



CICS Transaction Server V4.2

9606: CICS ... It's Not Just COBOL: Java Support

Matthew Webster

matthew_webster@uk.ibm.com

Important disclaimer

The information in this presentation is provided for information purposes only.

While efforts were made to verify the completeness and accuracy of the information in this presentation, it is provided “as is”, without warranty of any kind, express or implied.

In addition, this information does not form part of IBM’s current product plans and strategy, which are subject to change by IBM without notice.

IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this presentation or any other documentation.

Nothing contained in this presentation is intended to, or shall have the effect of:

- *Creating any warranty or representation from IBM (or its affiliates or its or their suppliers and/or licensors); or*
- *Altering the terms and conditions of the applicable license agreement governing the use of IBM software.*

IBM values your participation and suggestions but makes no commitment to any future product announcement or specific future product content.

Themes & Topics

- **Languages**
- **Environments**
- **Tools**

Motivation

- **Maturity**

- Am I taking full advantage of the capability CICS provides?
- Am I using the right application programming language(s) and environment(s)?
- Am I using the right tools?

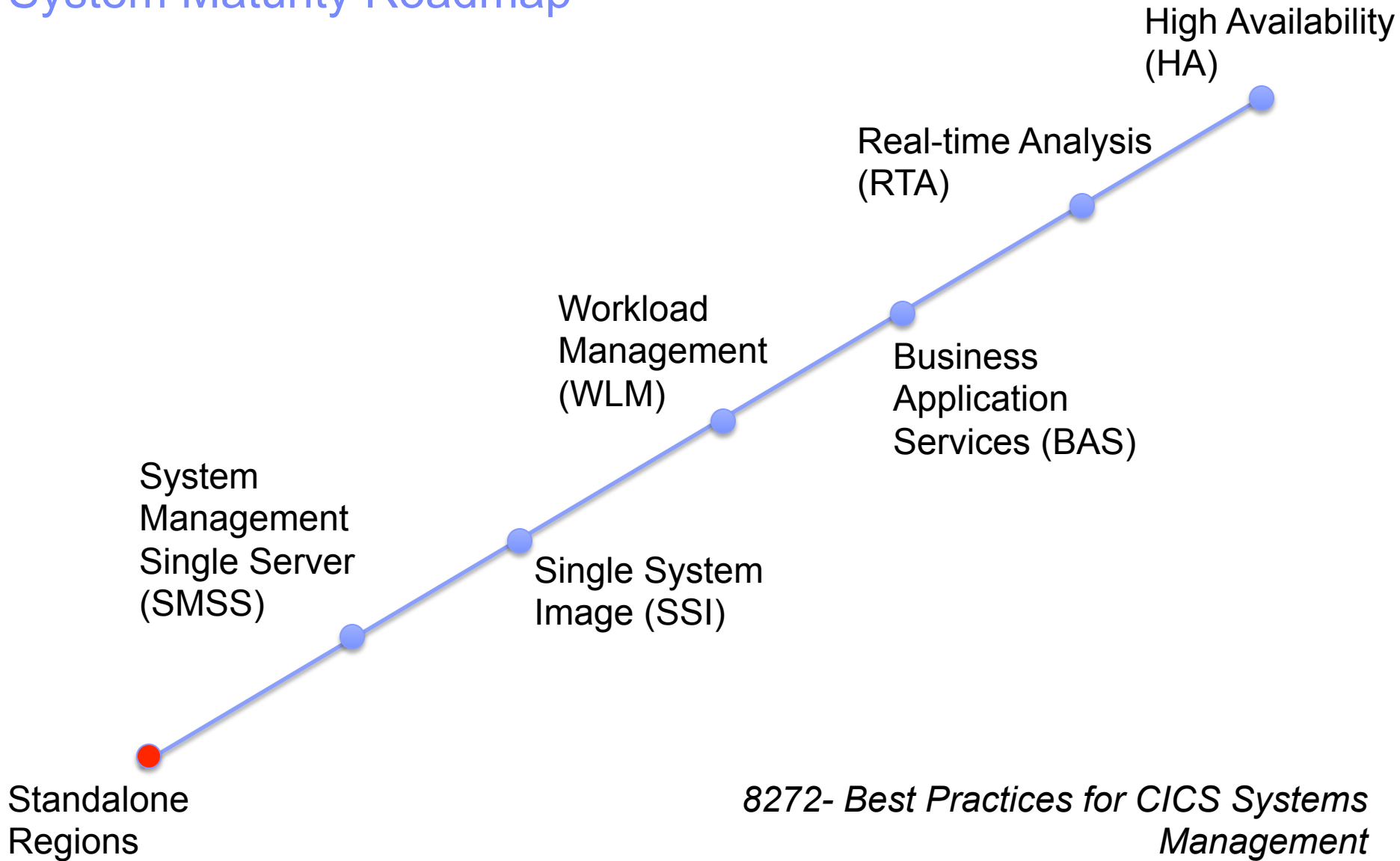
- **Consumability**

- Am I making the best use of skills?
- Am I providing good value to my stakeholders (developers, consumers)?
- **9600: CICS TS and the Cloud**

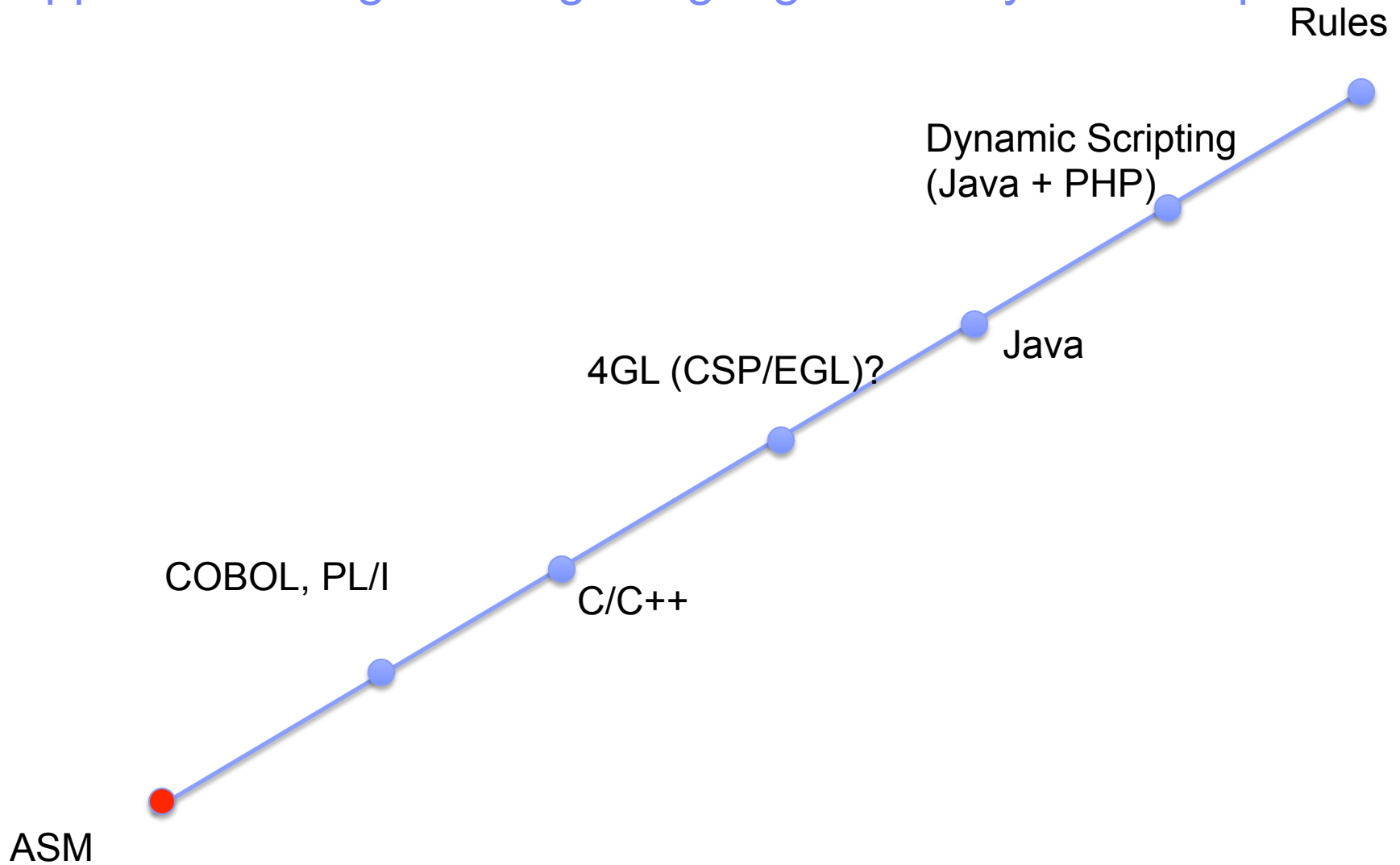
- **Agility**

- How quickly can I bring a new application or version of an application to production?
- How quickly can I repurpose an existing application?

