



IBM Software Group

Leverage existing assets by migrating old code to newer technology

2008



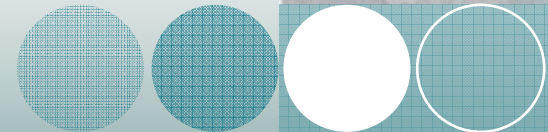
Ed Gondek
Business Unit Executive
Application Transformation & Modernization
IBM Rational Software
egondek@us.ibm.com

ON DEMAND BUSINESS™

© IBM Corporation

Goals for Today

- Discuss Application Transformation as a means for Modernization
 - ▶ What Issues does IT face in dealing with Legacy Systems?
 - What are the Challenges?
 - ▶ Spectrum of Modernization Options
 - ▶ Proposition: Application Transformation
 - It's about reuse
 - ▶ Why Rational Business Developer?
 - ▶ Summary



Key Messages

Businesses need to change to stay viable, and IT must enable this flexibility.



IT must overcome challenges to enable flexibility

Challenges, such as:

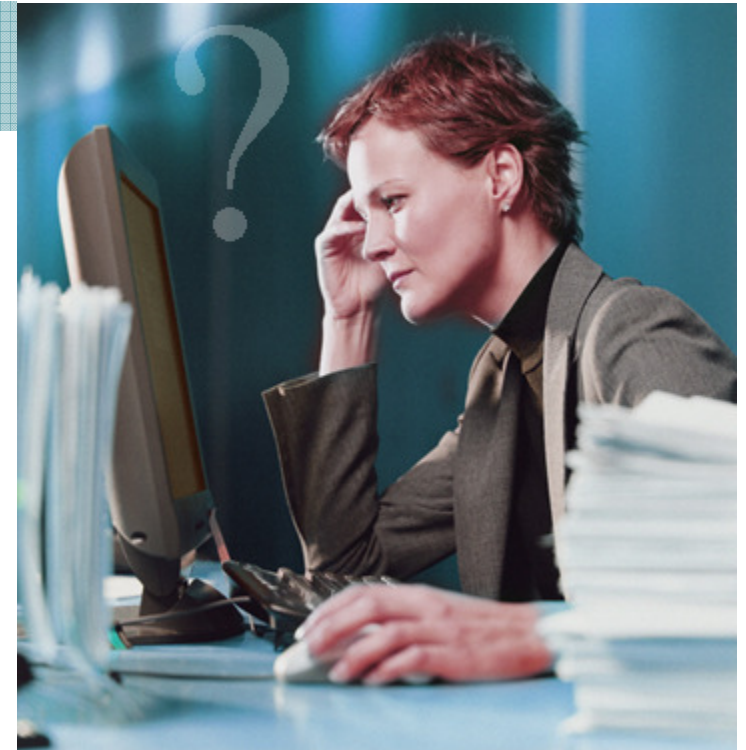
- ▶ Reuse & leverage core systems
- ▶ No inventory of current IT assets
- ▶ Complex, tightly coupled architectures
- ▶ IT Skills lock-in
- ▶ Islands of development
- ▶ No flexibility for new investments
- ▶ High Cost of Maintenance

The **IBM Rational Software Delivery Platform** helps companies overcome these challenges with comprehensive support for **enterprise modernization**.



IT Modernization Challenges

- ▶ Major investment in older applications (time, money, effort)
 - This is tested, proven, business-critical code – it runs the business
 - Advantageous to reuse (harvest) as much as possible
 - **Escalating software vendor maintenance** fees for obsolete environments
- ▶ Maintaining separate tools for each platform
- ▶ **Retraining, Retaining and Attracting talent**
 - Advantageous to leverage existing IT staff and skills



Why Modernize?

