



**zEnterprise –  
The Ideal Platform For  
Smarter Computing**

Eliminating Redundant Software

# Eliminating Redundant Software Is Harder To Do Than Infrastructure Consolidation

... but potential savings are greater!

To minimize software redundancy:

- ▶ **Find redundant software** in your infrastructure, then consolidate it
- ▶ **Drive down redundant code**, and prevent it from coming back

# Identifying Existing IT Assets Is Key To Eliminating Redundant Software

Only 40% of IT assets are discovered and are understood.

-- Finance Week

If 60% of our IT assets are NOT understood, **How Can We Know:**

- ▶ *What is deployed in our data center?*
- ▶ *How are deployed items interrelated?*
- ▶ *How are they configured? How are they changing?*
- ▶ *Which business services run on which servers?*
- ▶ *If the servers support the business services properly?*
- ▶ *How to avoid changes that cause problems, requiring a costly rollback?*

**Tivoli Application Dependency Discovery Manager (TADDM)**

**Tivoli Asset Discovery for z/OS**

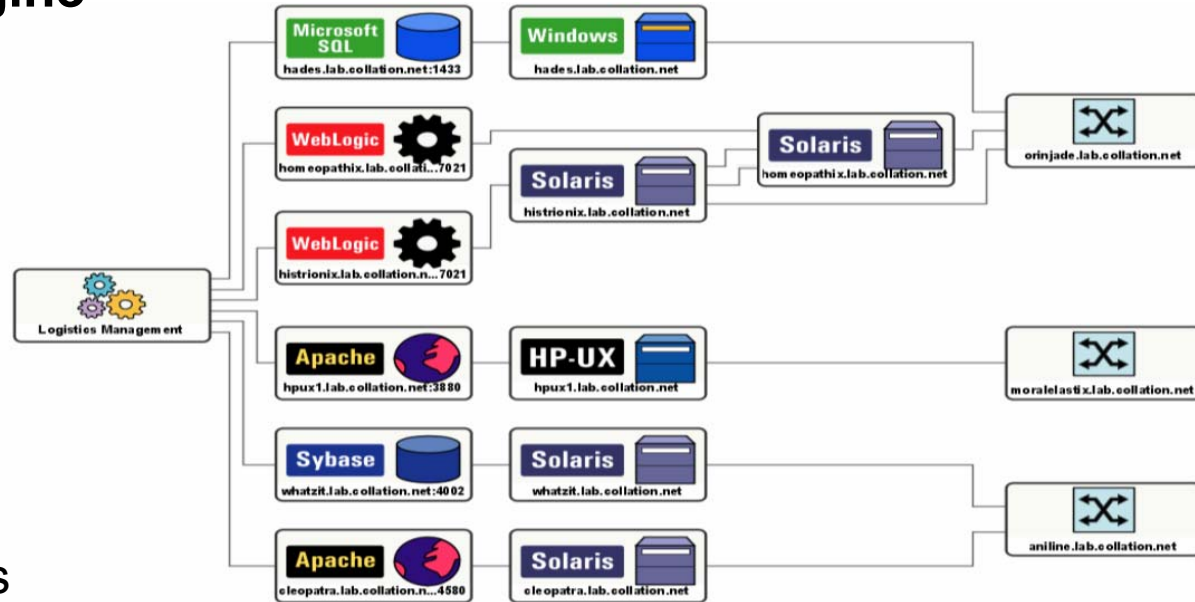
**Tivoli Asset Discovery for Distributed**

can help answer these questions to eliminate redundancy

# TADDM Shows Existing Components And Finds Dependencies Between Them

## Universal Discovery Engine

- ▶ Discovers
  - Servers
  - Operating systems
  - Middleware
  - Applications
  - Network devices
- ▶ Reconciles names
- ▶ Normalizes data
- ▶ Creates topology views

