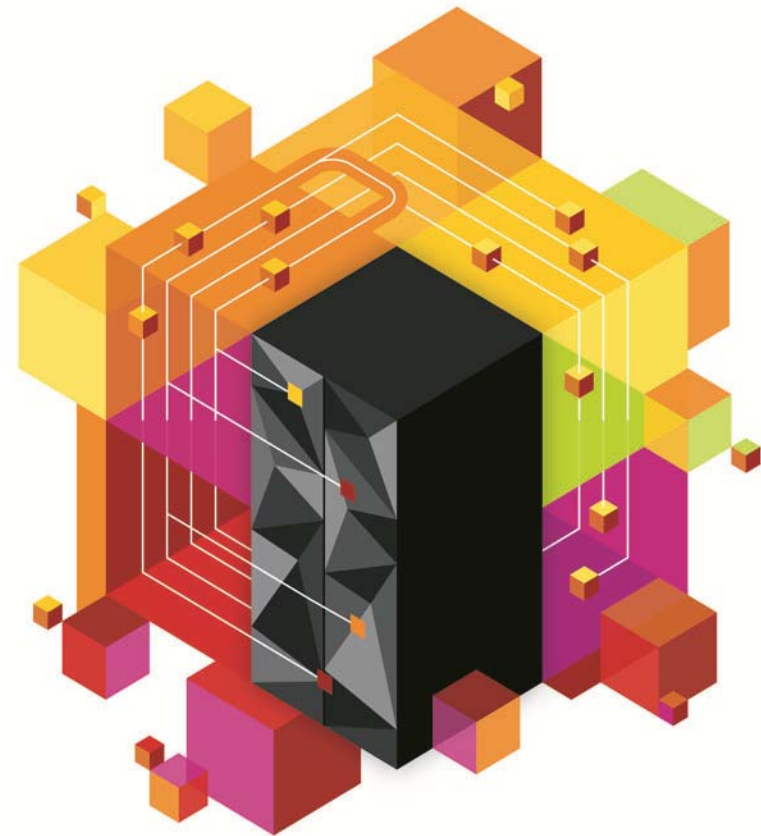# IBM zEnterprise Technology Summit

Presenter – Title

Date

Please Note:

•IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal 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
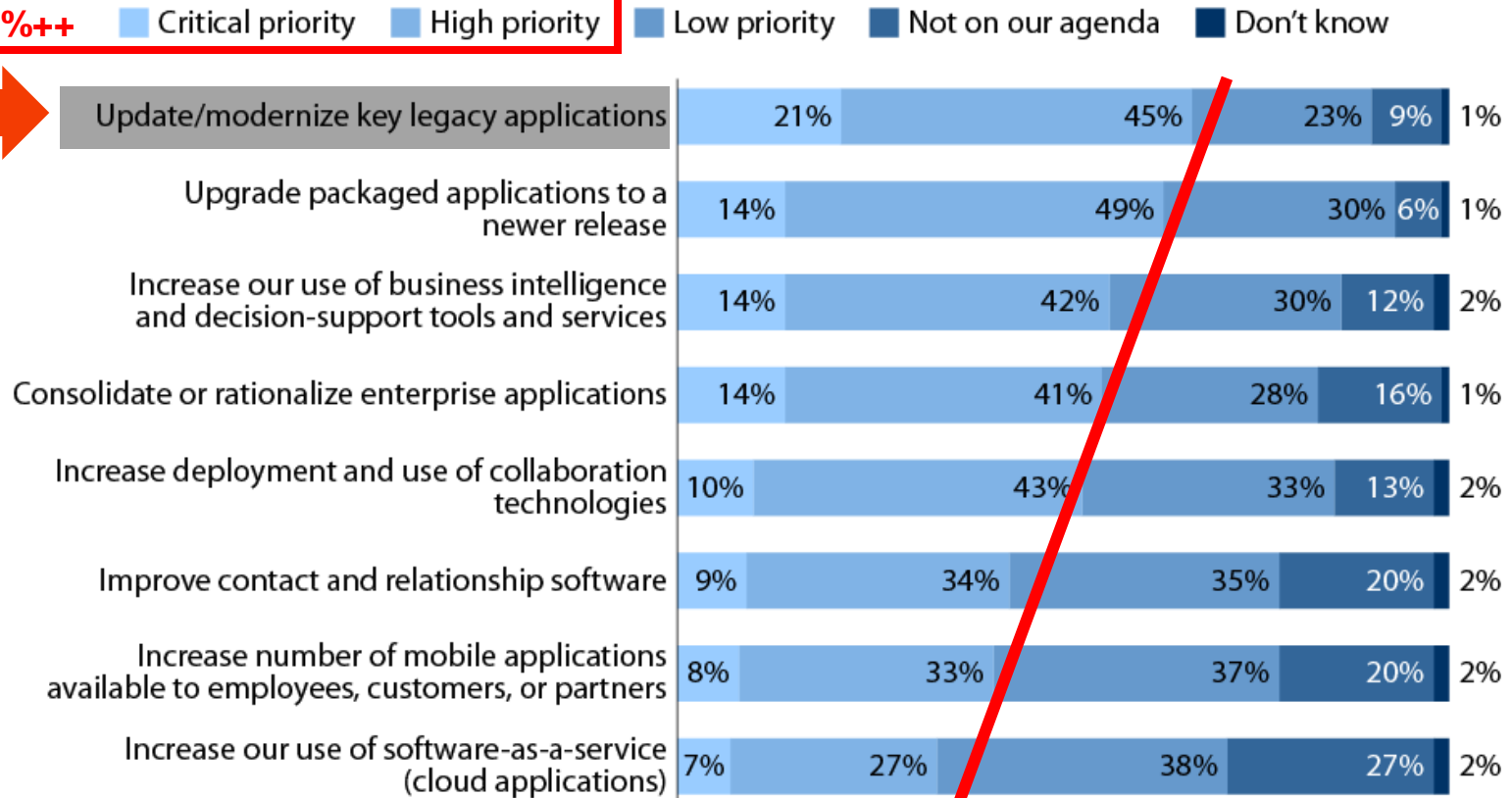
7 - Optimizing Tools and Compilers

- Challenges IT execs are facing today

- Rational Enterprise Modernization Solution Overview

- Function highlights and Strategic directions of COBOL, PL/I and C/C++ compilers

- Modern System z development tools

- Q & A

Applications are causing BIG headaches for execs

"Which of the following initiatives are likely to be your IT organization's top software priorities over the next 12 months?"

