EM03

EGL and IBM Rational Business Developer: An Executive Overview

Todd Britton
Director, Enterprise Tools & Compilers
tbritton@us.ibm.com

IBM Rational Software Development Conference 2008

WHERE TEAMS ARE R-HEROES

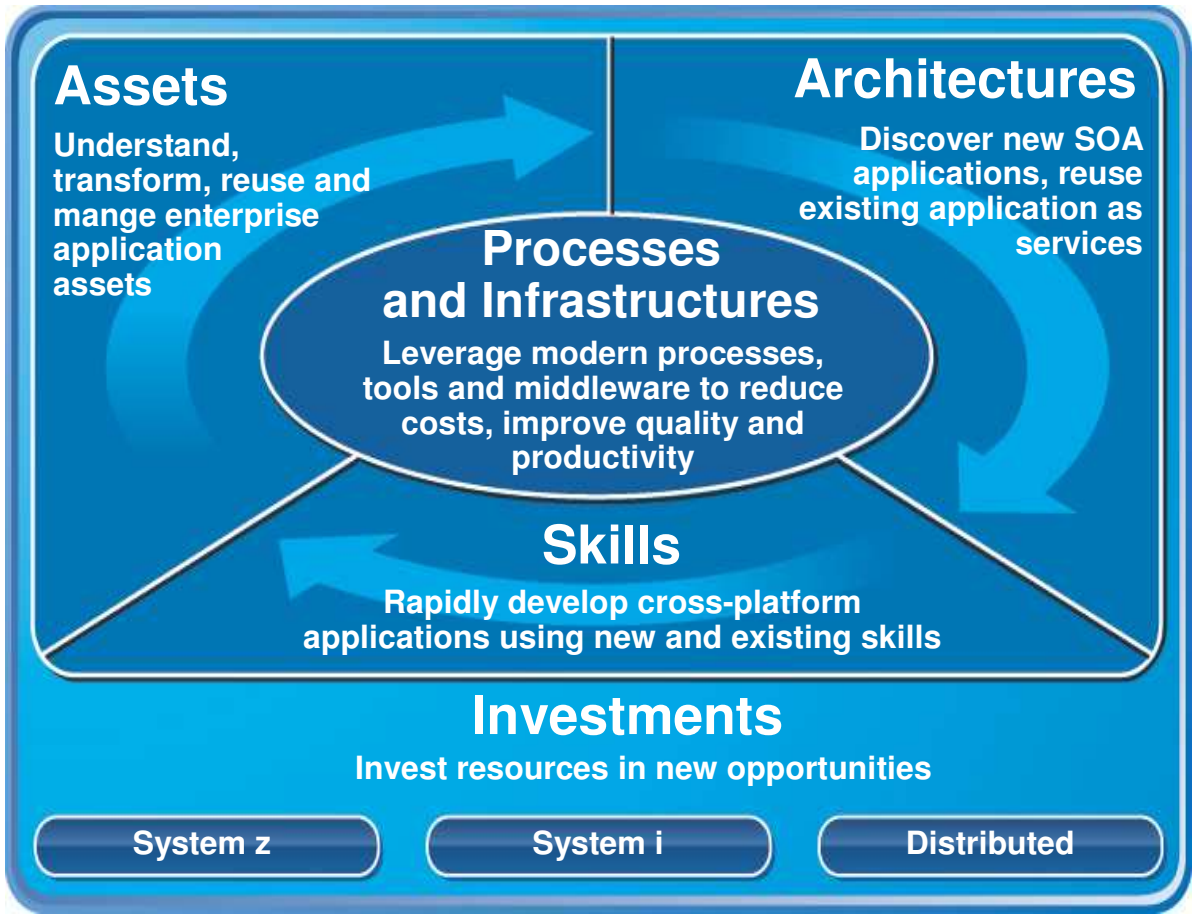
IBM

Rational. software

Agenda

- Today's application construction challenges
- Rational Business Developer and EGL
  - Capabilities and value proposition.
- Application Transformation and Modernization
- Customers success
- Product Demonstration
- Summary

Enterprise Modernization – Solution Overview



Leverage value in existing assets

Drive innovation with SOA and web technology advancements

Leverage existing and new staff on multi-platform projects

Improve quality and flexibility with consolidated team infrastructure

Reduce maintenance costs

http://www-306.ibm.com/software/info/developer/solutions/em/

Key architecture management challenges

SAP HR system
Custom .Net applications
Back-office legacy systems
Home-grown line of business apps.
Oracle Siebel CRM

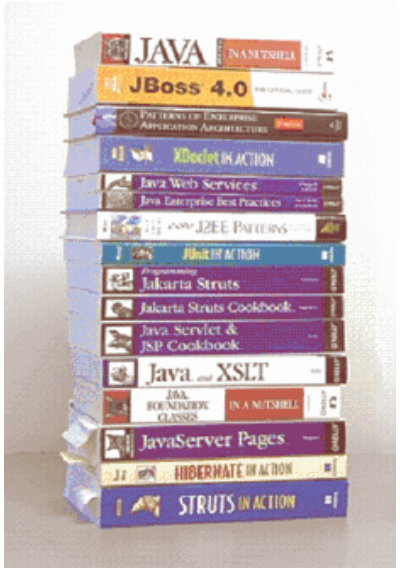
Windows
.Net
AIX
Linux
Java
COBOL/zOS
5iOs

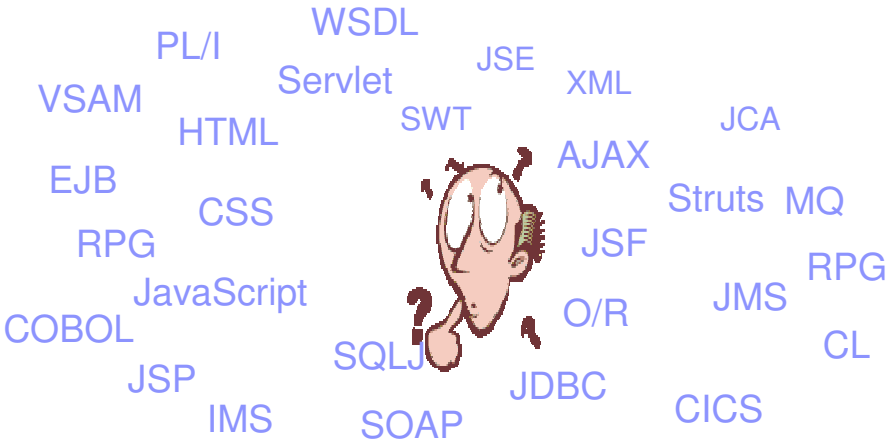
Existing Applications	Skills	Platforms / Middleware
▪ Costly to maintain	▪ Skills silos	▪ Proliferation
▪ Monolithic	▪ Skills mismatch	▪ Coexistence
▪ Hard to reuse in new ways	▪ Erosion of legacy platforms skills	▪ Complexity

High costs — **Compromise** — **Slow response**

A Closer Look at Construction Challenges

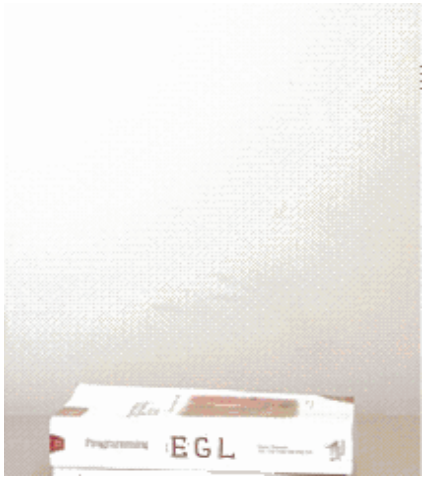
Building applications today is not easy





- Learn and master a myriad of middleware programming interfaces.
- Understand new programming paradigms, frameworks, libraries.
- Cross-platform solutions require totally different programming skills.
- Constantly emerging new technologies, frameworks, standards, etc.

