DevOps Track 4

Multi-platform IDEs to accelerate development and testing

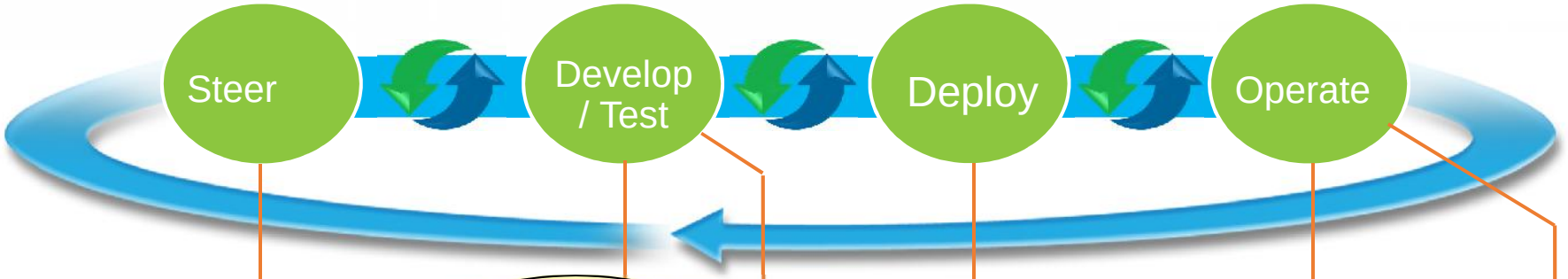
Speaker Name
Speaker Title

Updated Jan 09 2015

- 10:15 – 11:15 - Regi
- **Use multi-platform IDEs to accelerate development and testing**
- This scenario-driven session highlights the ease of using an Integrated Development Environment to accelerate multi-platform development and testing (from zUnit, to integrated debugger to off-host testing).
- We show how Rational Developer for the Enterprise supports the design, creation, deployment and maintenance of traditional transactional applications and modern composite applications running on IBM z/OS.
- We highlight the integrated debugger, which provides full edit, compile and debug capabilities. These capabilities can remove the need for additional debug and code coverage products.

IBM DevOps – Broad set of DevOps capabilities

Address bottlenecks across the application delivery lifecycle



Continuous Business Planning
Rational Team Concert

Collaborative Development
Rational Collaborative Lifecycle Management
Rational Developer for System z
Rational Compilers
Worklight Studio

Continuous Release and Deployment
UrbanCode Release
UrbanCode Deploy with Patterns
SmartCloud Orchestrator

Continuous Monitoring
SmartCloud Application Performance Management
SmartCloud Control Desk

Continuous Testing
Rational Quality Manager
Rational Test Virtualization Server
Rational Test Workbench
Rational Test Workbench – Mobile Test Edition
Rational Development and Test Environment for System z
Worklight Quality Assurance
InfoSphere Optim Test Data Management
Security AppScan

Continuous Customer Feedback & Optimization
Tealeaf CX
Digital Analytics
Worklight Quality Assurance

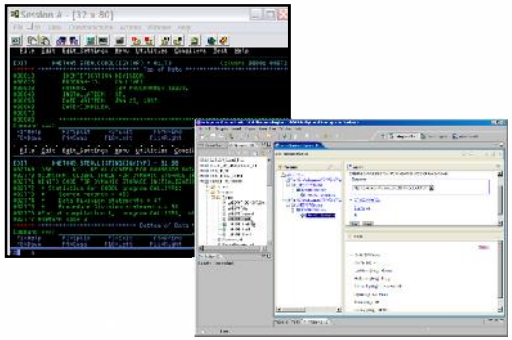
Bluemix DevOps services

- Agile planning and tracking
- Application auto-scaling
- AppScan mobile analyzer
- Continuous delivery pipeline
- Git hosting
- Mobile application security
- Mobile data
- Mobile quality assurance
- Monitor & analytics
- Push
- RapidApps (beta)
- Server-side code
- Web IDE

Hardware have been improved.. How about our Software?

In September 1956, IBM launched the 305 RAMAC, the first computer with a hard disk drive (HDD).
The HDD weighed over a ton and stored 5MB of data.”

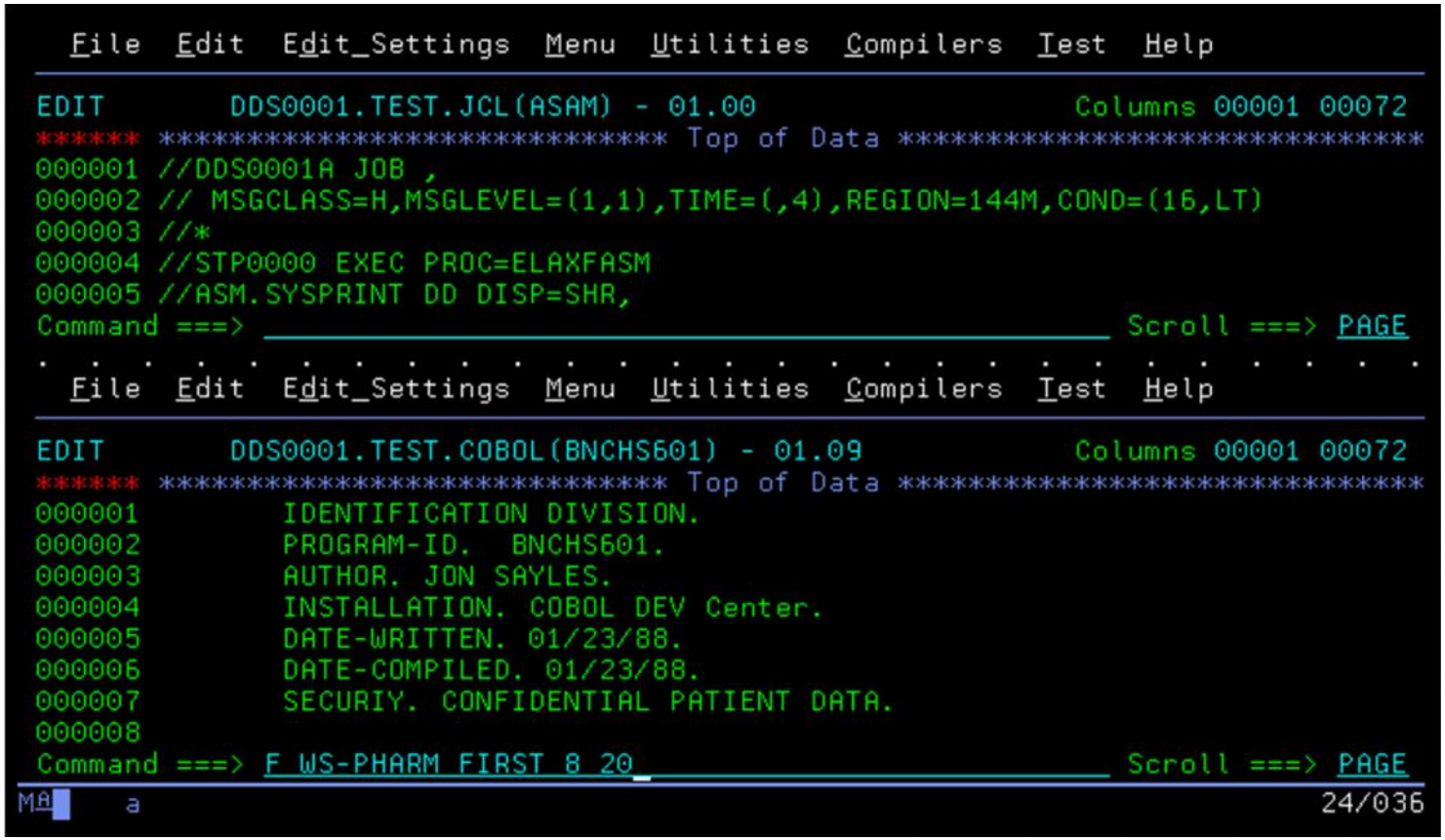
Makes you appreciate your 32 GB jump drive, doesn’t it?



→ Software also needs Improvements….

