

repackage the PHP from the sMash
copying the PHP files into a directory inside the example WAB

Repackage Java application as OSGi bundle
Define it as dependency in PHP project.

CSS, java script
Copy into Web application project.

Limitation
Don’t support all the Smash functions

- ZRM
- CLI
- Ivy

Summary

Simplicity and speed

Reuse

Strong fundation and good expandability

CICS Performance and Optimisation

Please Note... ESPECIALLY HERE!!!

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM's sole discretion.

Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

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.

Some time in the future...

"CICS performs 5% faster out-of-the-box (for most workloads)"

"CICS can handle a 25% growth in your transaction rates"

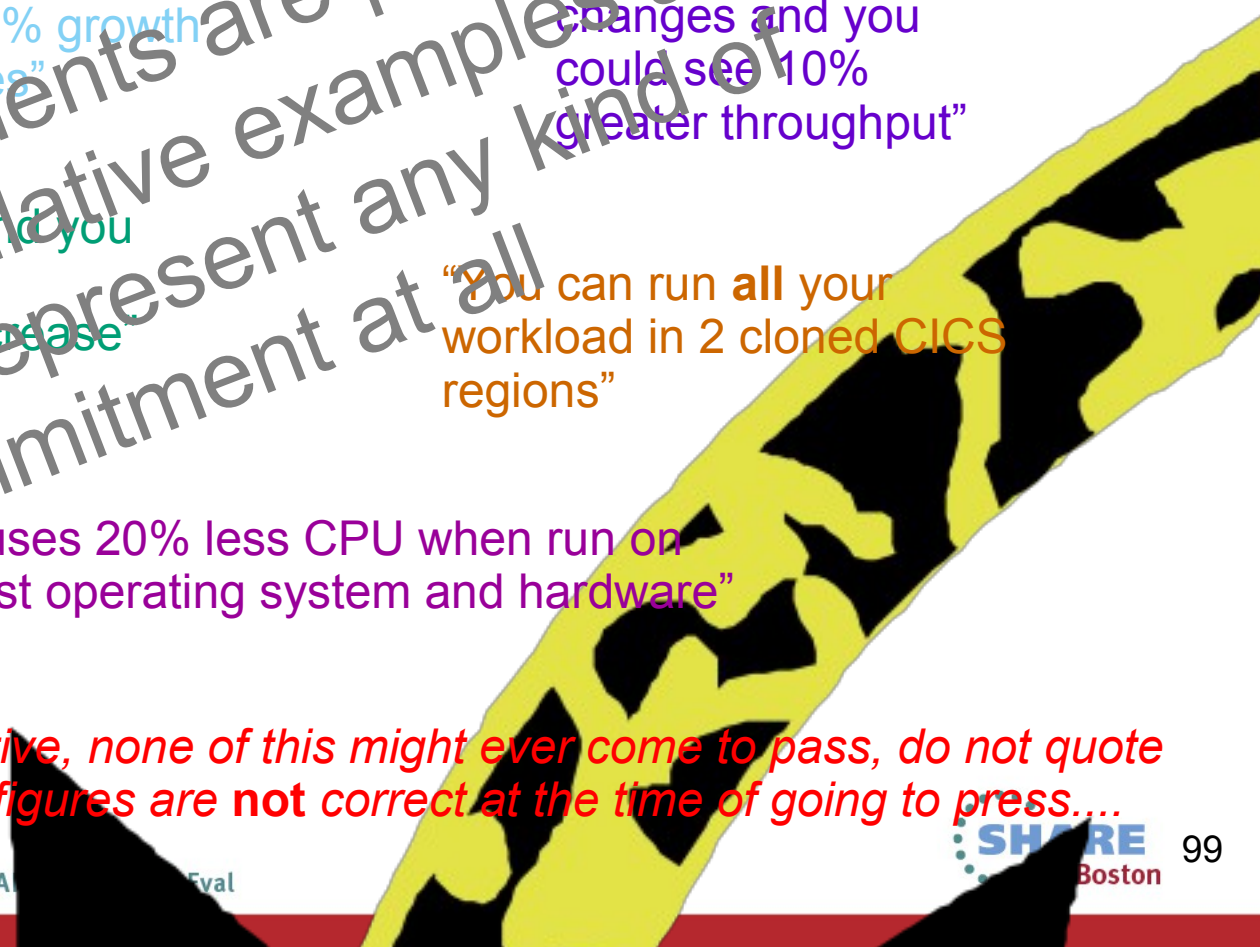
"Make these changes and you could see 10% greater throughput"

