



IBM Software Group

Overview of IBM Compilation Technology

Yesterday, Today, and Tomorrow

Steve Hikida

Program Director, Compilation Technology Development
email: hikida@ca.ibm.com

Rational software



Agenda

- ▶ IBM Compiler Team and the Product Portfolio
- ▶ Changing Landscape of the Mainframe
- ▶ Why are compilers important?
- ▶ Strategic Imperatives
- ▶ Highlights of System z Compilers
- ▶ Value of other development tools
- ▶ What is the future?



IBM Compiler Team

- **50+ year history of innovation in and contributions to the fields of languages and compilers**
- **Located primarily in IBM's Toronto, CANADA and Silicon Valley, USA Labs for the past 25 years**
- **Roughly 350 engineers in development, test, service and ID roles**
 - ▶ One of the largest compiler development teams in the world
- **Developers are key representatives on most International Language Standard's committees and consortiums**
- **Responsible for COBOL, PL/I, C/C++, Fortran, and Java JIT compilers for a wide range of hardware and OS platforms**
- **Close ties with**
 - ▶ IBM Research teams in Watson, Tokyo and Haifa
 - ▶ Academic Research (e.g. UPC, UT Austin, U of Toronto, U of Alberta...)
 - ▶ IBM CPU design teams in Austin and Poughkeepsie
 - ▶ Industry CPU design teams at AMD and Intel



IBM Compiler Optimization Technology

- **Few compilers can match our optimization**
 - ▶ XL compilers and IBM Power hardware have produced many world's fastest benchmark numbers (e.g. SPEC) even when compared to "faster" machines
- **We work directly with IBM hardware designers to ensure we fully exploit the chips and the chips contain features our compilers need**
- **Our compilers build IBM-critical software like z/OS, AIX, Lotus Domino, and DB2**
- **Close collaboration with IBM Research facilities to target emerging programming models**



IBM Compiler Product Portfolio

C/C++

- z/OS XL C/C++ (optional priced feature of z/OS)
- XL C for AIX
- XL C/C++ for AIX
- XL C/C++ for Linux on Power Systems
- XL C/C++ for Blue Gene (PRPQ)
- XL C/C++ for Multicore Acceleration
- C/C++ for z/VM

Fortran

- XL Fortran for AIX
- XL Fortran for Linux on Power Systems
- XL Fortran for Blue Gene (PRPQ)
- XL Fortran for Multicore Acceleration

COBOL

- Enterprise COBOL for z/OS
- COBOL for AIX
- COBOL for Windows (component of RD/z)