50%++ | Critical priority | High priority | Low priority | Not on our agenda | Don't know

Initiative	Critical priority	High priority	Low priority	Not on our agenda	Don't know
Update/modernize key legacy applications	21%	45%	23%	9%	1%
Upgrade packaged applications to a newer release	14%	49%	30%	6%	1%
Increase our use of business intelligence and decision-support tools and services	14%	42%	30%	12%	2%
Consolidate or rationalize enterprise applications	14%	41%	28%	16%	1%
Increase deployment and use of collaboration technologies	10%	43%	33%	13%	2%
Improve contact and relationship software	9%	34%	35%	20%	2%
Increase number of mobile applications available to employees, customers, or partners	8%	33%	37%	20%	2%
Increase our use of software-as-a-service (cloud applications)	7%	27%	38%	27%	2%

Base: 2,124 software decision makers
(percents may not total 100 because of rounding)
Source: Forrsights Software Survey, Q4 2010

Forrester Inc. February 2011 "Application Retirement — It's Time To Put The Elephant In The Room On A Diet"

IBM Enterprise Modernization solutions

Achieving greater value and performance from your IT investments

Revitalize Applications

- ✓ **Cut maintenance costs and project risks by understanding and managing application portfolios**
- ✓ **Increase flexibility by modernizing and reusing existing application assets**

Empower People

- ✓ **Boost developer productivity with a common IDE and modern business language for multiplatform applications**

Unify teams

- ✓ **Increase organization agility by unifying teams with a collaborative team infrastructure for all platforms**

Optimize Infrastructure

- ✓ **Free up development MIPS for production use by offloading development and test activities**
- ✓ **Reduce capital expenses by maximizing application performance on existing hardware**

Empower

Unify

Enterprise Modernization

Optimized Systems

Revitalize

Optimize

jazz
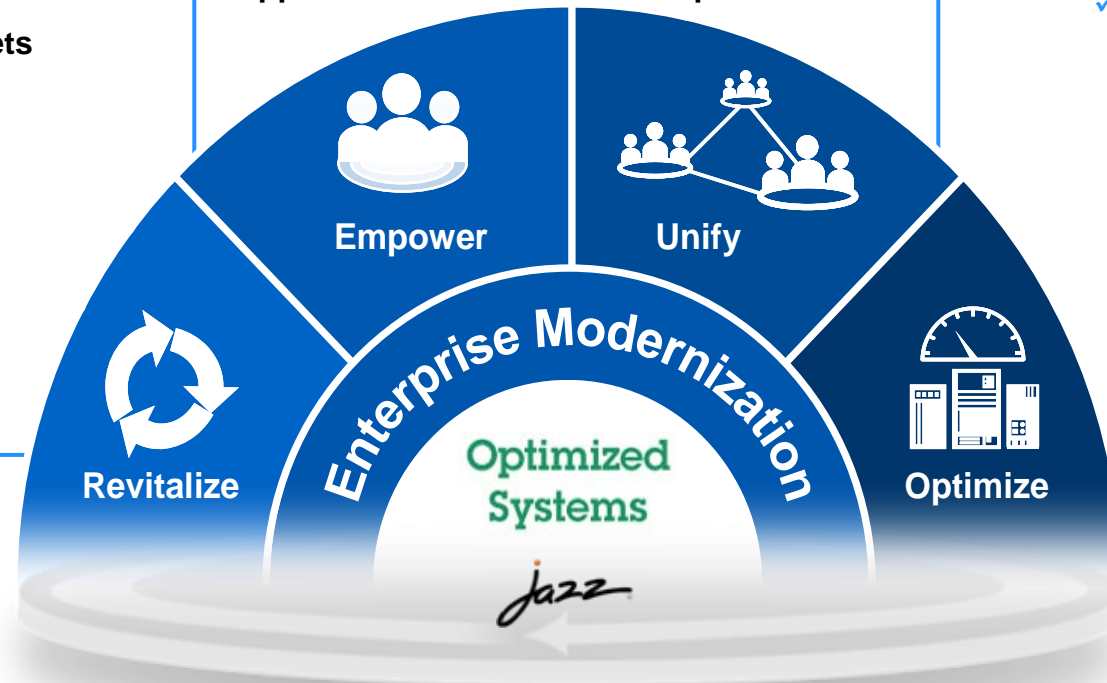
Strategies to reduce cost, boost performance and productivity
Modernize the your System z environment

Upgrade Hardware Infrastructure

- Workloads
  - z/OS
  - zLinux
  - AIX
  - Linux
  - Windows

Upgrade to Latest Compilers and Middleware

- Upgrade
  - COBOL
  - PL/I
  - C/C++
  - CICS
  - DB2
  - IMS

Performance

Compiler & Middleware Version

Leverage Modern Mainframe Development Tools

- Modernize
  - Design
  - Develop
  - Test
  - Deploy
  - Optimize
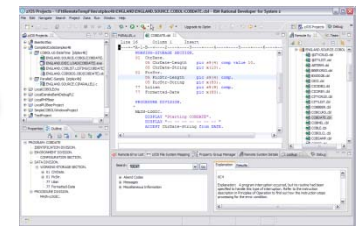
Rational Developer for System z

Compiler Business Values

- Increase return on investment
  - Maximize application performance on System z
    - *Exploit z/Architecture and middleware*
    - *Leverage advanced optimization technology*
    - *Reduce total cost of ownership*