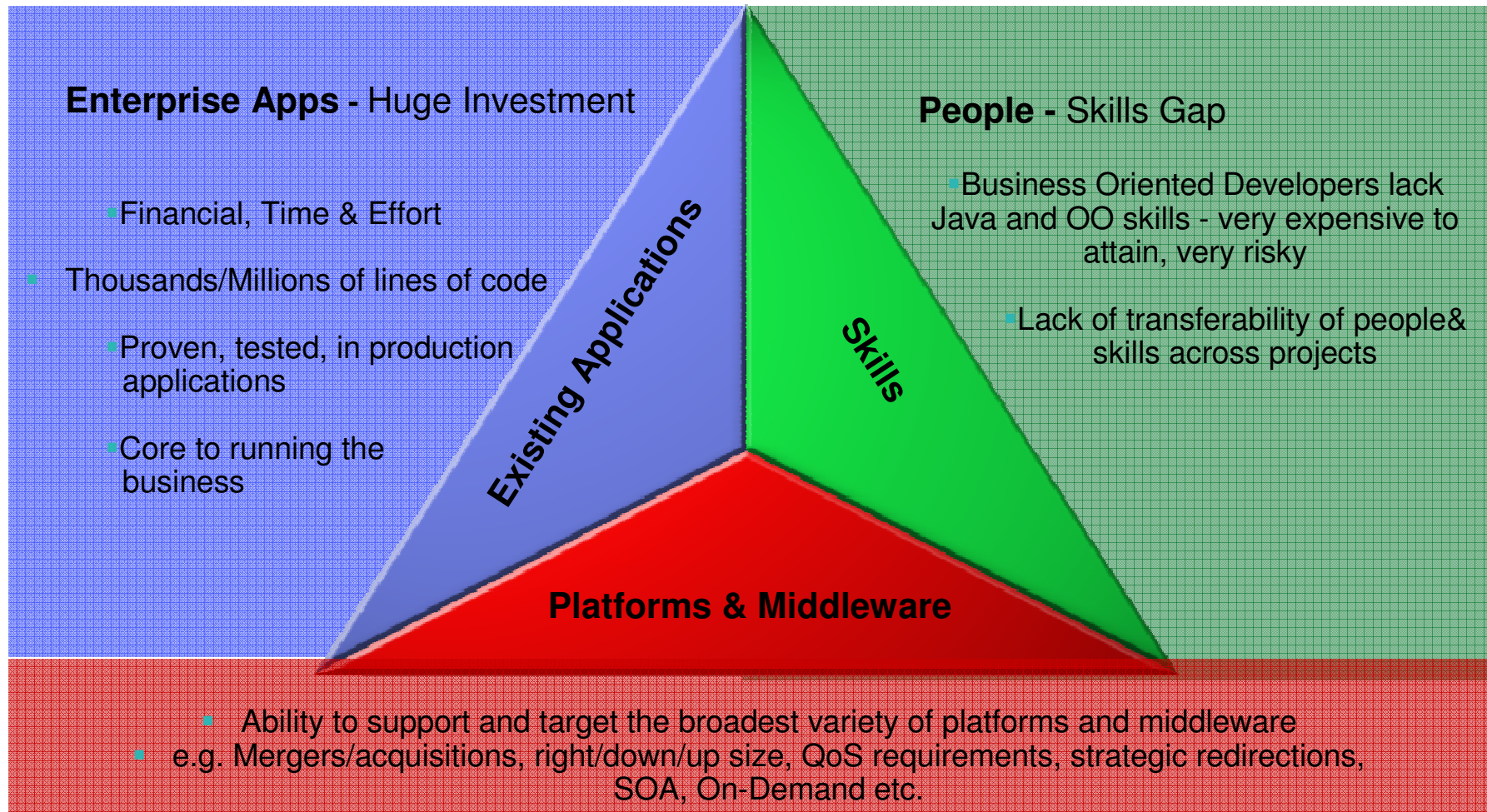
To take advantage of latest technology innovations like Web, RCP, integrated (composite) applications and SOA:

- ▶ **Gain competitive advantage and speed to market**
 - These systems run the business
 - They're not necessarily broken
- ▶ **Reduce costs and align IT with business goals**
 - The cost of completely replacing them is unrealistic
- ▶ **To deliver real value to the business**
 - It's all about reuse and leveraging investment
- ▶ **Remove Inhibitors**
- ▶ **It's doable**