Building today's applications *CAN* be easy



EGL

Provides programming "abstractions" that increase productivity and quality

Supports both modern and traditional runtimes

Very easy to learn

Powerful comprehensive language (not your father's 4GL!)

Shields programmers from middleware programming complexity

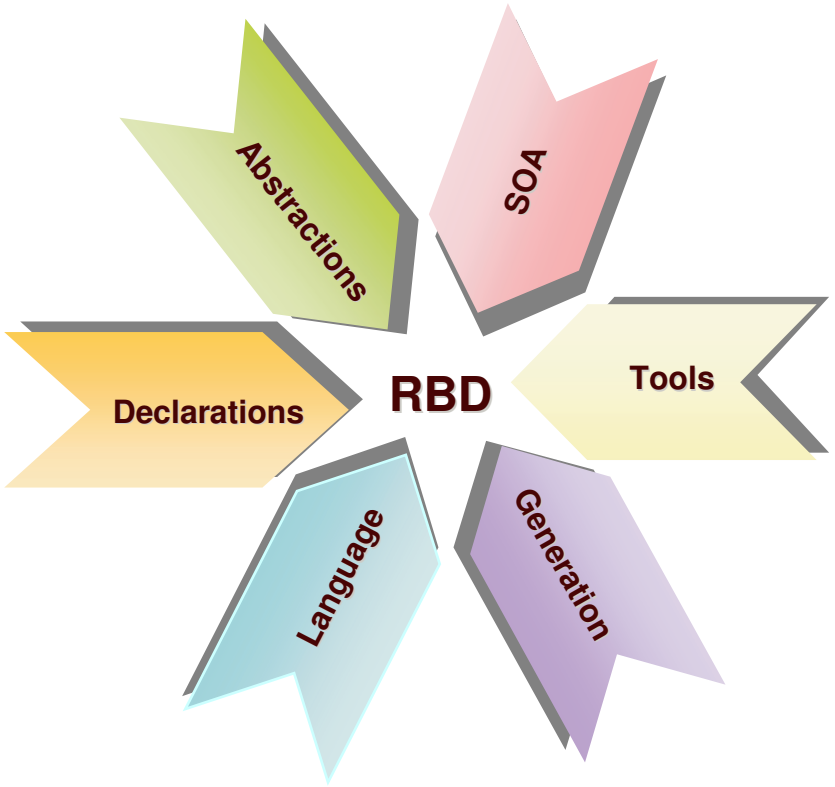
Inter-operates with legacy

EGL Principles

- Decouples application specification from runtimes
- Is built with “extensibility” in mind
- Immediately useable by developers of any background
- Hides technical complexity
- Supports emerging standards and technologies
- Allows optimal (native) deployment to any runtime (new and traditional)
- Ensures easy inter-operability with legacy
- Delivers productivity without compromising flexibility:
  - Language simplicity
  - Language robustness
- Enables agile, iterative development

Rational Business Developer

A simple, robust, unified approach to end-to-end construction of application and services that shields developers from intricacies of runtimes and middleware



Rapidly Deliver…

Modern Innovative Solutions…

With the available Skills

The power of abstractions

Data access:

"Records" provide access to:

SQL, Indexed, Relative, Serial, DL/I, Service data, Message Queues

Common Verbs for data access (**Get, Add, Replace, Delete**).

Allows complete access to SQL statement if needed.

Common Error Handling.

Remote Invocation:

Call COBOL, RPG, C, Java.

Linkage information separated from code.

Data mapping, protocol invocation all resolved at runtime, NO code necessary!

User Interface:

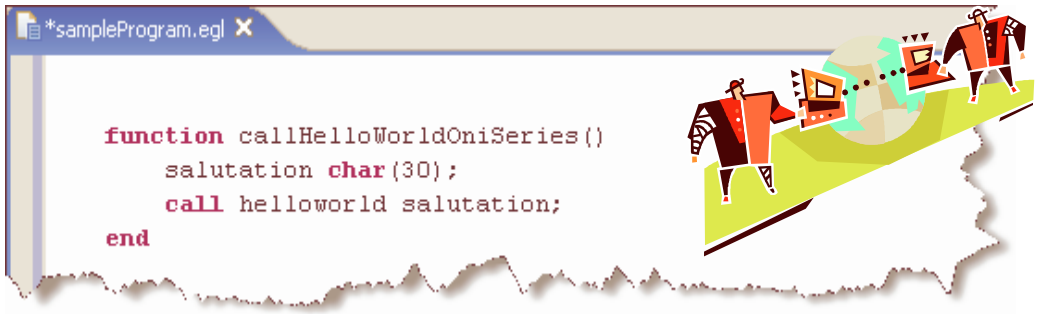
EGL "handlers" hide complexities of

Web (JSF) UI

Reports (BIRT)

Portlets

* Rich Web (Ajax)



* Tech Preview

The power of declarative programming

Validation/Editing Rules:

- Via properties in “Data Items
- Define formatting & validation rules in a common place
- Reuse data items for records, screens, web pages, reports

*sampleProgram.egl

```
DataItem SSN char(9) {
    validatorFunction = "ValidateSSN()",
    displayUse = secret,
    pattern = "XXX-XX-XXXX",
    displayName = "Social Security No",
    inputRequired = yes}
end
```

The power of the Language

Simple and familiar…. yet robust and complete

Rich data types

- Simple (int, string, boolean, etc.) or Complex (any, static arrays, dynamic arrays, dictionaries, array dictionaries, etc.)

Keywords

- Case, if-then-else, while, for loop, for loop cycling through a database result set, etc.

High power language capabilities

- Automated Casting (e.g. using AS operator)
- Mixing data types in assignments and expressions
- Exception handling

Rich libraries of built-in functions

- Math, string, date/time, system, i5/OS resources (Data Areas, Data Queues) access, …

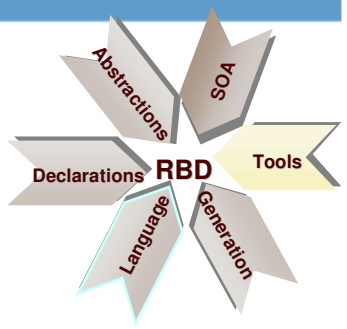
Robust integration with existing investment or access to low level APIs

- Call RPG, COBOL, C, etc.
- Full Java interoperability
  - Invoke Java from EGL (map Java classes with EGL External Types)
  - Invoke EGL from Java

hello.egl

```
1  // Hello World basic program
2
3  program hello type BasicProgram
4
5      // Data Declarations
6      name string = "World";
7
8      function main()
9          writeStdOut("Hello " +name);
10     end
11
12 end
13
```

The power of tools: Robust Page Design



First Class integration with Page Designer and JSF tools

- Drop EGL data structures on JSP:
  - Validation, editing, formatting rules from EGL Data Items applied.
  - Appropriate UI controls rendered pre-bound to data declared in EGL Page
- Server-side event handlers in EGL within context of page designer

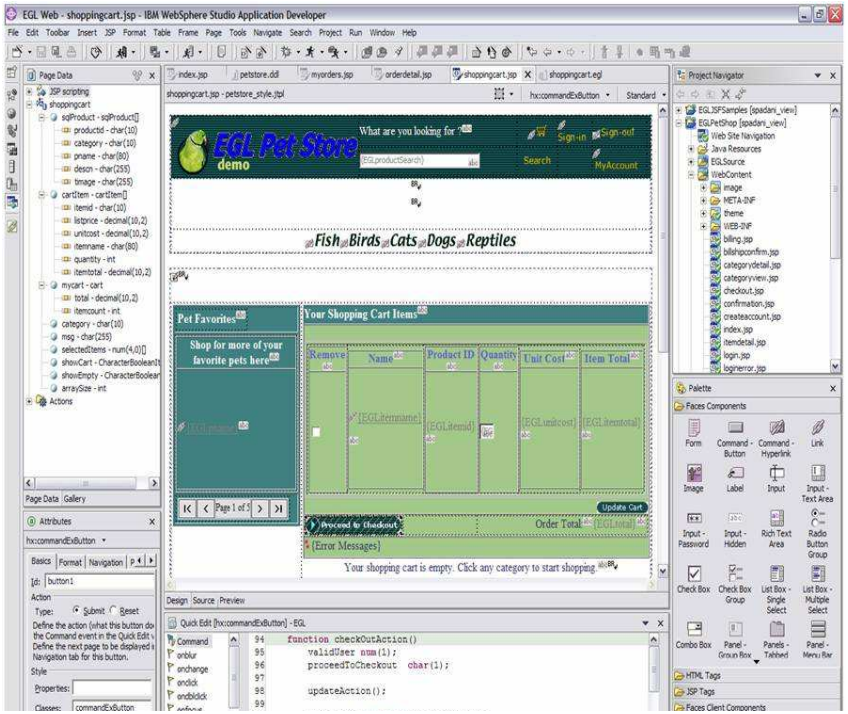
Integration is totally seamless.