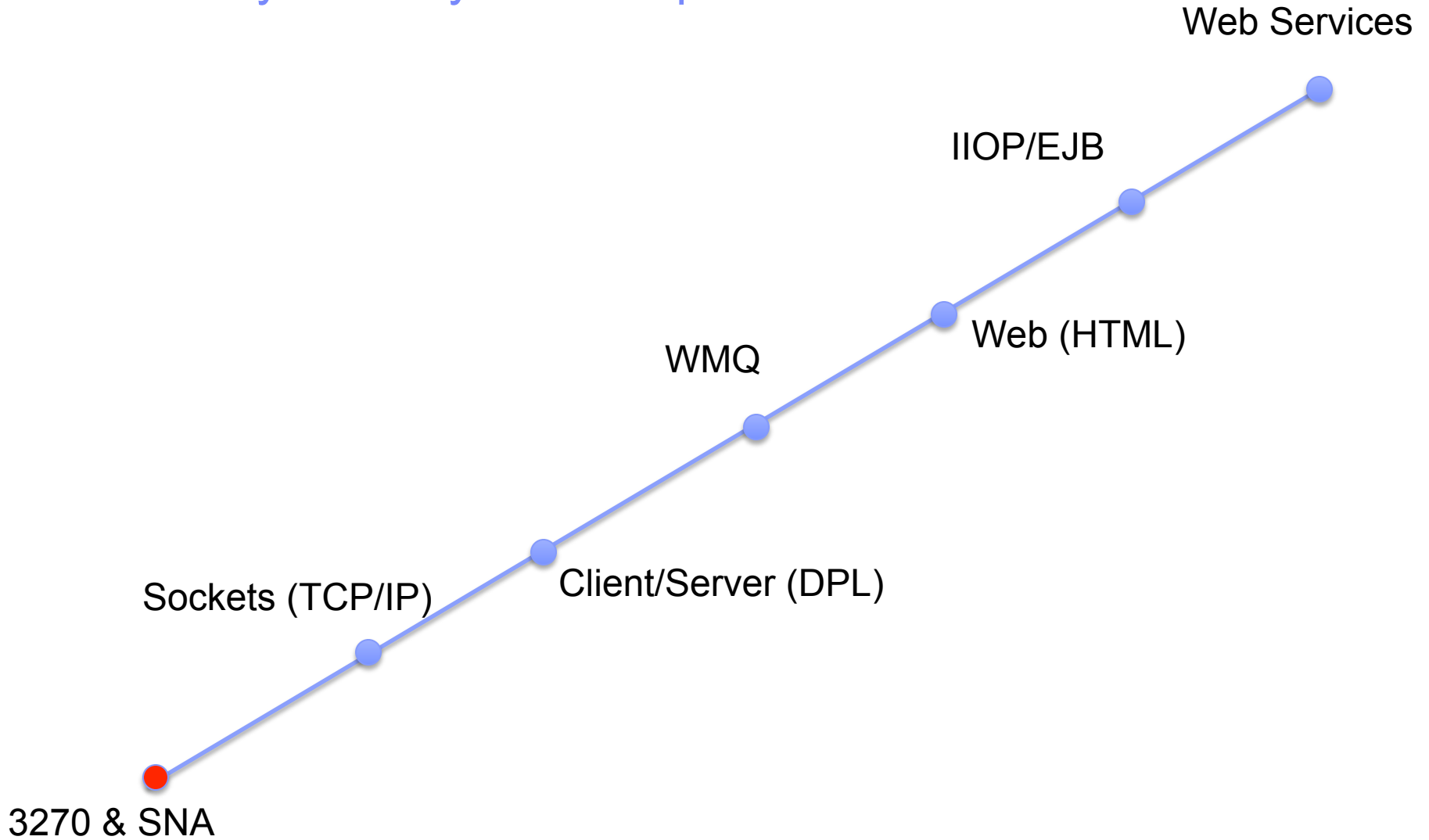
System Maturity Roadmap



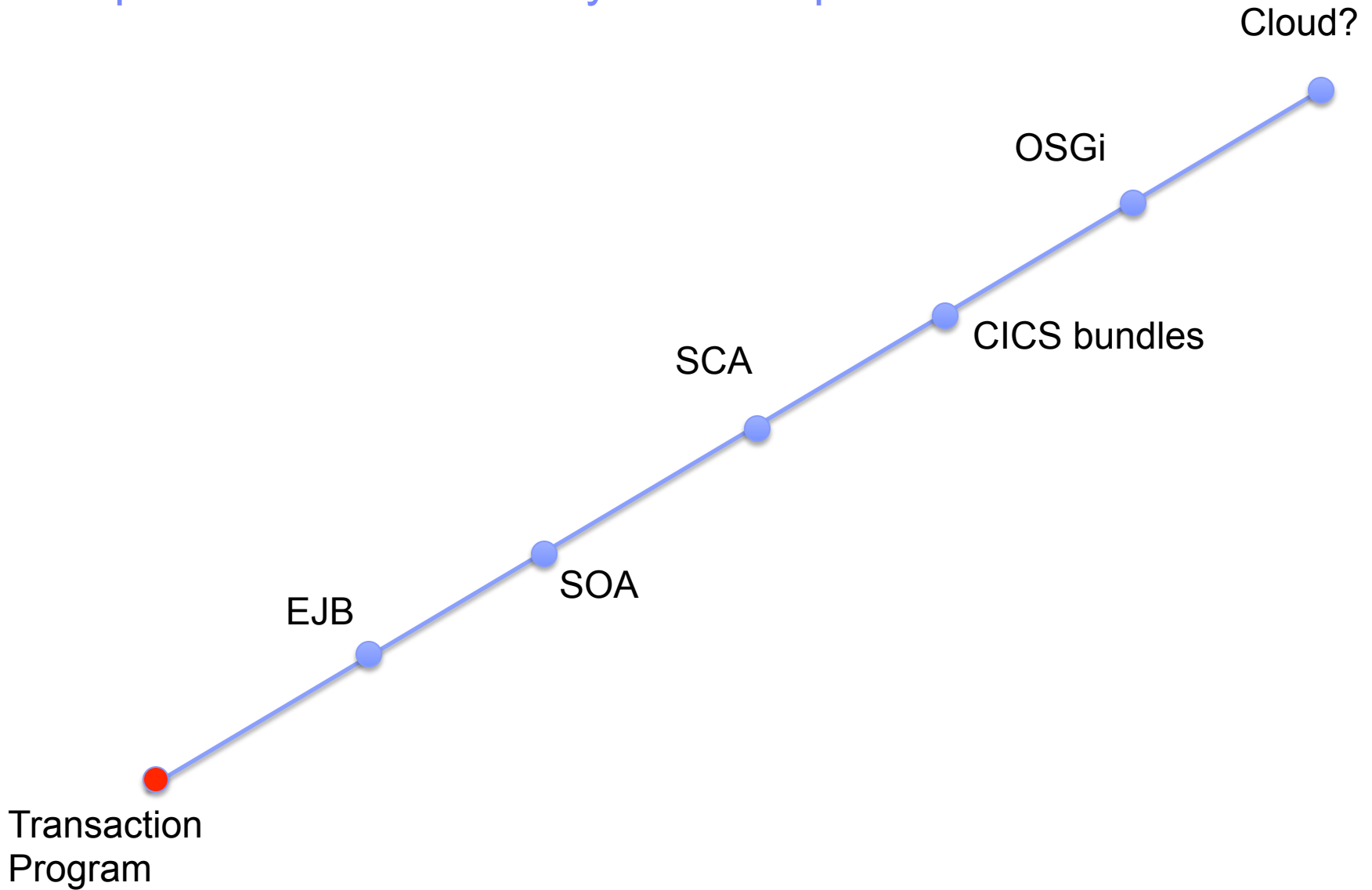
Application Programming Language Maturity Roadmap