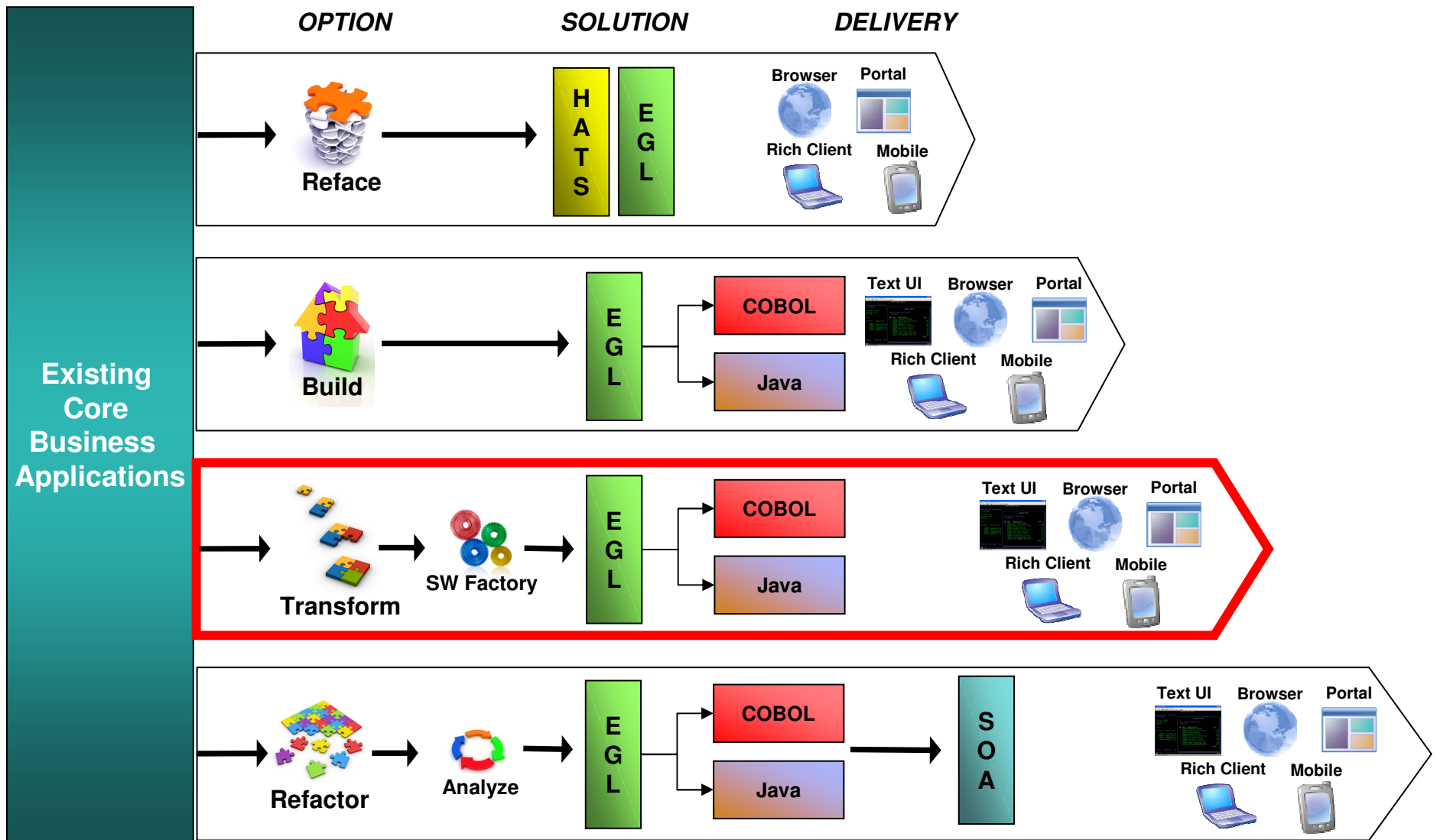
This requires new software development tools and capabilities that reduce the risk and costs associated with updating IT environments.