PL/I

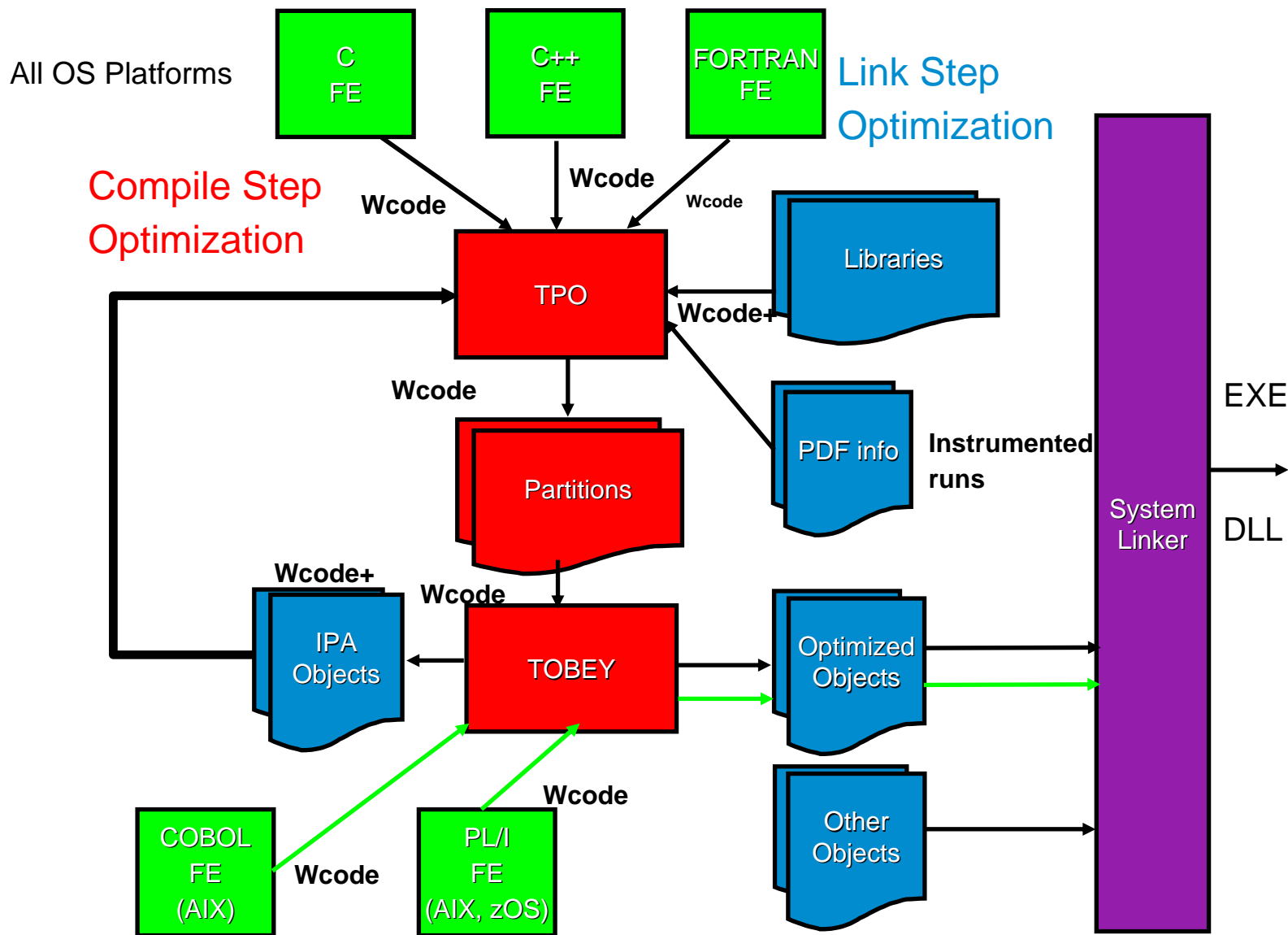
- Enterprise PL/I for z/OS
- PL/I for AIX
- PL/I for Windows (component of RD/z)

PL/X

- PL/X-390



IBM Compiler Architecture - Leveraging Common Components



Changing Landscape of the Mainframe

- **Commitment to commercial application development**
 - ▶ Support Batch Processing and legacy z/OS file systems (ISAM, VSAM, ...)
 - ▶ Integrate with middleware (CICS, IMS, DB2, ...)

- **Customer need to Modernize their Enterprise investment**
 - ▶ Extend legacy applications to integrate with modern technologies including Web Services, Java, and XML to support SOA

- **System z hardware design increasing focus on high performance**
 - ▶ High Performance Computing is no longer just for the distributed market

- **Increase of applications and middleware being ported to System z from other platforms**
 - ▶ Compilers are key enablers for the platform



Success of any Platform dependent on Compilers

- **COBOL critical to commercial market**
 - ▶ 75% of world's business data
 - ▶ 90% of financial transactions are processed in COBOL
 - ▶ 1.5 to 2.0 million developers
 - ▶ 200 billion lines of code
 - ▶ 5 billion lines of NEW COBOL source code yearly
 - ▶ Language is simple, easy to read, and self documenting
 - ▶ Applications are portable and easy to maintain
- **C/C++ usage significantly increasing as z/OS development language**
 - ▶ C/C++ is a dominant and growing implementation language of portable applications and middleware
- **Compilers supporting industry language standards and extensions lower the barrier to entry by enabling application portability**
- **Compilers maximize performance of hardware and OS architecture**



