16520: Application Development for z/OS - Not your Father's Green Screen

Rosalind Radcliffe

Distinguished Engineer, IBM Academy of Technology Member

IBM Corporation

rradclif@us.ibm.com

@RosalindRad

#SHAREorg

SHARE is an independent volunteer-run information technology association
that provides **education**, professional **networking** and industry **influence.**

SHARE in Seattle 2015

Abstract

Ask most people how they write and maintain applications on z/OS and you hear "oh, you use this thing called a green screen" followed by a chuckle.

In reality, application development for zEnterprise applications has been transformed over the past several years to the point where application developers enjoy the same or better features from integrated development environments as programmers who work on other platforms.
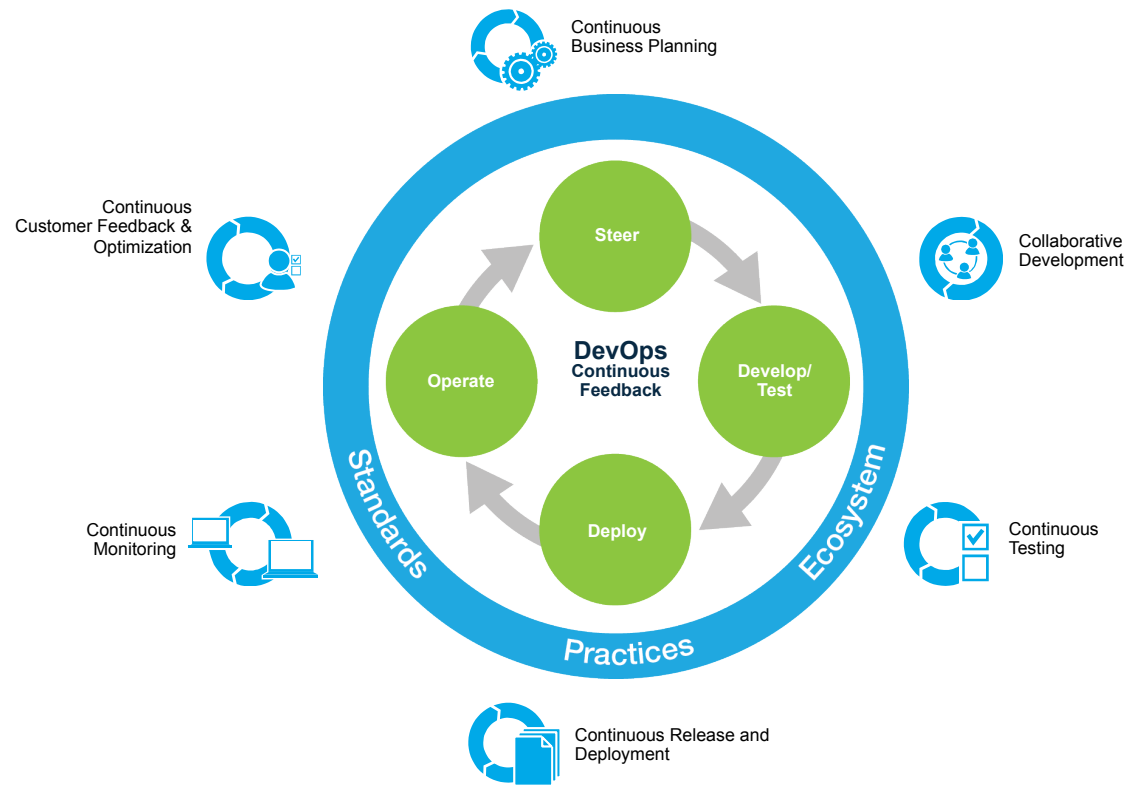
Advances in remote system communication and interaction, syntax-highlighting, parsing, and code understanding for Assembler, PL/I, C/C++, and COBOL source code, as well as programming assists such as code snippets and templates are all available to application programmers. Interactive debug of applications, written in multiple programming languages and running in various runtime environments is also possible and can greatly boost programmer productivity.

Come and learn about how these features can enable application developers who are new to the mainframe to interact with, update, and efficiently enhance mainframe applications.

IBM DevOps point of view

Enterprise capability for continuous software delivery that enables organizations to seize market opportunities and reduce time to customer feedback

Accelerate software delivery – *for faster time to value*

Reduce time to customer feedback – *for improved customer experience*

Balance speed, cost, quality and risk – *for increased capacity to innovate*

Continuous Business Planning

Continuous Customer Feedback & Optimization

Collaborative Development

Continuous Monitoring

Continuous Testing

DevOps Continuous Feedback

Steer

Develop/Test

Operate

Deploy

Standards

Ecosystem

Practices

Continuous Release and Deployment

Clients are achieving **measurable business outcomes with DevOps**

Collaborative Development

Integrated Application Lifecycle Management for all platforms

- Accelerate **time to delivery** with real-time planning
- Improve **quality** with lifecycle traceability
- Maximize **time to value** with in-context collaboration
- Refine **predictability** with development intelligence
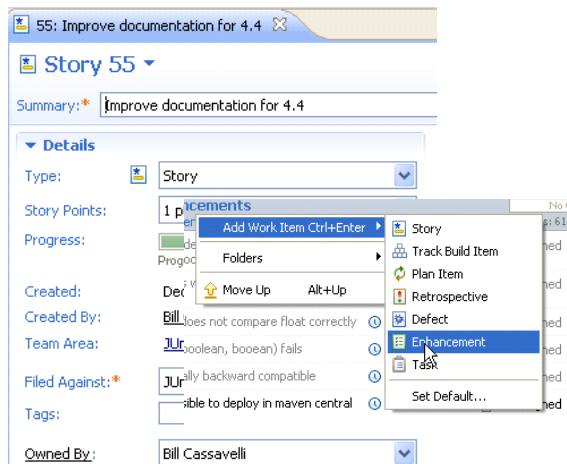- Reduce **costs** with continuous improvement

✓**Manage many sources - JavaScript to COBOL**
✓**Single work item can span technologies & solutions**
✓**Quickens agile adoption on the mainframe**
✓**Integrates existing SCMs and deployment tools**

▪ IBM Rational solution for Collaborative Lifecycle Management (CLM)
   ▪ IBM Rational Team Concert
   ▪ IBM Rational Doors Next Generation
   ▪ IBM Rational Quality Manager

Collaborative Development

Project Managers / SCRUM Masters		Quality and Test
Administrator		Agile Teams
LoB / Business Analyst		Developers

Continuous Delivery

Open Lifecycle and Service Management Integration Platform

jazz

5
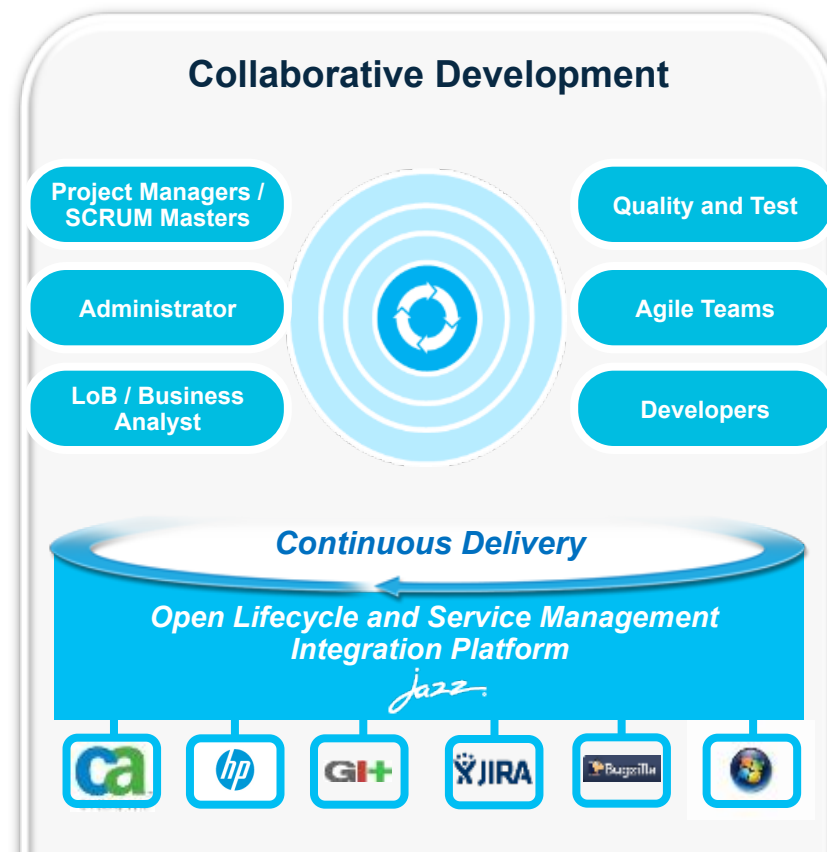
Rational Team Concert – A single tool, many capabilities