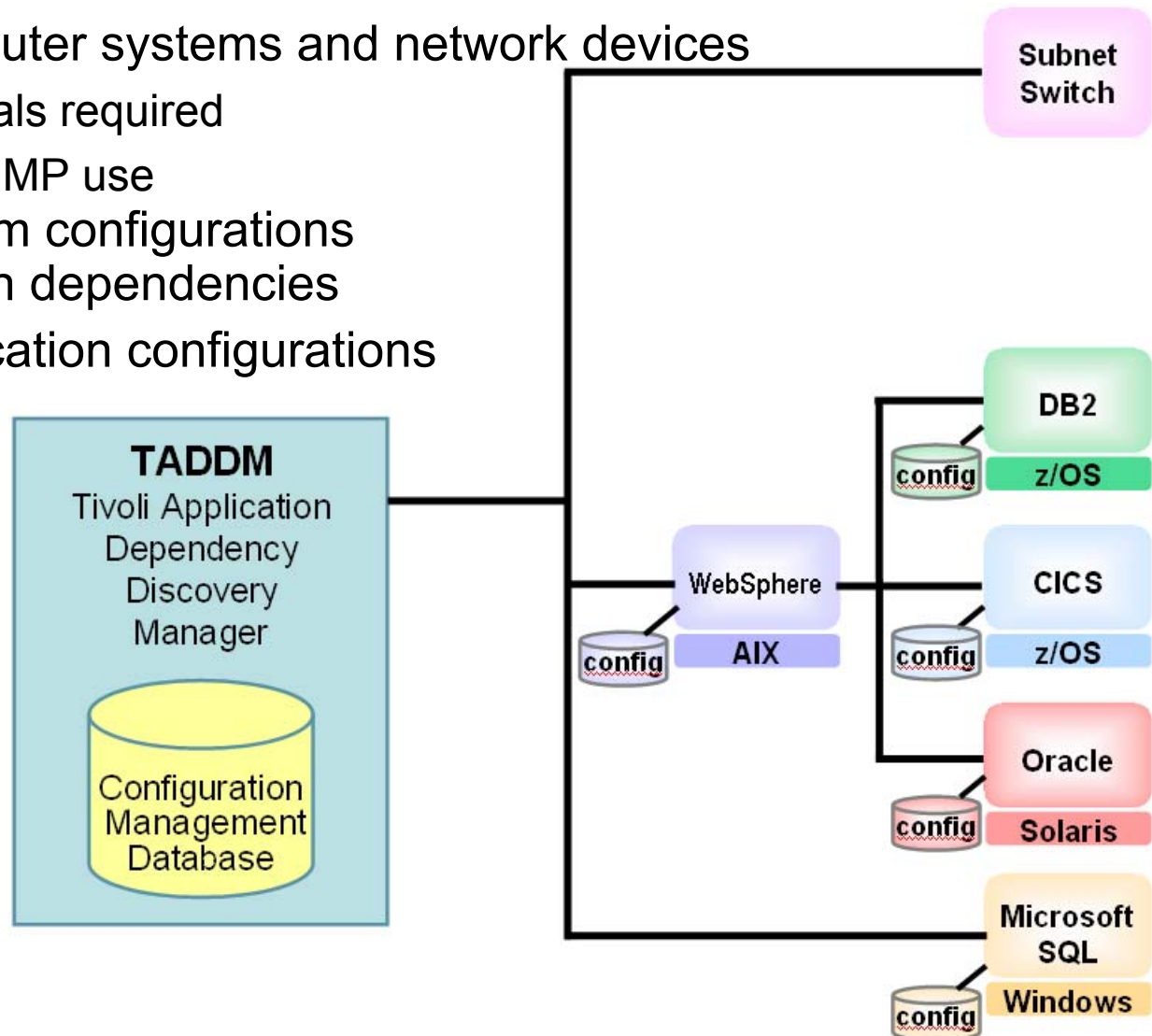


## Agent-less Application Mapping

- ▶ Finds dependencies between discovered items

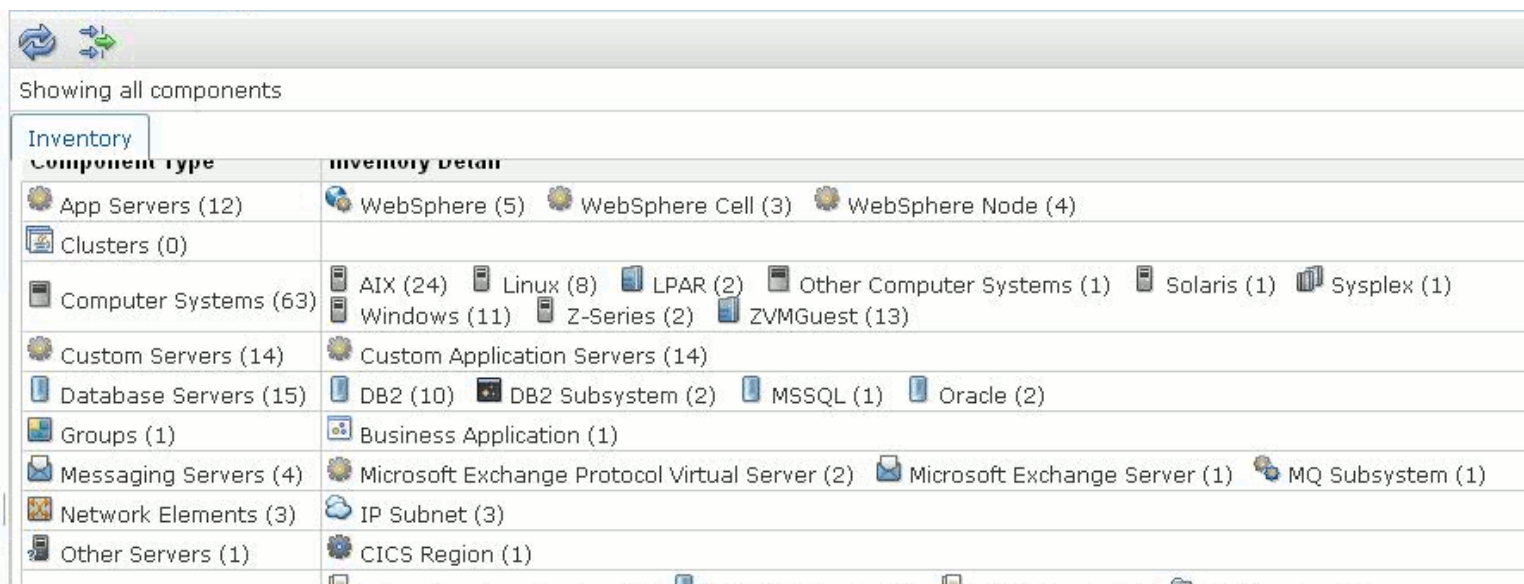
# TADDM Has A 3-Step Discovery Process

1. Discover computer systems and network devices
  - ▶ No credentials required
  - ▶ Optional SNMP use
2. Discover system configurations and application dependencies
3. Discover Application configurations



# DEMO: Using TADDM To Identify Redundant Software

1. The IT department knows there are multiple database types and instances in use. License costs can be reduced by consolidating the various databases
2. Use TADDM to discover database servers, software and instances
3. Use TADDM discovery results to identify potential software consolidation targets



Showing all components

Component Type	Inventory Detail
App Servers (12)	WebSphere (5) WebSphere Cell (3) WebSphere Node (4)
Clusters (0)	
Computer Systems (63)	AIX (24) Linux (8) LPAR (2) Other Computer Systems (1) Solaris (1) Sysplex (1) Windows (11) Z-Series (2) ZVMGuest (13)
Custom Servers (14)	Custom Application Servers (14)
Database Servers (15)	DB2 (10) DB2 Subsystem (2) MSSQL (1) Oracle (2)
Groups (1)	Business Application (1)
Messaging Servers (4)	Microsoft Exchange Protocol Virtual Server (2) Microsoft Exchange Server (1) MQ Subsystem (1)
Network Elements (3)	IP Subnet (3)
Other Servers (1)	CICS Region (1)

# TADDM Output Helps Identify Redundant Servers And Software For Consolidation

- A quick view of server usage for server consolidation
  - ▶ What is this server used for?
- A complete view of installed software