Strategic Imperatives for IBM's Compilers

- **Commitment to Long Term investment and Development for IBM's compilers**
 - ▶ COBOL, PL/I, C/C++, Fortran are critical languages to IBM and our customers
 - ▶ Leverage common components

- **Commitment to Language Standards Conformance**
 - ▶ Support existing and new programming languages

- **Maximize IBM Hardware and OS Exploitation through Compiler Performance**
 - ▶ Help IBM achieve price-performance leadership goals
 - ▶ Actively participate in new processor design
 - ▶ Deliver new & innovative optimization technologies to maximize the performance of applications and industry benchmarks
 - ▶ Extend support for parallelization

- **Commitment to Superior Customer Service**

- **Strengthen the System z platform**



Few Interesting Facts about IBM Compilers

- **Industry Leadership in Optimization Technology**
 - ▶ IBM Compilers are key contributors in IBM's success among the World's Fastest Supercomputers (www.top500.org)
 - ▶ IBM XL C/C++ and XL Fortran compilers provide industry leadership in SPEC integer and floating-point performance benchmarks (www.spec.org)

- **IBM PL/I first release in 1999 with a new release every year since (10 consecutive years in a row)**

- **Nearly 49 years since IBM's first COBOL release in 1960 on OS/VS**
 - ▶ COBOL is the flagship of IBM's Compiler Portfolio
 - ▶ Modern enhancements include support for Web Services, Java, and capability to offload COBOL XML parsing to zAAP specialty processors



Compiler Optimization with the z10

Supporting System z innovation, raising the bar and taking System z to the next level of...

... Compiler Optimization and Performance

- Exploit latest hardware without need of expert knowledge of architecture
 - Enables users to exploit performance edge of hardware without source code changes
- Exploit 36 NEW z10 instructions from the General-Instructions-Extension facility
- Exploit IEEE Decimal Floating-Point (DFP)
- Exploit Additional Floating-Point Registers (AFP)
- Exploit 64-bit instruction set and registers even in 32-bit code
- Support IEEE Binary Floating-Point which eases platform portability
- Maximize application performance using new & innovative optimization technologies
 - Reduces total cost of ownership
 - Up to 10-25% Performance Improvements²



¹ Individual features in the content list may not be applicable to all IBM compiler languages. Check specific language documentation for details.

² Performance improvement results based on select benchmarks. Results will vary depending on application.



Compiler Release Highlights

- **z/OS XL C/C++ V1.9 Update¹, simultaneous GA with z10 in February 2008**
 - Exploit NEW z10 instructions and architecture
 - Exploit NEW DFP instructions
 - Introduced low level system programming (Metal C) as alternative to HLASM
 - Significantly increases developer productivity
 - Enhance instruction scheduling capability to reduce execution latency caused by AGI delays
 - Exploit condition code set by the A and S instructions removing unnecessary compare instructions
 - Increase usage of PDF information improving performance
 - Eliminate implicit memory usage limits during whole-program optimizations
 - Increase optimization scope

- **z/OS V1.10 Preview Announce including XL C/C++ V1.10**
 - ▶ Additional enhancements and tuning to exploit z10 architecture
 - ▶ Support for mixed 32- and 64-bit addressing modes for the METAL option
 - ▶ Standard C++ Library enhancements
 - ▶ Increase performance

¹ Content list includes features introduced with base z/OS XL C/C++ V1.9 which GAed Sept 2007.