- Work Items

- Planning
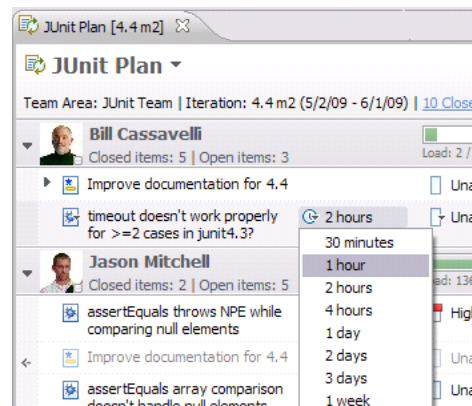
- Source Control
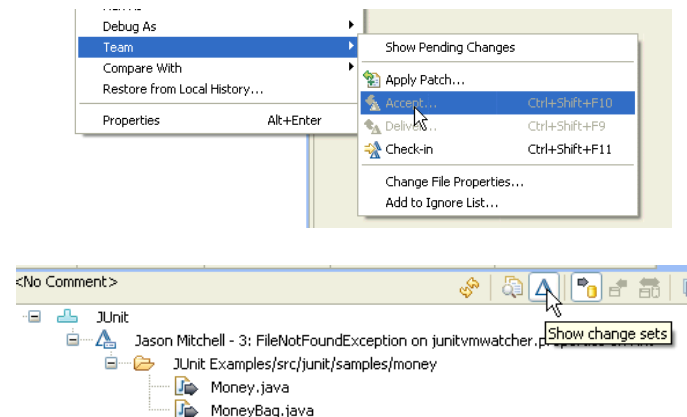
- Builds – Continuous
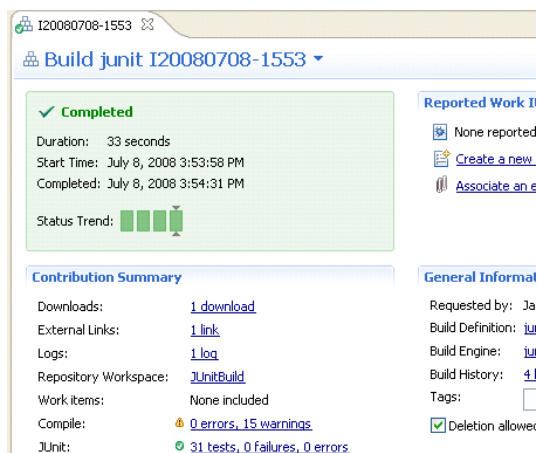
- Dashboards & Reporting
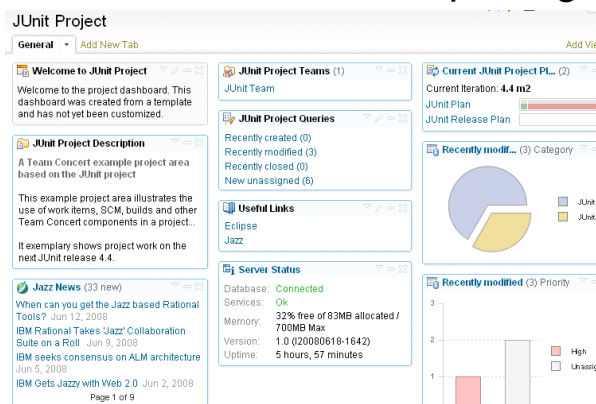
- Method Enforcement and Automation

Rational Developer for System z:
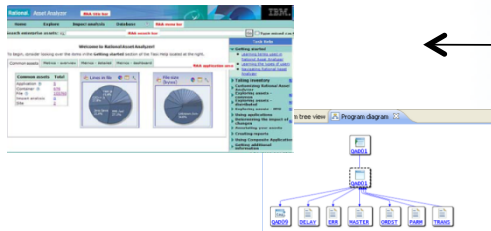## An Integrated Development Environment for System z

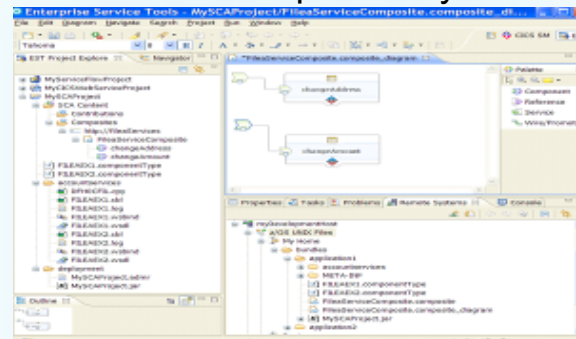Integration with Team Concert for Lifecycle and Source Management
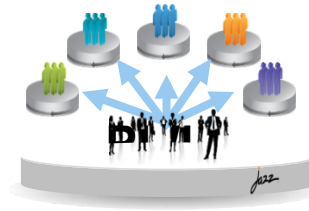
Access to typical System z sub-system functionality in z/OS, CICS, IMS, DB2, WAS

Integration with Asset Analyzer for Application Understanding and Impact Analysis
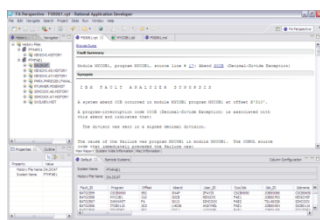
Rational Developer for System z

New

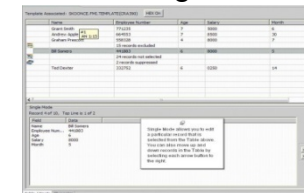**Out of the Box debugger and code coverage capabilities**

A modern IDE for productive development of cross-platform applications written in COBOL, PL/I, ASM, Java, EGL or C/C++ in System z CICS, IMS, DB2, Batch applications

Integration with Fault Analyzer for Dump Analysis

Integration with File Manager for file and test data handling

Integration with RD&T for flexible access to System z environment

The Benefits of RDz

Instead of maneuvering to access panels and working **sequentially**,
in RDz the functionality you need is always in-focus – you work **concurrently**

Edit a program

Access Datasets + Dataset Management

File Compare

File Search

Submit a Compile

Dataset Statistics

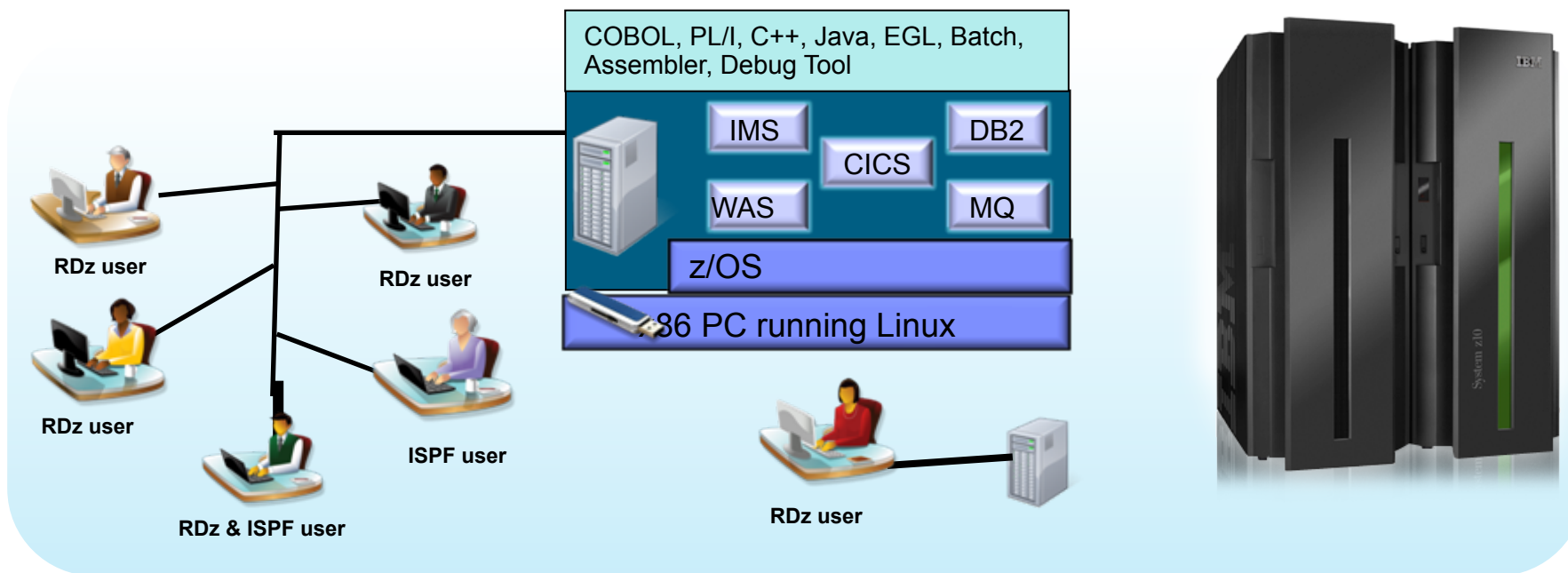**Access Jobs (Outlist facility)**

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

8

Rational Development and Test Environment for System z
The ultimate in modern application development for System z

- Increase availability of z/OS testing environment and resources
    - Liberate developers to rapidly prototype new applications
    - Develop and test System z applications anywhere, anytime!
    - Eliminate costly delays by reducing dependencies on operations staff
- Improve quality and lower risk via automation, measurement, and collaboration
- Focus on what is required for the change at hand, then scale