No Java coding required to wire EGL data to JSF.

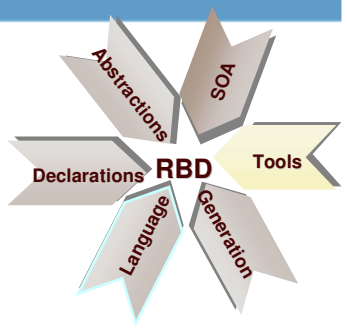
EGL logic can be used to handle user interaction with the JSP.

AJAX capability built-in…partial refresh, etc…

Portlet support



The power of tools: Debugger



Debug entire application regardless of ultimate deployment targets:

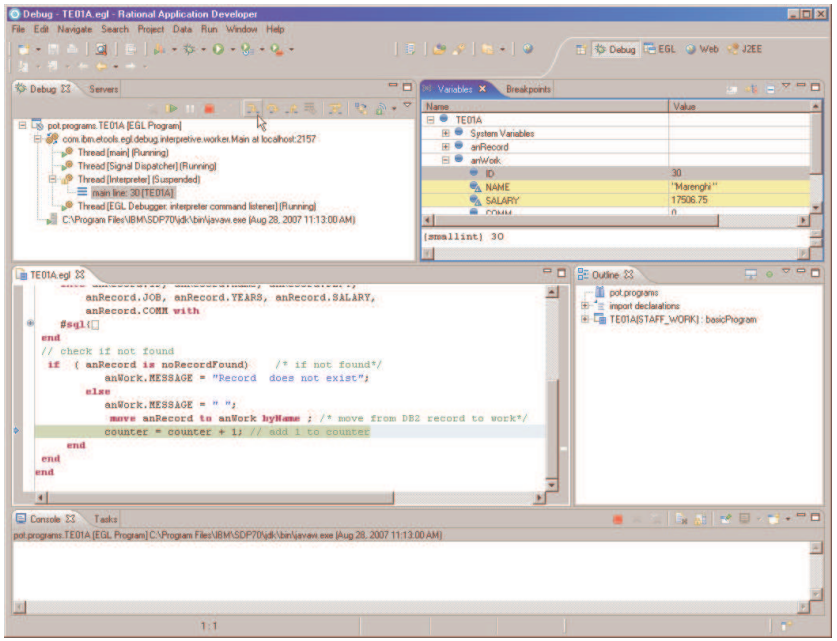
- Transition from debugging JSP's to EGL code to Java to … and back.

EGL source debugger:

- Breakpoints
- View and modify variables
- Jump-to line
- Hot-swapping
- Extends base Eclipse debugger

Remote data access

- Relational DB, VSAM files, DL/I data, DQ, DA

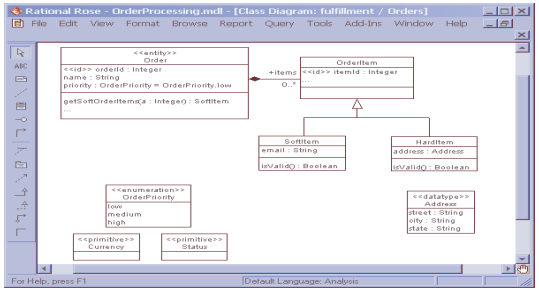


Great debugger = great productivity!

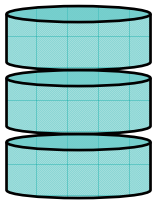
The power of tools

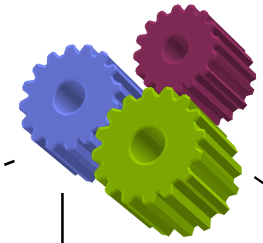
Model Driven EGL Development

UML Class Diagram

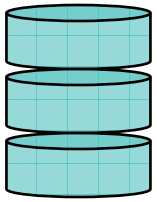


DB Schema

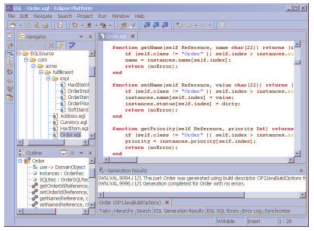




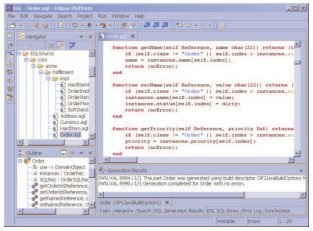
Transformation Engine



DB Schema



EGL Records and Items



EGL Functions
Libraries or Services

JSF Pages

The power of Services

Built into the language

Service part:

a generatable part containing code that will be accessed:

from EGL code via local or TCP/IP connection (*EGL Service*)
from any code via HTTP connection (*EGL Web service*)

```
customerService.egl
// service
Service CustomerService
        Function getCustomer(custid String) returns (string)
//          ...
        end
//      ...
end
```

Interface part:

Used to access external Web services as EGL services

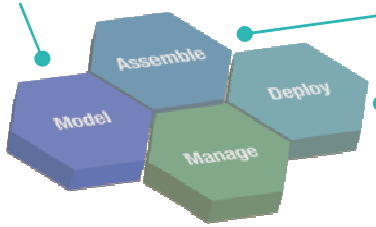
WSDL drives automated generation of EGL artifacts
EGL takes care of XML to EGL dynamic conversion
No need to master SOA protocols and technologies

```
customerService.egl  creditCheck.egl
// interface
Interface creditCheck

    function checkCredit(SSAN string in) returns (string);
//  ...
end
```

At development time…

- Focus on the business logic
- Implement SOA design elements: services and interfaces
- Leverage existing business developers for new SOA development
- Ignore deployment targets/technology while coding/testing



Leverage external web services…

- EGL Interfaces
  - represent external web services
  - are created via import from WSDL
  - allow the EGL developer to stay within the context of EGL programming

Deploy EGL services…

To any platform (Java to WAS/Tomcat, COBOL to CICS, i5/OS)

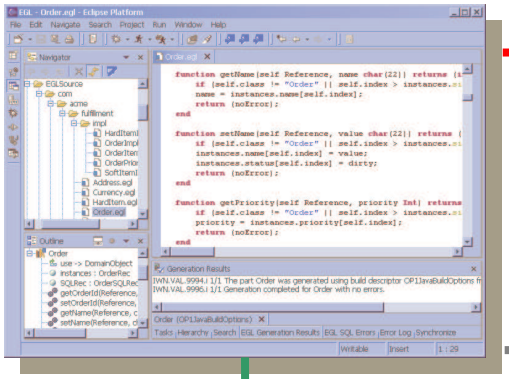
- As a Web service (uses SOAP)
- As a private service (uses CICS ECI or TCP)