System z development environment and tools

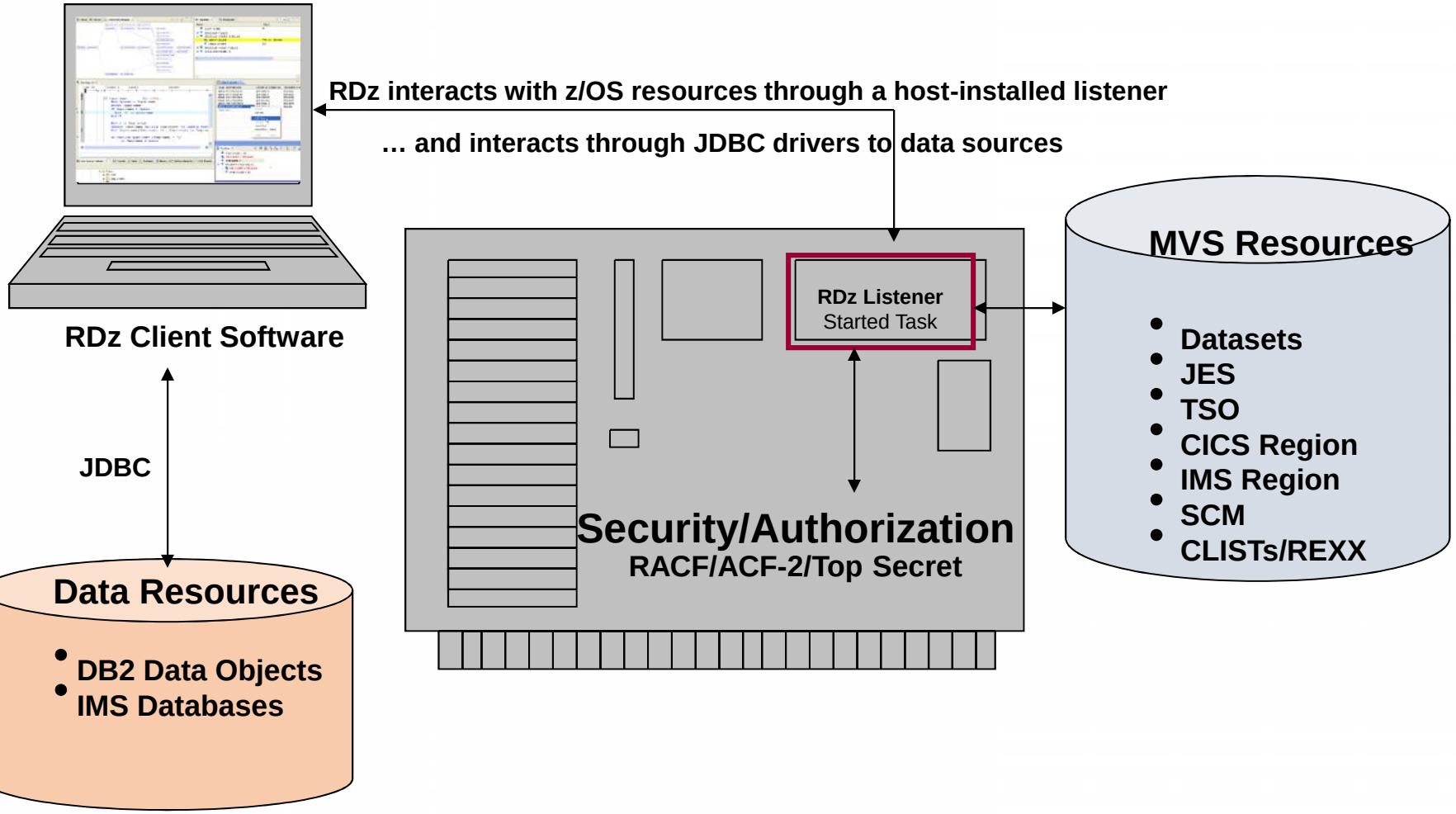
ISPF DB2/SQL Development



Drawbacks:

- Typing speed & accuracy == productivity ceiling
- Limited use of "screen real estate"
- No language-sensitive intelligent development tooling
- No advanced application development tools for maintenance programming
- No integration with other development tools

What is Rational Developer for System z (RDz)?

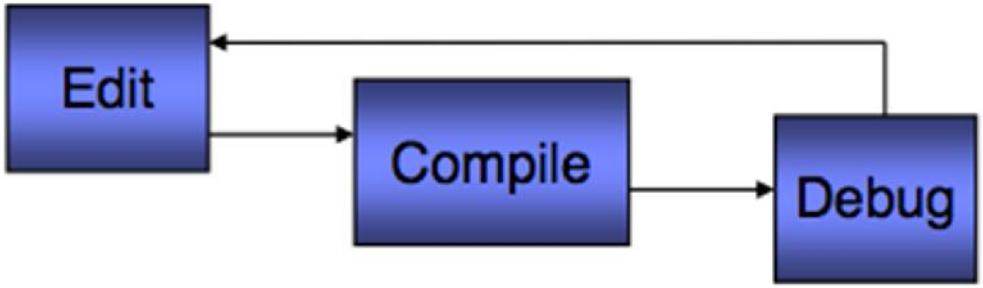


RDz also interacts with data sources
(DB2 tables/views, IMS database segments)
through efficient JDBC access

IBM Rational Developer for System z (RDz)

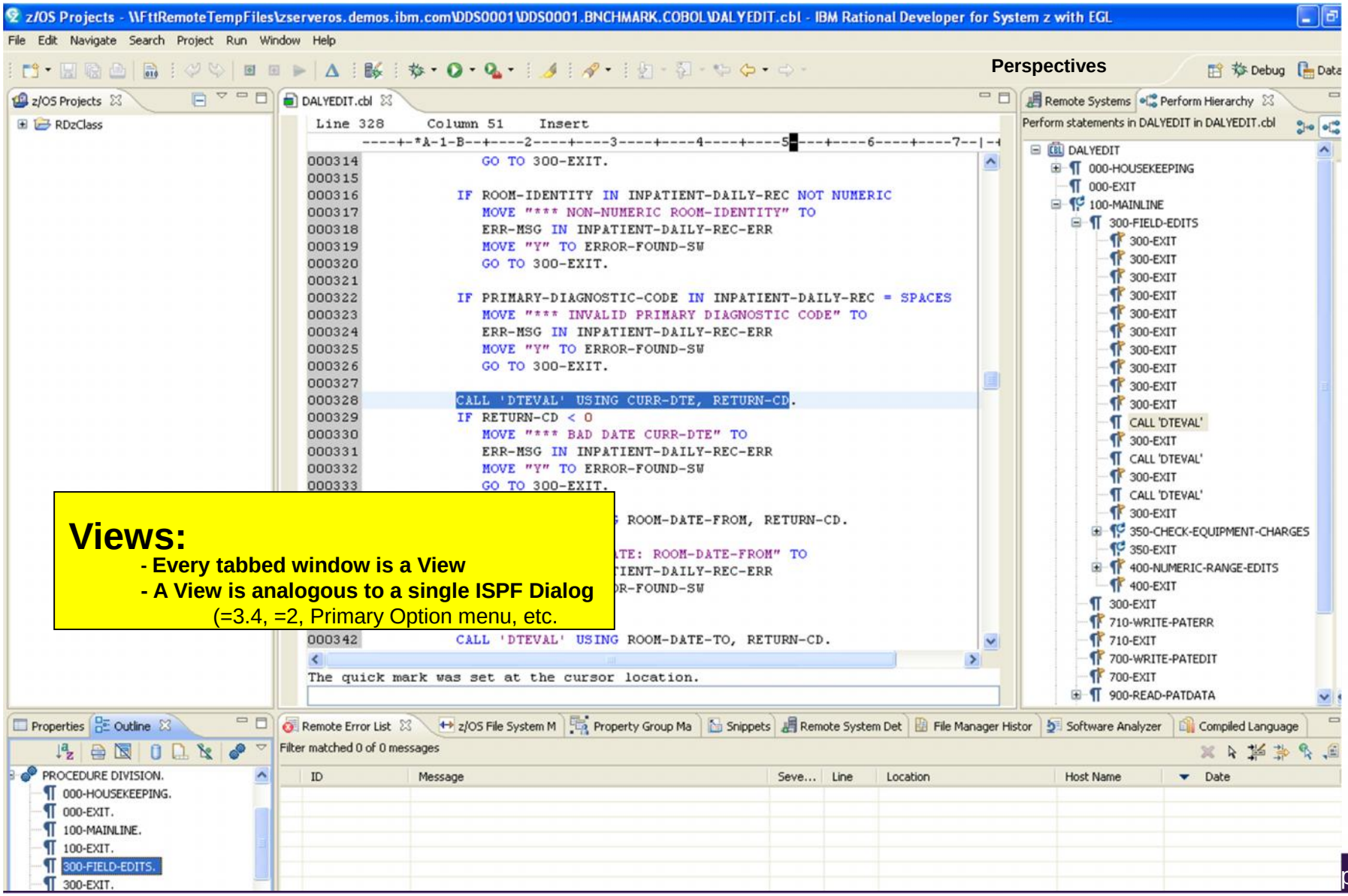
An integrated development environment to boost your Edit/Compile/Debug productivity:

- ✓ Rich source editing and navigation
- ✓ Code review for code governance
- ✓ Integration with the platform's latest compiler technology (including the recently released Enterprise COBOL for z/OS, V5.1)
- ✓ … and a new RDz Integrated Debugger to provide a seamlessly integrated & complete Edit/Compile/Debug solution out-of-the-box



The RDz Workbench – Terms and Concepts

The Workbench is based on graphical tools and the Eclipse framework. Many terms and concepts will be familiar to younger developers who used Eclipse-based tools in college