Note: This Program is licensed only for development and test of applications that run on IBM z/OS. The Program may not be used to run production workloads of any kind, nor more robust development workloads including without limitation production module builds, pre-production testing, stress testing, or performance testing.
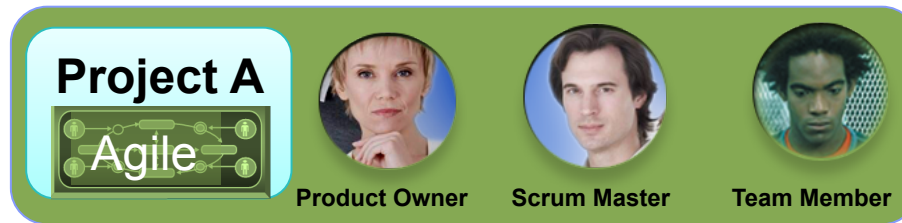
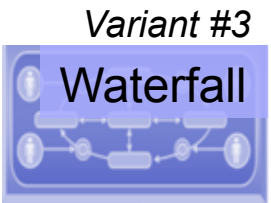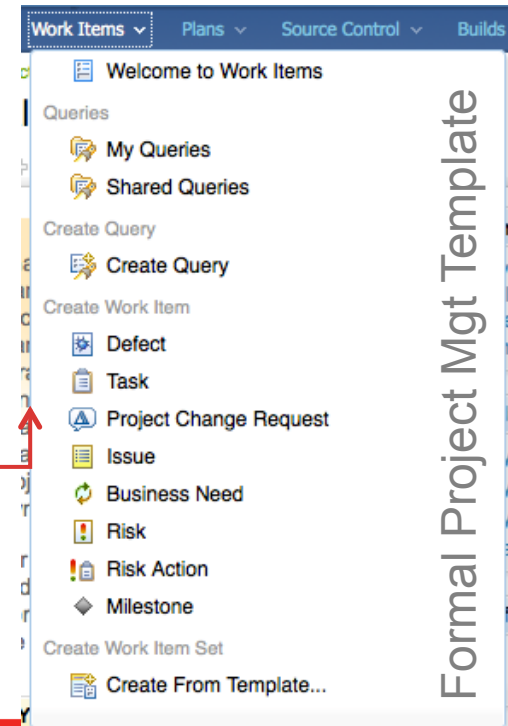# Any process: Executable and repeatable
Use ONE tool to support both agile and non-agile

Process Architect

Project Manager

Variant #1

Scrum

Variant #2

Iterative

Variant #3

Waterfall

Project A

Agile

Product Owner

Scrum Master

Team Member

Project B

Iterative

Analyst

Developer

Quality Professional

Release Engineer

Proj

Waterfall

Analyst

Developer

Quality Professional

Release Engineer

Scrum Template

Work Items ∨   Plans ∨   Source Control   Build

- Welcome to Work Items

Queries
- My Queries
- Shared Queries

Create Query
- New Query

Create Work Item
- Defect
- Task
- Story
- Epic
- Track Build Item
- Impediment
- Adoption Item
- Retrospective

Create Work Item Set
- Create from Template

Formal Project Mgt Template

Work Items ∨   Plans ∨   Source Control   Builds

- Welcome to Work Items

Queries
- My Queries
- Shared Queries

Create Query
- Create Query

Create Work Item
- Defect
- Task
- Project Change Request
- Issue
- Business Need
- Risk
- Risk Action
- Milestone

Create Work Item Set
- Create From Template...

Complete your session evaluations online at www.SHARE.org/Seattle-Eval

10

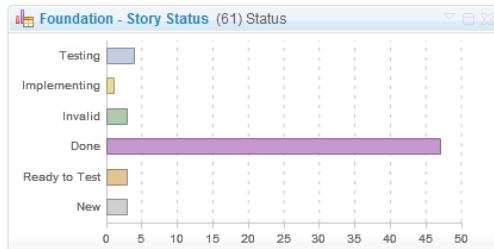# Multiple plan views facilitate continuous planning

Complete your session evaluations online at www.SHARE.org/Se

Progress Tracking - Everyone can see live project status
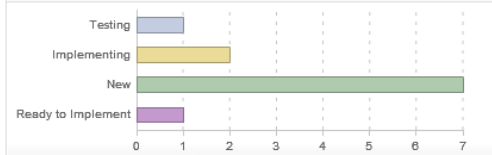
Story Status Current Iteration

Shows the status of all stories planned for the current iteration

Foundation - Story Status (61) Status

Status	
Testing	
Implementing	
Invalid	
Done	
Ready to Test	
New	

0 5 10 15 20 25 30 35 40 45 50

Foundation - Current Stories (61)

- 155918: Provide TVT testcases and testcase updates to TVT team
- 155399: [Repository - M14] Performance, quality and robustness enhancements
- 154407: [Continuation] Manage and repair DB inconsistencies during 3.0.1 M14
- 155024: Track the fixing of hangs occurring on shutdown of the CLM server
- 155033: Explore servlet dispatching enablement as a backup option
- 155030: [Application SDK - 14] Performance, quality and robustness enhancements
- 155120: LPA Web UI fixes and polish
- 111848: User can validate consistency of configuration information across contexts
- 155168: Fix bugs and improve the quality of LPA services
- 155624: update JREs for 3.0 iFix1 and 3.0.1

Page 1 of 7

CCM - Story Status (11) Status

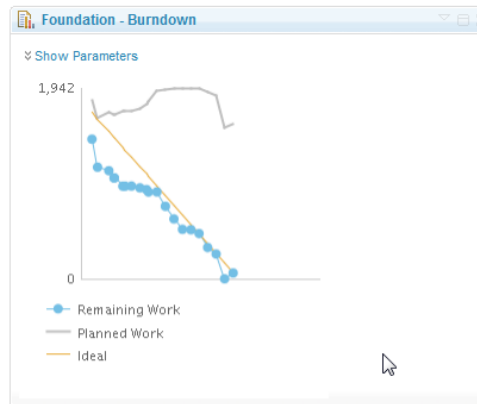| Status	
Testing	
Implementing	
New	
Ready to Implement	

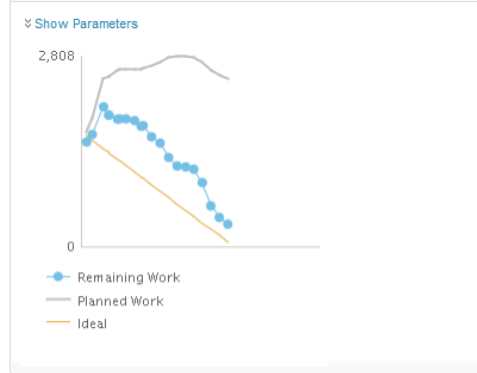0 1 2 3 4 5 6 7

CCM - Current Stories (11)

- 150244: address "run as administrator" issues on win7
- 142127: As a user I want to see all extended state groups in taskboards
- 150856: Improve the performance of loading a work item

Burndown Current Iteration

Shows the remaining amount of estimated work in hours of work items planned for the current iteration.
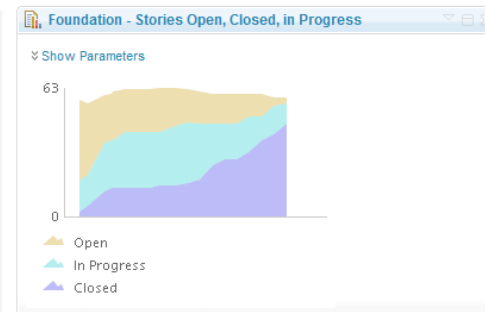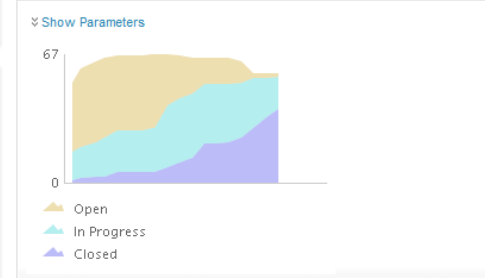
Foundation - Burndown