The power of Generation

Platform Flexibility

ClearCase, CVS, Other

EGL program/library/service



System z

- WebSphere
- USS
- Linux
- Batch
- CICS
- IMS

COBOL

Windows, Linux, Unix

- WebSphere
- Tomcat
- Native

System i

- WebSphere
- Tomcat
- I5/OS Integrates App Server
- Native i5/OS
- Native i5/OS

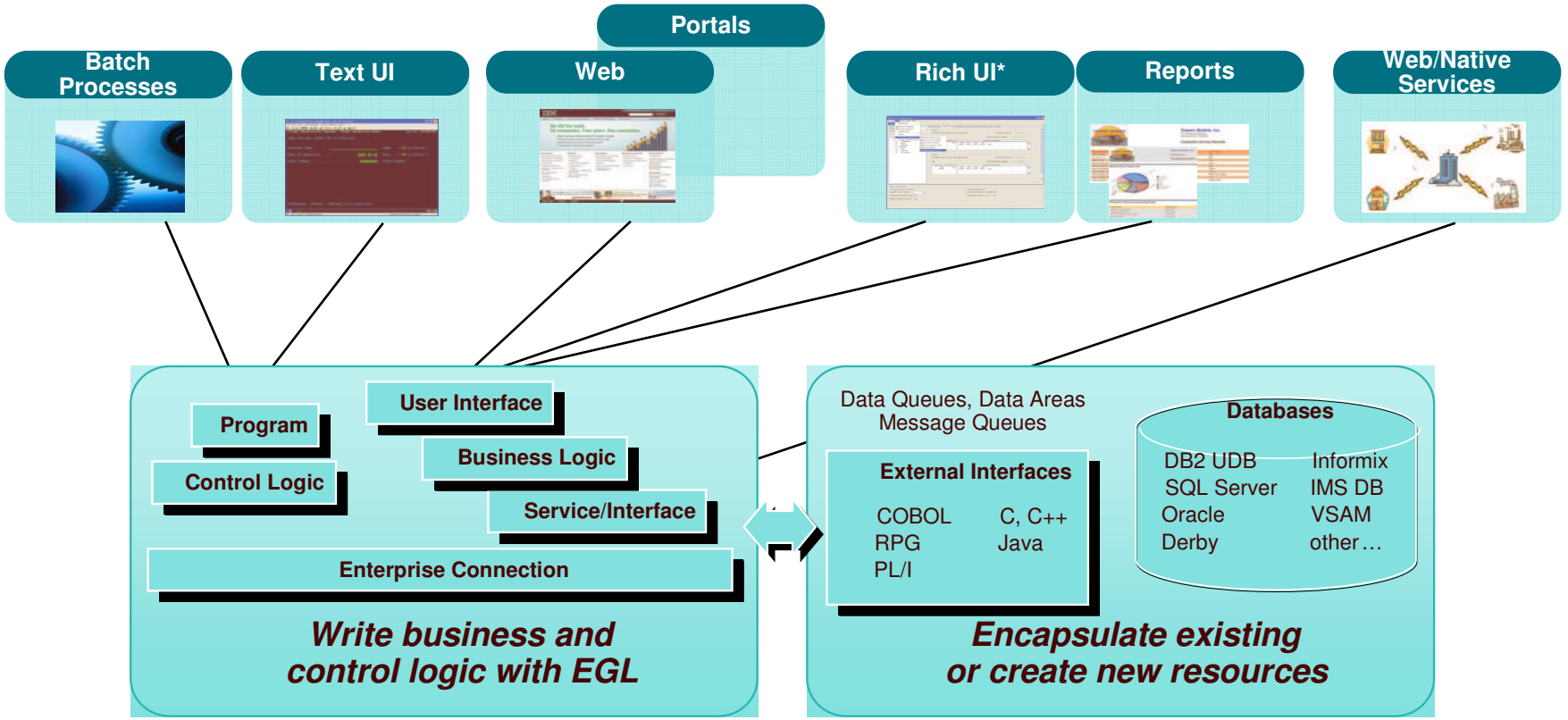
COBOL

JavaScript

Browser*

- IE*
- Firefox*
- Safari*

Application flexibility



Rich Internet Applications

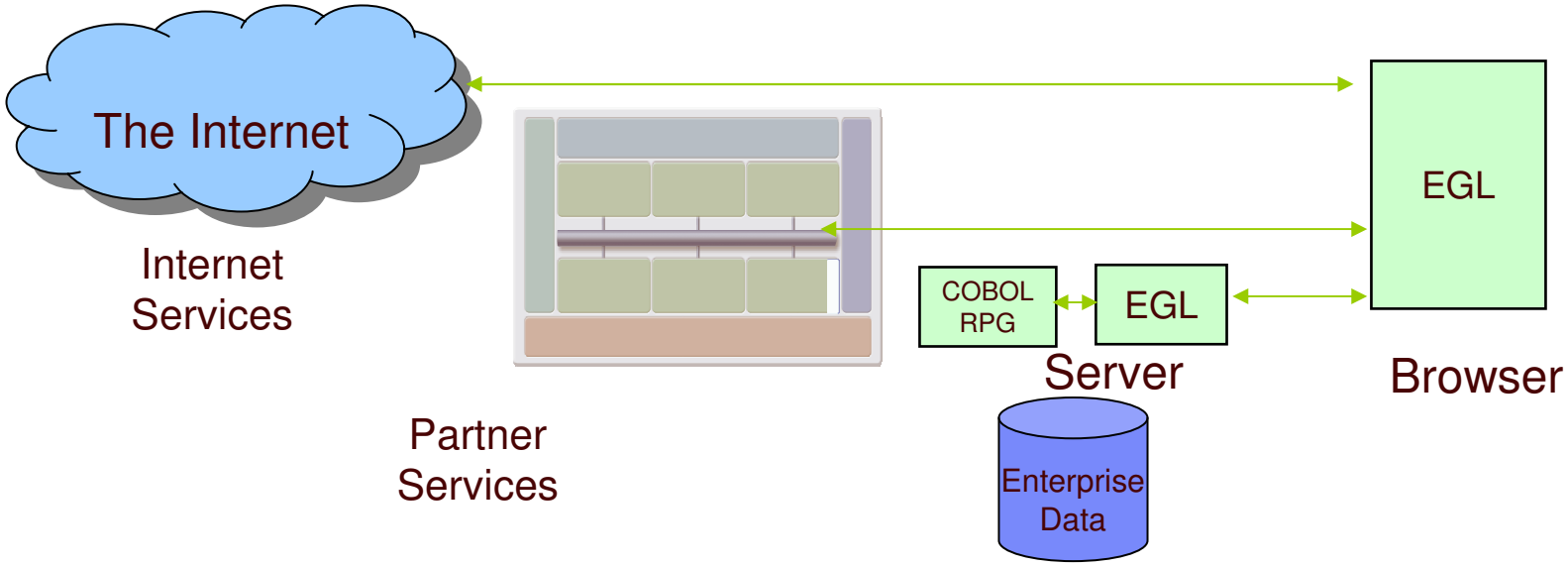
Why RIA?

- Applications are responsive and user friendly (user productivity)
- Natural paradigm for leveraging internet resources
- Get back to Web architecture basics (REST)
  - Browser keeps state, Server is stateless thus….
  - Better performance/response (caching)
  - Frees up server CPU and Memory, increase scalability
  - Less complex server-side stack, lower software and administrative costs

Why is it hard to create RIA?