  - ▶ Which servers support a specific application or middleware?
  - ▶ Highlights opportunities for consolidation or simplifying backups
- Detailed views of application interactions
  - ▶ What would be the impact of moving or eliminating this application?
- Locate department 'islands' or clusters of redundant software that is not part of the overall IT plan
  - ▶ Identify assets that are 'flying under the radar'
- TADDM output is a key resource for quickly identifying software redundancies from mergers or acquisitions

*Typical TCO  
savings of  
30% to 70%*

# Case Study – Cineca



*Italy's largest research computing center limits cost, improves efficiency and aligns IT management with business priorities using Tivoli Automation*

## **Business Challenges**

- Update infrastructure to limit costs
- Improve service delivery and support

## **Solution**

- TADDM helps CINECA
  - ▶ Understand configurations
  - ▶ Map applications and changes
  - ▶ Address compliance measures
- Configuration Management Database (CMDB) is used as the information base for service management

## **Business Benefits**

- Manage 93% of the infrastructure components better
- Automatically track changes in the configuration of the infrastructure components
- Successfully conduct inventory reporting and topology mapping



# Service Oriented Architecture Can Help Drive Down Redundant Code By Pooling Services

Many businesses re-implement the same function over and over again

- ▶ Leads to software redundancy
- ▶ Drives up software licenses
- ▶ Creates software sprawl

Redundancy can also be caused by mergers and acquisitions

To drive down redundancy...