Compiler Release Highlights (continued)

- **Enterprise PL/I V3.8, GA October 2008**
 - 10th release in 10 years
 - Exploit NEW z10 instructions and architecture
 - ARCH(8) / TUNE(8) support
 - Exploit NEW DFP instructions
 - GOFF object support
 - HGPR option to allow more exploitation of 64-bit registers in 32-bit code
 - Increase user control of built-in function behaviour
 - Enhanced SQL support
 - Provide built-in language support for high speed XML parsing and generation



Compiler Release Highlights (continued)

▪ **Enterprise COBOL V4.1**

- Provide built-in language support for high speed XML parsing and generation
 - COBOL XML parsing can be offloaded to zAAP specialty processors
- Supports z10 architecture¹
- Improve Unicode Performance
- Enhance DB2 support
 - Support new SQL data types and syntax provided by DB2 9
 - DB2 options are shown in the compiler listing (DB2 9 only)
 - SQLCA and SQLDA control blocks are expanded in the compiler listing
 - New compiler option SQLCCSID is provided to coordinate the coded character set ID (CCSID) between COBOL and DB2

¹ Current releases of COBOL support z10 architecture. Exploitation of z10 instructions and architecture targeted for future releases.



The Rational Software Delivery Platform

Accelerating software innovation

Manage enterprise wide assets and change with common shared repository
Rational ClearCase

Understand business process & IT system mapping. Refactor existing assets for reuse or migration.
Rational Transformation Workbench
WebSphere Studio Asset Analyzer
Rational Asset Analyzer

Process & Portfolio Management

Collaborate across all teams, customize processes, painless status
Rational Team Concert

Change & Release Management

Governance & Lifecycle Management

Integrated Requirements Management

Create architectural model from business process model. Manage reuse.
Rational Software Architect
Rational Asset Manager

Produce coordinated, traceable, automated, and cross-platform builds
Rational Build Forge

Enterprise Modernization

Architecture & Construction

Accelerate development of web apps, traditional RPG, COBOL, C, C++, PL/I, EGL apps, web services, and XML-based interfaces
Rational Developer for z
Rational Developer for i
Rational Business Developer

Quality Management

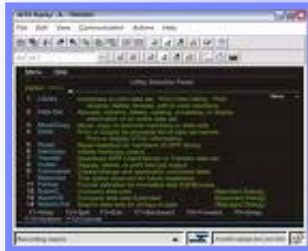
Reduce time and guarantee repeatability with automated testing including 3270 and 5250
Rational Functional Tester

Quickly and easily extend your 3270 and 5250 applications to the web
Host Access Transformation Services



Achieve high productivity, attract new talent with modern IDEs

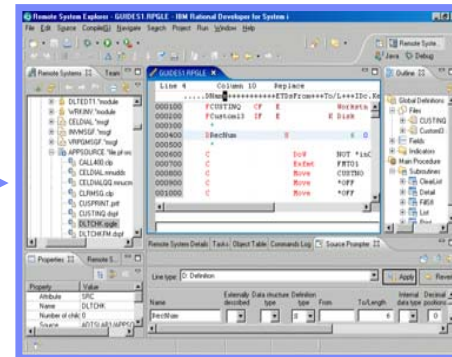
- ✓ Attract new talent with Eclipse-based IDEs
- ✓ Develop new workloads faster using built-in high productivity features
- ✓ Create web services quickly from existing and new COBOL, PL/I, Java, or C/C++ code
- ✓ Reduce System z MIPS by offloading testing to developer workstations



Rapid application development



- ✓ Sought development tools to increase productivity and position the company for growth
- ✓ Experienced COBOL developers made switch to new RDz IDE while sustaining workload
- ✓ Younger developers appreciated RDz productivity tools and how easily mainframe could be navigated