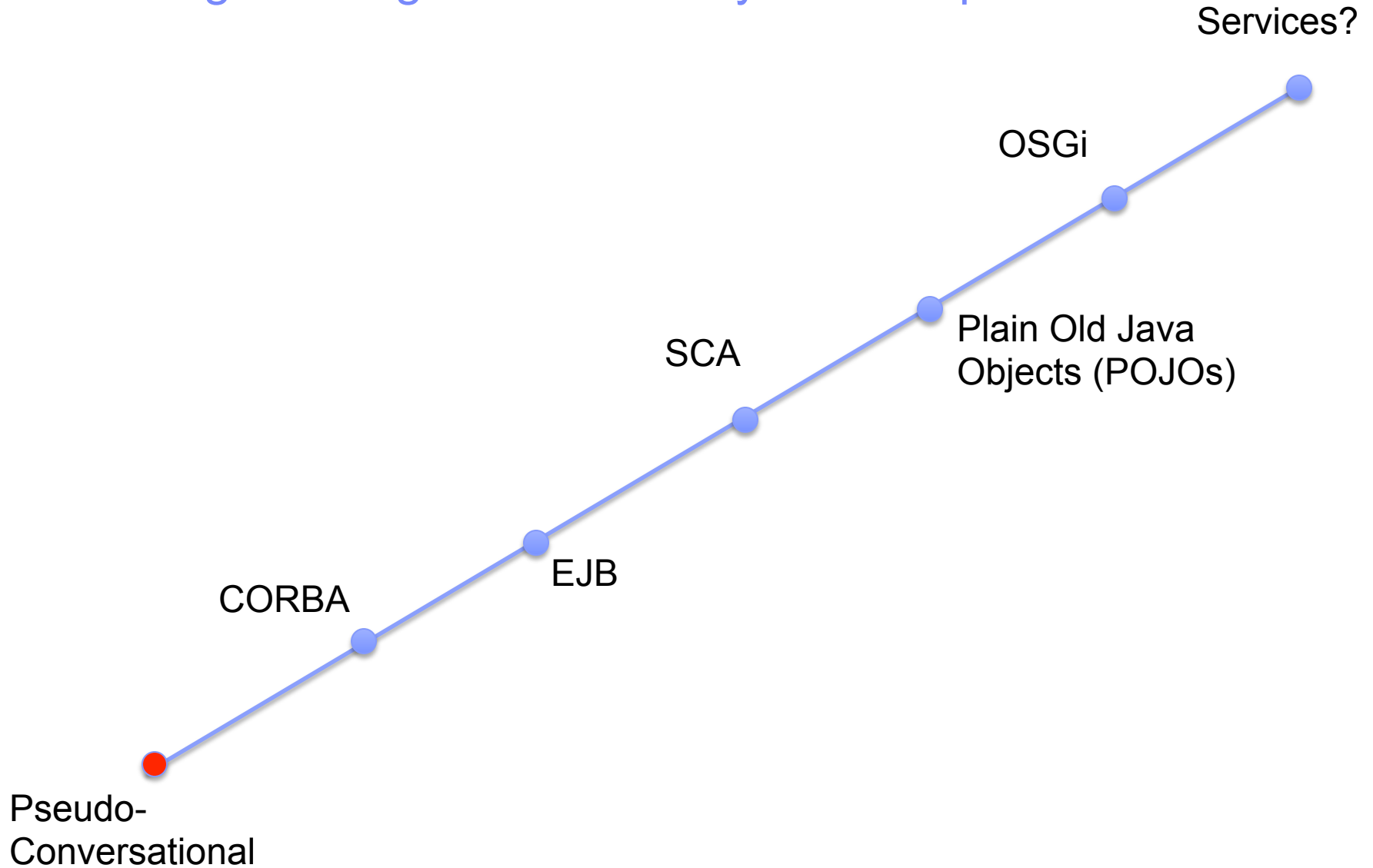
Connectivity Maturity Roadmap



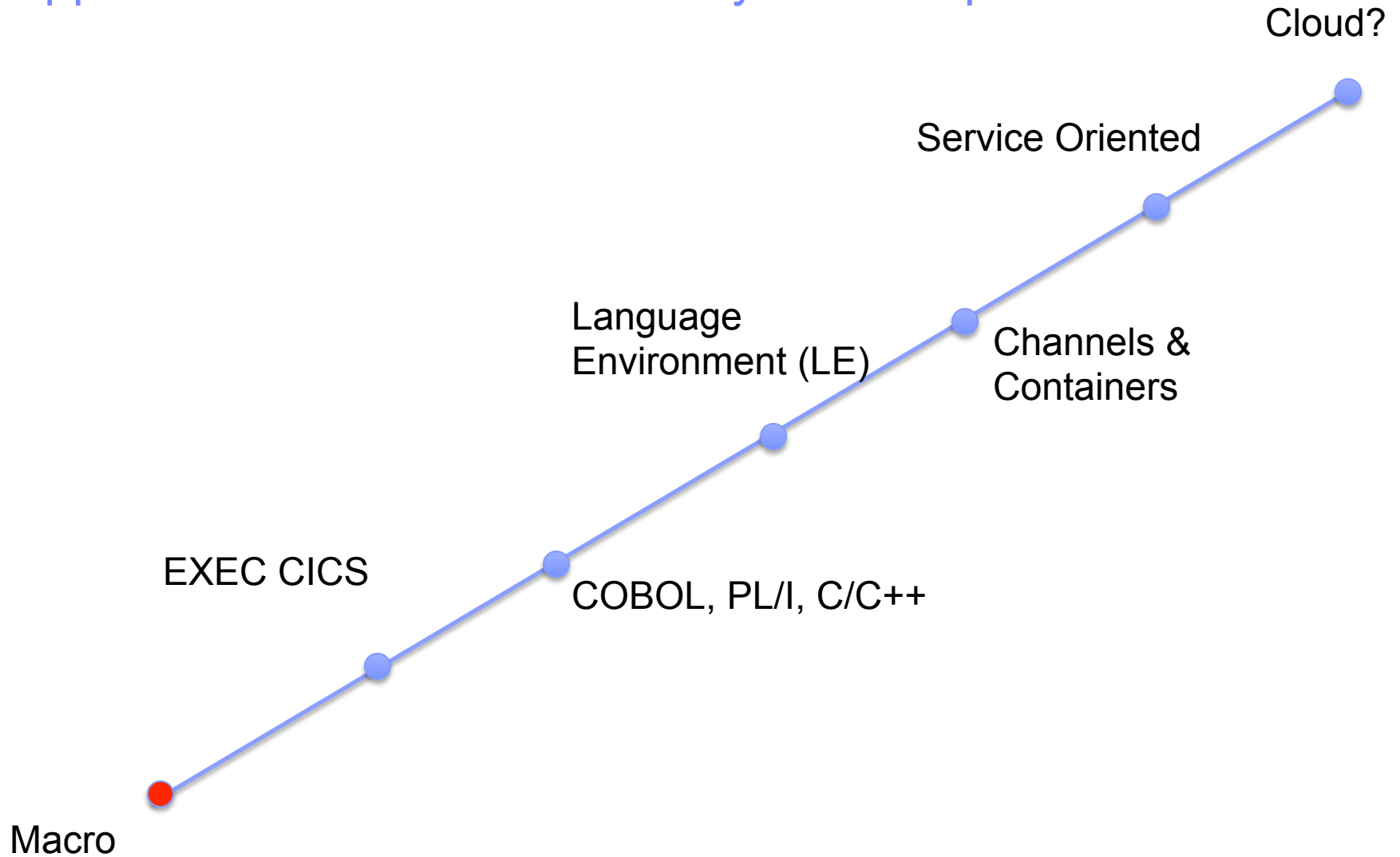
Component Model Maturity Roadmap



Java Programming Model Maturity Roadmap



Application Environment Maturity Roadmap



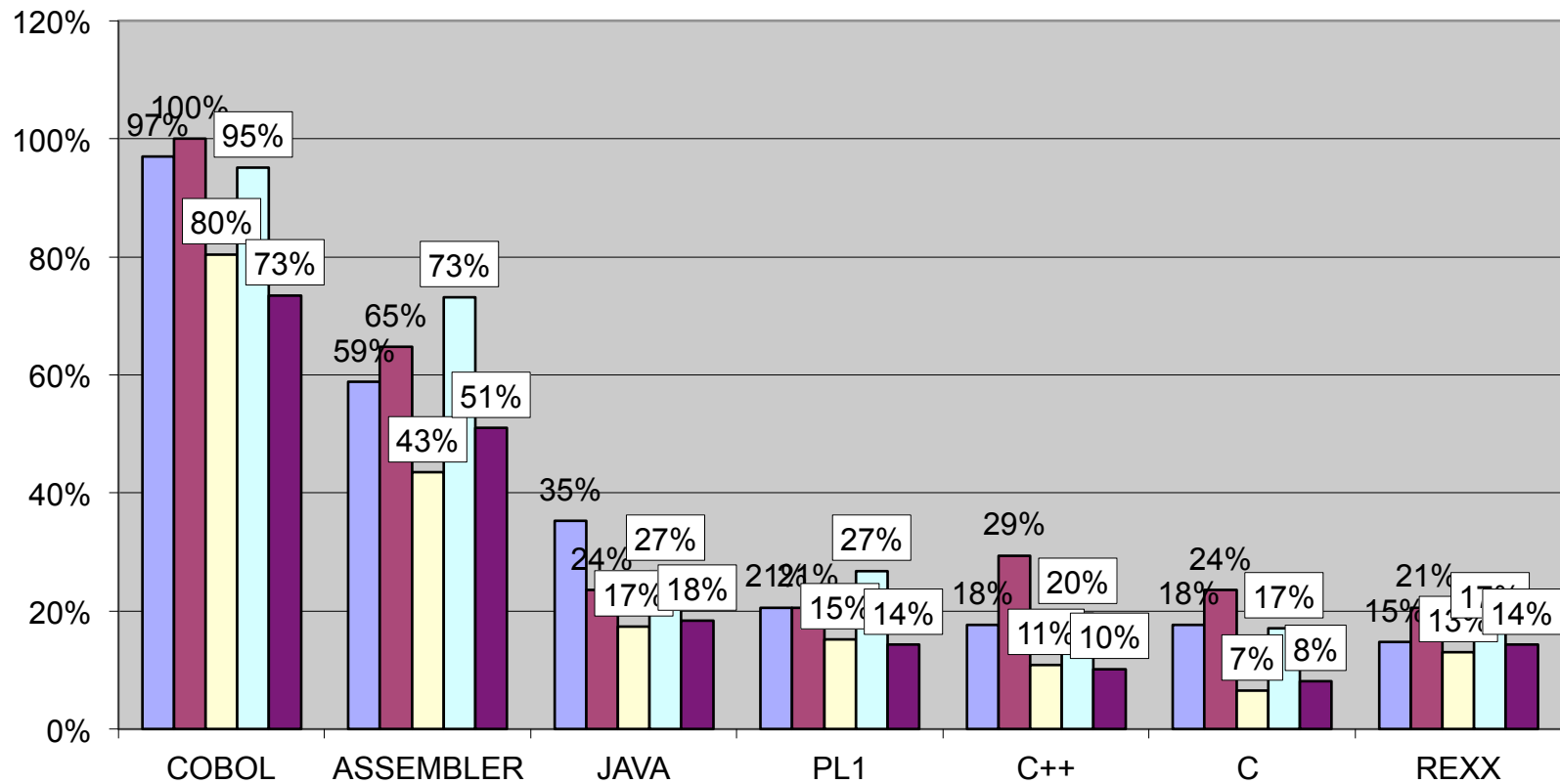
Languages

Languages

- **ASM**
- **COBOL, PL/I, C/C++**
- **REXX**
- **Java**
- **Dynamic Scripting (PHP)**
- **Rules**
- **Event Processing**
- **Anything that I missed?**

SHARE Questionnaire

Languages %



ASM

■ Questions

- Who is using still ASM for applications rather than just user exits?
- Why (legacy, performance, ...)?

■ Characteristics

- High performance
- High skill

COBOL (, PL/I & C/C++)

- **Questions**

- Does anyone use more than one?

- **Characteristics**

- Right balance between performance and skill
- Right balance between abstraction and control
- Full access to EXEC CICS API & SPI
- Well understood performance characteristics
- Mature lifecycle (SCM) & problem determination tools
- Increasingly scarce/expensive skill

REXX

■ Questions

- Is anyone using REXX online?
- Is REXX used for API or just SPI?
- What about CICSplex SM?
- Has anyone looked at Dynamic Scripting (PHP)?

■ Characteristics

- The first dynamic scripting
- Great for automation

■ References

- **9327: [CICSplex What Is It, and Why Do I Care?](#)**

Java

▪ Questions

- Who has Java in production today?
- Who is considering Java for the future & why?

▪ Characteristics

- Probably still the most widely used programming language
- (Perceived) low performance
 - but can now match COBOL with zAAP
 - Half of Watson processes implemented in Java
- Highly sophisticated (free) development tools
- Less familiar lifecycle (SCM), problem determination & management tools
- Only a subset of the CICS API & SPI (JCICS)
 - but Access to wide range of high quality 3rd party libraries & frameworks
- Widely available skills
 - transferable from other platforms e.g. WAS

▪ References

- <http://share.confex.com/share/116/webprogram/Session8274.html>
- **9607: CICS for Java Developers and Java for System Programmers - Two Sides of the Same Coin**
- **9575: Beyond Watson: Technology Implications Today and In the Future**

Java

- **Supported since CICS TS V1.3 (JDK 1.1 & HPJ)**
- **Updated in CICS TS V4.2 (Java 6 + JVM server + OSGi)**
- **Environments**
 - High Performance Java (HPJ)
 - JVM pool
 - Batch (Compute Grid & JZOS)
 - JVM server
- **JRE**
 - Resettable JVM & shared classes
 - Java 5 (since CICS TS 3.2) concurrent & type accurate GC
 - 64-bit (since CICS TS V4.2)

Dynamic Scripting

- **Supportpac CA1S: REST support in CICS using PHP for use with CICS TS for z/OS V3**
 - Just PHP
- **CICS TS and WebSphere Application Server Feature Packs for Dynamic Scripting (V4.1)**
 - Full sMASH (PHP, Groovy & Java)