The IT Challenge - Leverage & Reuse



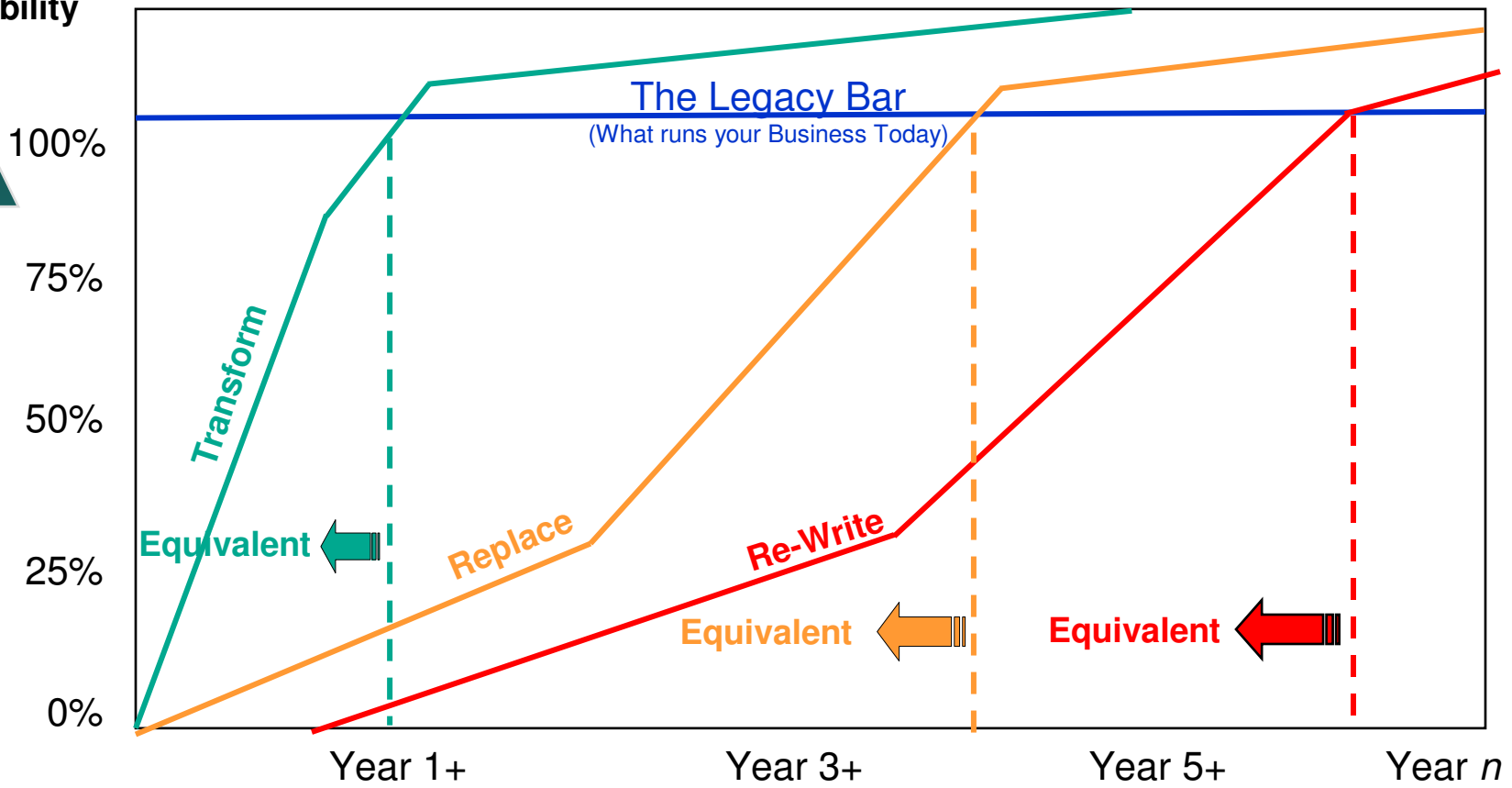
Application Modernization Options



Proposition: Application Transformation

meet and rapidly exceed the present application functionality, performance and stability – the most cost-effective approach

Functionality
Performance
and Stability



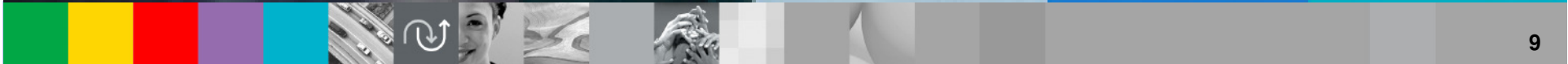
Application Transformation Approach

High-level Process Flow

Phase I
Discovery and Analysis
(DNA)