- ▶ Find the functions
- ▶ Categorize them
- ▶ Rationalize them

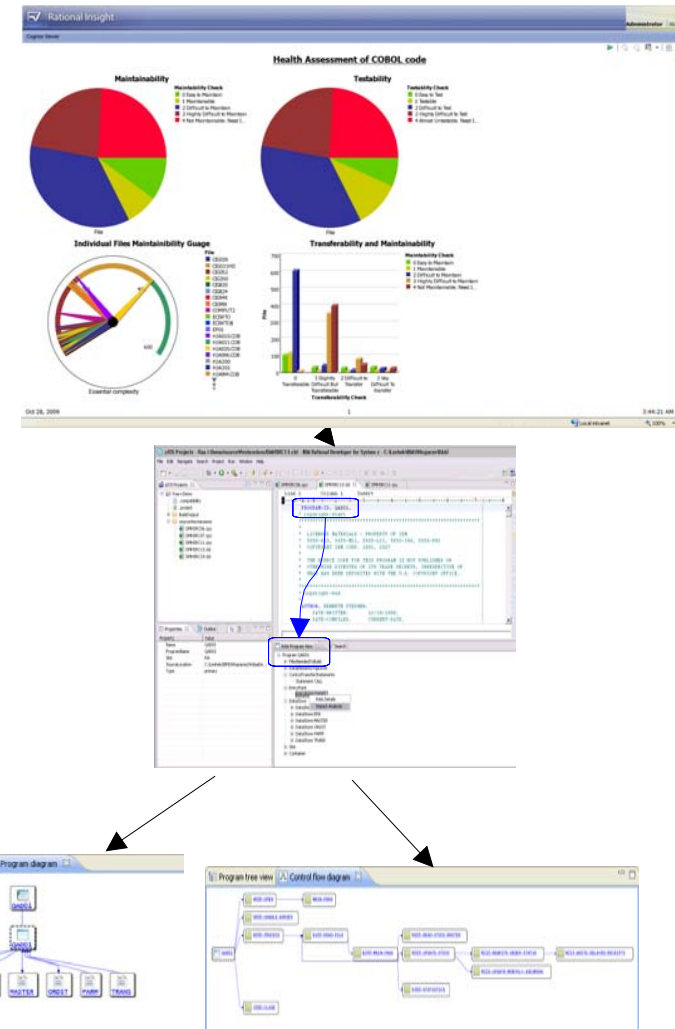
...then **pool** services by identifying repeated functions, and

- ▶ Choosing one as the strategic function
- ▶ Deploying it as a Web service
- ▶ Modifying each solution to use the strategic service
- ▶ Eliminating the non-strategic functions
- ▶ Ensuring that all new solutions use the strategic service instead of creating a new one

**Fewer lines of code means lower maintenance costs,  
so more resources are available for new strategic development!**

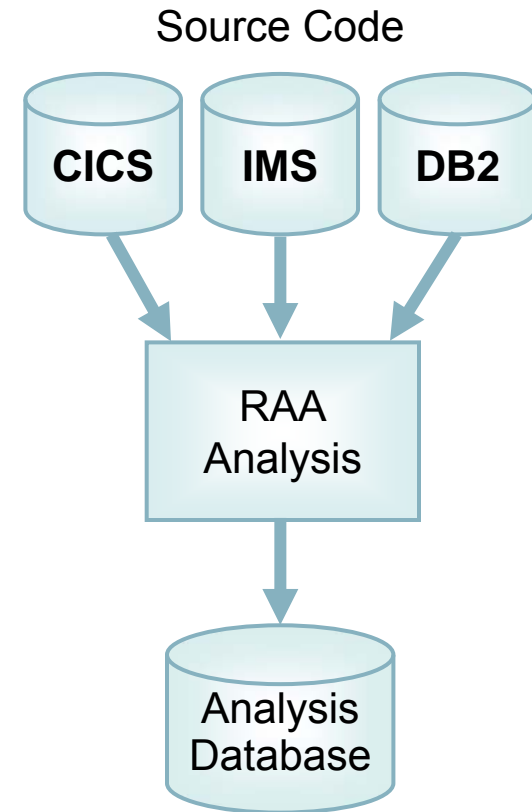
# Rational Asset Analyzer (RAA) Helps Find Reusable Functions In Source Code

- RAA scans source code to determine program and data flow, and show dependencies between code elements
  - ▶ COBOL, PL/I for CICS, IMS, DB2
  - ▶ z/OS JCL and High Level Assembler
  - ▶ Java, Java EE, C/C++ applications
- RAA creates an application knowledge base showing relationships, program structure, flows, and run times
  - ▶ Provides browsable view of architectures and artifacts
  - ▶ This forms the basis for intelligent SOA-based design using services built from existing code



# RAA Analyzes CICS, IMS And DB2 Assets For Potential Reuse As Web Services

- Some assets in particular have potential for reuse as services:
  - ▶ CICS online regions and transactions
  - ▶ Existing Web services in CICS
  - ▶ IMS transactions
  - ▶ DB2 access and stored procedures
- Accessing existing functions as services makes modernization easy and flexible