- Improve programmer productivity
  - Simplify programming
  - Improve usability
  - Reduce risk, cost, and development time

- Protect investment in business critical applications
  - Modernize business critical applications
    - *Reduce risk, reduce cost*
  - Maintain release-to-release compatibility
  - Support Industry programming language standards and extensions

Compilers on System z

- Enterprise COBOL for z/OS

- Enterprise PL/I for z/OS

- z/OS XL C/C++

COBOL, PL/I and C/C++ are strategic to System z platform

IBM z/OS XL C/C++

- Optionally priced feature of z/OS
  - Enables development of high performing business applications, system programs and low level C applications
- IBM has been delivering highly optimizing C/C++ compilers on z/OS for over 20 years
  - Every release sets new standard for performance
  - Includes advanced optimization technology originally designed for HPC applications, and innovations to improve programmer productivity
- Provides system programming capabilities with Metal C option
  - Allows developers to use C syntax to develop system programs and low level free standing applications on z/OS without coding in HLASM
    - Significantly shortens the learning curve
    - Leverage advanced optimization technology to generate high performance optimized code

What in z/OS v1.13 XL C/C++?

- Ships with z/OS v1.13

- Improved exploitation of zEnterprise 196 processor
  - Improve support for new instructions with new New ARCH(9) functions

- Improved application performance[1]
  - 4% over v1.12 for a compute intensive integer benchmark suite
  - 7% over v1.12 for a compute intensive floating point benchmark suite

- Added language support to enable straightforward porting of C/C++ applications to z/OS

- "Metal C" functional and performance enhancements
  - Enabled advanced optimization with IPA and HOT options

- Improved debugging and programmability support

[1]Results are based on a compute-intensive integer and floating point benchmark suites compiled with z/OS C/C++ V1R13 executing on a System zEnterprise 196 server. Performance gains from other applications may vary

C/C++ Historical Performance Improvement
CPU Intensive Integer computation
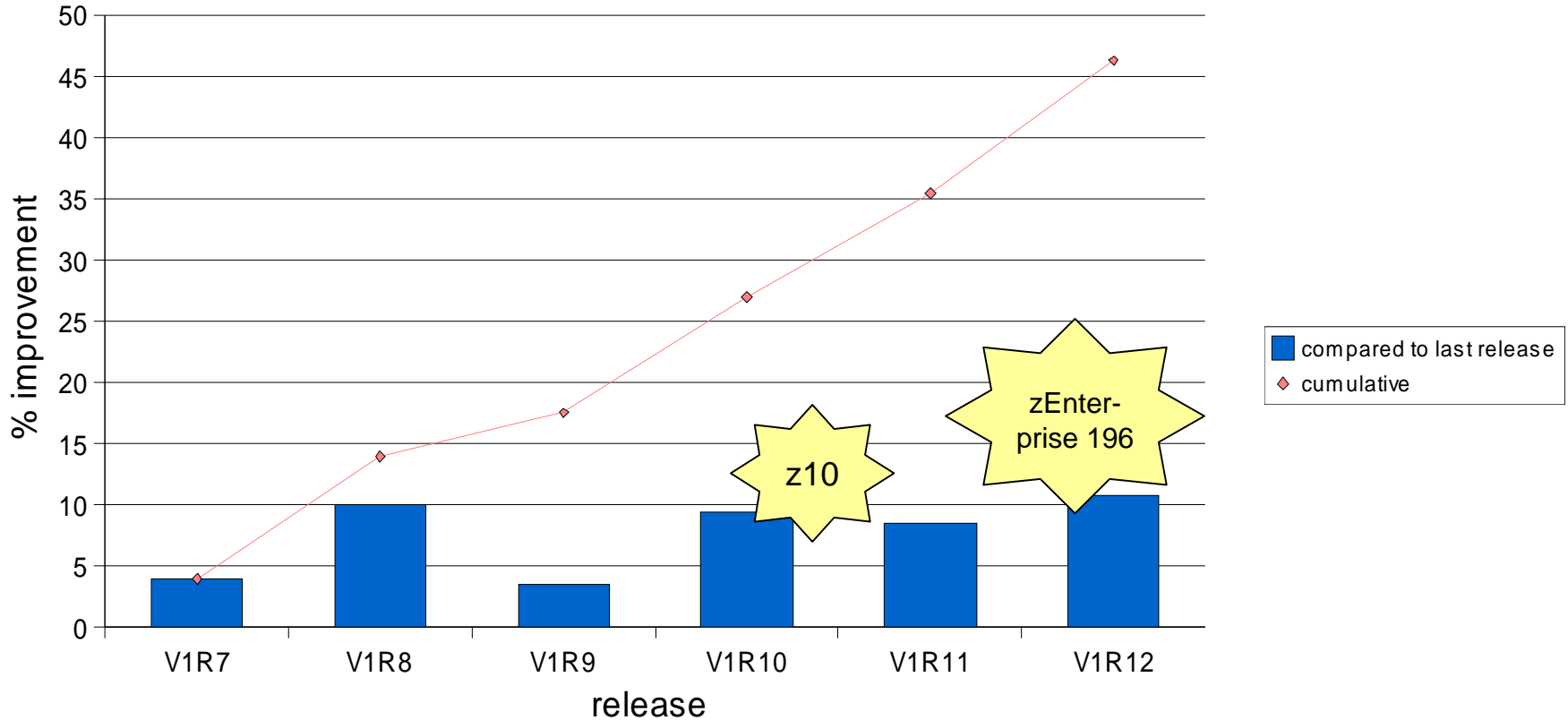
Results are based on a compute-intensive integer benchmark suite compiled with z/OS XL C/C++ compiler. Performance gains from other applications may vary

z/OS v1.13 XL C/C++ support for zEC12

- z/OS XL C/C++ will align with z/OS schedule
  - Follow a new two-year release cycle announced in the April 2012 z/OS SOD

- z/OS XL C/C++ V1.13 will provide initial support for zEC12 in the Sept. 2012 PTF