Phase II
Code Transformation,
Education &
Knowledge Transfer

Phase III
Implementation, Testing
& Deployment



Application Transformation Phases

- Phase I
 - ▶ DNA – Discovery aNd Analysis of Natural Environment
 - Identify all Application Assets
 - Inventory, Measure, Analyze, Scoping *
 - ▶ Database Migration Assessment
 - ▶ Define Project Scope
 - ▶ Define Target platform – System z, System i, Distributed, COBOL, Java, Transaction Processor etc.
- Phase II
 - ▶ Application Conversion/Transformation* (off-site)
 - ▶ Education/ Knowledge Transfer/ Training
 - ▶ Establish Development & Test Environment
 - Software
 - Database
 - Application Server
- Phase III
 - ▶ Implementation & Deployment of application(s)*
 - ▶ Data migration
 - ▶ Testing, Testing, Testing <<< *can be the longest window of time*
- Post-implementation assessment & support

*iterative



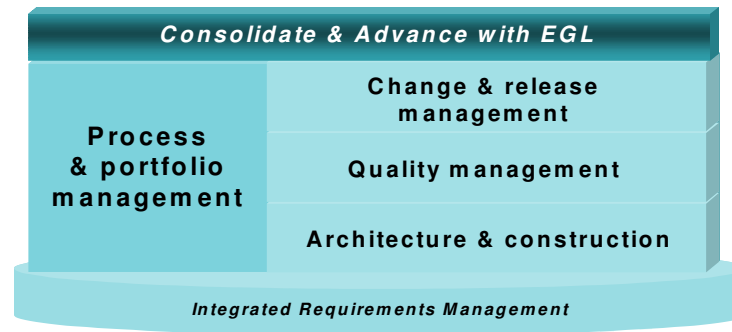
Modernize Development Investments

Transform your legacy applications...

Non IBM Technologies	
HPS/AppBuilder	
COBOL	 COOL: GEN COOL: Enterprise
Maestro IDMS ADS/Online *	
natural ADABAS	CSP
SYNON *	Visual Basi SYBASE OpenEdge * PowerBuilder
RPG	
IBM Technologies	
VA Gen PACBASE	Informix 4GL SOFTWARE

* Future Capability

- ▶ Migrate from out-of-date and expensive legacy development platforms onto **IBM SDP** via **Rational Business Developer** featuring **Enterprise Generation Language (EGL)**



Deploy to:

Linux
 AIX
 Solaris
 HP-UX
 System i
 System z
 Windows
 WAS

Why is Rational Business Developer (EGL) the Target?

Skills: Learning and Adoption



- Java – great language and technology; too difficult for many customers to learn and adopt



- C# is an OO language same issues as Java plus many API's

**RPG/
COBOL/
4GLs**
(Natural)

- Comfortable languages, great capabilities, but graying staff, no new developers

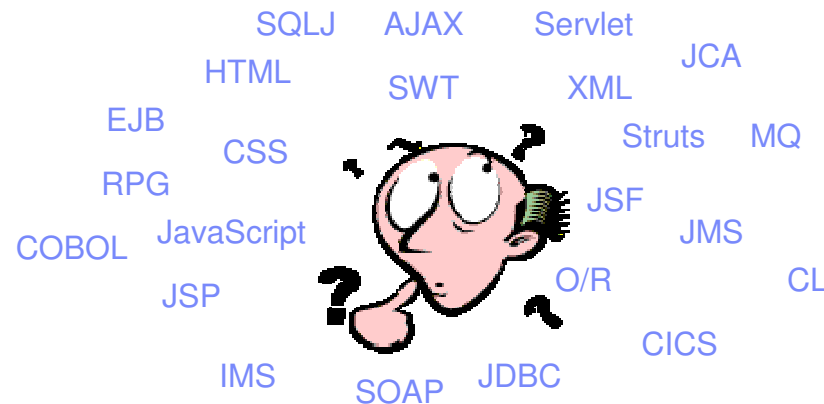


“Re-training COBOL developers to Java/J2EE costs over \$50K each, and only 12% may actually succeed” – Gartner Group

*“The task force had an initial meeting and identified challenges to teaching Java based on the literature and our experiences.” – ACM Education Board Java Task Force
<http://www.sigcse.org/topics/javataskforce/>*

An insurance company spent approximately \$250,000 to train 12 RPG developers. One out of 12 succeeded. This person had an MS in Computer Science and 8 years of C programming experience. – Bob Cancilla, IBM

Building and modernizing applications today is not easy...



but leveraging the IBM Rational Software Delivery Platform and standardizing on Rational Business Developer (EGL) can make the difference between success and failure

What problems does RBD (EGL) solve?