- **Characteristics**
 - Ideal for “situational” applications
 - Currently better suited to the “long tail”
 - Emphasis on presentation layer & Web 2.0
 - Presentation & business logic mixed
 - Only a subset of the CICS API & SPI (JCICS)
 - but can also use Java libraries & frameworks
 - Cloud-style lifecycle
 - Full JDBC
 - simplified ZRM (Zero Resource Model)
 - PHP Data Objects (PDO)
 - PHP leverages libraries that are compiled C code
 - a bit like Java was used to access existing (high performance) COBOL in the early days
- **References**
 - **9608: [CICS Dynamic Scripting](#)**
 - <http://www.ibm.com/support/docview.wss?rs=1083&uid=swg24021196>
 - <http://www.ibm.com/software/htp/cics/scripting/>

CICS Dynamic Scripting

Strategic, long-lived applications

Strategic Value, Cost, Complexity, Usage

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**

Enterprise Applications

Departmental Applications

Team/Project Applications

Personal Applications

Time to value is more important than enduring value

Java EE / CICS Traditional

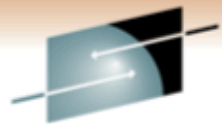
WebSphere sMash & CICS Dynamic Scripting

Number of applications

Rules

- **Supportpac CA0A: CICS channels and container support utility for ILOG Rules for COBOL**
- **Characteristics**
 - Business logic in rules not code
 - Visual development
 - Opportunity for end user programming & testing
 - Cross platform reuse
 - Generates linkable COBOL program
 - Agility
- **References**
 - <http://share.confex.com/share/116/webprogram/Session8282.html>
 - <http://www.ibm.com/support/docview.wss?rs=1083&uid=swg24024202>

Rules Authoring Delivered to Business



SHARE
Technology • Connections • Results

Eligibility.xlsx - Microsoft Excel

A	B	C	D	E	F
A	City	Min	Max	D	E
		≤ 22.00		FALSE	The customer's age is below the minimum for rent
New Hampshire	X	23	70	TRUE	The customer is eligible to rent in New
		≥ 71.00		FALSE	The customer's age is past the maximum for rent
		≤ 20.00		FALSE	The customer's age is below the minimum for re
Rhode Island	A	21	70	TRUE	The customer is eligible to rent in Rho
		≥ 71.00		FALSE	The customer's age is past the maximum for re

Rules

ROW4

Definitions...

If

- all of the following conditions are true:
 - the state of the pickup branch of the rental agreement is New Hampshire
 - the last name of the customer starts with X
 - the age of the customer of the rental agreement is between 23 and 70

Then

- set the rental agreement accepted status to True
- display the message: The customer is eligible to rent in New Hampshire.

Else...



Word Tutorial - copied.docx - Microsoft Word

Page 3

RuleDoc Outline

- Compute the Base Rate
- Check the Eligibility for the Default
- Define the Pricing of the Default
- Check the Eligibility for the Long Term
- Define the Pricing of the Long Term