C/C++ compiler	UK80670 and UK80671
C/C++ readme	UK80039
C++RT builtins.h	UK79899

  - New "Arch(10)" and "Tune(10)" options
  - Enable developers to exploit new Transactional Execution Facility via built-in functions

  - PTF Web site: http://www-01.ibm.com/support/docview.wss?uid=swg21108506

- Performance Improvements of up 23% for CPU intensive applications on zEC12
  - From new hardware, No recompilation required

Hardware Transactional Memory (HTM)

- *Allow lockless interlocked execution of a block of code called a 'transaction'*

  - **Transaction:** *Segment of code that appears to execute 'atomically' to other CPUs*
    - Other processors in the system will either see **all-or-none** of the storage up-dates of transaction

- *How it works:*

  - TBEGIN instruction starts speculative execution of 'transaction'
  - Storage conflict is detected by hardware if another CPU writes to storage used by the transaction
  - Conflict triggers hardware to roll-back state (storage and registers)
    - transaction can be re-tried, or
    - a fall-back software path that performs locking can be used to guarantee forward progress
  - Changes made by transaction become visible to other CPUs after TEND

CPU 0: Tran A

```
TBEGIN

    …

load Y

load X

    …

    TEND
```
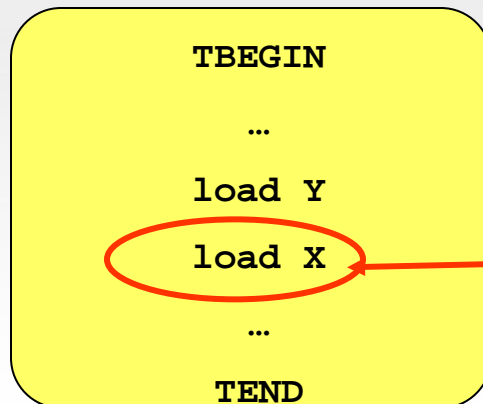
Storage conflict:
Tran A will abort
Tran B will commit
changes to X and Y

CPU 1: Tran B

```
X = Y = 0;

    TBEGIN

    X = 1

store X

    Y = 1

store Y

    TEND
```
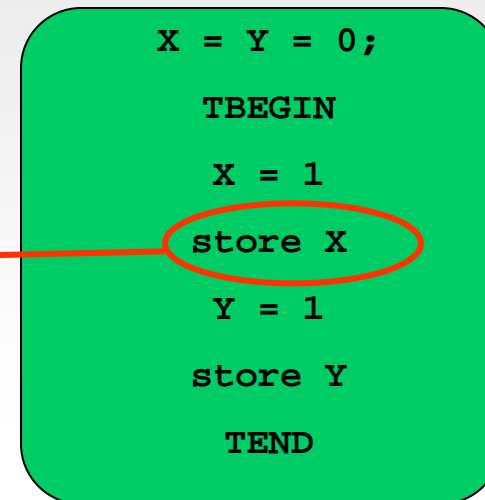
CPU 0 can only see $(X=Y=0)$ or $(X=Y=1)$,

cannot see $(X=1, Y=0)$ or $(X=0, Y=1)$

HTM with IBM XL C/C++

Complete set of built-ins

Enable exploitation of new
Transactional Execution Facility.

Provides function to:

- Start/End/Abort transactions
- Diagnose transaction failures
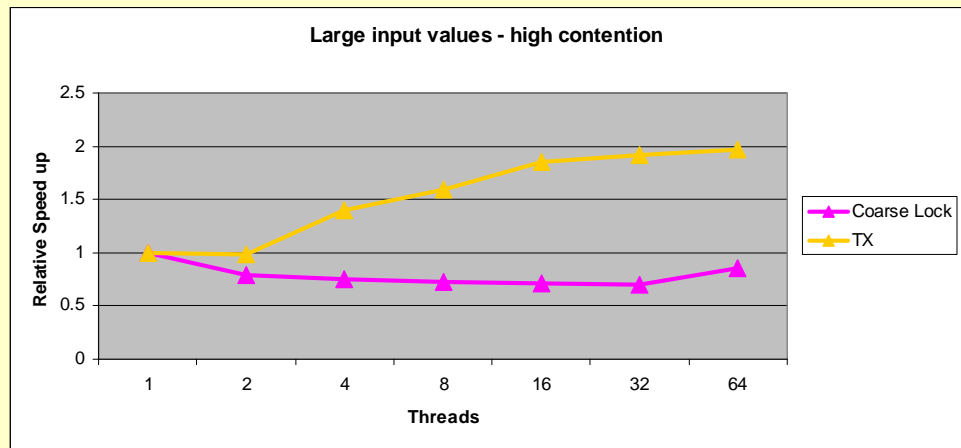- Detect transaction state (e.g. depth)

```
CASE ACTION_UPDATE_TABLES:
//..SOME CODE
  TM_BEGIN(MYID);

  …

  TM_END();
BREAK;
```