"Work with us and you could get a 15% performance increase"

"You can run **all** your workload in 2 cloned CICS regions"

"CICS uses 20% less CPU when run on the latest operating system and hardware"

Statements are purely speculative examples and do not represent any kind of commitment at all



Disclaimer: *Purely speculative, none of this might ever come to pass, do not quote me, your mileage* **will** *vary, figures are* **not** *correct at the time of going to press....*

CICS Performance & Optimizati… Focus

CICS has always focussed on improving perform…, but now...

- We want to look to deliver more o… …ge in perf…ce
- Some of it "out of the box", wi… …ng lots of c… the benefit

What do we mean by "performa… Good question.

- Gathering input from …rs → questionnaire on what aspects… performance are im… what areas of CICS
- tinyu…m/CIC… …mance

Understand wh… …atter t… …tomers

Aim to improve… …kloads

- Coul… …ud… …ation, partnering wi… stack… …ucts,

This is a fut… …sed effort ... …n't exp… …g soon

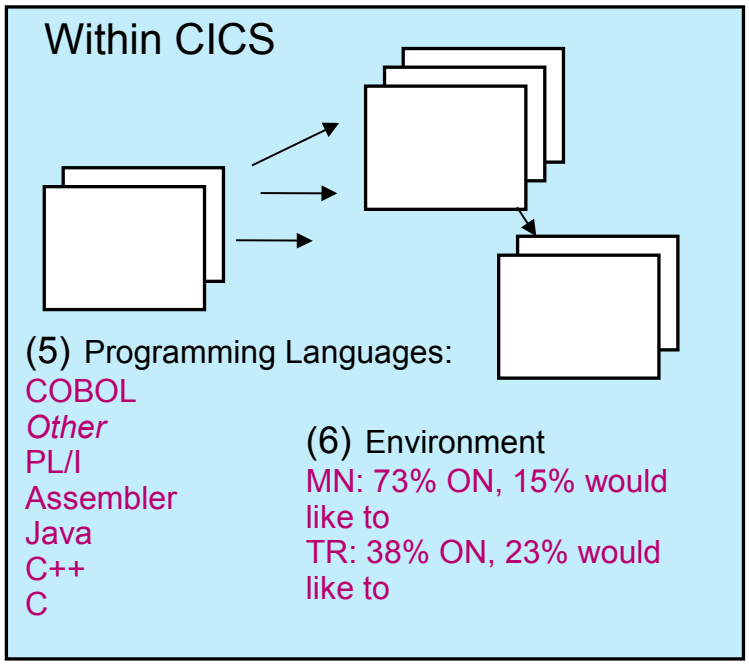
CICS Performance Questionnaire – early feedback

(1) Importance: Response Time > CPU > Throughput > Virtual storage use

(2) Current Ways into CICS:
- MQ
- Web services
- 3270
- CICS Sockets
- CICS TG
- LU6.2
- Web HTTP
- *Other*

Workload distribution

(3) Growing Ways into CICS:
- Web services
- MQ
- CICS TG
- Web HTTP
- CICS Sockets
- *Other (e.g. Java clients, IPIC)*
- 3270
- LU6.2



(4) Resource Manager Access
- DB2
- VSAM
- *Other*
- IMS

Requests out

(7) Constraints:
- CPU
- *Other (apps, encryption)*
- Contention (Locks)
- Data I/O
- 31-bit storage
- Outbound Requests
- 24-bit storage

Hot buttons (7, 8, 9)
- Solve poor application design
- More 64-bit use (for apps)
- CPU reduction
- RLS Performance
- Web services performance
- More speciality engine use
- Reduce DB2 contention
- SSL
- More self-healing

*These are very **preliminary** findings from 35 responses: **you** can input to this:*

tinyurl.com/CICSPerformance