IBM Rational empowers business oriented developers to rapidly deliver new services and applications and modernize existing investment

Easily integrate mainframe assets into web and service oriented solutions

IBM Rational provides development tools for rapidly creating new mainframe services or extending legacy assets to SOA and web



Leverage trusted mainframes QoS and achieve high level of reuse for new Service Oriented solutions

Break Skills Silos

IBM Rational helps achieve developer portability across projects with a unified easy to learn development approach



Maximize the use of programming resources and leverage their business domain knowledge

Simplify and accelerate cross platform development

IBM Rational helps developers quickly build business services and applications independently of the runtime platform



Reduce duplication of business components and services across heterogeneous platforms

Gain competitive advantage, reduce development costs and accelerate time to market with IBM Rational software innovative development technology

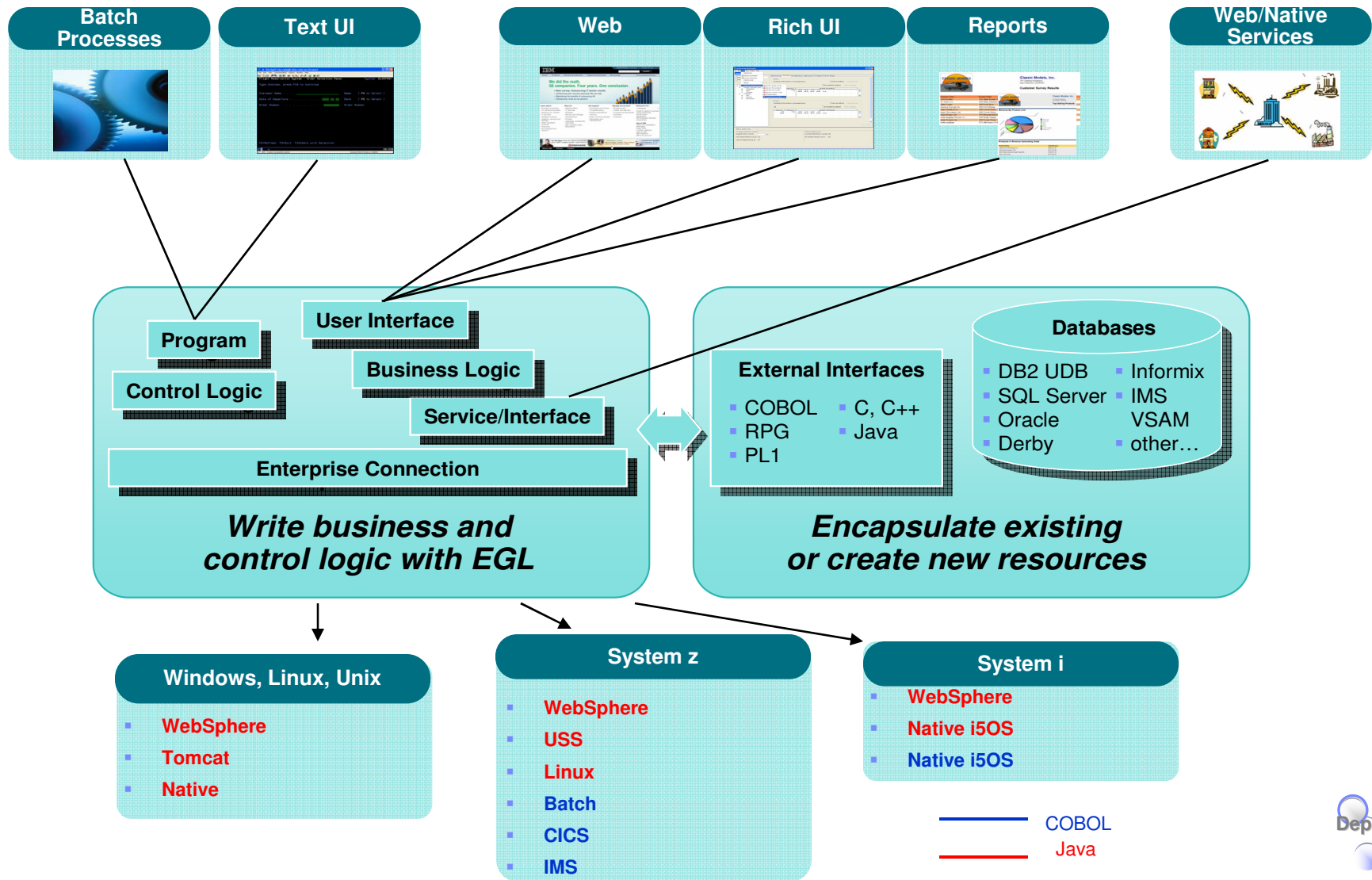


Why RBD (EGL) for Application Modernization?

- **Platform flexibility**
 - ▶ Broad choice of (new) target environments
- **Modern, Robust, Open, SOA ready**
 - ▶ Future-proof architecture allows to grow with business requirements
- **Easy to learn for Business developers**
 - ▶ No need to re-staff, productive in a very short time
- **Procedural nature of EGL has greater “affinity” with Business Languages (and business developers)**
 - ▶ More natural “mapping”
 - ▶ Easier to automate the transformation process
 - ▶ End result is understandable and highly maintainable
- **Eliminate costly runtime charges**



Respond to Broadest Application Needs



Rational Business Developer Advantage

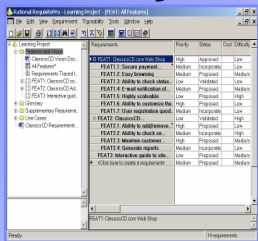
- RBD is a thoroughly modern and powerful programming language