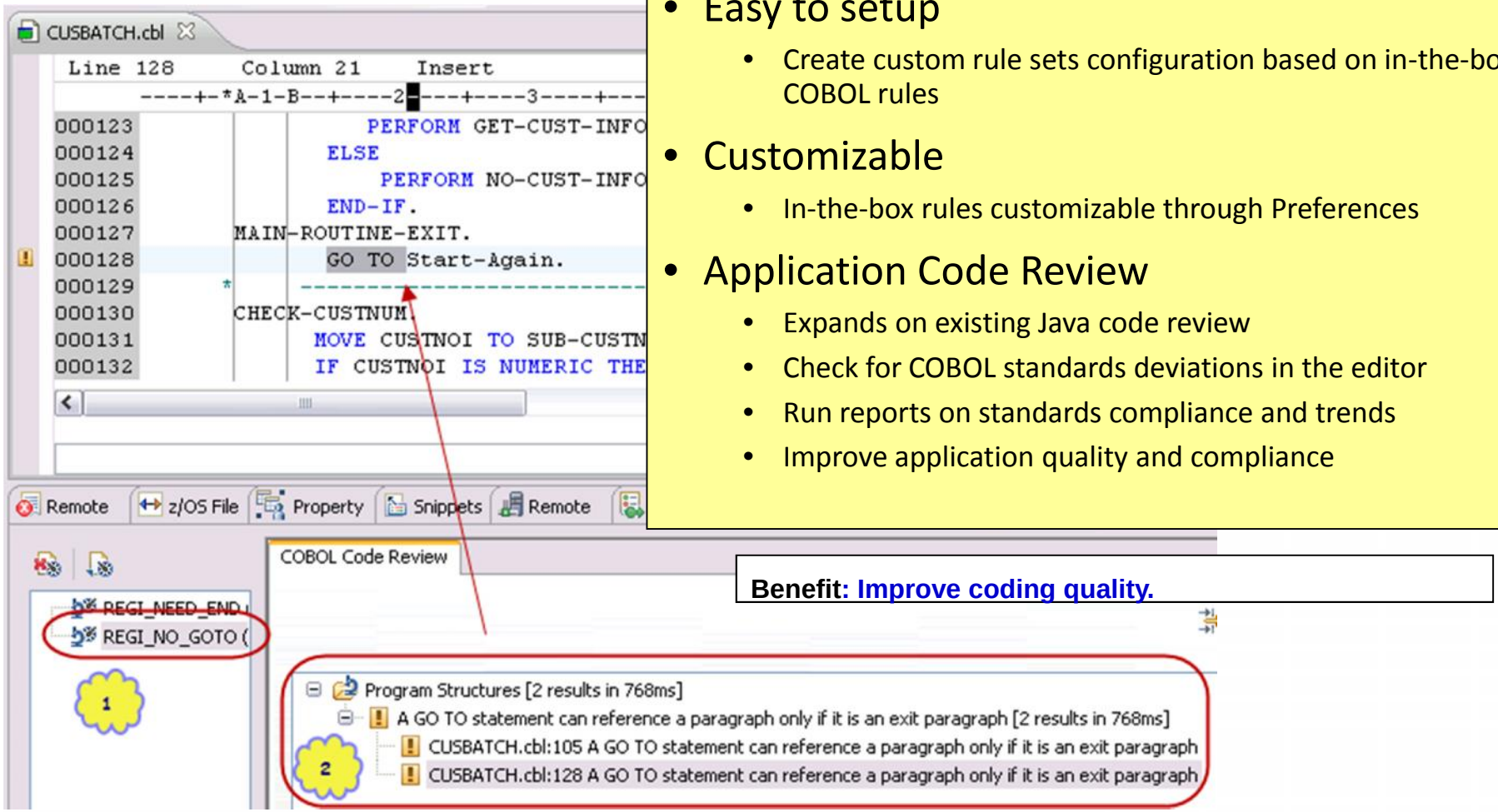


Views:

- **Every tabbed window is a View**
- **A View is analogous to a single ISPF Dialog**
  (=3.4, =2, Primary Option menu, etc.

Software Analyzer (Code Review)

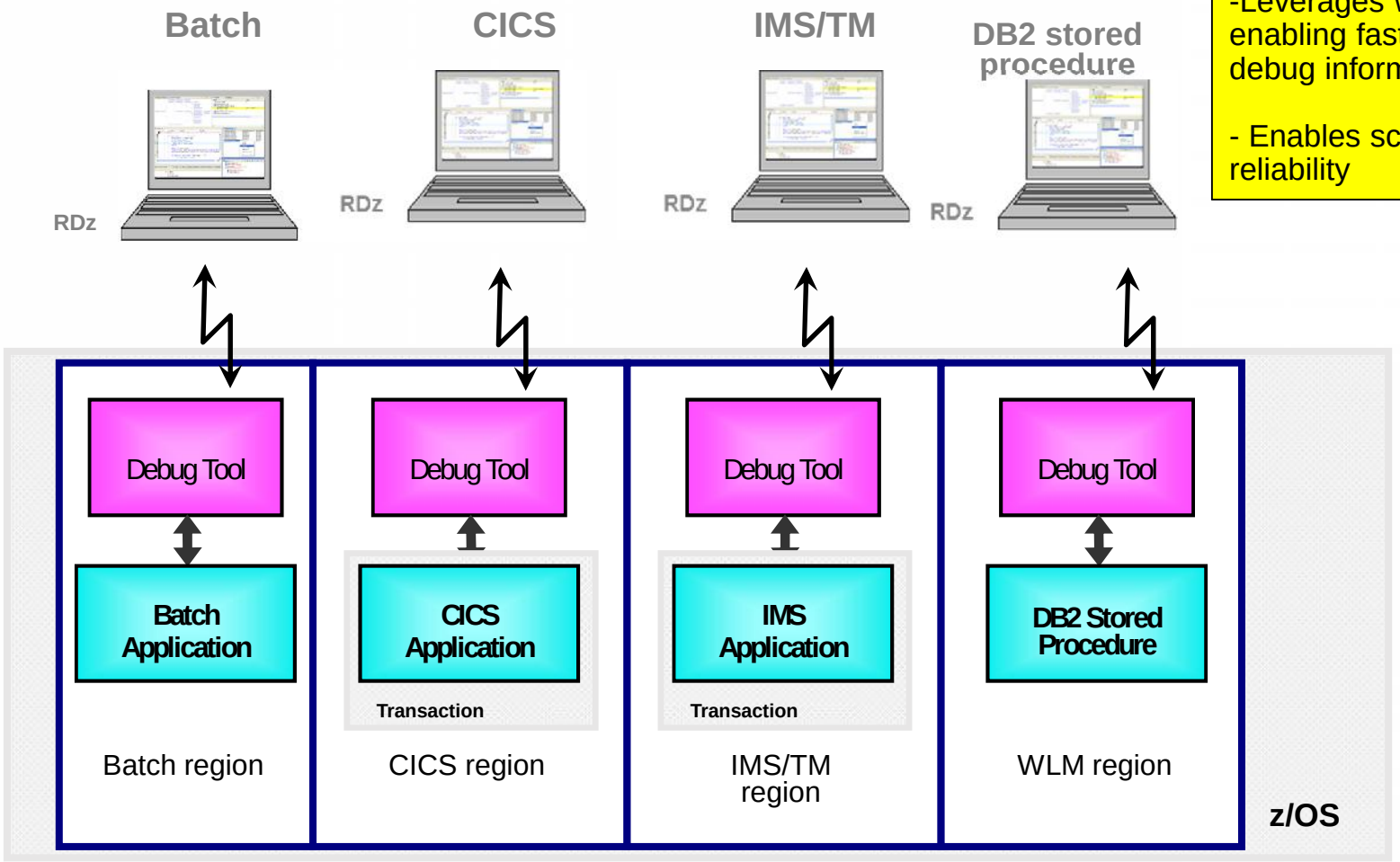
Also Running in Batch on z/OS



- Provides a means for you to enforce shop coding standards and best practices
- Easy to setup
  - Create custom rule sets configuration based on in-the-box COBOL rules
- Customizable
  - In-the-box rules customizable through Preferences
- Application Code Review
  - Expands on existing Java code review
  - Check for COBOL standards deviations in the editor
  - Run reports on standards compliance and trends
  - Improve application quality and compliance

Benefit: Improve coding quality.