tm_begin(MYID) –start a transaction

THREAD_MUTEX_LOCK –acquire a lock

tm_end(MYID) –end a transaction

THREAD_MUTEX_UNLOCK –release a lock
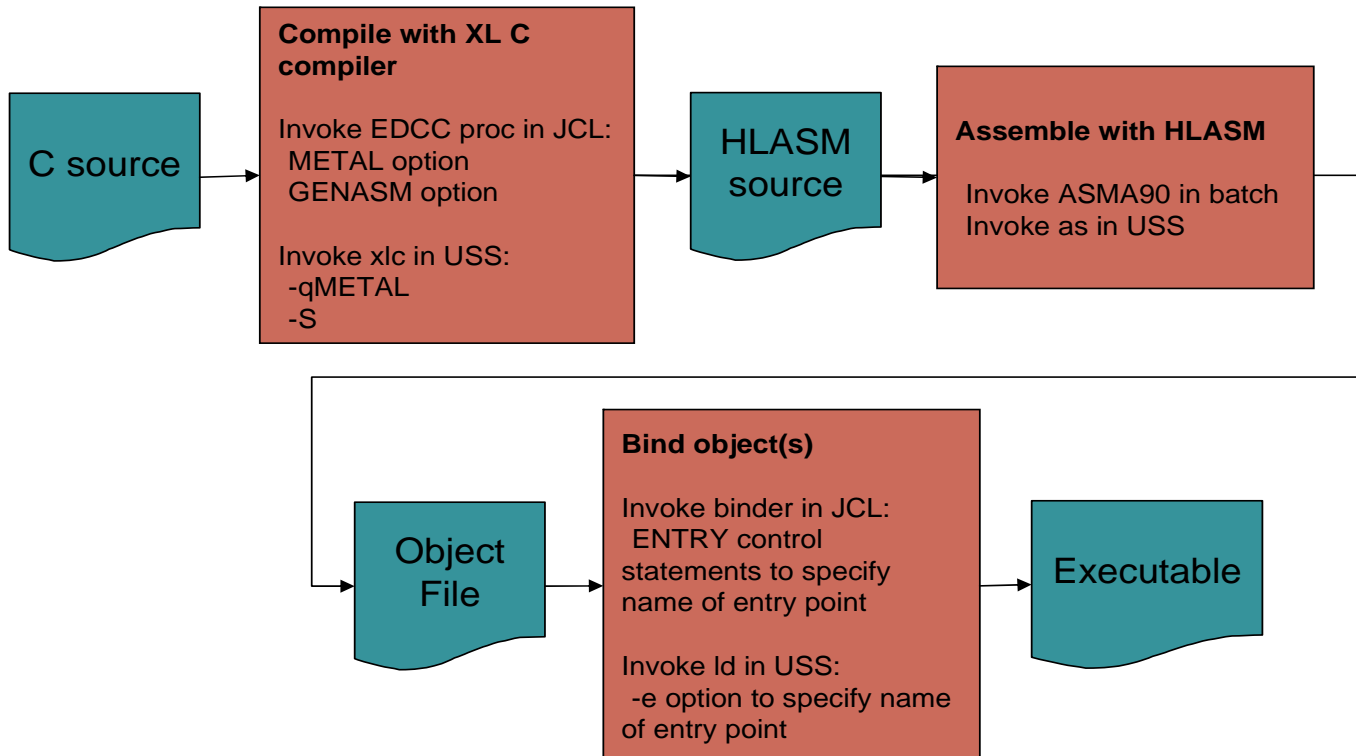
Sample Application**

An on-line transaction processing system that accesses a database concurrently. The work-load manages four tables representing car, hotel, flight and customers.

Large input values - high contention

Chart: X-axis "Threads" (1, 2, 4, 8, 16, 32, 64), Y-axis "Relative Speed up" (0 to 2.5). Series: Coarse Lock, TX

STAMP** http://csl.stanford.edu/~christos/publications/2008.stamp.iiswc.pdf

Metal C

- A new mode of code generation in the z/OS XL C/C++ compiler.
- Generates Language Environment independent code.
- Adopts the traditional linkage conventions for interoperating with existing assembler programs.
- Offers programmers the ability to embed their own assembler statements.
- Generates HLASM source code.
- Supplies a subset of C library functions.

C source →

Compile with XL C compiler

Invoke EDCC proc in JCL:
 METAL option
 GENASM option

Invoke xlc in USS:
 -qMETAL
 -S

→ **HLASM source** →

Assemble with HLASM

 Invoke ASMA90 in batch
 Invoke as in USS

→ **Object File** →

Bind object(s)

Invoke binder in JCL:
 ENTRY control statements to specify name of entry point

Invoke ld in USS:
 -e option to specify name of entry point

→ **Executable**

Rocket Software, Inc. increases development efficiency on the IBM System z platform
The Metal C feature makes it easier to leverage its C programming skills

The need:

Rocket Software, Inc. wanted to increase efficiency and improve time to market for its IBM® System z® operating system-based software products.

The solution:

Rocket Software used the Metal C feature of the IBM z/OS® XL C/C++ compiler to develop high performance system level programs. With the Metal C feature, programmers can write code in the C syntax while taking advantage of advanced optimization technology in the z/OS XL C/C++ compiler, resulting in high-performance code that works seamlessly with code written in IBM High-Level Assembler language (HLASM).

The benefits:

- Significantly increased development efficiency

- Reduced development time by half

- Enabled the company to leverage C programming skills

> *"Metal C in z/OS XL C/C++ is yet another powerful tool helping turn the economics of System z software development into a complete equation."*
> *—Joseph Devlin, managing director, R&D, Rocket Software*

Solution components:

- **IBM® z/OS® XL C/C++**
  - **IBM® System z®**

Link to case study:
https://www-01.ibm.com/software/success/cssdb.nsf/cs/CPAR-8WYKCC?OpenDocument&Site=swzseries&cty=en_us

Enterprise PL/I

- Strategic Programming Language
  - Significant use in business applications but also in some scientific and engineering applications
  - Introduced new version (v4) in 2010
- Advanced optimization technology
  - Shares optimizing back-end technology with z/OS XL C/C++
    - Enables timely delivery of leading edge optimization and hardware exploitation to PL/I customers
- Time proven
  - First Enterprise PL/I product released in 2001 (Enterprise PL/I for z/OS and OS/390 v3.1)
  - Latest release of Enterprise PL/I for z/OS (v4.3) is based on same architecture
    - Provides easy migration
- Shipped new release every year since 1999
  - Improved optimization technology, z/Architecture exploitation, usability, middleware support, and application modernization features.
  - Addressed customer requirements

What's new in Enterprise PL/I V4.3?