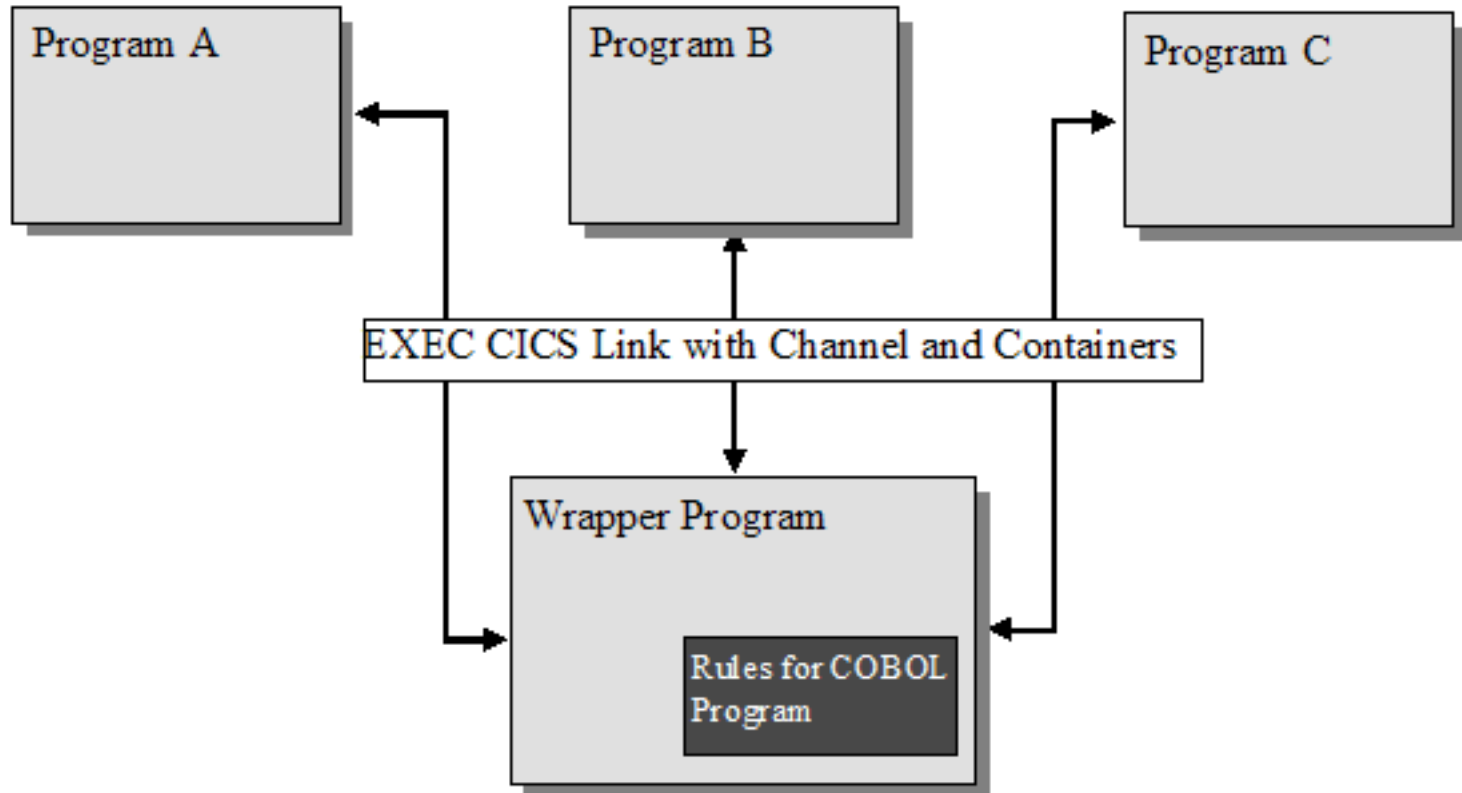
Rule Properties

Problem List

- The word "true" is expected in place



Rules for COBOL Generated Program



Event Processing

- **Available from CICS TS V4.1**
- **Enhanced in CICS TS V4.2**
- **Characteristics**
 - Orthogonal programming model
 - Product integration
 - Non-invasive
 - Agility
- **References**
 - **9330: [CICS Event Processing](#)**

Environments

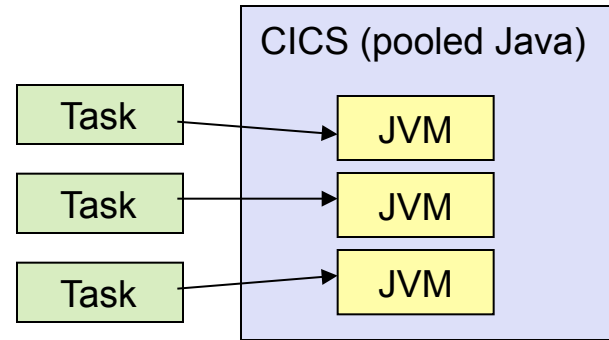
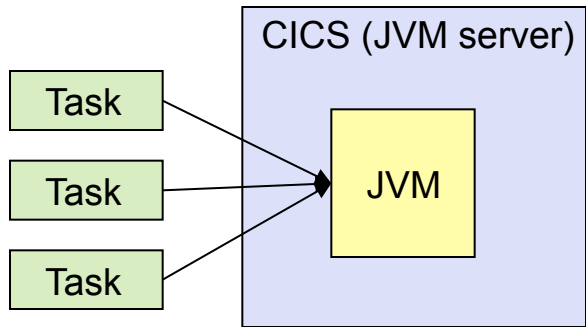
Batch

- **Online & offline**
- **Dual mode**
 - COBOL
 - Java (JZOS & JCICS)
- **EXCI**
- **Compute Grid**

Pooled JVM vs. JVM Server

- **Pooled JVM**
 - Discontinued in future release
- **JVM Server**
 - CICS V4.1: Dynamic Scripting & Compute Grid only
 - CICS V4.2: Any CICS application, OSGi development/deployment/management
- **References**
 - 9607: [CICS for Java Developers and Java for System Programmers - Two Sides of the Same Coin](#)
 - <http://www.ibm.com/software/htp/cics/tserver/v42/>