RDz Integrated Debugger - Environments



✓**Host-offload architecture:**

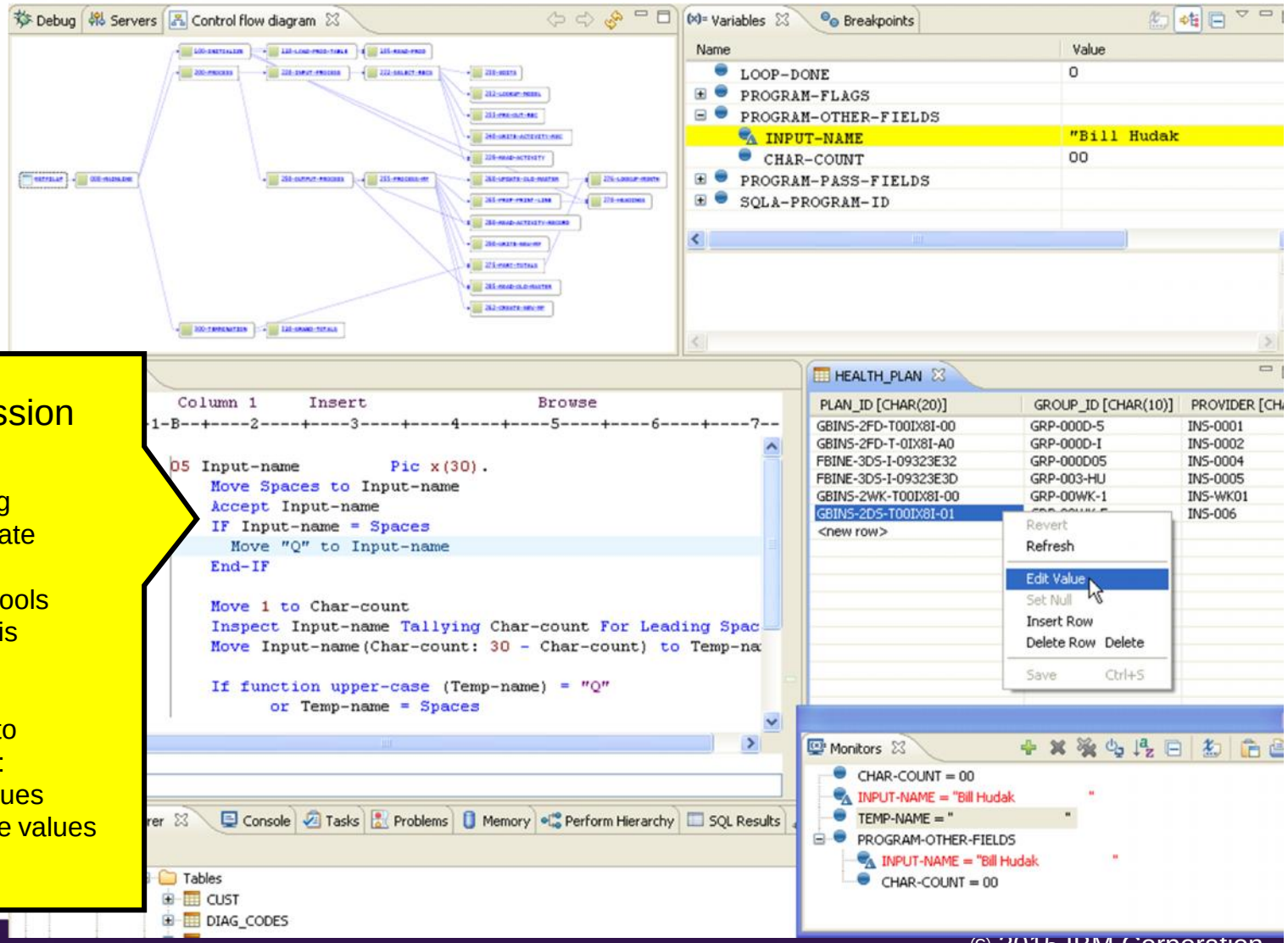
- Remote debugger with only a small footprint on the mainframe:

-Leverages workstation CPUs enabling faster processing of debug information

- Enables scalability and reliability

Debugging z/OS Applications

- Application debugging/testing tools – that scale to the complexity of your z/OS environment



An RDz Debug session

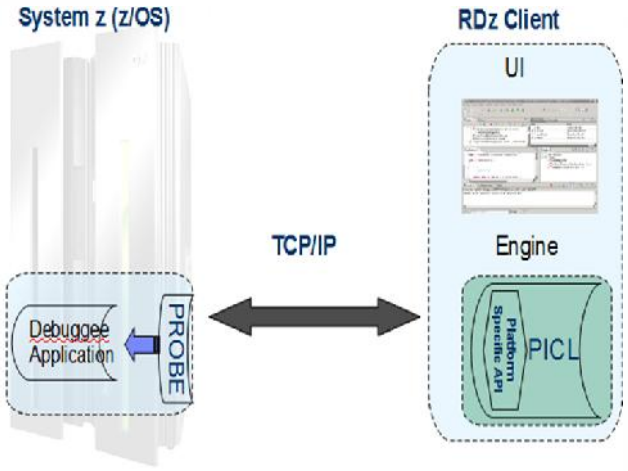
using:

- Variable monitoring
- Dynamic data update
- Breakpoints
- Access to source tools
  - Program analysis
  - Flow diagram
  - Navigation
- Real-time access to Edit and Browse:
  - DB2 table values
  - IMS Database values
  - VSAM files
  - QSAM files

RDz Integrated debugger....

- **A GUI-based multi-platform, multi-language debugger**
  - Full asynchronous mode
    - Thread-level control of multi-threaded applications

- RDz Supports**:**
  - COBOL V5.1, V4, V3.4
  - PLI v4.x, v3.9
  - C/C++ V1R13, V2R1
  - IMS TM
  - DB2 Stored procedures
  - Batch, Batch IMS, Batch DB2, CICS 5.2, 5.1, 4.2, 4.1
  - Interactive Code coverage – Out of the box



RDz Code Coverage

- Tracks tested lines of code
- Improves application quality
- Focuses testing resource usage
- Reports on tested code and trends

SAMJON2.jcl | Code Coverage Report (Nov 1, 2011 1:16:09 PM) | USER170.TEST.SYSDEBUG(SAM3).cob

```
363  024400      PERFORM 710-READ-TRAN-FILE.
364  024500
365  024600      IF WS-TRAN-EOF NOT = 'Y'
366  024700          COMPUTE NUM-TRAN-RECS = NUM-TRAN-RECS + 1
367  024800          MOVE 'Y' TO WS-TRAN-OK
368  024900          IF TRAN-KEY < WS-PREV-TRAN-KEY
369  025000             MOVE 'TRANSACTION OUT OF SEQUENCE' TO ERR-MSG-DATA1
370  025100             MOVE SPACES TO ERR-MSG-DATA2
371  025200             PERFORM 299-REPORT-BAD-TRAN
372  025300          ELSE
373  025400            EVALUATE TRAN-CODE
374  025500               WHEN 'UPDATE'
375  025600                   PERFORM 200-PROCESS-UPDATE-TRAN
376  025700               WHEN 'ADD    '
377  025800                   PERFORM 210-PROCESS-ADD-TRAN
378  025900               WHEN 'DELETE'
379  026000                   PERFORM 220-PROCESS-DELETE-TRAN
380  026100               WHEN OTHER
381  026200                   IF TRAN-COMMENT NOT = '*'
382  026300                     MOVE 'INVALID TRAN CODE:' TO ERR-MSG-DATA1
383  026400                     MOVE TRAN-CODE TO ERR-MSG-DATA2
384  026500                     PERFORM 299-REPORT-BAD-TRAN
385  026600                   END-IF
386  026700            END-EVALUATE
387  026800          END-IF
388  026900          MOVE TRAN-KEY TO WS-PREV-TRAN-KEY
389  027000          IF WS-TRAN-OK = 'Y'
```

SAMJON2.jcl | Code Coverage Report (Nov 1, 2011 1:16:09 PM)

Code Coverage Report

Code Coverage Summary

Code coverage report, generated Nov 1, 2011 1:16:09 PM

Element	Coverage	Covered Lines	Total Lines
SAM3	49%	106	213
SAM4	60%	34	57
USER170.TEST.SYSDEBUG(SAM4).cob	60%	34	57
SAM4()	60%	34	57
SAM3	46%	72	156
USER170.TEST.SYSDEBUG(SAM3).cob	46%	72	156
SAM3()	46%	72	156