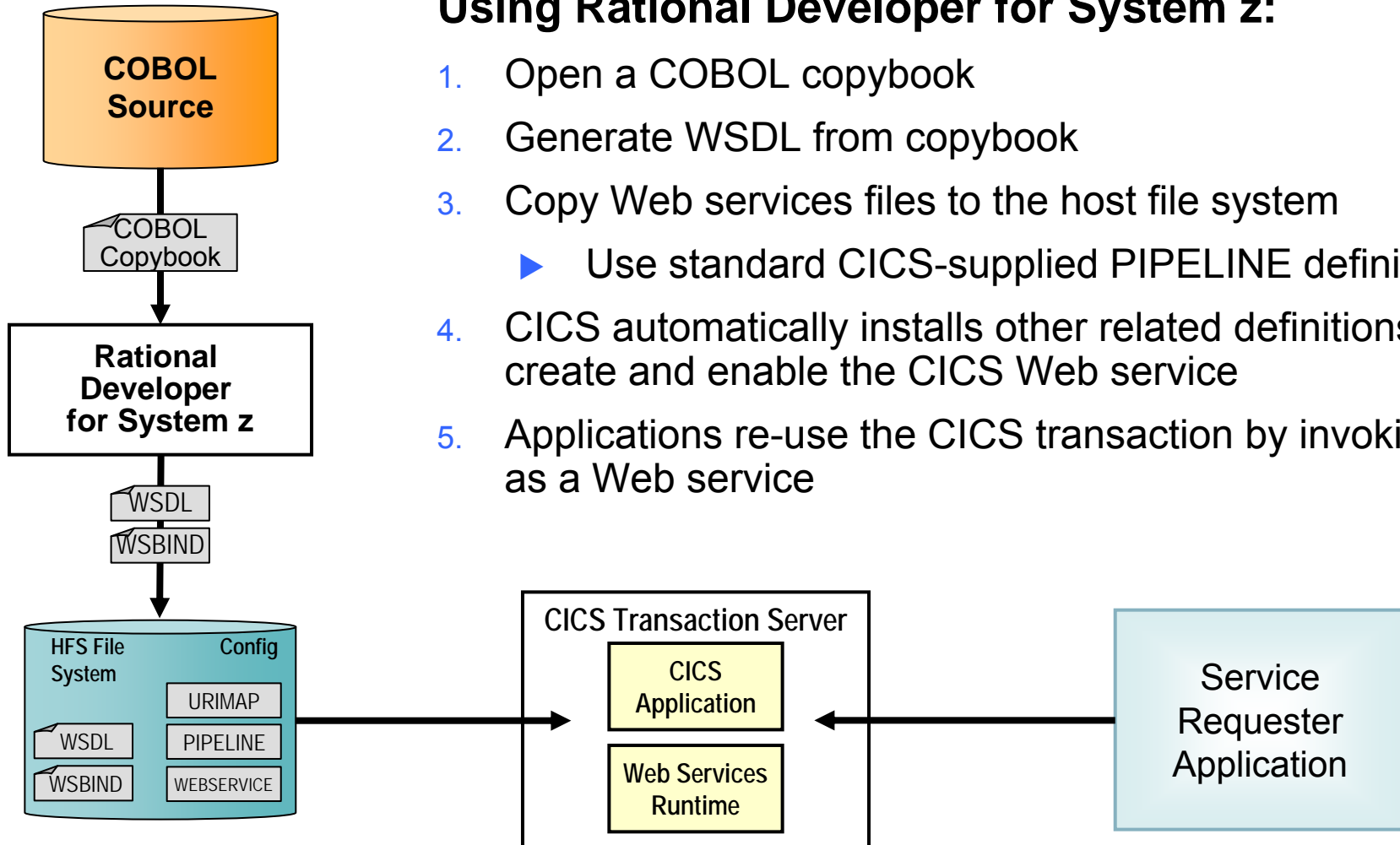


**Rational Developer for z makes it easy to create Web services from any of these – without writing any code!**

# Use Rational Developer For z To Create Services From Existing CICS Transactions Without Coding

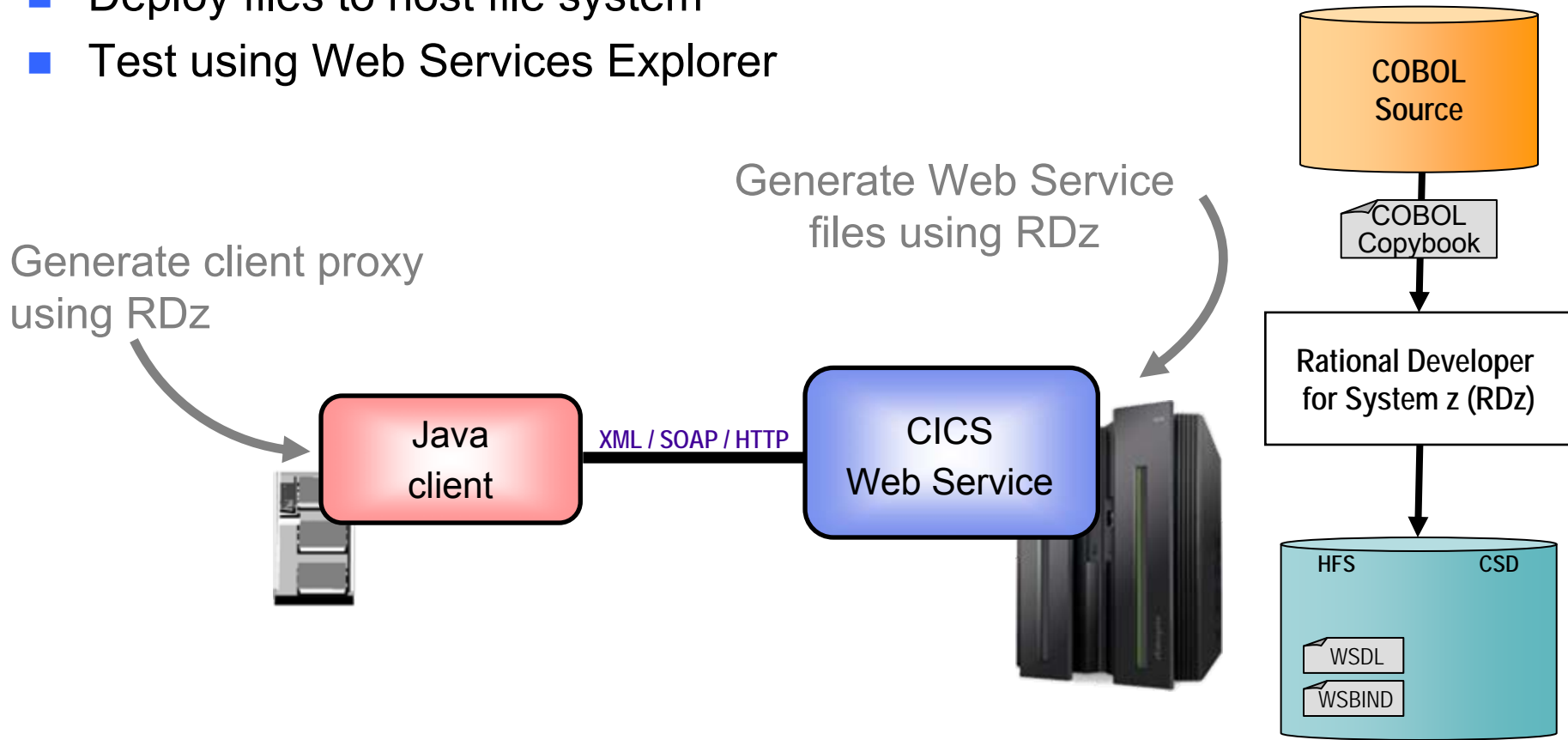
## Using Rational Developer for System z:

1. Open a COBOL copybook
2. Generate WSDL from copybook
3. Copy Web services files to the host file system
  - ▶ Use standard CICS-supplied PIPELINE definition
4. CICS automatically installs other related definitions to create and enable the CICS Web service