- Need to know many technologies
  - JavaScript, HTML, CSS, Ajax, Dojo, SilverLight, SOAP, Flex, XML, JSON, PHP/C#/Java
  - Less App Server control = more application responsibility managing multi-user transactions
  - Required technologies operate on a very detailed, low level
  - These technologies use different metaphors and abstractions, hard to mix and match
  - Require highly skilled adaptable software hackers (not typical of traditional business developers)

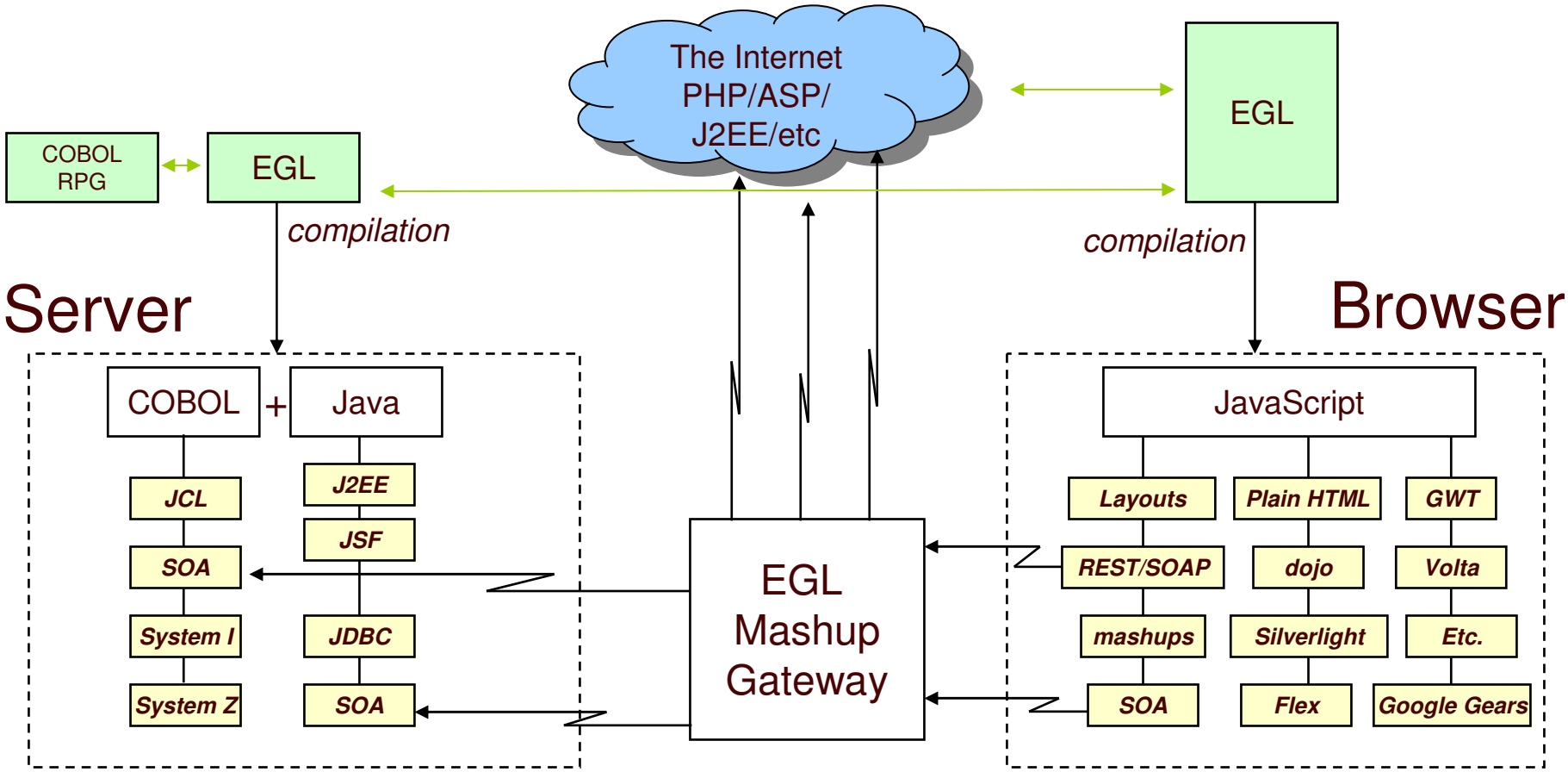
Simplifying Innovation: EGL Web2.0 Support*



- One language end-to-end: EGL
- One unifying IDE: RBD
- Mobility of Skills
- Innovative Editing Approach

* Tech Preview : http://www.alphaworks.ibm.com/tech/reglrws

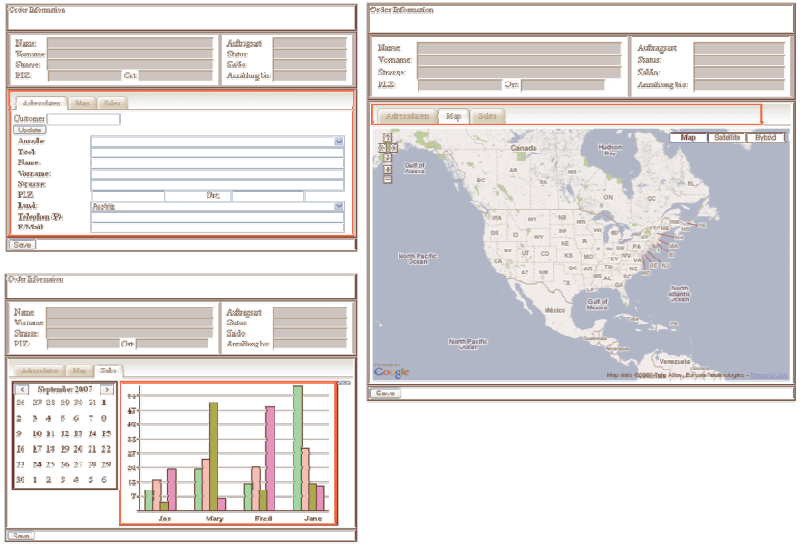
EGL Web2.0 Support Architecture



EGL Rich Web Support Highlights

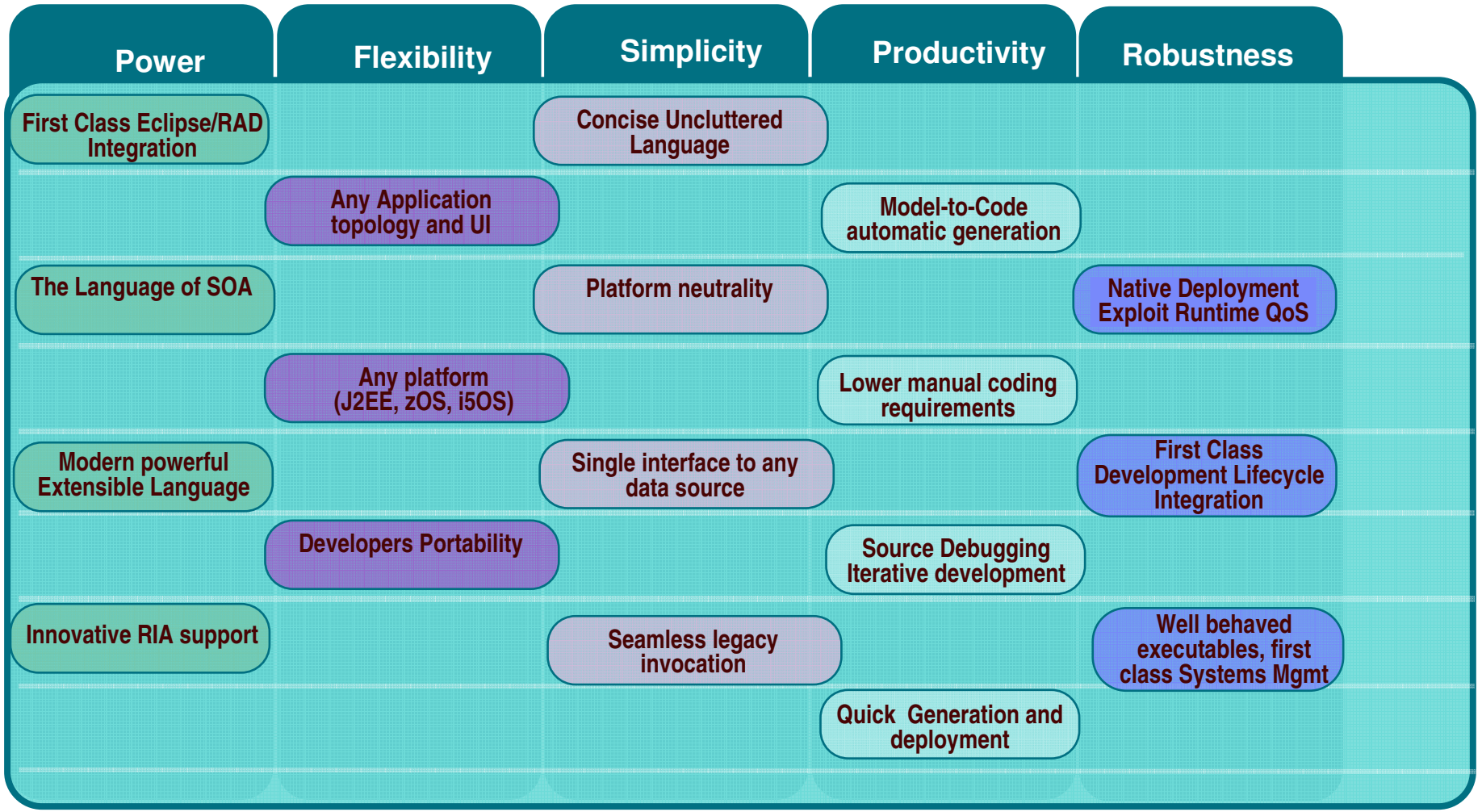
New EGL notation and tools to support development of Rich Internet Applications