The value of early and extensive testing

"80% of development costs are spent identifying and correcting defects" **

During the Coding or Unit Testing phases

$80/defect

During the BUILD phase

$240/defect

During Quality Assurance or the System Test phases

$960/defect

Once released into production

$7,600/defect

+

Law suits, loss of customer trust, damage to brand

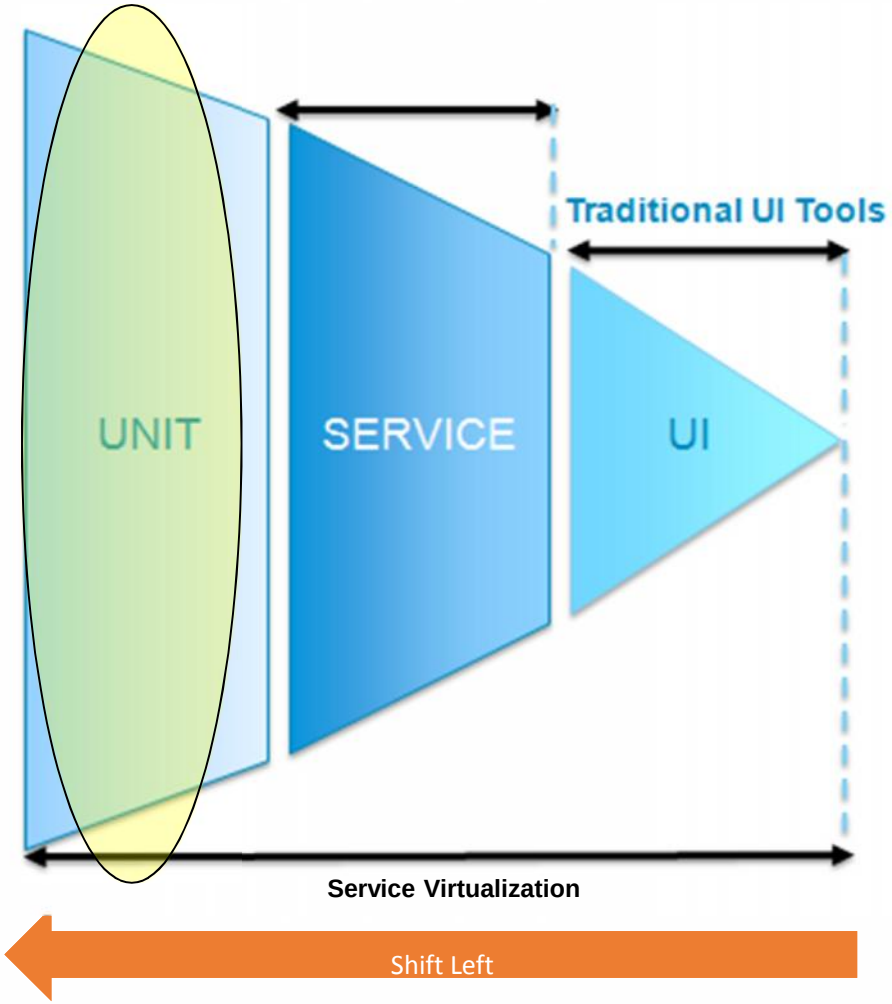
****National Institute of Standards & Technology**

Source: GBS Industry standard study

Defect cost derived in assuming it takes 8 hours to find, fix and repair a defect when found in code and unit test. Defect FFR cost for other phases calculated by using the multiplier on a blended rate of $80/hr.

Shift-Left

Are the testing efforts directed at the highest risks?



Automated Integration Testing

Why Unit Test?

By testing individual logic routines in your programs:

- You can move through the lifecycle more quickly, because you have precise feedback about separate logic routines
  - So you can better understand cause & effect
  - And you know **that** your code works and you know **how** your code works, which gives you confidence to make enhancements and modifications
- Because you execute zUnit Tests through JCL:
  - The testing can be automated
  - The end-to-end process takes less time than interactive debugging.
  - And it can be more systematic
- Because zUnit generates such a high % of the test code there's a relatively low Total Cost of Ownership

All of the above benefits allow you to catch errors earlier in the lifecycle…

How do you Unit Test - with zUnit

Using zUnit you:

1. **Develop Unit Tests for your programs**
2. **Execute the Unit Tests through JCL or interactively**
3. **Interpret/Analyze your Unit Test results**

The Unit Tests you develop are COBOL programs or PL/I procedures

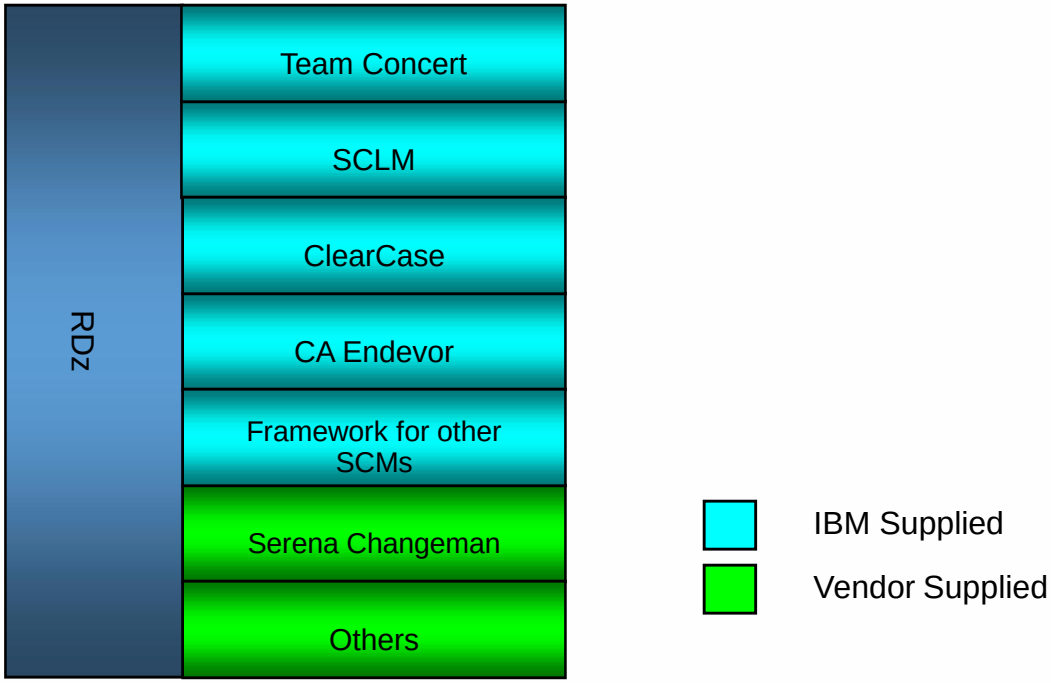
- **RDz generates 95+ percent of the COBOL and PL/I Unit Testing code**
- RDz provides the test harness (Test Runner) as an installed run-time
- And RDz provides both an interactive and batch test execution workflow

The zUnit development process is:

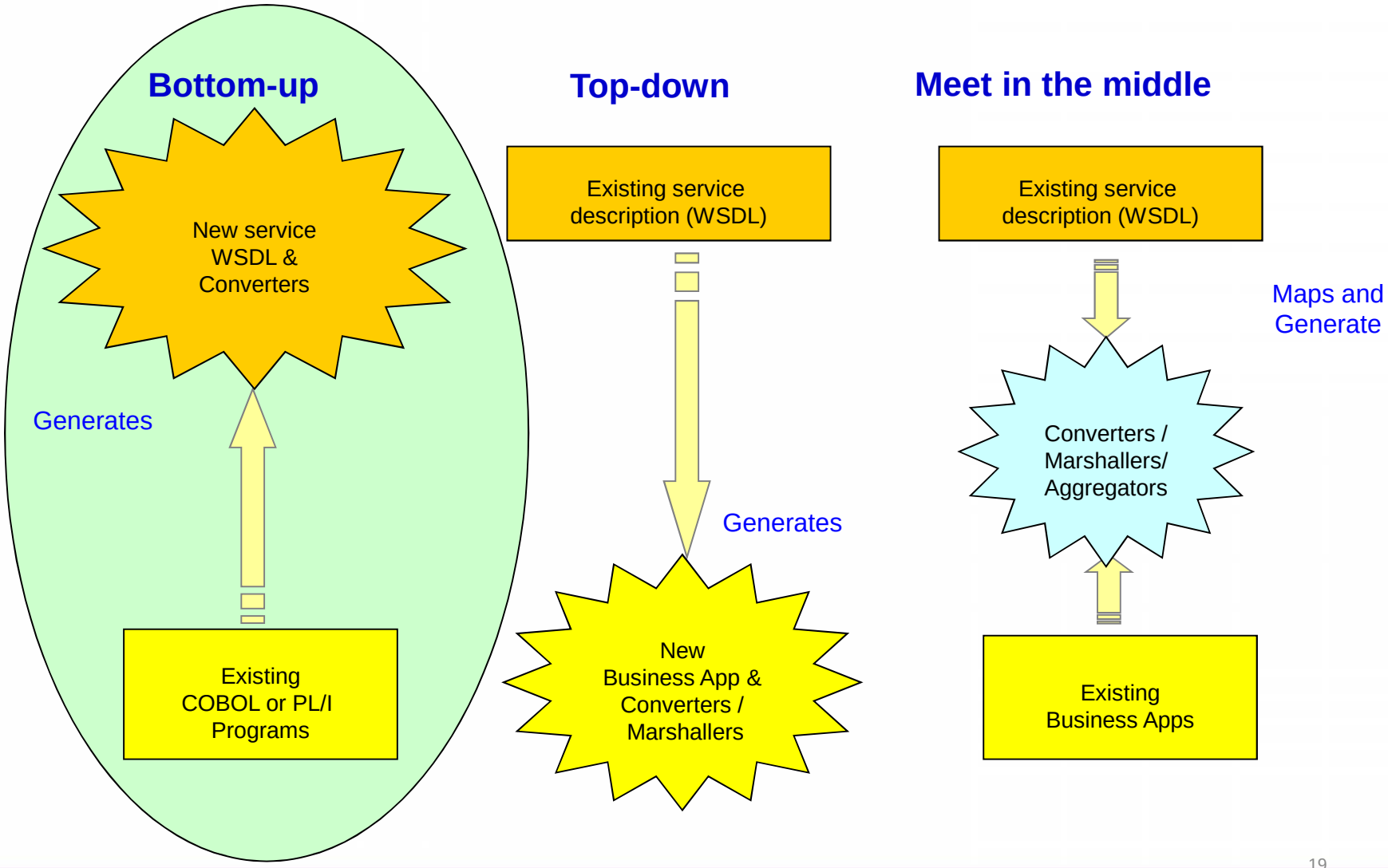
- **Wizard-driven**
- **Evolving** - as customers adopt, and feedback next-generation enhancements and suggestions