5. Applications re-use the CICS transaction by invoking it as a Web service



# DEMO: Use RDz To Expose CICS Program As Web Service

- Generate WSDL and WSBIND files
- Deploy files to host file system
- Test using Web Services Explorer



# Use WebSphere Service Registry And Repository To Prevent Redundancy From Returning

## Define An SOA Governance Plan

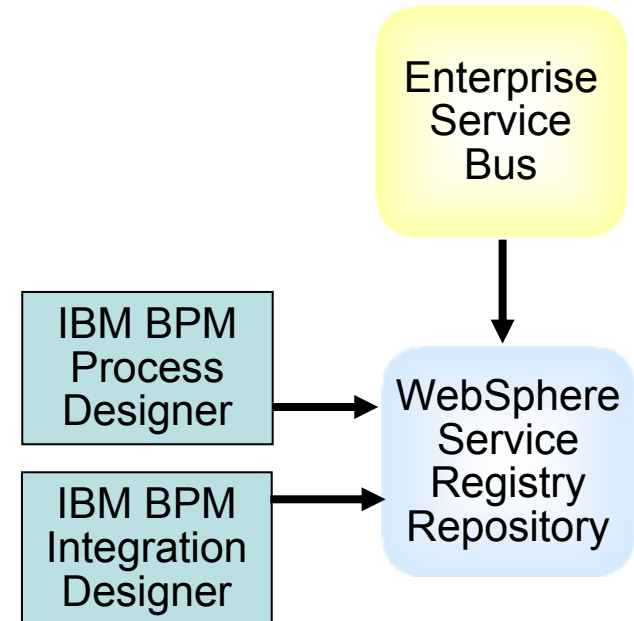
- ▶ Create a development process for reviewing new application and service deployment proposals
- ▶ Enable enforcement of policies