- Consistent with EGL philosophy, hides all Ajax complexities with a higher level abstraction
- Includes WYSWIG and scripted composition tools, and libraries of Rich UI widgets (EGL rendering of DOJO, YUI, GWT, EXT, etc)
- Fully open and extensible: JavaScript experts can easily create EGL rendering of any JS widget and enrich the library
- Easily integrate/consume any service (REST or SOAP, external or created in EGL)
- Single language end-to-end, simple and productive to deliver RIA-to-mainframe solutions
- Combined with EGL legacy interoperability easily extends legacy to Web 2.0



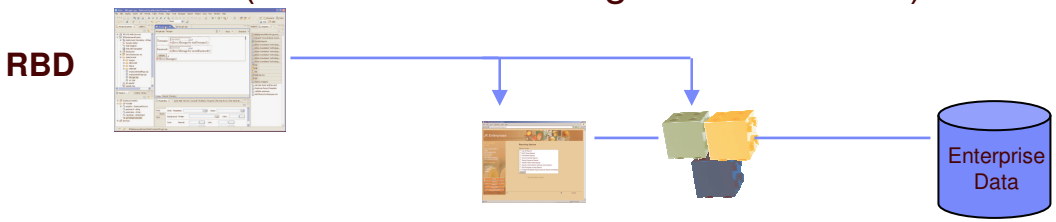
Check out the RSDC Personal Planner for iPhone written entirely in EGL RUI

The RBD Difference

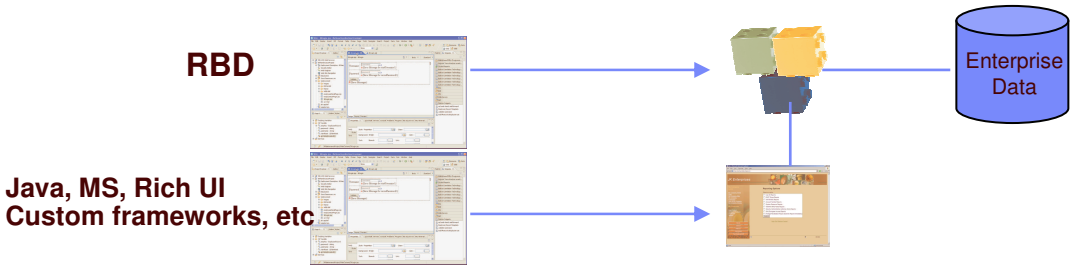


EGL Use Cases: Developing new Systems

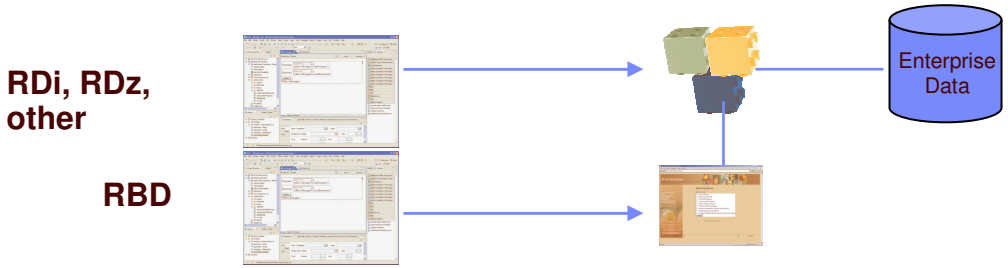
EGL end-to-end (web-to-backend logic and services)



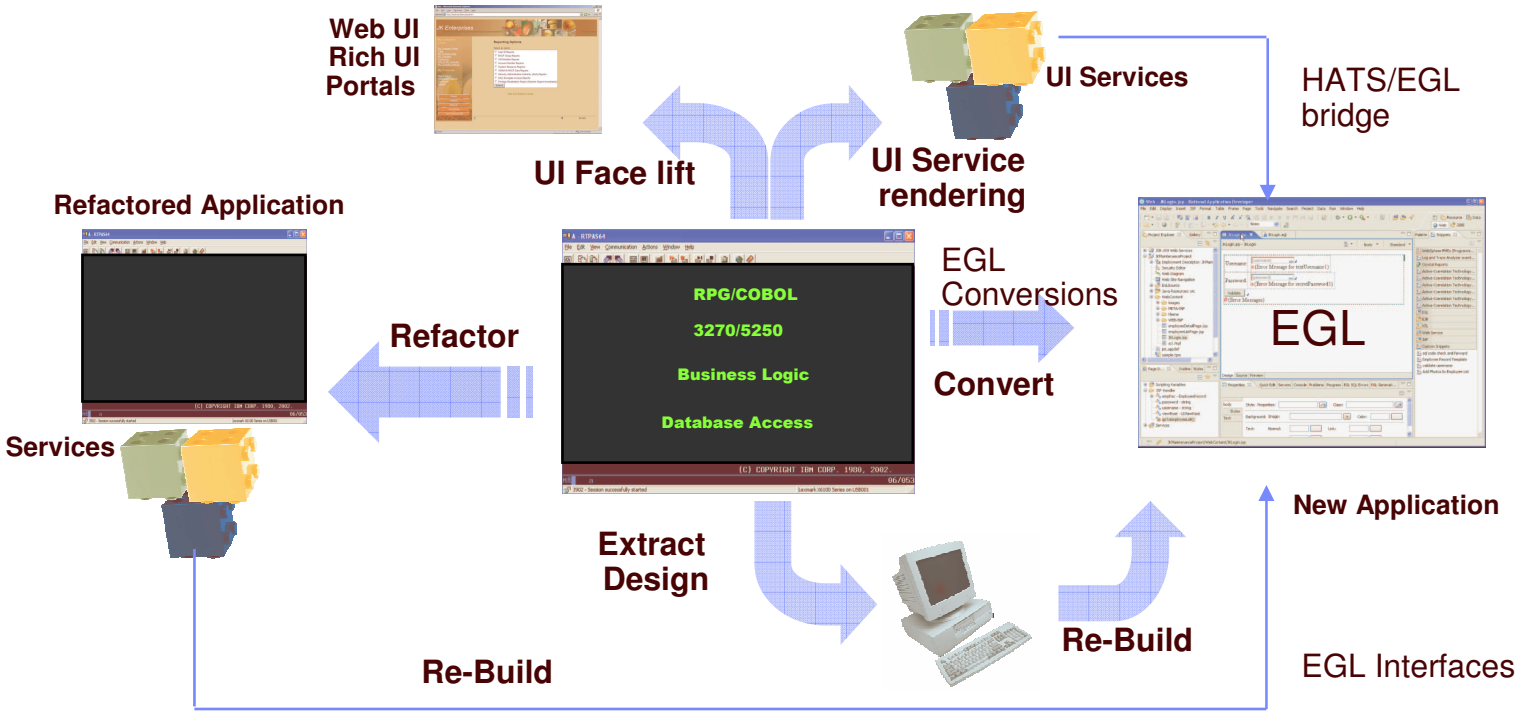
EGL as a SOA lingua franca for enterprise business services