Accessing z/OS Repository…

- Rational Developer for System z offers integration into a variety of Source Code Management (SCM) tools as well as a framework for creating SCM integration on your own
- Variety of vendors supply plug-ins to Rational Developer for System z to provide easy access to processes and source code controlled by their products



Web Service Enablement Styles



Why use RDz for Traditional TSO/ISPF Development?

1. Developer Productivity

RDz has an enormous assortment of high-value tools that:

- **Complement the functionality of ISPF** – to automate, streamline and simplify the tasks of everyday z/OS maintenance, production support and development
- **Integrate with tools within and outside of the IBM solution set** – which allow you to tap into your site-specific trusted and mature development processes, and access high-end functionality from IBM and OEM solution providers running on Eclipse
- **Emulate the functionality of ISPF** – for fast on-ramping of veteran TSO developers

2. Code Quality

RDz also has capabilities and features that improve:

- Code maintainability
- Production application run-time efficiency

3. Development and Application Modernization

RDz's wizards for Web Service generation and integration are unparalleled, for depth and breadth of functionality and ease-of-use.

Net: RDz can be your standard platform for:

- z/OS development, maintenance and production support for: **COBOL, PL/I and Assembler** applications that use: **DB2, IMS, CICS, VSAM/QSAM, REXX**
- **Java/J2EE** and C++ development
- z/OS application and **SOA modernization**


Questions

Learn More About RDz

- Contact your IBM account representative, and ask about a Proof-of-Technology
- Take a class:
  - From IBM/Rational Education
    - http://www-01.ibm.com/software/rational/education/
  - From an IBM Business Partner
    - http://www.clearblade.com/Product%20PDFs/ClearBlade%20RDz%20Offering.pdf
    - www.soforte.com
    - http://qgrp.com/english/
  - RDz Distance Learning – IBM sponsored web-based workshops
    - https://www.ibm.com/developerworks/mydeveloperworks/wikis/home?lang=en#/wiki/Learn%20RDz-Learn%20COBOL
- Attend an IBM / Rational event:
  - Innovate
    - http://www-304.ibm.com/jct03001c/services/learning/ites.wss?pageType=page&c=a0008413
  - RDz Online User Group Meeting
    - https://www.ibm.com/developerworks/mydeveloperworks/groups/service/html/communityview?communityUuid=22eac60d-8bab-44e2-a5b8-a4fe1c1aecad
- Learn about RDz online:
  - RDz Product Page on the web
    - http://www-01.ibm.com/software/rational/products/developer/systemz/
  - IBM InfoCenter
    - http://publib.boulder.ibm.com/infocenter/ratdevz/v8r0/index.jsp?topic=/com.ibm.etools.getstart.wsentdev.doc/helpindex_rdz.html
  - IBM / Rational "Education Assistant" – an assortment of individual feature tutorials
    - http://publib.boulder.ibm.com/infocenter/ieduasst/rtnv1r0/index.jsp
  - IBM Business Partner videos
    - http://vimeo.com/channels/clearbladetv#36157534
  - YouTube
    - www.youtube.com – search on RDz

ibm.com/devops

Acknowledgements and disclaimers

Availability: References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates.

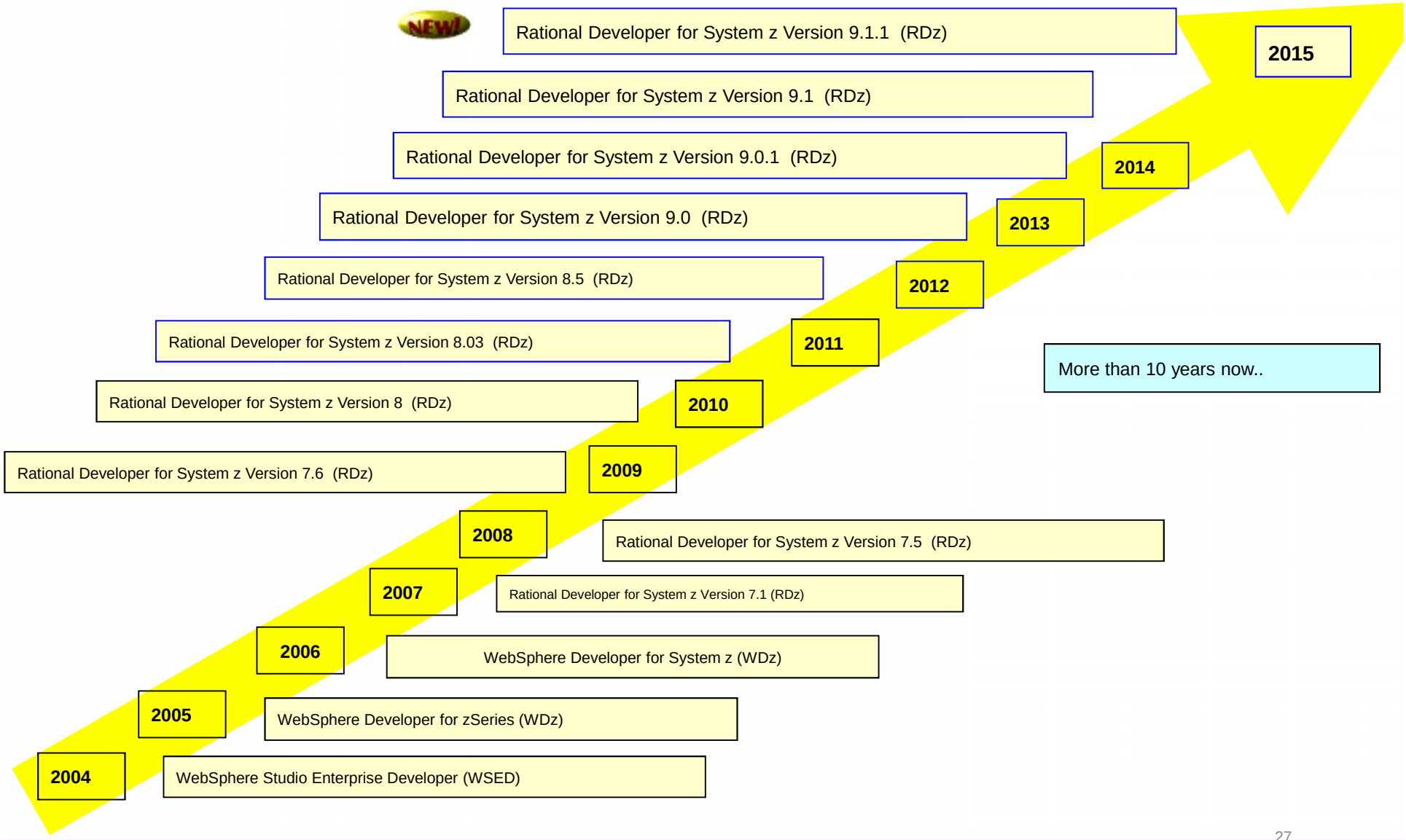
The workshops, sessions and materials have been prepared by IBM or the session speakers and reflect their own views. They are provided for informational purposes only, and are neither intended to, nor shall have the effect of being, legal or other guidance or advice to any participant. While efforts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