 - hides complex J2EE, SOA, middleware and runtime technologies
 - designed for developers who need to solve business problems, not technology problems
 - Targeted at business developers building business applications for business platforms...BUSINESS IS THE KEY
- RBD is extremely easy to learn
 - weeks instead of months/years
 - provides a unified programming paradigm regardless of the target runtime platform
- RBD is flexible & productive
 - applications can be deployed across any platform or operating system
 - Proper balance between high level abstractions for productivity and access to the details for flexibility
- RBD is a strategic part of IBM's Software Delivery Platform



RBD – Integral Part of the Rational Software Delivery Platform



Analyst



Rational RequisitePro

Rational ClearQuest

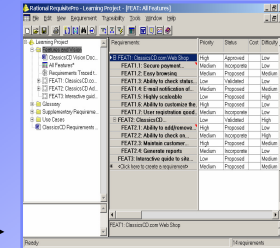
Requirements Coverage

Defect and Change Tracking

Test Planning and Tracking



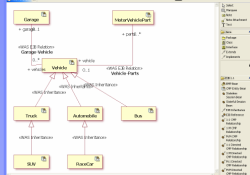
Tester



Rational Test Tools



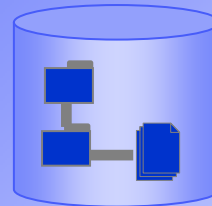
Developer



Rational Business Developer

To Do List

Unified Change Management



Versioned Artifacts
Components & Subsystems
Baselines & Builds

Rational ClearCase



Summary

- Enterprise Application Transformation of Legacy systems is a **Viable, Cost-Effective** approach for modernizing these aging application technologies
 - ✓ Preserve your huge investment in these systems
 - ✓ Maintain the full functionality of these systems
 - ✓ Replace the part of the system that “inhibits”
- Allows for **Reuse of proven functionality** extracted from tested, production-ready systems
 - Enables the **Redeployment of business-critical, stable functionality** to be used by SOA
- Provides maximum deployment flexibility and use of IT resources staff
- Enables **Rapid** new application development via RBD
- Eliminates Inhibitors prepares IT organizations for future development



Thank
you