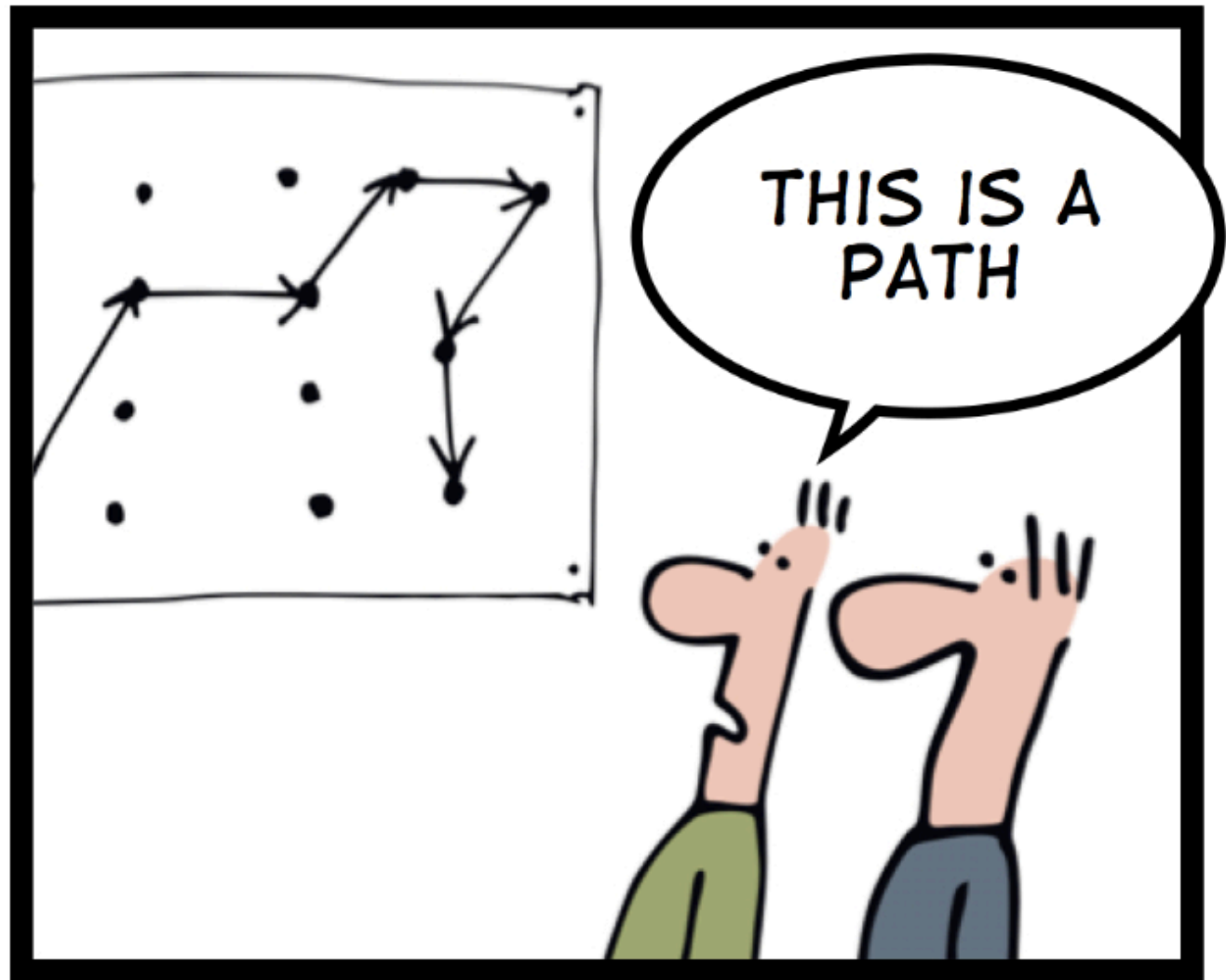
JVM server vs existing Java support



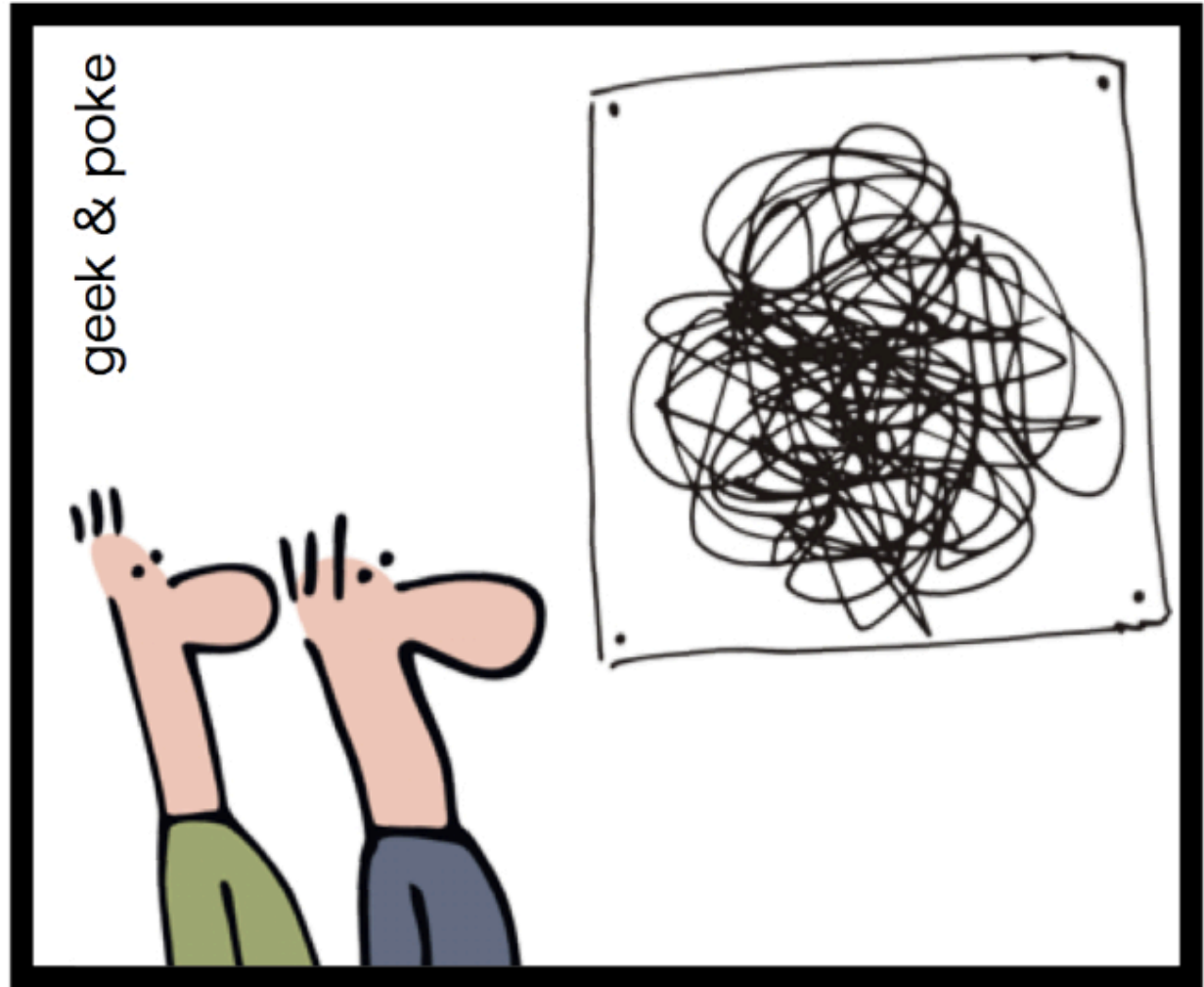
Single JVM - serves many tasks (reduced storage) (concurrent, multi-threaded, up to 256 threads per JVM server)	Pool of JVMs - each serves only a single task. Java Program Isolation
T8 (CICS key)	J8 (CICS key), J9 (User key)
MAXTHRDTCBS (automatically calculated), up to max of 1024 per region	MAXJVMTCBS, SIT parm
More standard Server model (+ data-sharing)	Difficult, convoluted to share data and state.
Dynamic update and replace of modules	JVMs must be restarted to effect changes

Why OSGi?

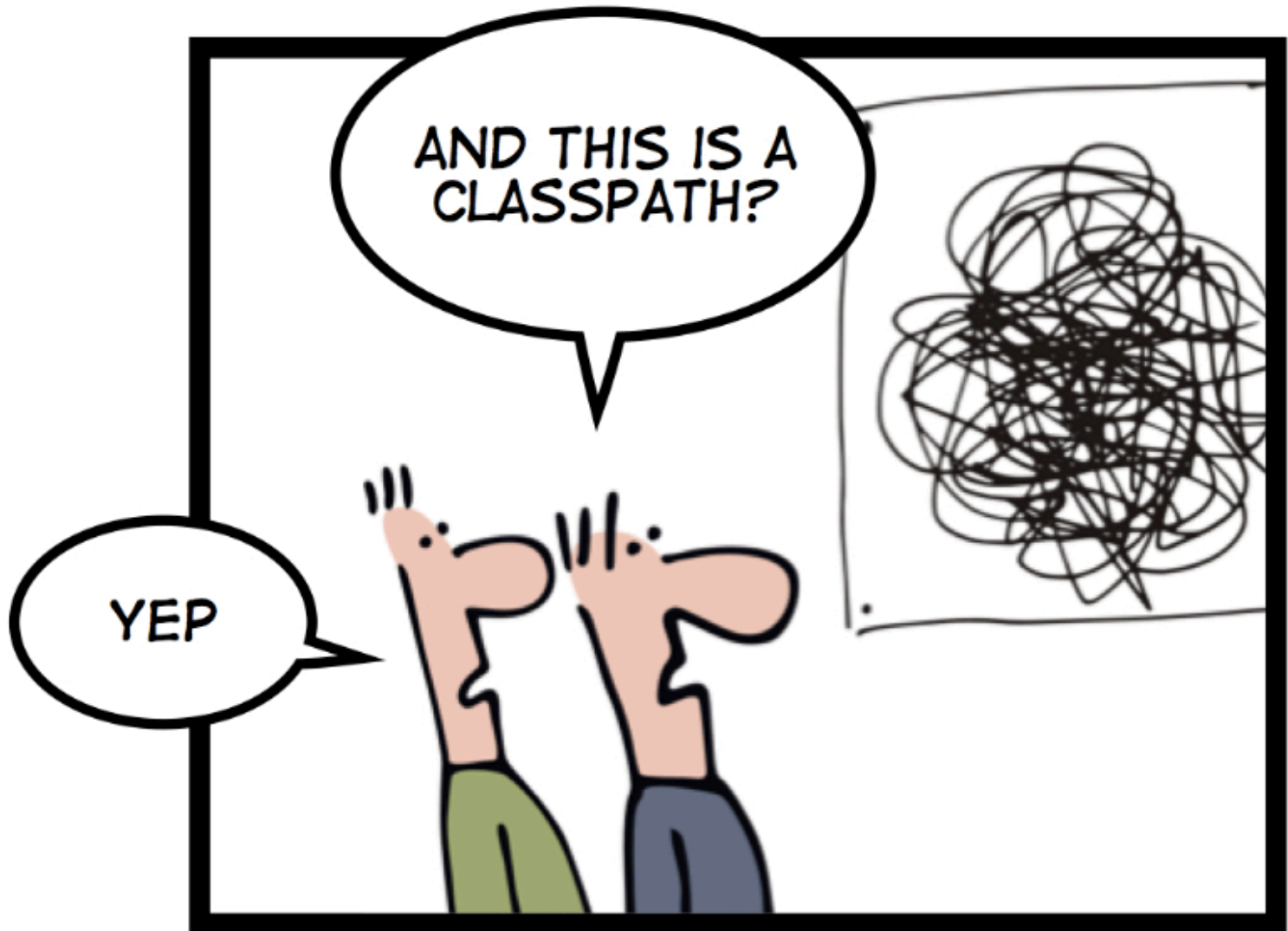
Why OSGi?



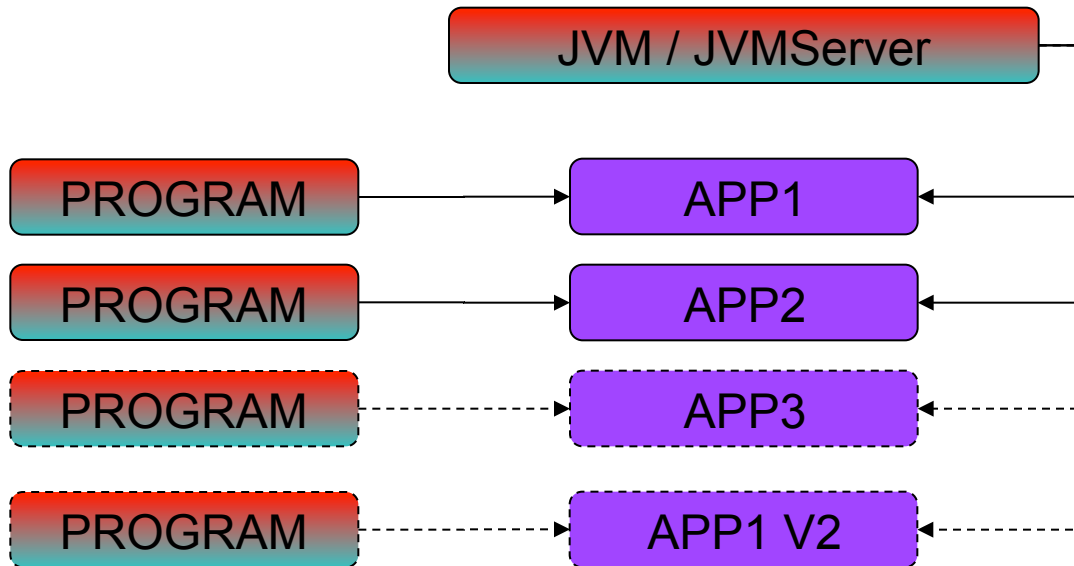
Why OSGi?