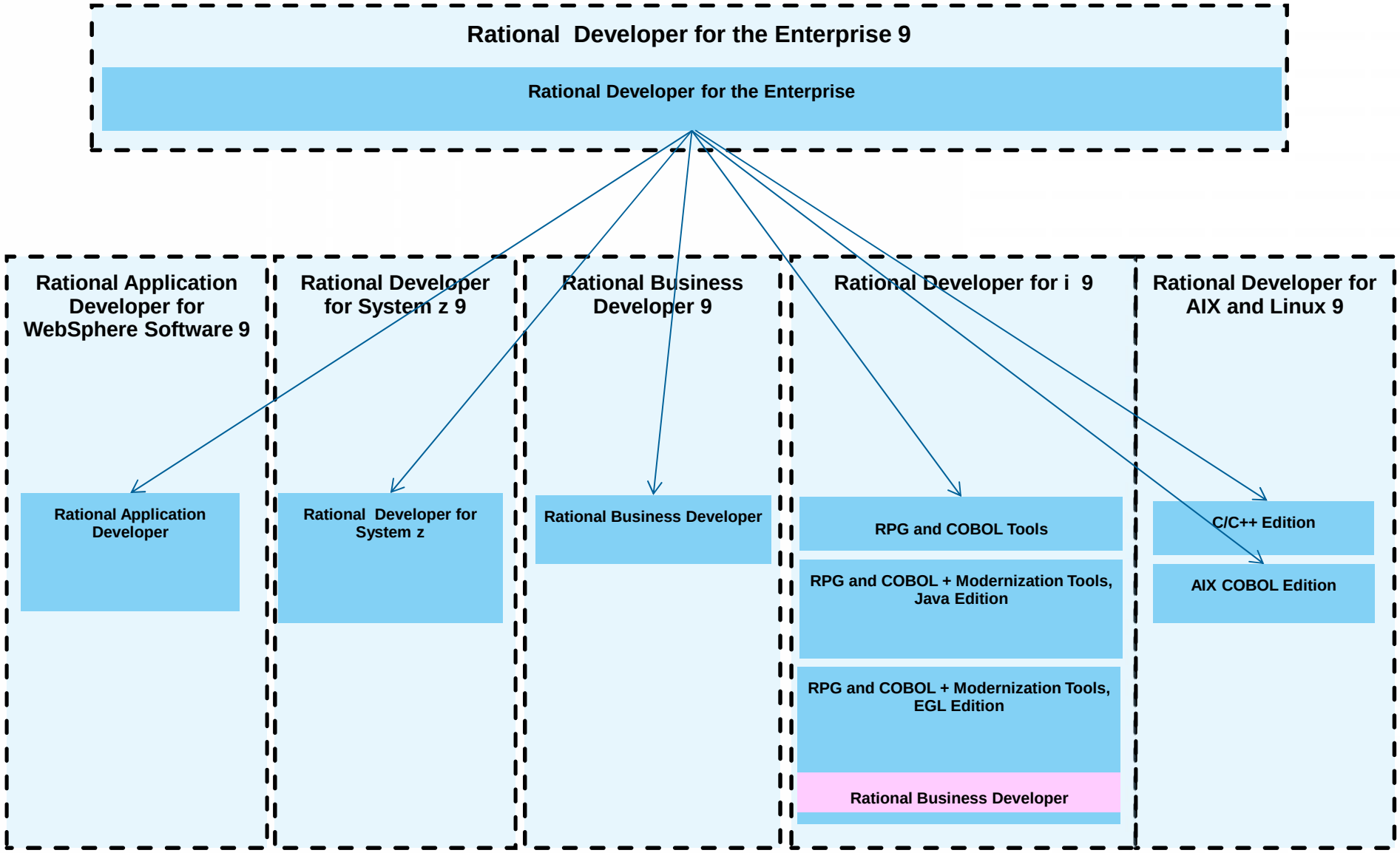
All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.

END

- Remaining Slides are Optional

History of Rational Developer for System z





RDz Functional Taxonomy – a Partial List

z/OS Development, Maintenance and Production Application Support

Access Datasets/Source Files

- SCM functional integration
- PDS Support
- Migrate/Recall Support
- Local and Remote file support
- Tooling support in single or across multiple LPARs
- Windows metaphor
- Edit/Browse/View
- "Favorites" – "Most recently used"
- Source and PDS Search
- QSAM Data File Search
- Browse Load Module
- Search Load Library
- Use of Regular Expressions

Program Analysis

Source Navigation
- Windows (Standard) Navigation
- ISPF PF-keys + extensible Hot-keys
- Outline View
- Hover
- Open Declaration / Arrow keys
- Open copybooks

Program Logic tools
- Control Flow Analysis
- Data Flow Analysis
- Where used/Where Referenced

Windows Screen Real Estate
- Size-able views
- Multi-window development
- Source Filters
- Collapse/Expand paragraphs/sections

Source Development

ISPF and RDz Source Editing
- PF-Keys
- Hexedit
- Prefix Area Commands
- Command Line Commands
- Colorized statement support
- Local History
- PC Source editing functionality
- Code refactoring
- Wizard-driven DB2 Stored Procedure generation
- Comment/Un-comment multiple lines
- Access to 3270 Emulation within Eclipse
- All development options "preference-enabled"

Languages
- COBOL, PL/I, Assembler, Java, C/C++
- JCL/CLIST/REXX
- SQL
- BMS/MFS
- WSDL, HTML, XML
- 4GLs supported with Eclipse Tooling

Syntax Check and Build
- Real-time validation
- Local and Remote Syntax Checking
- Integration with z/OS Build Process

Editing Data Sources
- QSAM File Editor
- DB2 Table Editor
- IMS Segment Editor
- VSAM File Editing with File Manager
- Integration with File-Aid Plug-ins

Test and Debug
- Integration with PD Tools/Debug Tool
- Integration with Xpeditor and Intertest

Content Assist
- COBOL, PL/I, Assembler
- SQL: Embedded, Interactive
- CICS statements

Submitting/Managing Jobs

- Submit and Locate Job
- Integration with JES
- Job Organization options (Filters)
- Show JCL
- Cancel/Purge

Dataset Management

- Allocate/ Rename/Delete
- Create GDG Model
- Create VSAM Dataset
- Search
- Compress

Copy Files
- Within an LPAR
- Across LPARs
- LPAR ←→ PC

Code Quality

- Code Review
- Source Format
- File Compare
- All of the above functionality

Enterprise Modernization

CICS Web Services

Generate:
- WSDL
- WSBIND file
- XSD files
- Deployment manifest
- Stub modules

• Test and Deploy WSDL

• Use Cases:
- Bottom Up
- Top Down
- Meet in the middle

IMS Soap IMS Web 2.0

Generate
- XML/WSDL
- COBOL/PLI converters
- Manifest files

Use Cases:
- Bottom Up
- Top down (PL/I only)
- Meet in the middle

CICS Service Flows

- 3270 "screen scraping"
- Aggregate transactions
- Automate processes
- Expose as web services

RDz Product Integration

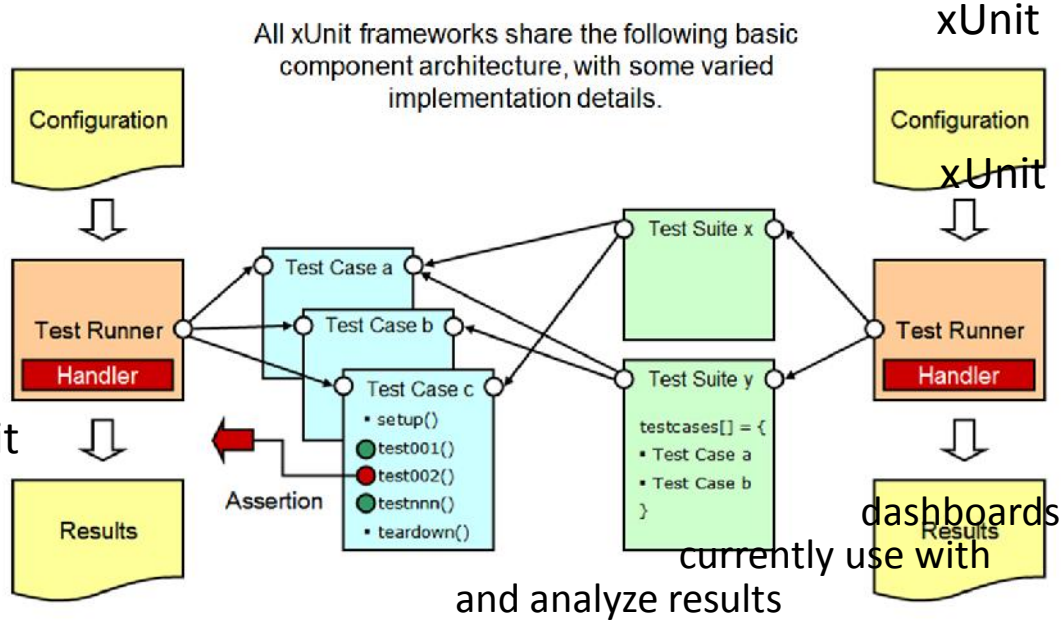
SCM:
- **IBM:** Team Concert, SCLM, ClearCase
- **CA:** Endevor, Panvalet, Librarian,
- **Serena:** Changeman
- **ISPW**

Functional Integration with z/OS REXX/CLIST/3rd Party Tools:
- Menu Manager
- HATS
- Eclipse Plug-in Integration

zUnit and the xUnit Standard

- **xUnit** is an industry-standard, program testing framework - that helps create and automatically run repeatable, self-checking unit test cases.
  - **xUnit provides:**
    - Assertion-style test validation capabilities and result reporting.
    - Automated execution validation - instead of independent/interactive testing activity
  - At run-time, the xUnit framework distinguishes between failures and errors:
    - A failure is an anticipated problem (e.g., actual output does not match expected).
    - An error is an unexpected, catastrophic problem (e.g., protection exception, null pointer)