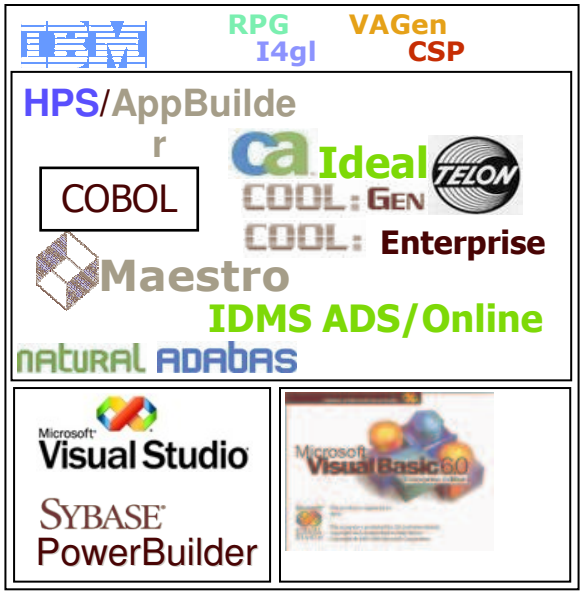
EGL front-end to COBOL or PL/I backend

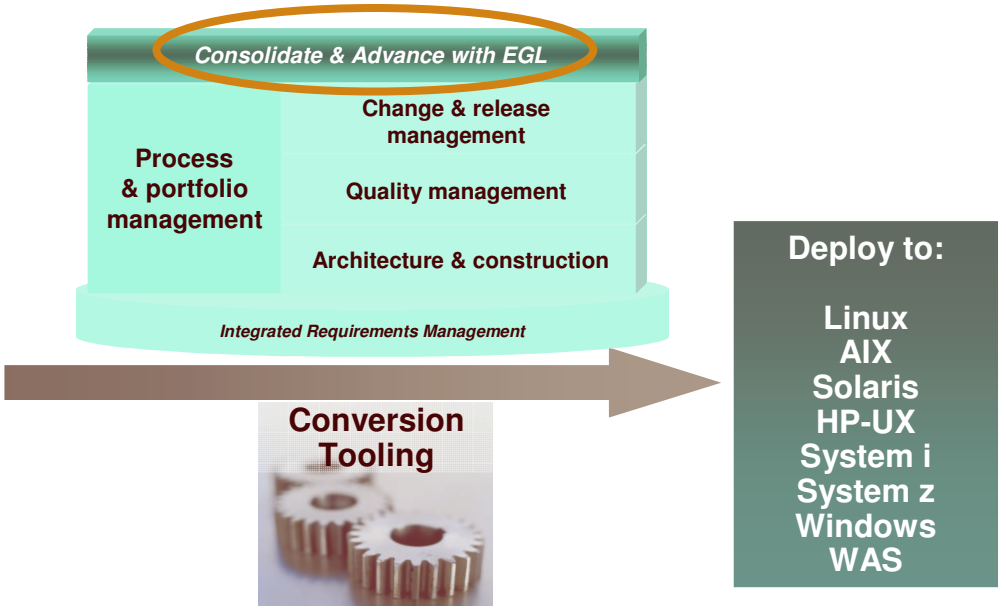


EGL Use Cases: Modernizing existing Systems



EGL Conversions





FBD (Natural/ADABAS to EGL DB2)

Xact (COBOL/CA:Gen/CA Ideal/CA Telon/Maestro to EGL

PKS, Databorough (RPG to EGL)

Synchrony Systems Inc. for TUI to Web transformation

Accelerating Enterprise Modernization

Rapid cross-platform Web and SOA development

Rapidly integrate existing mainframe assets into service oriented solutions

Leverage trusted mainframes quality of service

Reduce mainframe and server maintenance and new development cost

Break Skills Silos

Maximize use of programming resources and business domain knowledge

Reduce retraining costs

Simplify cross platform development

Respond to business needs independent of underlying technology implementation

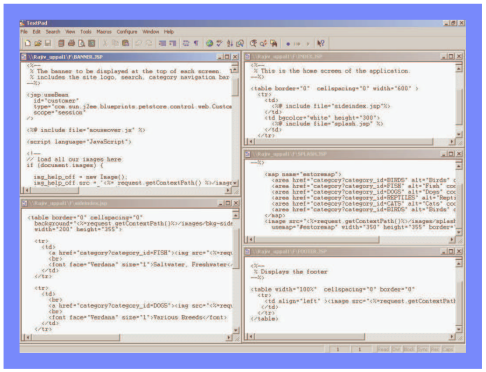
Improved quality and architectural integrity

Leverage IBM Rational's commitment to open technology and modularity

Rational Business Developer is the right solution tailored to specific mainframe and cross-platform needs

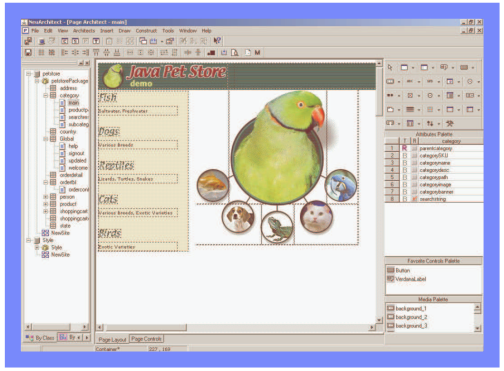
Accelerating Application Delivery

Hand-Coded



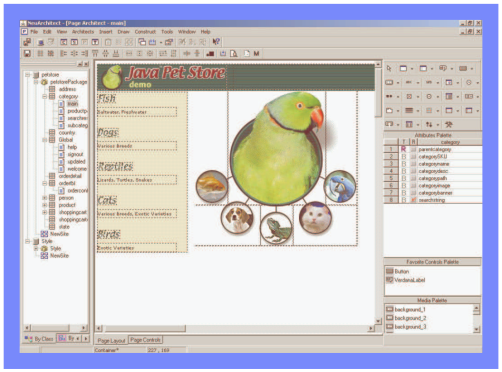
507 Hours

Java/J2EE IDE



330 Hours

RBD



60 Hours

- RBD is **dramatically faster** than traditional development*
- RBD is more productive than MS VS 2005**

* *Internal benchmark using Sun PetStore application*

** *Branham Study April 2008*

Nordisk Copyright Bureau

Deliver copyright registration system over the web

Challenge:

Registration of copyrighted material in databases was a manual, cumbersome process

Slow, inefficient reporting copyright information to Producer's inquiries

Solution:

Allow customers to get info through a web portal (reduce manual data entry and costs)

Results:

Total processing time for new registrations dropped significantly

EGL's short learning curve, high level of abstraction, and automatic code generation resulted in substantial productivity gains, time savings, and on-schedule project completion.

"A team of three developers completed WebCover application development on schedule, in just three months. If we had used another approach and not JSF/EGL it would have taken much longer than we would like."

- Stephan Kristensen, project leader NCB

Community Health and Counseling Services

Deliver Prior Authorization System over the web

Challenge:

Respond to government plans for behavioral health managed care system within regulatory compliance demands

Requirement to track prior authorization for patient visits and integrate it with electronic medical record (EMR)

No Web development experience nor Java J2/EE skills (only RPG skills)

Results:

Learned EGL in less than a month

Developed and delivered web based prior authorization tracking system in less than 3 months

No need for external resources and total self sufficiency in responding to new application requirements

> *"It would have been impossible to meet our deadline if we had to learn Java™ before developing the application."*
>
> *- Valery Levy MIS Manager CHCS*

Morpheus

Transform a manual/paper process into a web system

MORPHEUS eBusiness Solutions

Challenge:

The manual, paper based insurance quote process for major automobile dealerships clients (BMW, Renault, Peugeot, etc.) was becoming a competitive handicap.