Why OSGi?

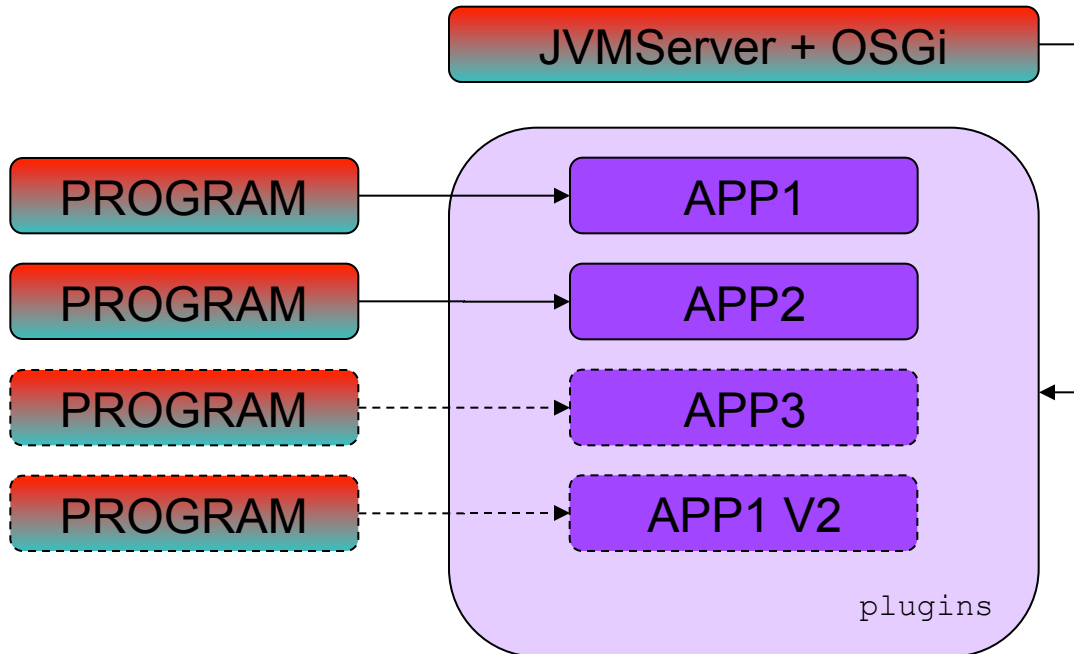


Pooled JVM – Application Lifecycle

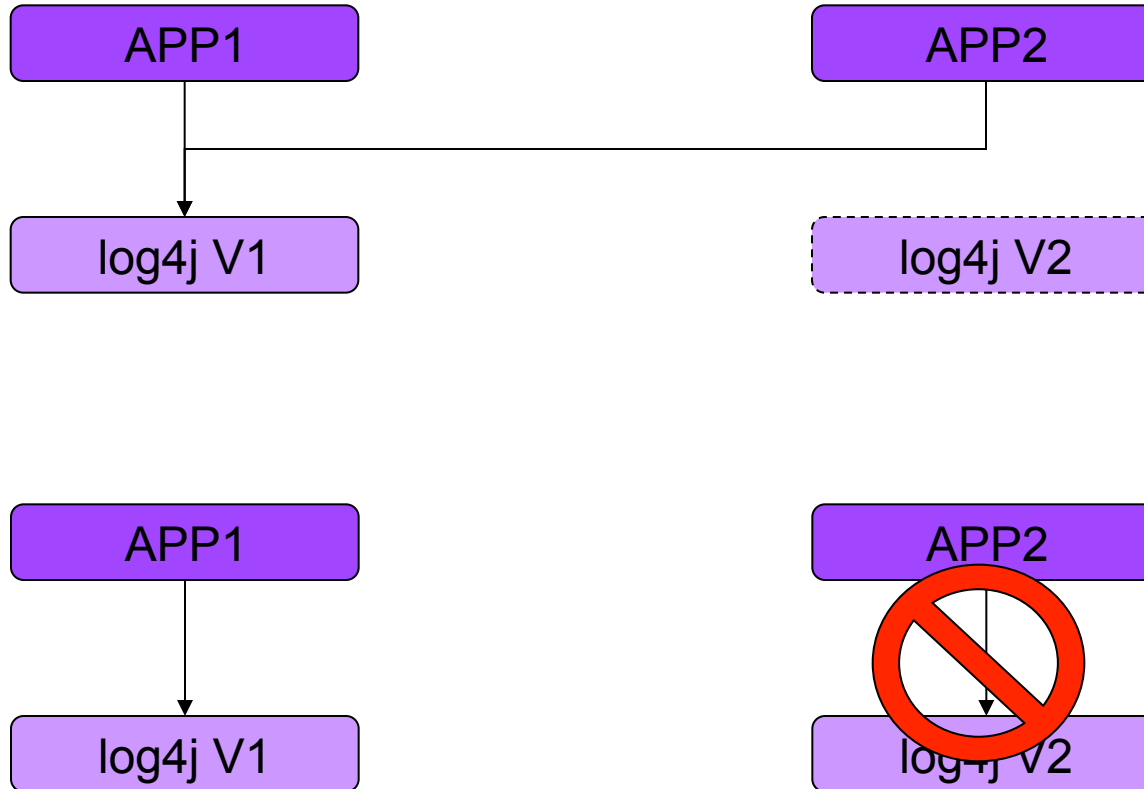


Must restart the JVM pool

JVM Server – Application Lifecycle

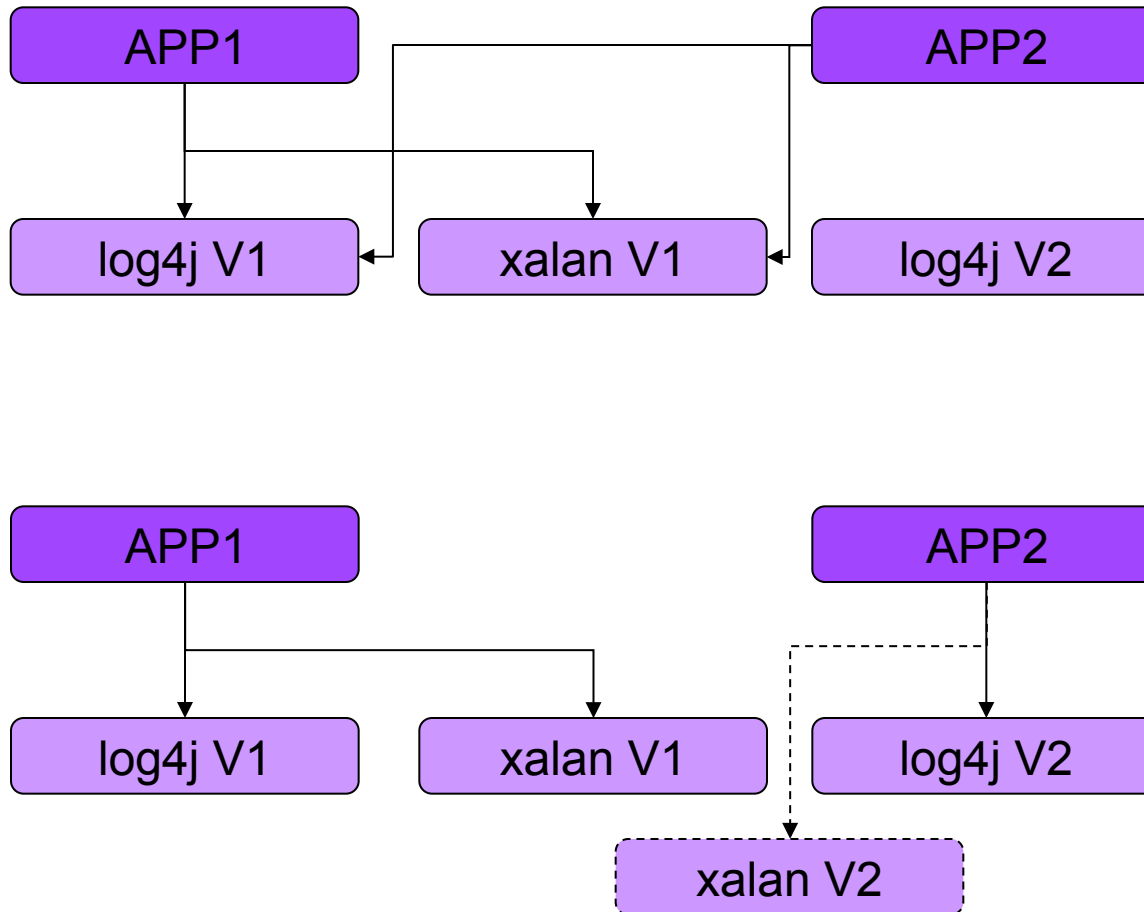


Pooled JVM – Application/Library Versioning



Cannot support 2 versions of a library

JVM Server – Application/Library Versioning



Tools

Rational Developer for z

- **Questions**

- What tools do you use for traditional application development?