Show Parameters

1,942

0

- Remaining Work
- Planned Work
- Ideal

CCM - Burndown

Show Parameters

2,808

0

- Remaining Work
- Planned Work
- Ideal

Burndown RM
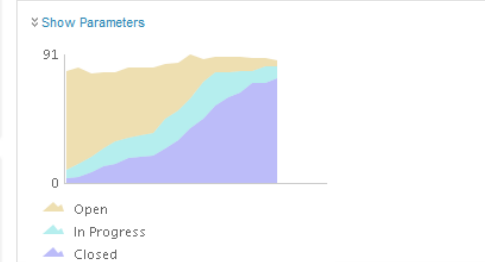
Show Parameters

3,932

Stories Open/Closed/In Progress

Shows the number of stories which are open, in progress, done during the iteration.
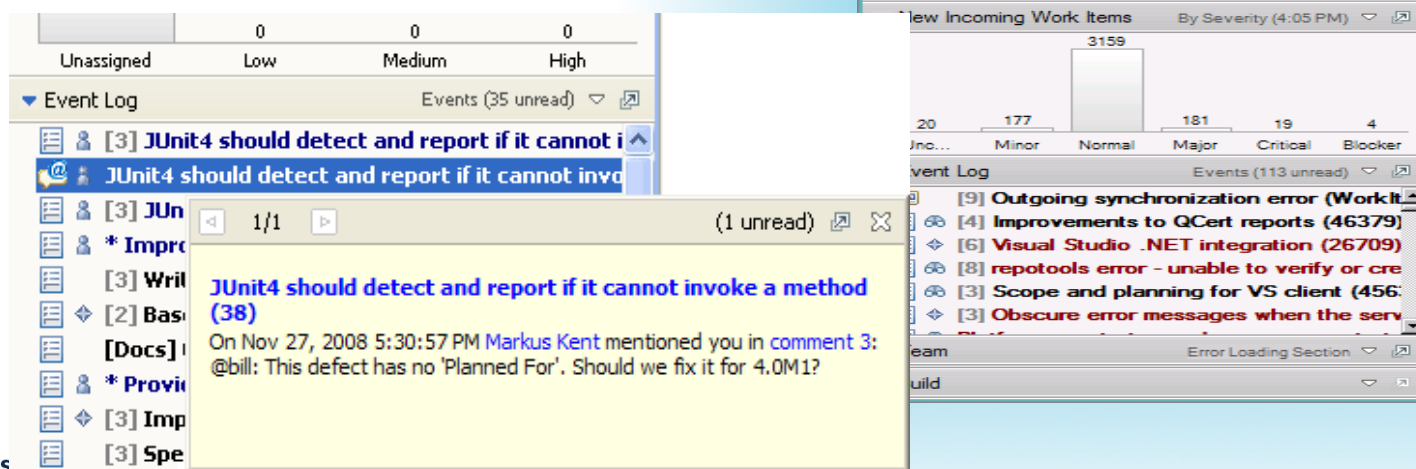
Foundation - Stories Open, Closed, in Progress

Show Parameters

63

0

- Open
- In Progress
- Closed

CCM - Stories Open, Closed, in Progress

Show Parameters

67

0

- Open
- In Progress
- Closed

RM - Stories Open, Closed, In Progress

Show Parameters

91

0

- Open
- In Progress
- Closed

In-context Collaboration – Team View

Team Central

- Shows what is happening on project:
  - News & events
  - What's being worked on
  - Changes
- Configurable (RSS feeds) - New kinds of information easily added
- Personalized, Persistent - Each team member can tailor to their needs

Source Control Management
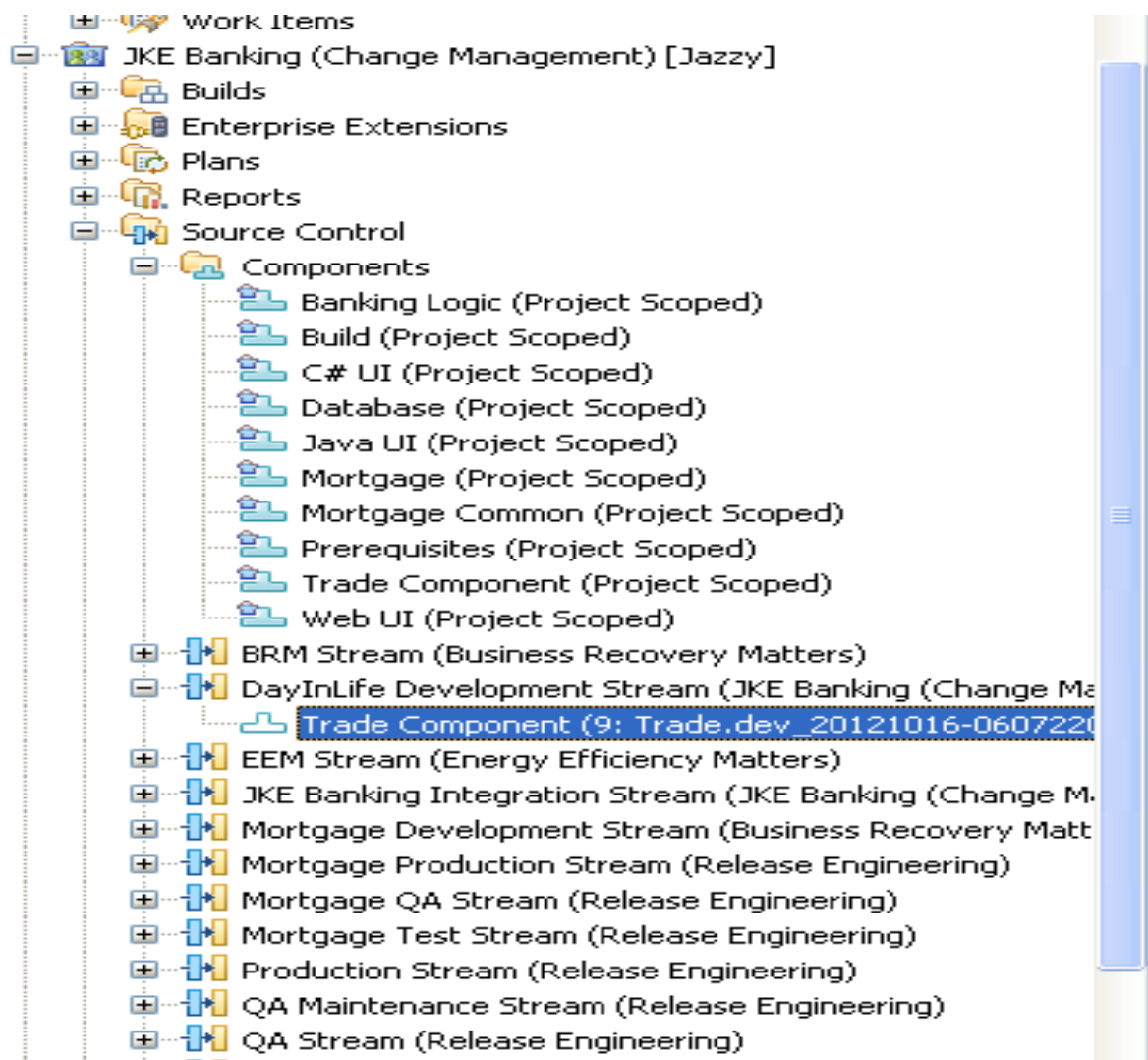
Load the source artifacts

RDz and RTC together

- Once the project is loaded, it will appear in the RDz z/OS projects view

- RDz augments the development productivity & experience
    - Appropriate editors (COBOL, maps, etc.) and functions (content assist, real time syntax check, etc.)
   - High value functions (Enterprise web services, SFM, Code review, Unit testing, program analysis/control flow etc.)
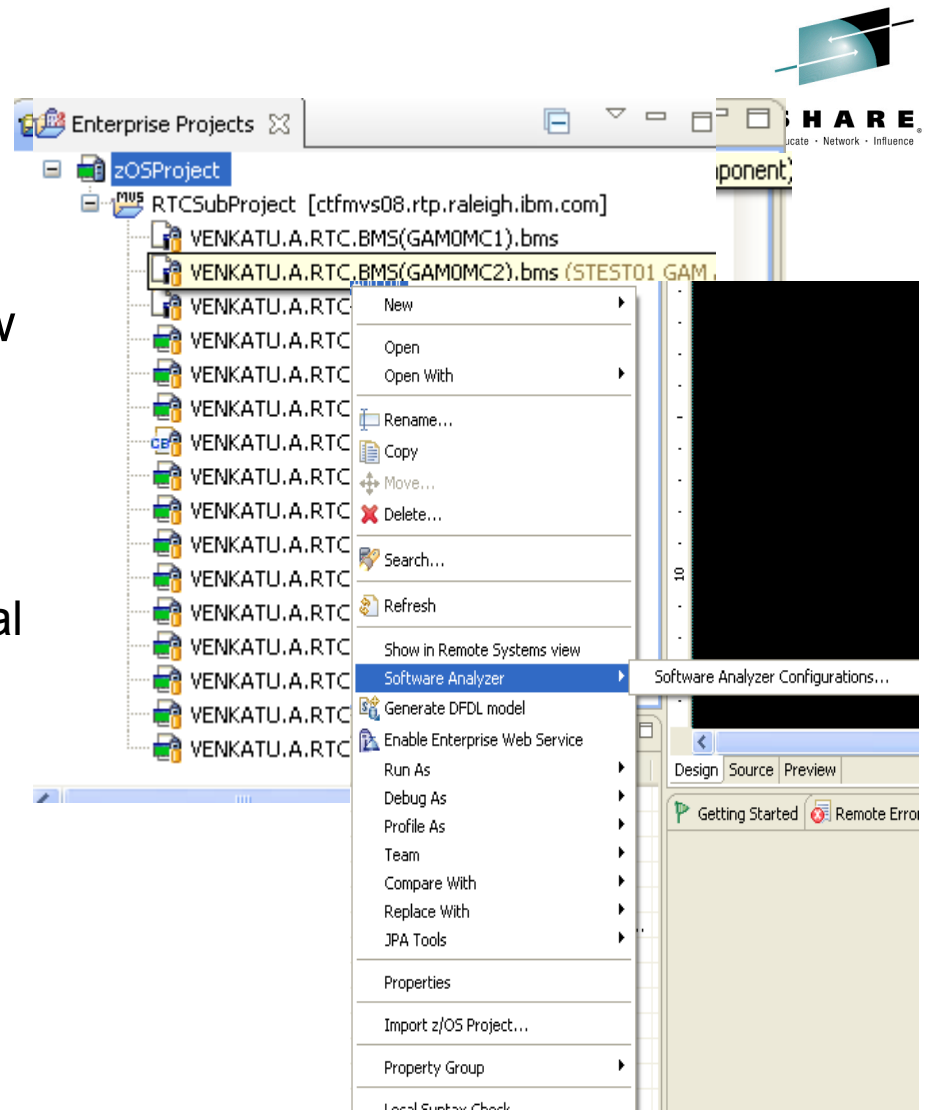
Navigate datasets and jobs live on zOS