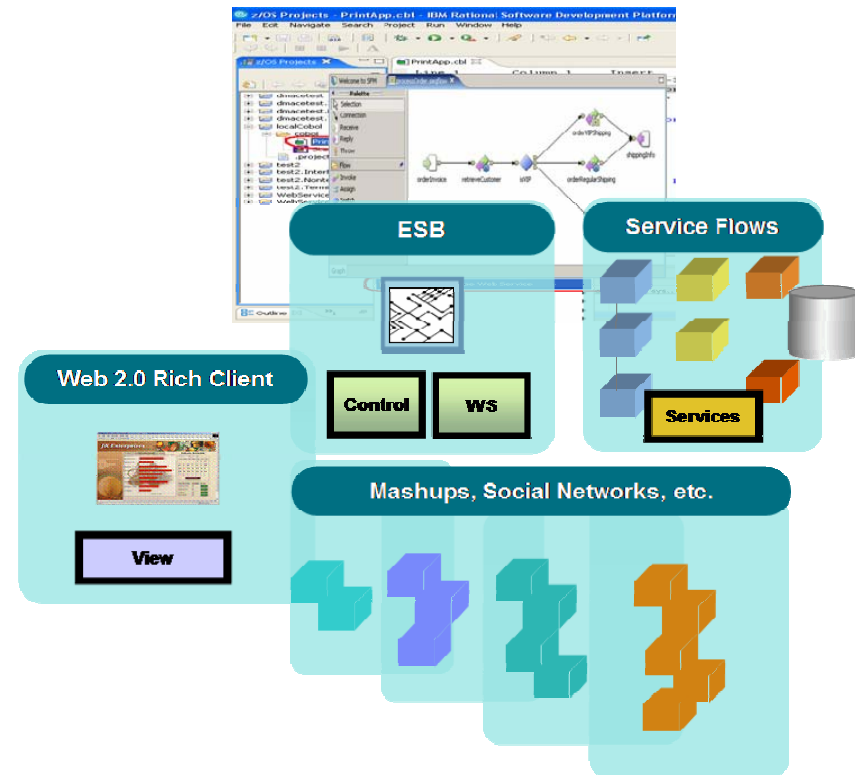


Accelerate delivery of traditional and modern workloads

IBM Rational Developer for System z v7.5



- ✓ **Accelerate development of Core Systems**
 - COBOL, PL/I, CICS, IMS, DB2, Batch, UNIX System Services, and WebSphere environments
- ✓ **Increase developer productivity and flexibility**
 - New and existing skills can collaborate, navigate the mainframe and work with core IT applications
- ✓ **Simplify SOA and Web service construction**
 - Improved Service Flow Modeler now supports simplified screen flow recording to integrated deployment
- ✓ **Speed model-driven development**
 - COBOL design and development using UML
- ✓ **Integrated application lifecycle**
 - Enhanced z/OS projects management with Rational ClearCase
 - Enhanced Problem determination tool integrations
- ✓ **Automate CICS service deployment & testing**
 - Integrated CICS explorer and Application Deployment Manager to manage CICS resource definitions
- ✓ **Two new offerings for greater choice and flexibility**
 - IBM Rational Developer for System z with EGL v7.5
 - IBM Rational Developer for System z with Java v7.5



IBM Rational Developer for System z v7.5



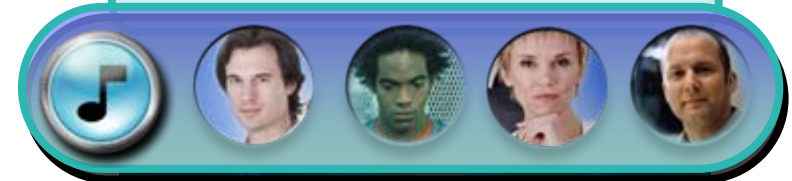


Increase collaboration between dispersed software teams

IBM Rational Team Concert for System z v1.0

- ✓ Real time, in-context team collaboration
 - Make software development more automated, transparent and predictive
- ✓ "Think and work in unison"
 - Integrated source control, work item and build management
- ✓ Assess real-time project health
 - Capture data automatically and unobtrusively
- ✓ Automate best practices
 - Dynamic processes accelerate team workflow
 - Out-of-the-box or custom processes
- ✓ Unify software teams
 - Integrate a broad array of tools and clients
 - Extend the value of ClearQuest and ClearCase
- ✓ Exploit Quality of Service of z/OS
 - Hosted on z/OS with DB2 & WAS
- ✓ z/OS development support (coming)
 - Cobol support, EGL support, z/OS builds, RDz integration, existing SCM integration