## Encourage Reuse

- ▶ Publish new services for future reuse
- ▶ Quickly locate required services directly from development tools for new applications and business processes

## Enhance Connectivity

- ▶ Locate required services for runtime requests, including alternates
- ▶ Enable dynamic and efficient interactions between services at runtime with an Enterprise Service Bus



# zEnterprise Supports All SOA Components

## Service Requesters

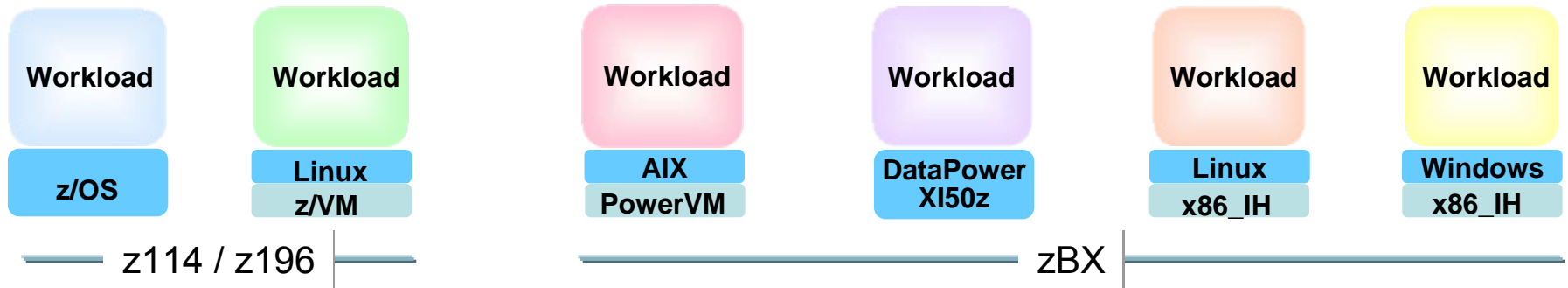
BPM Process Server  
 WebSphere Application Server  
 CICS Applications  
 DB2 Stored Procedures  
 IMS Applications  
 COBOL applications  
 Java batch applications  
 z/OS Batch Jobs

## Service Intermediaries

Enterprise Service Bus  
 WebSphere Service Registry Repository  
 Web service front ends to CICS/DB2

## Service Providers

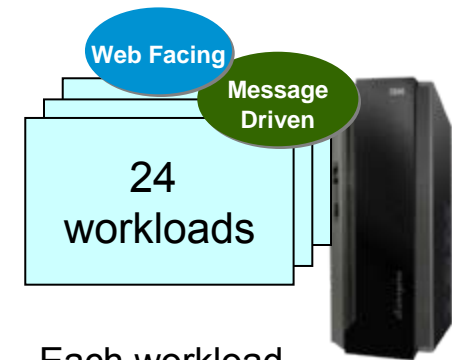
BPM Process Server  
 WebSphere Application Server  
 CICS Transactions as services  
 DB2 Stored Procedures as services  
 IMS Transactions and Data as services  
 COBOL procedures as services  
 z/OS batch jobs  
 Java batch applications



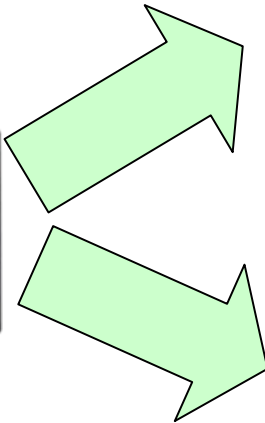
\*All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represents goals and objectives only.  
 05 - Eliminate Redundant Software V2.5