- **Improved performance**
  - ✓ zEC12 exploitation with new ARCH(10) option
    - ▪ Exploits new Decimal-Floating-Point Zoned-Conversion Facility
      - • Up to 40% faster for PICTURE to FIXED BIN conversions
      - • Up to 4X faster for PICTURE to FLOAT DEC conversions
    - ▪ CPU-Intensive PL/I benchmarks running on zEC12 see an improvement of up to 31% over zE196

- **Enhanced middleware support**
  - ✓ Support latest middleware CICS, DB2 and IMS
  - ✓ SQL improvements
    - ▪ Support for ONEPASS option
    - ▪ Improved display of EXEC SQL statements in listings
    - ▪ Usability improvements

- **Increased productivity (Requirements from users)**
  - ✓ New built-in functions to improve UTF-8 support
  - ✓ New Assert statement
 - ✓ New options for compiler messages and enforcing coding rules.

Decimal-Floating-Point Zoned-Conversion Facility

Before zEC12

- There are no instructions that perform arithmetic on zoned decimal

- Instructions are available to convert from zoned to packed decimal and packed back to zoned decimal

- Packed decimal data cannot be held in register - Hinders optimization

New Decimal-Floating-Point-Zoned Conversion facility in zEC12

- Provides new set of instructions for converting between decimal-floating-point (DFP) and zoned decimal data types

- H/W Instructions are available to perform arithmetic for DFP

- DFP data can be held in registers – Helps optimization

- PL/I V4.3 enables applications that uses zoned decimal data to take advantage of DFP unit in zEC12 without any source changes
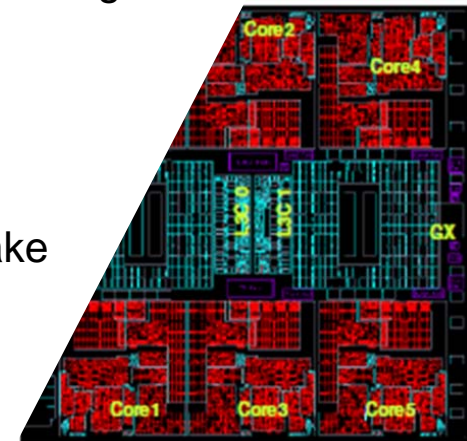
SQL Improvements in PL/I V4.2

- PL/I V4.2 shipped a smaller, faster and more powerful SQL preprocessor
  - ✓ Up to 8X smaller and 40% faster in processing SQL source
  - ✓ Improved support for multi-row fetch

COBOL

- COBOL celebrated its 50th birthday in September 2009.

- COBOL is still a dominant programming language for processing critical business transactions around the world.

- COBOL programs are simple, readable and very maintainable.

- Today, most business transactions are still processed with COBOL on IBM System z servers.
  – There are 200 times more COBOL transactions per day than Google Searches worldwide[1]

- COBOL is strategic
  – IBM is investing in improving the underlying technology to bring more value to customers.

[1] eWeek.com: 20 Things You Might Not Know About COBOL

Enterprise COBOL for z/OS v4.2
GA Sept. 2009

- Validated on zEnterprise 196 server with IBM's latest middleware
  - Support latest Middleware: CICS, IMS, and DB2

- Provides significant improvements to UNICODE performance

- Enables the integration of existing applications with web applications
  - Supports Java interoperability by object-oriented COBOL syntax
  - Supports access to enterprise beans that run on WebSphere Application Server or J2EE-compliant EJB server
  - Supports Java 5, Java 6 SDK

- Built-in language support for high speed parsing and validating of XML documents:
  - Offloading of XML parsing to zAAP specialty processors
  - Encoding in UTF-8, UTF-16, and various EBCDIC codepages

- Improved Debug Tool support for dynamically debugging optimized production programs

Enterprise COBOL VNext

Our intent:

■ Incorporate leading-edge optimization and code-generation technology to Enterprise COBOL for z/OS

- *Improve performance, and maximize machine utilization*

- *Reduce cost of ownership*

- *Initiate release to release performance improvements seen in C/C++, Enterprise PL/I and JAVA*

■ Improve delivery of z/Architecture exploitation

- *Provide solid foundation to support new hardware features (e.g. decimal floating point, 64 bit…), and future System z processors*

■ Continue to improve capabilities for modernizing business critical applications and creating new applications

- *XML enhancements*

- *Usability and problem determination enhancements*

Performance

IBM reserves the right to change strategy and plans at any time.

Compatibility

Our intent:

- Provide Source and binary compatibility
  - No need to recompile entire application
    - "Old" and "new" code can be mixed within an application, and communicate with static or dynamic call
  - Most correct COBOL programs will compile without changes
    - Correct programs will run to produce the same results

- Plan to remove some old/obsolete language extensions and options

IBM reserves the right to change strategy and plans at any time.
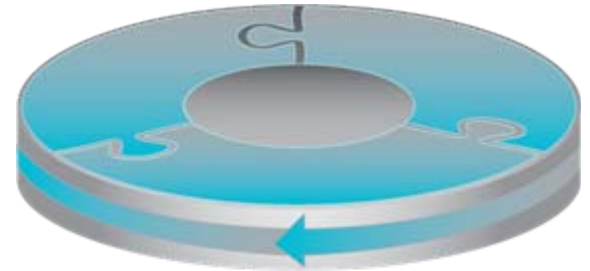
Capabilities

Our intent:

- New capability to select optimization and z processor architecture levels
- Raise data item sizes
- Improve capability for modernizing business critical applications
- XML" parse" and "generate" enhancements, web-services, Java interoperability
- Minimize administration overhead with SMF
- Improve programmability of UTF-8 applications
- New program listings and debugging interface
- Support latest Middleware (CICS, DB2, IMS)

IBM reserves the right to change strategy and plans at any time.