IBM Rational Team Concert for System z



transparent *integrated presence*
 wikis OPEN real-time reporting
 chat automated hand-offs Web 2.0
custom dashboards automated data gathering
EXTENSIBILITY *Eclipse plug-ins* services
 architecture **FREEDOM TO CREATE**

JAZZ TEAM SERVER

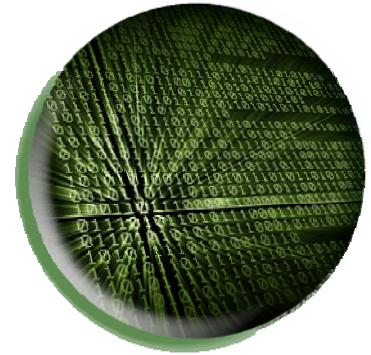
Open and extensible on



- ✓ Collaborate in context
- ✓ Right-size governance
- ✓ Day one productivity



Looking Forward



Languages and compilers continue to evolve

- Critical to ensure compatibility while significantly increasing capability

Growth in Workload Consolidation

- Consolidating infrastructure applications and databases for increase efficiencies
- Compilers will increase the exploitation of offload engines with the growth for content-rich, real time workloads
- Greater importance on Language Portability

Industry moving to ~~faster~~ more processors

- Greater focus on compilers to increase parallelization and multi-core exploitation
- Languages and Compilers will simplify programming model to new architectures

Increase leverage of common componentries

- Leverage optimization technology and innovation across languages and platforms



Final Thoughts

Languages and Compilers are critical and a key business for IBM

- IBM has a huge investment in Compilers and it has been growing

Innovation on Optimization Technology far from maturity

- Significant pipeline of innovation on increasing performance and exploitation of new architectures

We've been in the compiler and language business for over 50 years, we are looking forward to the next exciting 50 years! 😊



QUESTIONS



**Learn more at:**

- [IBM Rational software](#)
- [IBM Rational Software Delivery Platform](#)
- [Process and portfolio management](#)
- [Change and release management](#)
- [Quality management](#)
- [Architecture management](#)
- [Rational trial downloads](#)
- [Leading Innovation Web site](#)
- [developerWorks Rational](#)
- [IBM Rational TV](#)
- [IBM Business Partners](#)
- [IBM Rational Case Studies](#)

© Copyright IBM Corporation 2008. All rights reserved. The information contained in these materials is provided for informational purposes only, and 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, these materials. Nothing contained in these materials 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. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. IBM, the IBM logo, Rational, the Rational logo, and other IBM products and services are trademarks of the International Business Machines Corporation, in the United States, other countries or both. Other company, product, or service names may be trademarks or service marks of others.



Maximize application performance with the latest compilers

IBM Compilers on System z

New and enhanced products and resources

▶ **z/OS XL C/C++ V1R10 (Available now)**

- ▶ Exploit the new z10 processor including the Decimal Floating Point module.
- ▶ Middleware (DB2, CICS, IMS) integration
- ▶ METAL C Compiler option

▶ **XL C/C++ for z/VM V1.2 (Available now)**

- ▶ C/C++ development for z/VM programmers
- ▶ Supports the C99 programming standard
- ▶ New optimization levels to improve performance