- Connect to multiple hosts concurrently
- Respects existing security configurations and user IDs
- Search, filter, browse, edit, compare, migrate, and allocate new MVS datasets and USS files
- Copy source code, members, or datasets between systems with a few mouse clicks.
- Access JES queues submit jobs, view job state, and open output spools
- Submit TSO or USS commands
- Add datasets and members into projects to group applications and work items together logically
- Open an emulator in the IDE to configured hosts
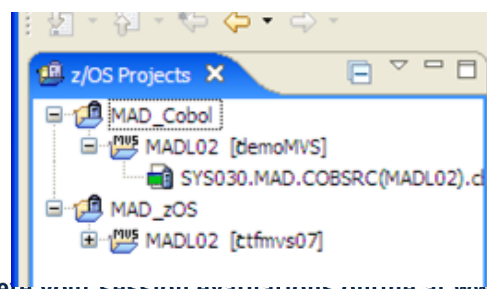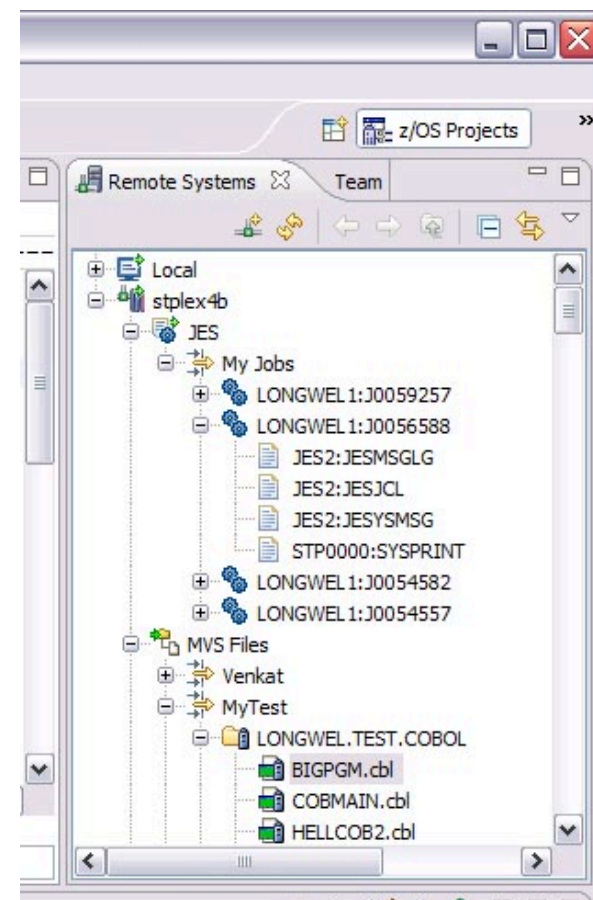
Edit capabilities in RDz

- RDz at a high level has different types of editors

  - LPEX Editor
    - Supports editing of COBOL, PLI, HLASM JCL, C/C++, Rexx etc.
    - Provides ISPF like edit experience including prefix commands, command line and even look and feel
    - Supports advanced edit functions for COBOL, PLI and HLASM like real time syntax checking, content assist

  - COBOL, PLI, and JCL advanced editors
    - Based on the Eclipse editor infrastructure, provide more advanced edit capabilities like quick fixes, hyper-linking, hover, easy navigation between various edit sessions or within the same edit session.
    - Supports real time syntax checking, content assist, key word highlighting etc.

```
Line 49          Column 1         Insert
======+--*A-1-B--+----2----+----3----+
000027          *
000028          ****************************
000029          IDENTIFICATION DIVISION.
000030          PROGRAM-ID. GAMOVDB.
000031
000032          DATA DIVISION.
000033
000034          WORKING-STORAGE SECTION.
000035
000036          COPY GAMOBMD.
000037          COPY GAMOBDD.
000038          COPY GAMOBED.
000039          COPY GAMOBCD.
000040          COPY GAMOBPD.
000041          01  DATABASECONNECTION.          P:
000042              49  SCHEMA
000043              49  PIC X VALUE '.'.
000044
000045          01  ITABLEPREPARE.
000046              49  ITABLE-VAR-LEN     PIC S
000047              49  ITABLE-VAR-TXT     PIC
000048
i2              01  INVENTORYPREP.
000050              02  PIC  X(12)  VALUE  'INSERT
000051              02  INVNTDBCON PIC X(9).
000052              02  TABLENAME PIC X(11).
```

```
PROCEDURE DIVISION.

    INITIALIZE INPUTS-OUTPUTS.
    INITIALIZE INVO.
    INITIALIZE LOCINVO.
    MOVE LOW-VALUE TO LOCINVO.
    MVE LOW-VALUE TO INVO.
    ↪ Change to "MOVE"
                                          EA
                                          TPUTS

                                          SE
```
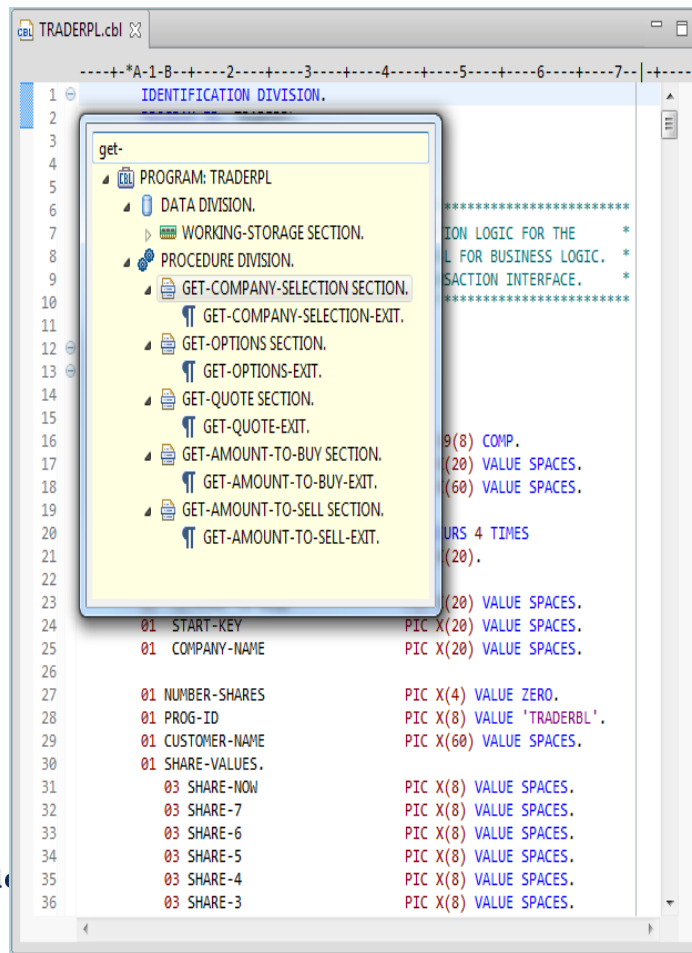
COBOL, PL/I and JCL Editor improvements
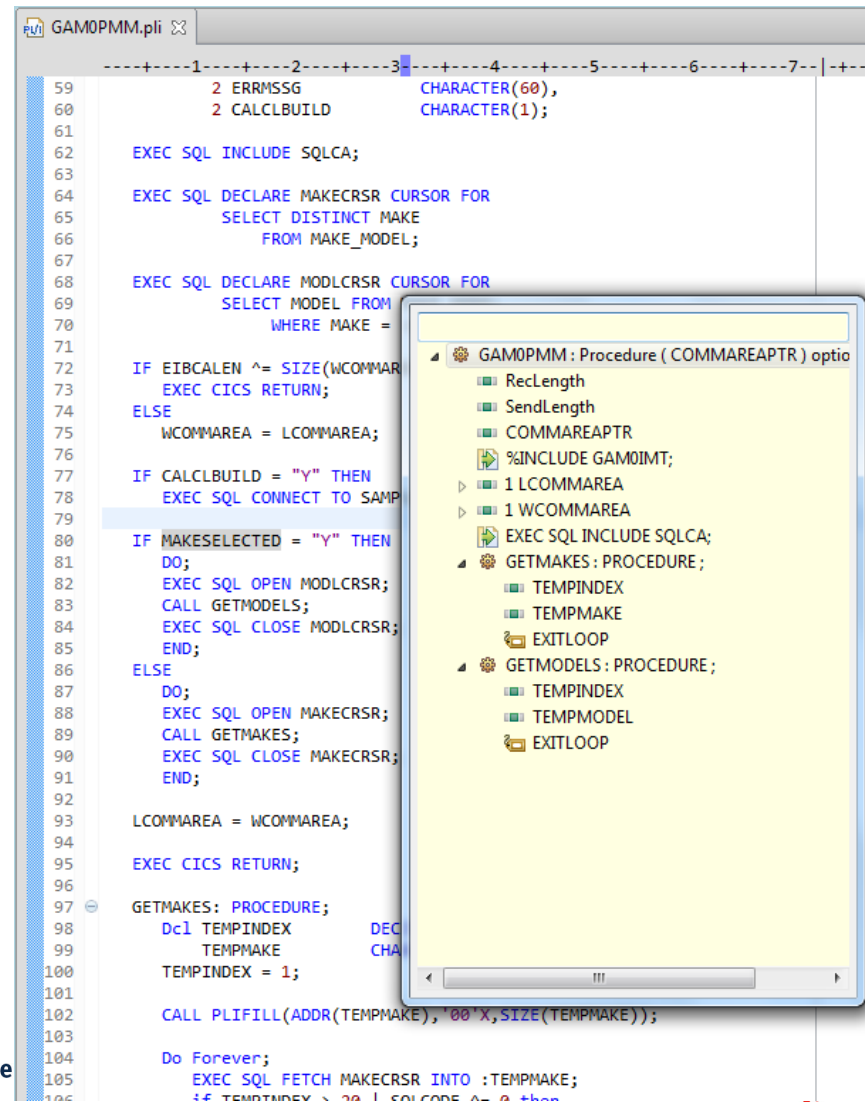
- **Quick Outline**