- **jUnit** is an instance of the architecture for Java
- **zUnit** is IBM's implementation of the architecture for System z
- **zUnit** provides the same industry-standard output reports as jUnit
  - You can use the same XML-reporting and products that you jUnit to review



xUnit

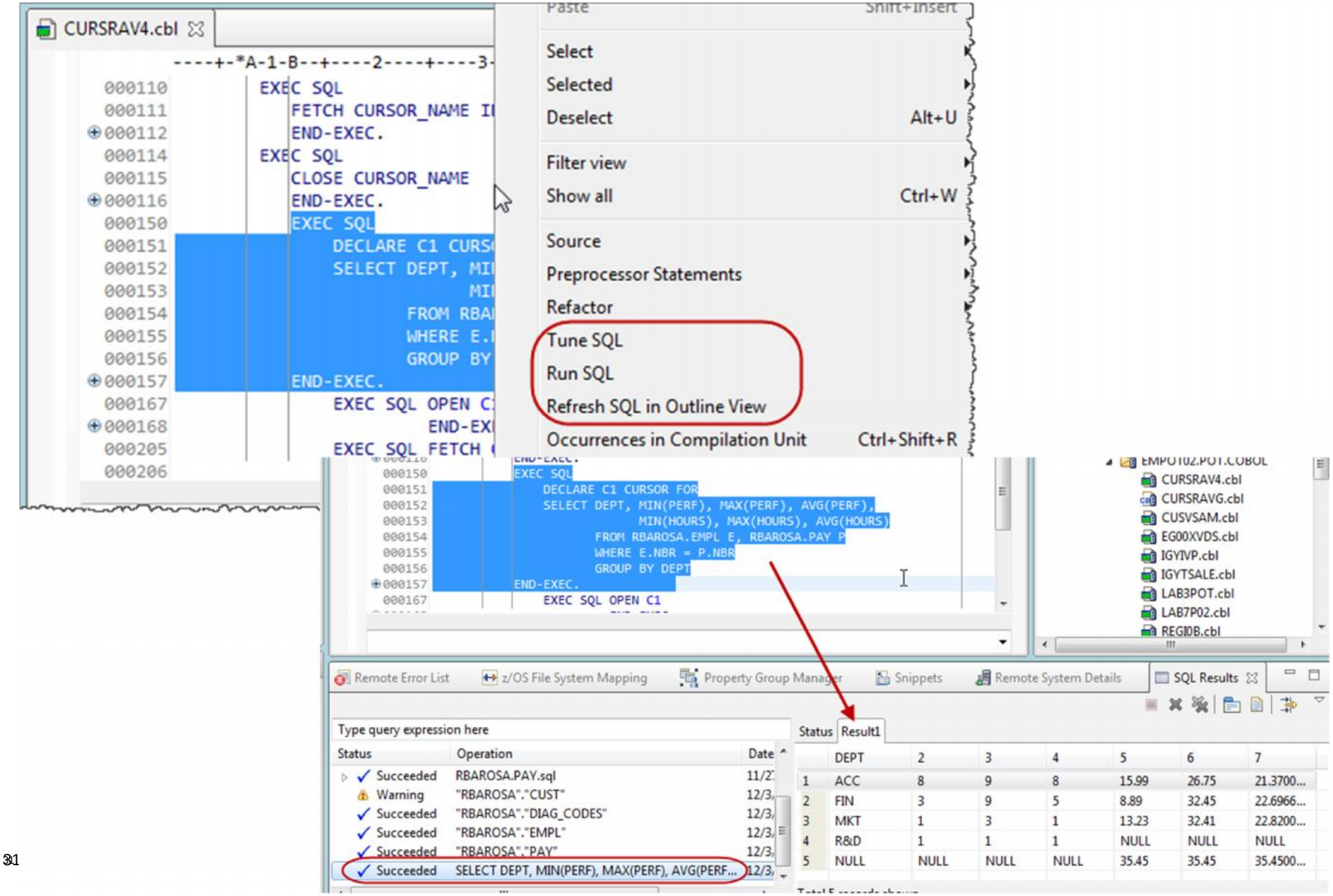
xUnit

dashboards currently use with and analyze results

Other Demonstration Use Cases

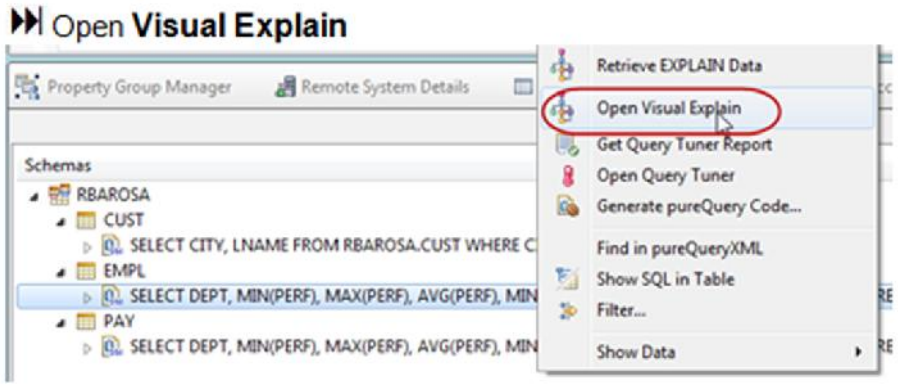
DB2 Tools integration with COBOL Editor

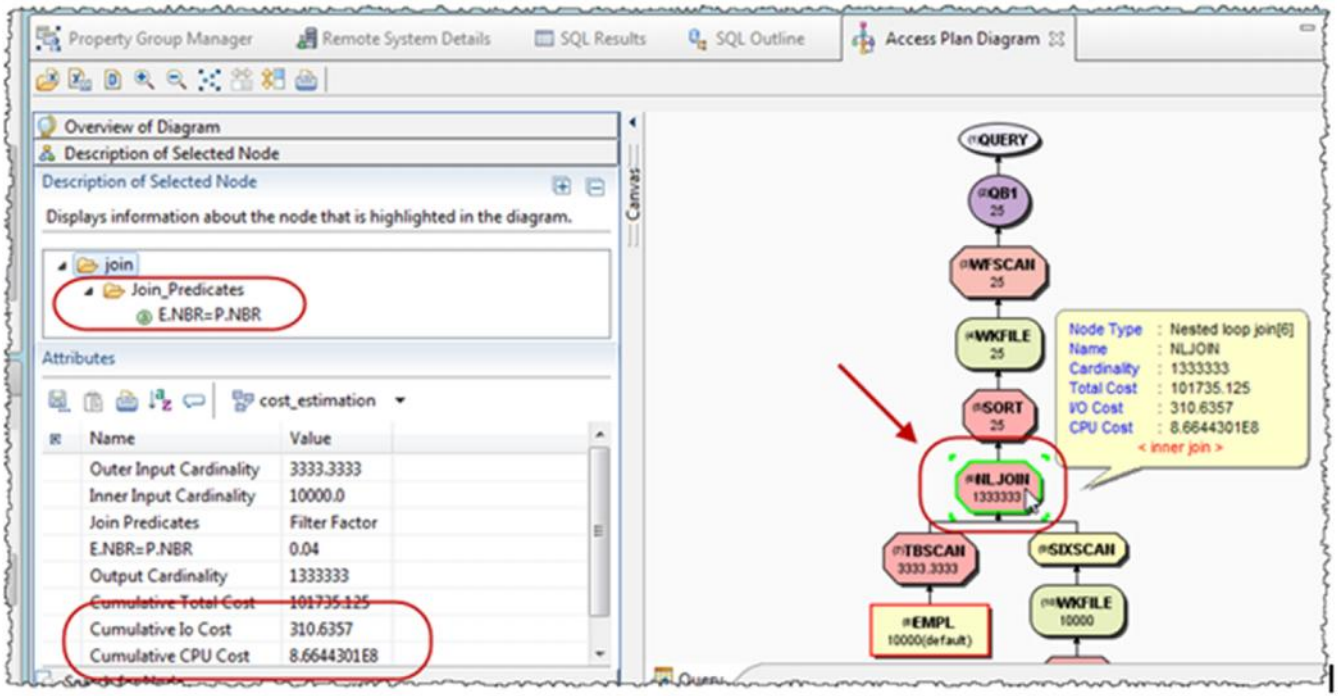
See: ..\..\RDz9.0\MOVIE_DB2_Deep_Dive_RDz_Users_Group\RDz User Group Meeting - Oct. 8th.wmv



31

DB2 Tools integration with COBOL Editor





Other Demonstration Use Cases

▪DB2 Tools integration with COBOL Editor