COBOL VNext Managed Beta Program

- Objectives
  - Provide early access to ISVs to enable their tools to support the new COBOL compiler
  - Provide early access to *Enterprise COBOL v3 and v4* customers to enable them to preview the new product and validate product acceptability
  - Collect feedback/suggestions on areas of improvement

- Requires active participation
  - COBOL development team will be directly involved
  - Regular calls will be held to discuss progress and exchange technical information

- 4 Stages (i.e. code drops) planned:
  1. Tool Interface   Completed
  2. Compatibility   Completed
  3. New features + Performance   Completed
  4. *Performance + Quality*   *(Started 01/31/2013)*

- Program contacts:
  - Marie Bradford mabrad@us.ibm.com   Roland Koo   rkoo@ca.ibm.com

    - Nomination Form: https://www-304.ibm.com/software/support/trial/cst/forms/nomination.wss?id=3869

Compilers and middleware
New releases of COBOL, PL/I and C/C++ provide improved support for middleware

- Integrated CICS and SQL translators
  - COBOL, PL/I and C/C++
  - Enterprise PL/I v4.2 improved performance of processing SQL source by up to 40%

- Programming support for new middleware features
  - CICS co-processor options, DB2 features (e.g. multiple-row insert, multiple-row fetch…)
  - Support for new SQL new data types and SQL syntax first introduced in DB2 v9

- Problem determination support with program listings and Debug Tool
  - Display SQL and CICS options in effect in COBOL and PL/I listing
  - Debug COBOL, C/C++, and PL/I applications with CICS, DB2, and IMS
  - Debug optimized COBOL applications in production

- Java Interoperability
  - Support Java 5 and Java 6 runtimes
  - Execute COBOL programs in IMS Java region

- XML Support
  - COBOL and PL/I programs can send, receive and process XML documents from middleware
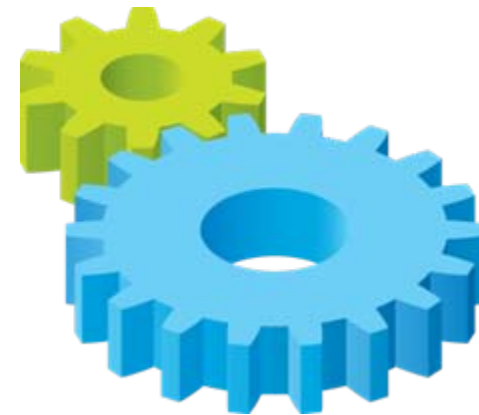
Best practices

- Upgrade compilers when you upgrade System z hardware, or Middleware (CICS, DB2, IMS)
  - Minimize quality assurance effort
  - Maximize performance
  - Leverage compiler support for new middleware features
  - Improve debugging and programmability

- Recompile only modified parts of the application

- Leverage new compiler features to modernize existing business critical applications
  - "Rip and Replace is expensive and risky
  - Modernization promotes reuse and delivery of new solutions faster, at lower cost and lower risk,

- Use Rational development tools to improve programmer productivity, and help attract new talent
  - Rational Developer for z, Rational Development and Test Environment for System z, Rational Team Concert

Strategic direction for System z Compilers

- Improve application performance on zEnterprise processors
  - Fine tune code generation and deliver new optimization features
- Provide enhancements to support modernization of COBOL and PL/I applications
  - XML, Java interoperability, Internationalization
- Support new C/C++ language features
  - Lower barrier for porting C/C++ applications to z/OS
  - New C++11 features, C1x features
  - GNU C/C++ language extensions
  - Provide Functional and performance enhancements to "Metal C
- Improve Programmer productivity
  - Deliver new usability and problem determination features
  - Provide up-to-date support for Programming tools (e.g. Debug Tool, File Manager, Fault Analyzer, RAD, WSAA)
- Support latest middleware (CICS, DB2, IMS) and z/OS
- Address customer requirements

# Boost productivity and accelerate innovation with modern skills

- **Modern IDEs for all platforms and languages**

Empower

Unify

Revitalize

Enterprise Modernization

Optimize

Enabling Product and Service Innovation

jazz

20 to 35% cost reduction in managing and maintaining existing application portfolio	**22-37%** improvement in developer productivity with modern tooling
30 to 50% cost reduction in renewal/re-architecture	**50-80%** reduction in host CPU usage
Increase Application Flexibility	*Boost Individual Productivity*
Maximize Team Productivity	*Optimize System Utilization*
15 to 20% decrease in development cycle time through common team infrastructure for collaborative application lifecycle management	**20 to 60%** improvement in application performance with latest compilers for IBM Systems

Solution: Modernize Application Development

Upgrade to modern Eclipse-based tools to develop and maintain enterprise applications spanning multiple platforms, languages, and technologies

EGL PL/I COBOL JEE C/C++

Rational Developer for zEnterprise

Unified IDE
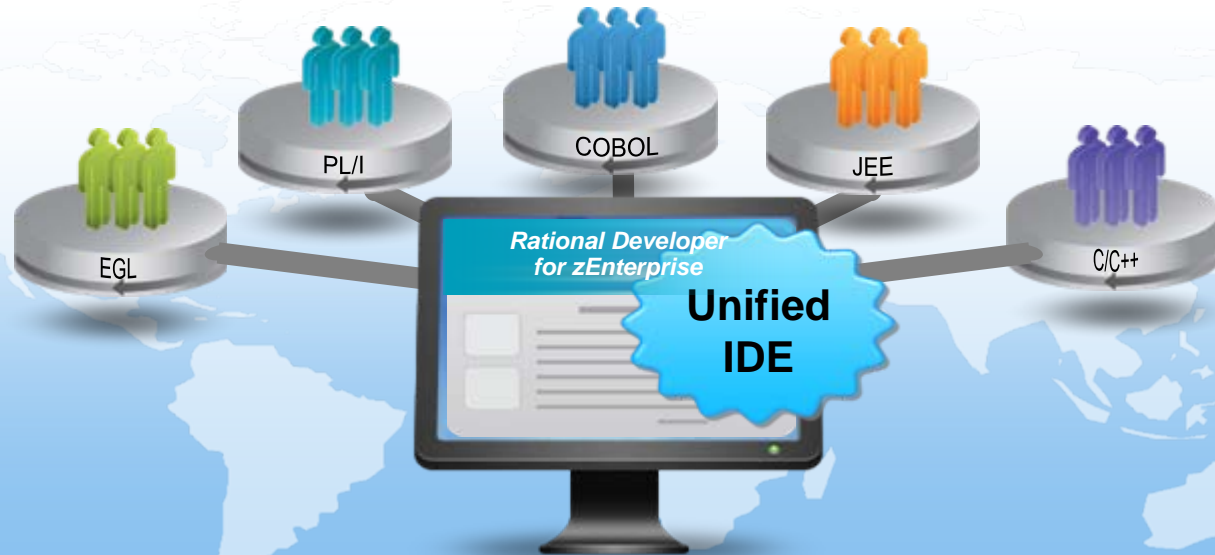
- Rational Developer for System z **NEW!**
  - Rational Business Developer
- Rational Developer for zEnterprise
  - Rational Programming Patterns **NEW!**
- Rational Developer for Power Systems
- Rational Application Developer

"Our team now develops, debugs and tests with more confidence, and it is clear that more and more people here will be using Rational Developer for System z."