  - Press Ctrl+O to activate

Editor Productivity Features – real time syntax checking

Real-time syntax check without requiring code compile or save

JCL Editor Content Assist

- **Keyword syntax proposals**
  - Press Ctrl+SPACE to activate

Enhanced Application Quality & Structure Analysis

- Application Analysis
  - Control flow diagrams for COBOL and **PLI** programs,
    - *Graphical representation of the program flow with links to the source*
  - Helps identify and highlight potential unreachable code
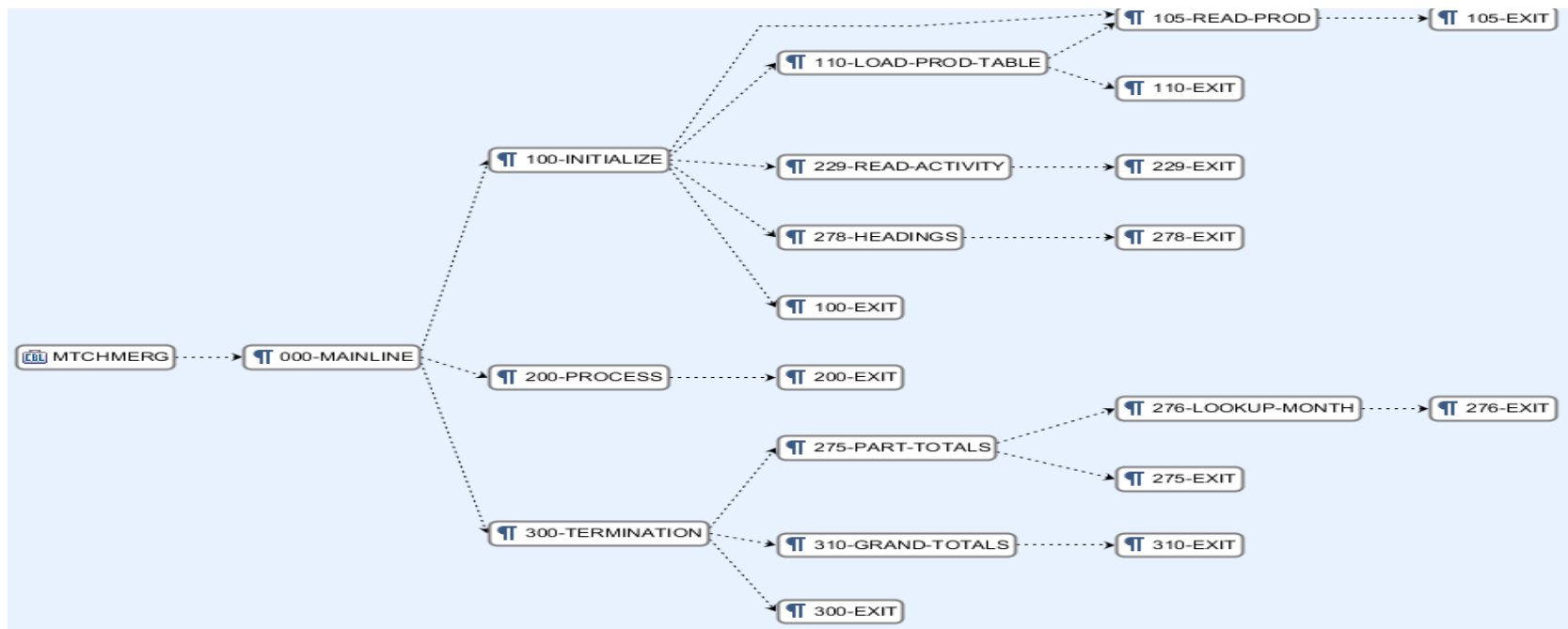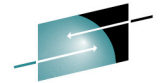
Enhanced Structure Analysis – Data Element Table

- A table representation of the user-defined data items and symbols in a program
  - Hyperlinks in the table are integrated with the editor allowing easy access to the declaration of the data items.

- Generated by showing the "symbol table" generated when RDz real-time syntax check parses the program

Data Element Table ✕

Showing data elements from WARDRPT.cbl   Search:

Data Item Name:	Data Type:	Level:	Top-Level Item:	Declaration:	Initial Value:	Line ...	Reference count:	Full Declaration:
PATLISTEST-S-ID	Data	10	PATIENT-MASTER-REC	PIC X(08)		378	0	10 PATLISTEST-S-ID PIC X
PATMSTR	File Descriptor	0	PATMSTR			116	4	FD PATMSTR RECORD CO.
PATMSTR-FOUND	Data	88	FILE-STATUS-CODES			134	1	88 PATMSTR-FOUND VALUE "0
PATMSTR-KEY	Data	5	PATMSTR	PIC X(06)		120	2	05 PATMSTR-KEY PIC X(06).
PATMSTR-REC	Data	1	PATMSTR			119	1	01 PATMSTR-REC.
PATMSTR-STATUS	Data	5	FILE-STATUS-CODES	PIC X(2)		133	3	05 PATMSTR-STATUS PIC X
PATPERSN	File Descriptor	0	PATPERSN			123	4	FD PATPERSN RECORD CO
PATPERSN-FOUND	Data	88	FILE-STATUS-CODES			136	1	88 PATPERSN-FOUND VALUE ".
PATPERSN-KEY	Data	5	PATPERSN	PIC X(06)		127	2	05 PATPERSN-KEY PIC X(06).
PATPERSN-REC	Data	1	PATPERSN			126	2	01 PATPERSN-REC.
PATPERSN-STATUS	Data	5	FILE-STATUS-CODES	PIC X(2)		135	3	05 PATPERSN-STATUS PIC
PATSRCH	File Descriptor	0	PATSRCH			98	4	FD PATSRCH RECORDING
PAYMENT-METHOD-TYPE	Data	5	PATIENT-PERSONAL-...	PIC X(02)		313	0	05 PAYMENT-METHOD-TYPE
PEDIATRICS	Data	88	INPATIENT-DAILY-REC			157	0	88 PEDIATRICS VALUE "1010".

User Build from RDz

"User build", is supported both in zComponent projects and RDz remote z/OS projects