- **Characteristics**

- Common development environment for COBOL, PL/I, C/C++, and Java
- Analyze, understand, edit, build, and unit test all from the same environment
- Remote source level debugging
- Code/meta-data generation and language integration wizards
- Integration with SCMs including Team Concert and Endeavor
- Almost infinitely extensible

- **References**

- <http://share.confex.com/share/116/webprogram/Session8657.html>
- **9586: IBM Rational Developer for System z V.8.0 (+)**
- **9767: Application Development for z/OS - Not Your Father's Green Screen**
- **10126: A System z Developer's Journey Through the Application Lifecycle - Lunch'n'Learn**



IDE-based development

- Common development environment for COBOL, PL/I, C/C++, and Java
- Simplified development with more information at your fingertips

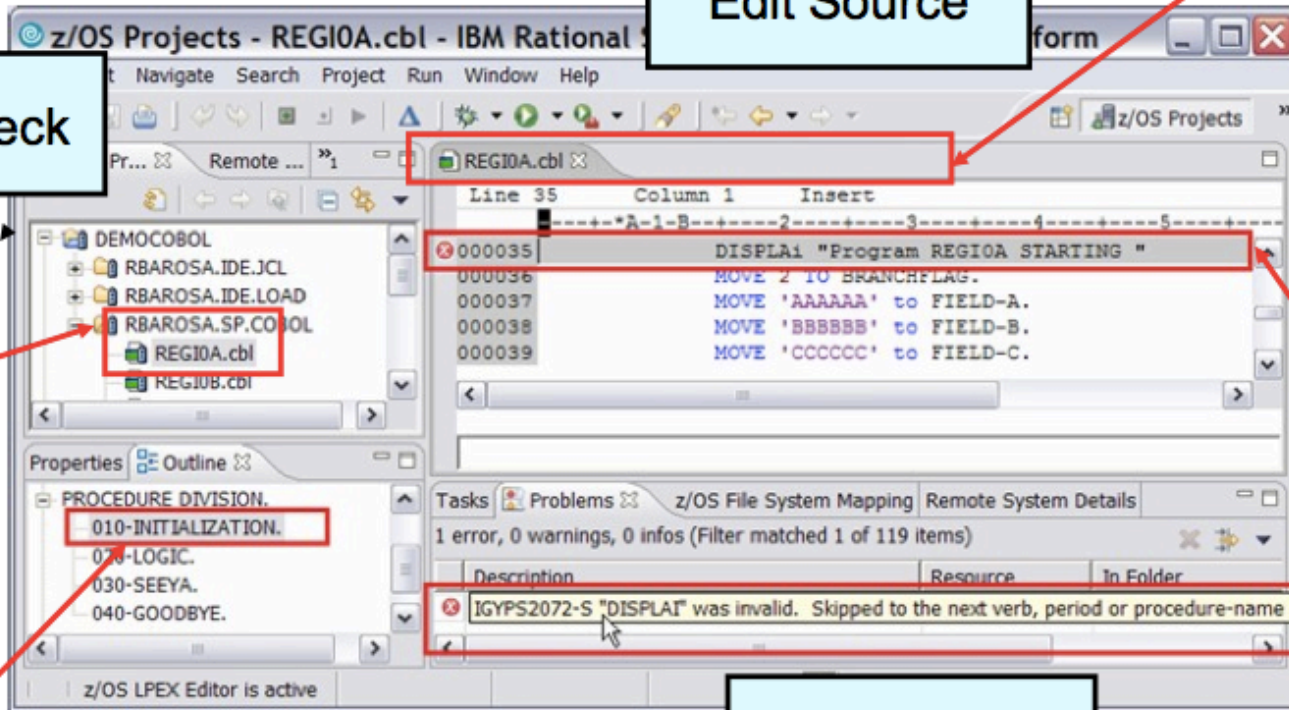
Open and edit multiple source and JCL members simultaneously

Edit Source

Syntax Check

Submit jobs, access job output, or open source members with a single click

Outline view presents COBOL structure



Statement in error indicated in source

Double-Click on the Error

Error list in Problems view



Multi-platform Application Development - the Nimble Programmer

Tools – A Silver Bullet (in more ways than one)

- Integrated Development Environments (IDEs have advanced incredibly in the past several years
 - Cross-platform differences are reduced
 - Access to multiple systems simultaneously is expected
 - Multiple language support is now common-place
 - Integration of multiple development tools into a single development environment is now reality
- At the same time, effective use of an IDE requires education and experience
 - On first sight, there is an overload of information
 - On second sight, there are “hidden” features – Where do I click?
 - On third sight, there are sometimes endless UI elements to interpret and understand
- But past the learning curve ...
 - Using an IDE allows programmers to concentrate on the application
 - Regardless of programming language
 - Regardless of runtime environment

The Nimble Programmer

■ **Application Developers**

- can't afford to be one trick pony
- probably need to know multiple languages
- should be able to target multiple execution platforms: mainframe, distributed even mobile

■ **Architects**

- don't just architect by language
- won't architect for a single system

■ **Project Managers**

- must allow skills growth

Rational Developer for System z Unit Test

■ Questions

- Has anyone got RDz UT?

■ Characteristics

- Bring up an entire z/OS software stack on your (Linux) laptop or blade
- Build and test applications locally
- Integration with RDz etc
- Only for development
 - not for stress testing or production

■ References

- <http://www.ibm.com/software/rational/products/developer/systemz/unittest/>

Eclipse IDE with CICS Explorer SDK

■ Questions

- Anyone doing Java development not using an Eclipse-based IDE?
- Why?

■ Characteristics

- Platform independent (Windows, Linux, ...)
- Can use the same environment for other “modern” workloads e.g. PHP
- Can also Eclipse-based IDE for COBOL, PL/I & C/C++ with RDz
- First class integration with source code management (version control) e.g. RTC, SVN, CVS, ...
- First class integration with workflow & planning tools e.g. RTC
- Test-driven development and continuous integration (Agile) with JUnit & mock objects

■ References

- [9607: CICS for Java Developers and Java for System Programmers - Two Sides of the Same Coin](#)

Eclipse IDE with CICS Explorer SDK

The screenshot displays the Eclipse IDE interface with the CICS Explorer SDK project open. The Package Explorer on the left shows the project structure, including the 'examples.hello' package containing 'HelloCICSWorld.java' and 'HelloWorld.java'. The main editor shows the source code for 'HelloCICSWorld.java', which includes a package declaration, an import for 'com.ibm.cics.server.CommAreaHolder', and a 'main' method that checks for a task and prints a message.

```

* @start_nonoco_copyright@
package examples.hello;

import com.ibm.cics.server.CommAreaHolder;

public class HelloCICSWorld
{
    public static void main(CommAreaHolder CAH)
    {
        Task t = Task.getTask();
        if ( t == null ) {
            System.err.println("HelloCICSWorld example: Can't get Task");
        }
        else {
            t.out.println("Hello from a Java CICS application");
        }
    }
}

```

The Problems view at the bottom shows a declaration for 'com.ibm.cics.server.Task', indicating that this class provides methods and variables corresponding to a CICS task. The version information is as follows:

- Since CICS TS version: 1.3
- Since package version: 1.0

The Help view on the right displays the 'Contents' of the CICS Java Developer Guide, listing various topics such as 'What you need to know about...', 'Java programming using JCICS', and 'Developing and deploying applications...'. The 'Go To' section includes links for Search, Related Topics, Bookmarks, and Index.

Summary

- **Languages**
- **Environments**
- **Tools**

Questions

Other Sessions

- **Monday**
 - 9586: [IBM Rational Developer for System z V.8.0 \(+\)](#)
 - 9322: [CICS TS V4.2 Technical Overview](#)
- **Tuesday**
 - 9327: [CICSplex What Is It, and Why Do I Care?](#)
 - 9600: [CICS TS and the Cloud](#)
 - 9605: [What's New in CICS V4.2 Systems Management Best Practices](#)
- **Wednesday**
 - 9319: [CICS Emerging Technologies Hands-on Lab Part 1 of 2](#)
 - 9320: [CICS Emerging Technologies Hands-on Lab Part 2 of 2](#)
 - 9608: [CICS Dynamic Scripting](#)
- **Thursday**
 - 9606: [CICS ... It's Not Just COBOL: Java Support](#)
 - 9607: [CICS for Java Developers and Java for System Programmers - Two Sides of the Same Coin](#)
 - 9613: [CICS Question Box and Pot Luck](#)
 - 9330: [CICS Event Processing](#)
- **Friday**
 - 9614: [CICS Nuts, Bolts and Gotchas](#)

More Information

- **IBM CICS Explorer**

- <http://www.ibm.com/software/htp/cics/explorer/>

- **SHARE (Past Conferences)**

- 8272: Best Practices for CICS Systems Management

- <http://share.confex.com/share/116/webprogram/Session8272.html>

- **Podcasts**

- <http://www-01.ibm.com/software/htp/cics/tserver/v42/library/index4.html>

- **Blog**

- <http://masterterminal.wordpress.com/>



- **developerWorks Forum**

- <http://www.ibm.com/developerworks/forums/forum.jspa?forumID=1475>