▶ **Enterprise COBOL for z/OS, V4.1 (Available now)**

- ▶ Supports high speed XML parsing and generation.
- ▶ COBOL XML parsing supports offloading to zAAP specialty engine
- ▶ Enhanced DB2 support through new SQL data types and syntax
- ▶ Integrated CICS translator and provide access to IBM IMS™ system
- ▶ Supports integration of COBOL and web-based business processes in Web services, XML, and Java™ applications

▶ **Enterprise PL/I for z/OS, V3.8 (Available now)**

- ▶ Exploit the new z10 processor including the Decimal Floating Point module
- ▶ PL/I XML parsing can be offloaded to zAAP specialty engine
- ▶ Powerful set of functions for the UTF-sensitive processing
- ▶ Support integration of PL/I and web-based business processes in Web services, XML, and Java™ applications
- ▶ Access to IBM DB2®, IBM CICS®, and IBM IMS™ systems

* Exploitation depends on language

▶ **Maximize System z10 Performance ***

Exploit new z10 instructions and architecture. Exploit new DFP instructions. Increase leveraging PDF information.

▶ **New Metal C support for System z**

Provides ability to generate code without Language Environment run-time dependencies and supports HLASM embedded source within C statements.

▶ **Support latest IBM XML parsing technology**

Enables offloading of COBOL and PL/I XML parsing to zAAP specialty engine

▶ **Support Application Modernization**

Legacy COBOL and PL/I can now be integrated to web-based business processes in XML, Web services, and Java applications.

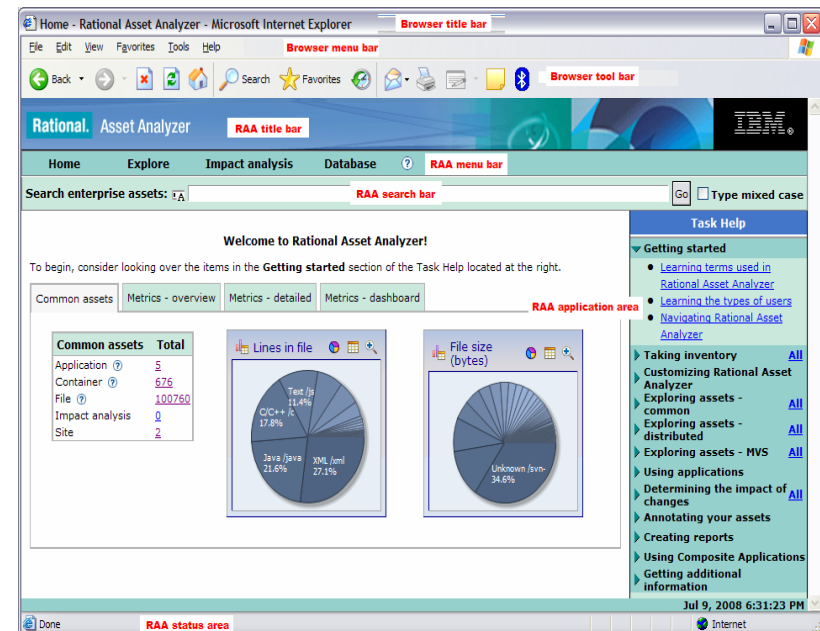




Automate application understanding and impact analysis

IBM Rational Asset Analyzer v5.5

- ✓ Gain business intelligence of software assets
 - Understand mainframe and distributed application assets and their inter-relationships
- ✓ Make code changes quickly, with less risk
 - Use impact analysis to quickly assess the impact of proposed code change
- ✓ Shorten learning curve
 - New developers can work independently with minimal support from subject matter experts
- ✓ Reduce software project complexity
 - Deliver up-to-date knowledge of application components from the code itself
- ✓ Improve process and team efficiency
 - Provide application insight to all team members via a web browser interface
- ✓ Offload application analysis
 - Installs on the Windows platform and offload mainframe processing
- ✓ Improve disaster recovery
 - Better application documentation and increased confidence that the location of all source for applications is known



IBM Rational Asset Analyzer v5.5