- Builds just one the single file selected, supports Error feedback

- Generates JCL based on the associated RTC Language definitions and Translators
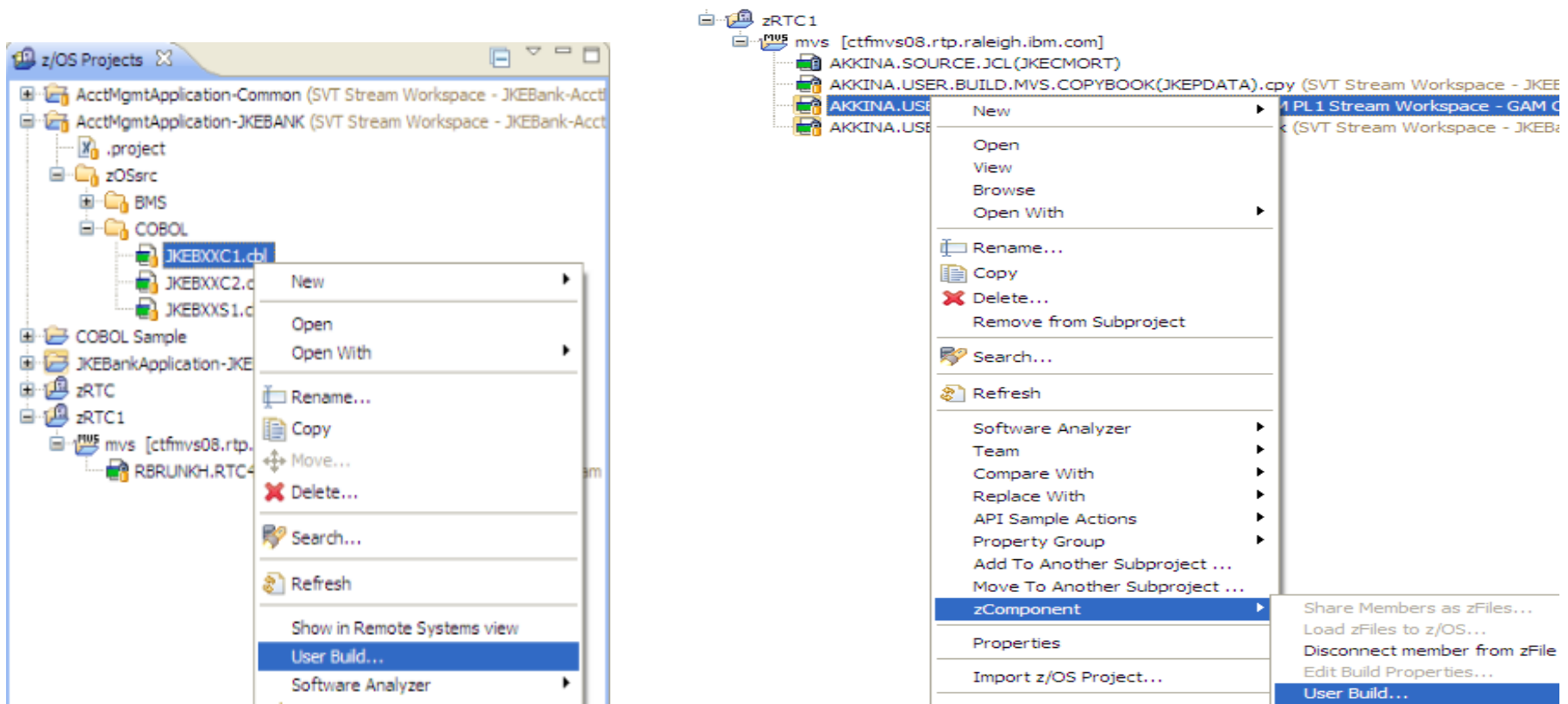
Pending Changes

- If you want finer grained control on your SCM operations, then the Pending Changes view is for you
  - Check in, deliver, accept changes
  - Suspend, resume, discard changes
  - Replace, reload out-of-sync
  - Resolve conflicts
  - Open change sets and work items via the web client
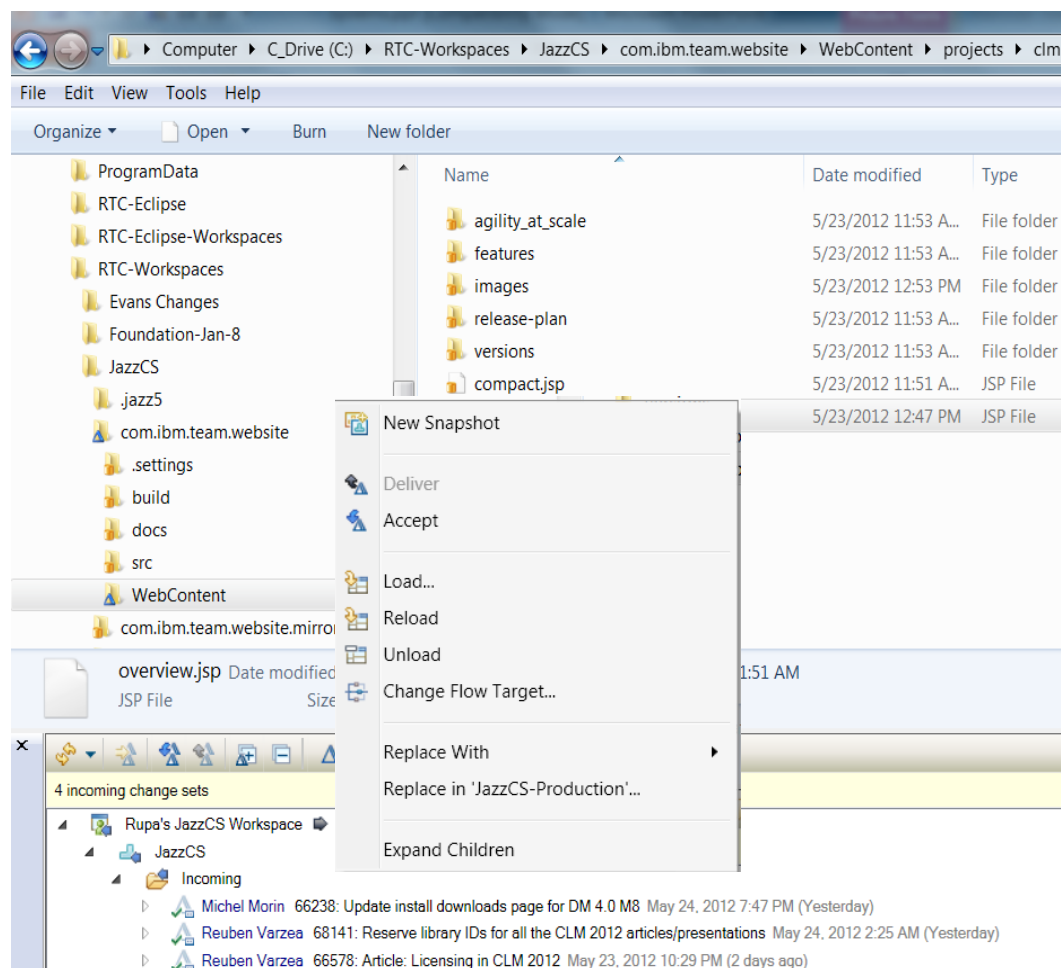
Traceability : Check-in History

- Someone made a costly mistake merging and you want to understand exactly where the mistake was made
  - Problem : Traditional history commands & UI only show before/ after & merge states for a change set … it does not show intermediates
  - Solution : Use Check-in history in Eclipse, CLI or .NET clients

Integrated Debugger

- **RDz introduced a new feature called Integrated Debugger**
  - ✓ **A GUI-based multi-platform, multi-language debugger**
    - ✓ Full asynchronous mode
      - ✓ Thread-level control of multi-threaded applications
    - ✓ Automonitor support
  - ✓ RDz v9.0.1 Supported**:**
    - ✓ COBOL V5.1, V4, V3.4
    - ✓ Batch, Batch IMS, Batch DB2, CICS 5.1, 4.2, 4.1
    - ✓ Interactive Code coverage – Out of the box
  - ✓ **RDz 9.1 added support for**:
    - ✓ PLI v4.x, v3.9
    - ✓ C/C++ V1R13, V2R1
    - ✓ IMS TM
    - ✓ DB2 Stored procedures

Integrated Debugger

✓ **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

  ✓ Debugger client is supported on Windows and Linux

✓ **Simple and Secure Connections:**
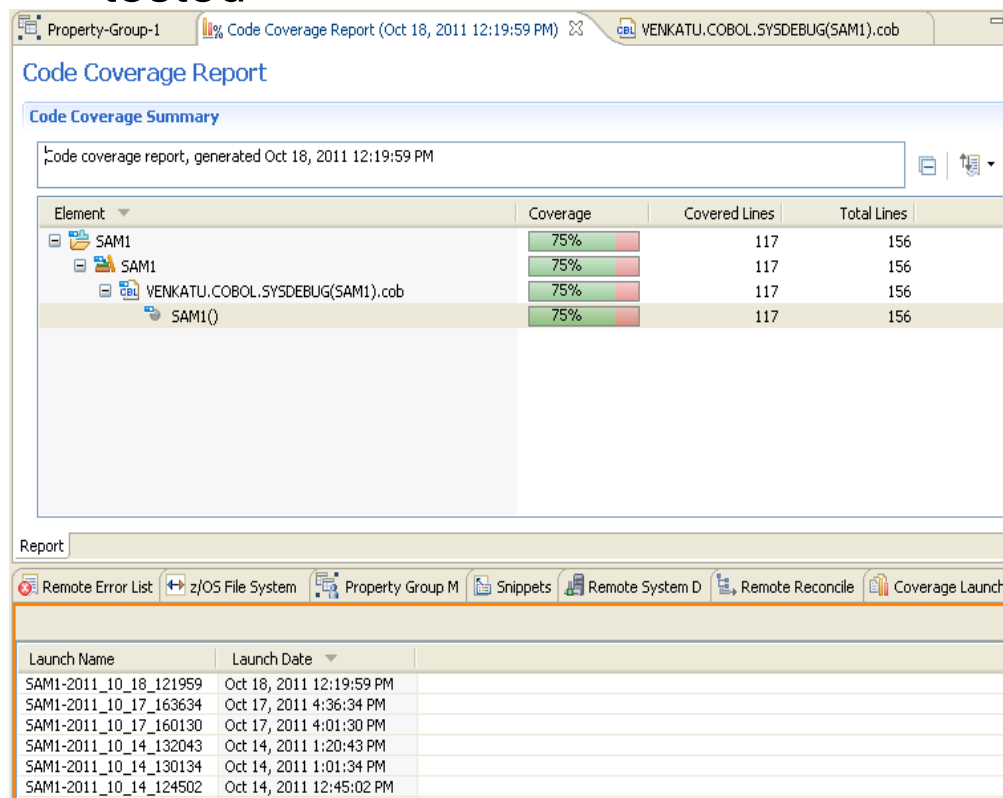
  ✓ Single client can handle multiple debug sessions on multiple hosts or an application the spans multiple systems