UniCredit

— Almut Geiger, product specialist at HVB IS

Rational Development and Test Environment For System z

The ultimate in modern application development for System z

NEW!

COBOL, PL/I, C++, Java, EGL, Batch, Assembler, Debug Tool

| IMS | CICS | DB2 |
| WAS | | MQ |

z/OS

x86 PC running Linux

RDz user

RDz user

RDz user

ISPF user

RDz & ISPF user

RDz user
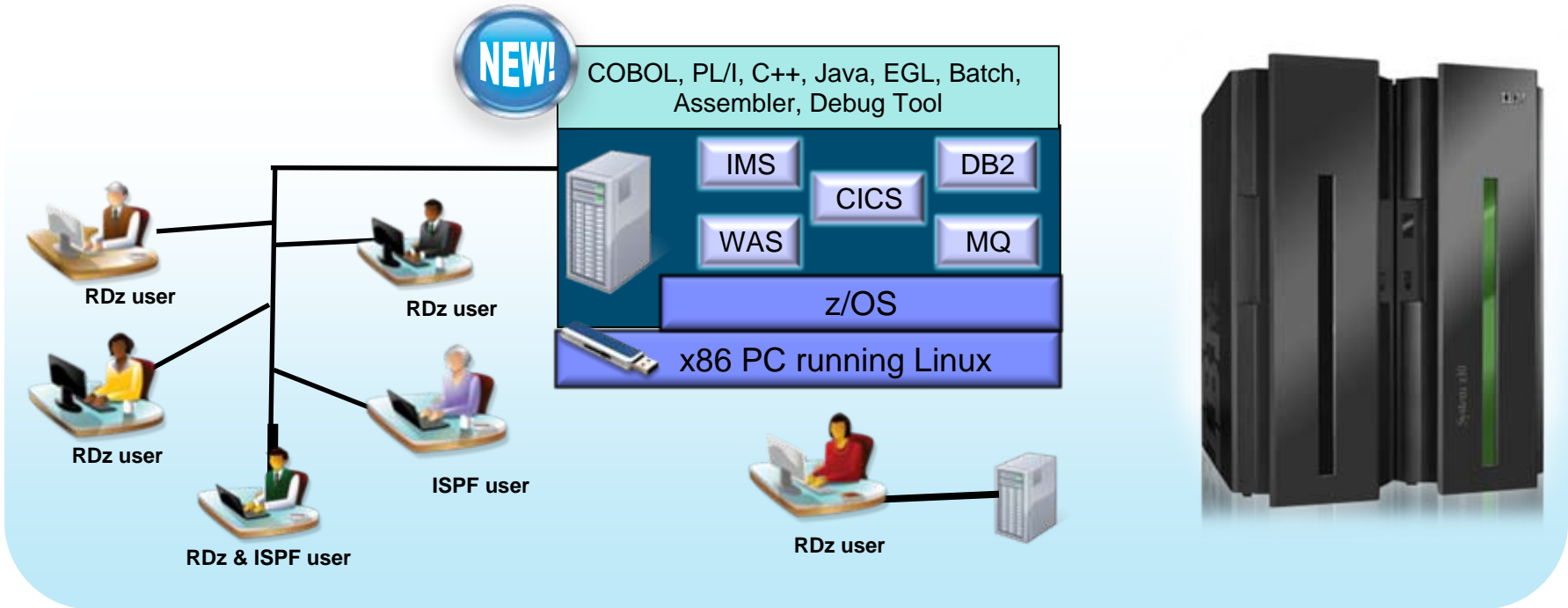
- Liberate developers to rapidly prototype new applications
- Develop and test System z applications anywhere, anytime!
- Free up mainframe development MIPS for production capacity
- Eliminate costly delays by reducing dependencies on operations staff

Note: This Program is licensed only for development, test, demonstration, and education 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.

IBM Software

Innovate2013
The IBM Technical Summit

June 2–6
Walt Disney World Swan and Dolphin Resort — Orlando, Florida

Join us in Orlando June 2-6, 2013!

Build the insights, skills and partnerships you need to deliver innovative new products faster than your competition, with lower costs and reduced risk.   Innovate 2013: The IBM Technical Summit will help you improve the practices, predictability and economics of software delivery. Register today and make your plans for June!

Visit us at the Technical Exchange Stream, Enterprise Development tracks !

A valuable opportunity you don't want to miss!

- Gain understanding of IBM's strategic vision in insightful general sessions
- Choose among over 450 breakouts with practical guidance you can apply to your work setting
- Hear first-hand experiences and learn best practices from hundreds of IBM clients.
- See innovative demonstrations from IBM and Business Partners at the high-energy Expo Hall.
- Network with over 4,000 of your peers in targeted education sessions, multi-use lounges, community receptions, and product user groups.
- Meet top strategic and technical talent.
- Gain insight into product roadmaps and the latest trends in software delivery.
- Develop new skills with self-paced, technical open labs
- Validate your product expertise with easy access to a wide variety of certification tests.
- Enjoy dynamic speakers and entertainment.

Learn more at ibm.com/innovate

Questions?

www.ibm.com/software/rational