# Web Front Ends Cost 59% Less On zEnterprise

Web front-end workloads



Each workload driving 3080 tps  
High availability  
Workload isolation



## Competitive Packaged System

24 Sun Fire X4170 M2 12-core Xeon servers in ¾ rack  
2 HP DL380 servers (for ESB)  
312 cores total



Competitor's system relies on physical workload isolation

Deploy on Sun hardware

**\$433K**  
per workload  
3yr TCA  
Front end HW+SW

## WebSphere App Server

24 POWER7 8-core blades  
2 DataPower XI50z  
in zBX  
192 cores total



Power blades in zBX

**\$177K**  
per workload  
3yr TCA  
Front end HW+SW

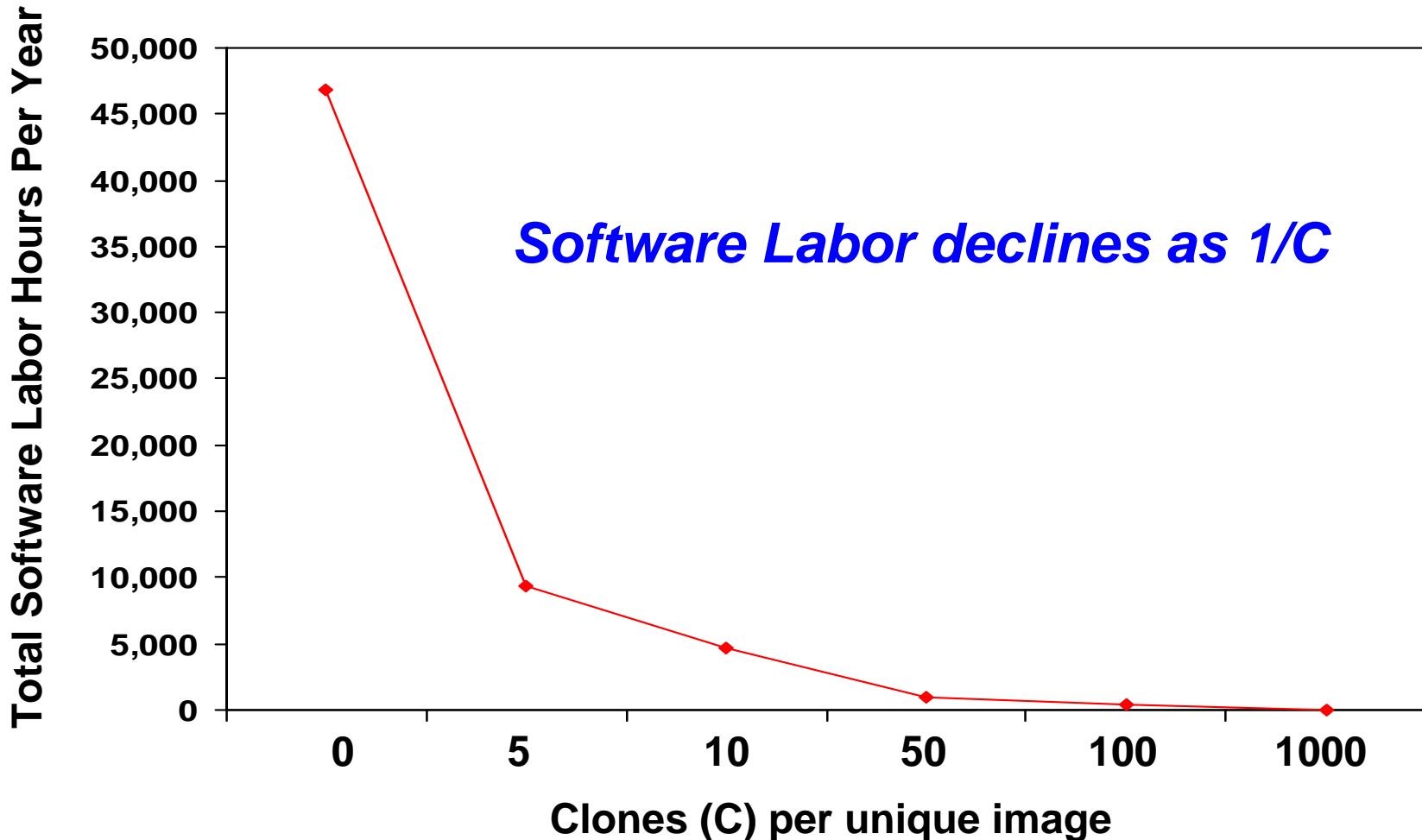
Source: IBM Internal benchmarks. Competitive Packaged System includes Competitive Application Server and Sun Fire X4170 M2 servers. 3 yr. TCA calculation includes hardware acquisition, maintenance, software acquisition and S&S. US list prices. Prices may vary by country.



# Standardization Of Software Stacks Can Also Eliminate Redundant Software

- A server needs a full set of software to run a workload
  - ▶ Operating System, Middleware, Applications
  - ▶ Patches, configuration specifications
- The combination of all this software is called a “software stack”
- Without controls, the variety of software stacks tends to proliferate, driving up labor costs
  - ▶ Different levels, patches, product selections, etc.
- Standardization of software stacks can reduce labor costs
  - ▶ Uniformity reduces the number of unique stacks to manage
  - ▶ Re-using a standard software stack is called “cloning”

# Benefit Of Cloning On Software Labor Costs In A Virtualized Environment



# Save Money By Eliminating Redundant Software

Start by assessing current environment

- ▶ Discover hidden “islands” of underutilized applications under departmental control
- ▶ Identify candidates for eliminating or consolidating applications and servers

Use Service Oriented Architecture to pool services and reduce redundant application code

- ▶ Reducing code yields reduced maintenance costs, enabling investment in new strategic development
- ▶ Use SOA Governance and a service registry to ensure redundant code does not return

Clone software stacks to significantly reduce labor costs