  ✓ Client initiated debug – no need to specify client IP address and port (v9.0.1.2)

  ✓ SSL/TLS support

Debug Multiple Runtimes

- Use the cross-platform debugger to debug end-to-end systems as they execute in the runtime
  - CICS
  - Batch
  - Java
- From the workstation:
  - View executing source code
  - Step through host code line-by-line
  - Set breakpoints
  - Alter working storage values
  - Alter register values
  - Etc…
- Debug zOS and distributed code in the same interface even stepping between runtimes and platforms!
- Leverage Integration with IBM Debug Tool for other runtimes

Enhanced Application Quality – Code Coverage

- Line Level Code Coverage - provides tools to measure and report on test coverage of an application
  - Leverages the Integrated Debugger technology
  - Indicating what source code lines were tested and remain to be tested

- Code Review/Governance - provides predefined rules and templates for COBOL and PL/I applications
  - Ensure adherence to corporate standards
  - Custom rules for COBOL and PL/I

Add Custom Rule

Select Template

Select a template to use for this rule

- An inline PERFORM statement cannot exceed the specified line number limit
- Avoid calling the specified routine
- Avoid including the specified copy book
- Avoid nesting IF statements deeper than the specified number of levels
- Avoid static calls except for the specified program name
- Avoid using the selected compiler directives
- EXEC SQL: Use a WHERE clause in selected statements
- Follow the specified naming convention for COBOL file names
- Procedure division statements cannot exceed the specified line number limit
- Use a scope terminator phrase with the specified COBOL statement type
- Use comments to describe all paragraphs
- Use comments to describe all sections
- Use the specified prefix with condition names

< Back   Next >   Finish   Cancel

COBOL Code Review [0/43]
- Naming Conventions [0/1]
  - Use a program name that matches the source file name
- Performance [0/9]
  - Avoid INITIALIZE statements. Use elementary MOVE statements or VALUE clauses.
  - Avoid OCCURS DEPENDING ON phrases
  - Avoid using subscripts to access a table. Use indexes.
  - **EXEC SQL: Avoid SELECT ***
  - **EXEC SQL: Use an ORDER BY clause when declaring a cursor**
  - **Specify 0 RECORDS for BLOCK CONTAINS clauses in file description entries**
  - Use an EVALUATE statement rather than a nested IF statement
  - **Use an odd number of digits in a COMP-3 or PACKED-DECIMAL data definition**
  - Use binary subscripts
- **Program Structures [0/33]**
  - Avoid ACCEPT statements
  - Avoid ACCEPT statements containing FROM CONSOLE or FROM SYSIN
  - Avoid ALTER statements
  - Avoid CALL statements with a literal program name
  - **Avoid CANCEL statements**
  - Avoid COPY SUPPRESS statements
  - Avoid CORRESPONDING phrases
  - **Avoid DISPLAY statements containing UPON CONSOLE**
  - Avoid ENTRY statements
  - Avoid EXIT PROGRAM statements
  - Avoid GO TO statements
  - Avoid GO TO statements, except those that reference an EXIT paragraph
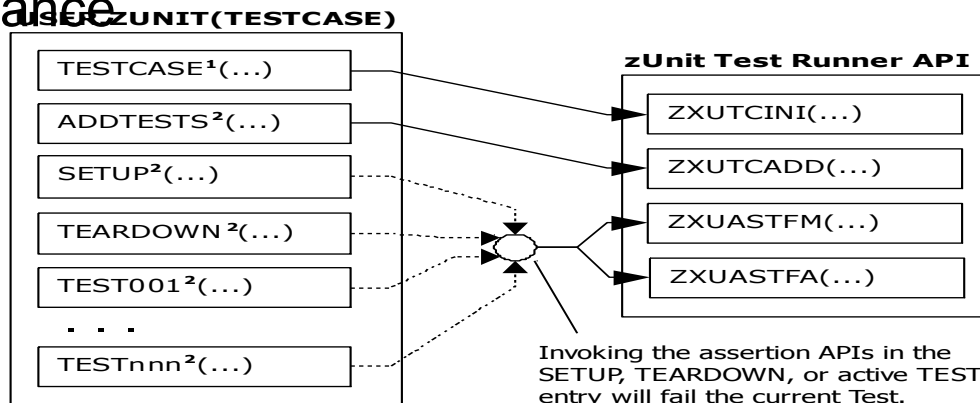 - **Avoid IF without ELSE**
  - Avoid NEXT SENTENCE phrases
  - **Avoid PERFORM, except PERFORM section**
  - Avoid RESERVE clauses in FILE-CONTROL paragraphs
  - Avoid STOP RUN and STOP literal statements
  - Avoid THRU phrases in PERFORM statements
  - **Avoid using level-88 entries in data descriptions**
  - **Avoid using more than one EXIT statement per section**
  - **Avoid using SECTION in the procedure division**
  - **Avoid XML PARSE statements**
  - **EXEC CICS: Check EIBRESP after NOHANDLE**
  - EXEC CICS: Use DFHRESP to check the return value
  - EXEC CICS: Use the RESP option
  - **EXEC SQL: Check the value of SQLCODE after an EXEC SQL statement**
  - **Use an EXIT paragraph in each section**
  - Use a WHEN OTHER phrase with an EVALUATE statement

zUnit – Unit testing framework for z/OS

- Frameworks that assist developers in writing code to perform repeatable, self-checking unit tests are collectively known as *x*Unit.

- xUnit defines a set of concepts that together provide a light-weight architecture for implementing unit testing frameworks.

  - JUnit, for example, is a very popular instance of the xUnit architecture.

- **zUnit** is a xUnit instance for System z

- Goal is to encourage the **continuous integration and continuous testing** methodology for System z Application development and maintenance

USER-ZUNIT(TESTCASE)

TESTCASE¹(...)
ADDTESTS²(...)
SETUP²(...)
TEARDOWN ²(...)
TEST001²(...)
. . .
TESTnnn²(...)

zUnit Test Runner API

ZXUTCINI(...)
ZXUTCADD(...)
ZXUASTFM(...)
ZXUASTFA(...)

Invoking the assertion APIs in the SETUP, TEARDOWN, or active TEST entry will fail the current Test.

¹Language-specific details:
- In COBOL, this is the first program appearing in the Test Case source file and it will be invoked by the Test Runner for Test Case initialization.
- In PL/I, the is the procedure declared with option(fetchable) in the Test Case source file and it will be invoked by the Test Runner for Test Case initialization.

²Language-specific details:
- In COBOL, these are expected to be subprograms (non-nested and therefore compatible with FUNCTION-POINTER).
- In PL/I, these are expected to be internal procedures that are declared at the package level (non-nested).

Snapshots for every build

Summary

- Many companies spend more than 70% on keeping lights on, and that amount is increasing

- IT organizations have problems modifying applications at speed of business

- IBM provides a structured approach to incrementally modernize your portfolio based on business priorities

- Change without a plan is chaos

- A Plan without change is stagnation

- Business goals change

  - applications need to change to address them

- Continual renewal is required

  - tools help to guide, govern, drive, and accomplish this change

Getting started
Next steps to modernize your enterprise applications

www.ibm.com/rational/modernization

Empower

Unify

Revitalize

Enterprise Modernization

Optimize

Enabling Product and Service Innovation

jazz

➢ **Try latest System z software for free**

➢ **Sign up for free web-based training**

➢ **Join IBM Rational Cafe Communities**

➢ **Get prescriptive service solutions**

➢ **Success stories**

➢ **Latest news on System z twitter**

➢ **Latest customer videos**

➢ **Latest skills: System z job board**

Complete your session evaluations online at www.SHARE.org/Seattle-Eval
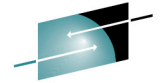
धन्यवाद
Hindi

多謝
Traditional Chinese

Grazie
Italian

ขอบคุณ
Thai

Gracias
Spanish

Merci
French

Спасибо
Russian

شكراً
Arabic

Obrigado
Brazilian Portuguese

Danke
German

多谢
Simplified Chinese

நன்றி
Tamil

ありがとうございました
Japanese

감사합니다
Korean

Notices and Disclaimers

Notices and Disclaimers (con't)

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products. IBM does not warrant the quality of any third-party products, or the ability of any such third-party products to interoperate with IBM's products. IBM EXPRESSLY DISCLAIMS ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents, copyrights, trademarks or other intellectual property right.

- IBM, the IBM logo, ibm.com, Bluemix, Blueworks Live, CICS, Clearcase, DOORS®, Enterprise Document Management System™, Global Business Services ®, Global Technology Services ®, Information on Demand, ILOG, Maximo®, MQIntegrator®, MQSeries®, Netcool®, OMEGAMON, OpenPower, PureAnalytics™, PureApplication®, pureCluster™, PureCoverage®, PureData®, PureExperience®, PureFlex®, pureQuery®, pureScale®, PureSystems®, QRadar®, Rational®, Rhapsody®, SoDA, SPSS, StoredIQ, Tivoli®, Trusteer®, urban{code}®, Watson, WebSphere®, Worklight®, X-Force® and System z® z/OS, are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at: www.ibm.com/legal/copytrade.shtml.