Solution:

A web based e-business application, providing direct access from dealerships to Insurance systems, to rapidly gather accurate quotes.

Results:

EGL's short learning curve, high level of abstraction, and automatic code generation enabled Morpheus, a System Integrator based in the UK, to leverage developers of different backgrounds in delivering the system to the Client in record time (less than 100 days!).

The new system virtually eliminated costly errors, and delivered quotes in record time, with great customer satisfaction.

"EGL allowed us to staff the application project with developers of different skills and deliver the system in just 100 days!! We are looking forward to our next project…"

- Bleddyn Wiliams, Director of eBusiness Solutions

KBC

Unify application dev across platforms and transaction managers

Challenge:

High costs and low responsiveness to business: redundancy and duplication across IT groups, fragmentation of languages and tools

Solution:

Leverage EGL to achieve the Cross-System components development and to eliminate skills silos among their dispersed development teams. Eliminate other platform specific languages

Results:

Created interchangeable developers. Shift from monolithic (3270) applications to browser based and open systems. Transform 600 mainframe developers to multi platform developers (Unix, mainframe, WAS, IMS).

Enable component based architecture. Shift to component based architecture, product factories and multi channel.

"… We want to avoid the 'skill silos,' what we really need is a large group of general developers who should not worry about target platforms and focus on developing business components, and only a small number of technology specialists, so that we can swiftly allocate general developers to upcoming business needs… EGL is helping us achieve this goal…"

- Lieven Gouwy, IT Architect, KBC, Redmonk Podcast

EGL Momentum

Hundreds of new EGL users, growing network of EGL Business Partners

Headlines


CIOZone
Network for IT Leadership
Change your HTTP MPH. ASAP.
at&t
Change your game. With AT&T's new Digital Media Solutions.
Home
Who Can Join?
Browse Members
Blogs
Forums
Groups
Featured Bloggers
About Us
search...
Login, Register or Get Started
Username
Password
Remember me
Login
Lost Password?
No account yet? Register
Featured Member
Dan Moen
How To Be A Better Leader
Ballmer becomes lone voice at Microsoft's helm
ASUG's Masney On SAP's Direction, Customer Care
How To Manage Multigenerational Workforce
News & N
HP purchase government
Ballmer bec voice at Mic
Tech majors against pate report
Samsung SDI million to up
How To Be A Better Leader
Stewart Friedman, founding director of the Wharton School's Leadership
the key to being an effective and satisfied leader is being successful in a
life: work, home, community and self. Here's his guide for becoming a to
Also See:
Why Followers Matter
4 Ways To Better Manage, Motivate The Team
Be first to comment this article
What Is EGL And Why Should CIOs Care?
If you have a staff chock-full of object-oriented programmers, you shoul
use Java for your Web development. On the other hand, if your staff is n
comfortable with Cobol or RPG, there's an alternative—IBM's Enterprise
Language.
Also See:
50 Top Open Source Resources CIOs Should Know (And Maybe Love)
Be first to comment this article
Business & Tech Leadership Forum
Sept. 23–24, 2008
REGISTER TODAY!
FORRESTER
CIOZone Highlights
Collaborative Case Studies
Research
IT-Director.com
TOTAL IMMERSION
IP address management - a latent need, not a market bandwagon
NORFOLK PUNT
Why oh why oh why oh (RSDC Fytte 3)
ARCHIVE
PAPERS
RESEARCH
EVENTS
NEWSWIRE
Member Login | Become a Member
BLOGS > THE NORFOLK PUNT
IBM RSDC - 2nd Fytte
By: David Norfolk, Practice Leader - Development, Bloor Research
Published: 5th June 2008
Copyright Bloor Research © 2008
So, I've got up at 6 for the analyst breakfast, after having heard Bob
son play at the RSDC Telelogic Welcome Party last night (he's not a pa
his Dad IMHO)—but what did I see at RSDC that I didn't expect to? Th
always the icing on the cake at these things, for me.
Well, the IBM Labs section, showcasing interesting research projects
Jazz and Eclipse platforms that may or may not make it into product,
great fun.
IBM Rational Business Developer was one such, using EGL (Enterprise
Generation Language) to generate client-side Javascript applications f
running in a browser.
Very slick and easy, but it's EGL that reallly interested me as an ex IBI
OS370 Assembler programmer, I was horrified at the return to
"assembler" (with C++) in the 1990s. My favorite language was then R
Now, EGL may signal a return to higher level languages. You write and
your EGL code and then generate the programme for 1 or many target
systems. So you write one piece of code and can produce Java, COBO
solutions ad lib. See this book for more details.
RSS Feeds
Add headlines to your personalized My Yahoo! Page
(About My Yahoo! and RSS)
YAHOO! FINANC
HOME
INVESTING
TRADEKING
$4.95 A TRADE | JOIN TODAY
Press Release
Report: IBM Outpaces Competitors in Application Development Software Market for Seventh Straight Year
Friday June 13, 8:00 am ET
ARMONK, NY--(MARKET WIRE)--Jun 13, 2008 -- IBM (NYSE:IBM - News) today announced that analyst firm Gartner, Inc.* and market research firm Evans Data Corp. have ranked IBM as the leader in the application development software market. These rankings come just as IBM is projecting more than 12,000 people will attend its 2008 IBM Rational Software Development Conferences in 13 countries around the globe.
For conference attendees using an iPhone, IBM is releasing a conference scheduler written in Enterprise Generation Language (EGL) to enable iPhone users to dynamically experience the IBM Rational Software Development Conference through an interface that they feel comfortable with. Using Web 2.0 and social engineering concepts, users can provide feedback on and chat about sessions, navigate the conference searching for tracks and events, and use inventive

Wisdom of the Crowd – EGL Cafe

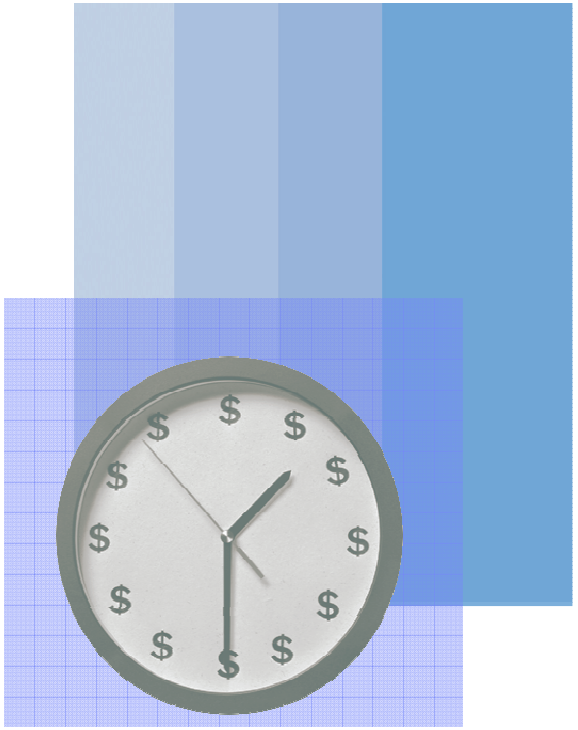
- Join
- Download
  - product trials
  - Sample code
- Learn
  - Documentation Corner
  - Jon's Corner – Learn
  - Articles, Books
  - About Products
  - Sandbox
- Participate
  - Forums
  - Blogs
  - Ratings
  - Reviews
  - Share code
- Partners
- GoingsOn

Summary

IBM newest business language, key to IBM Rational Enterprise Modernization strategy

RBD is a powerful solution that helps

- Break the skill silos and leverage business know-how of all developers.
- Build innovative state-of-the-art solutions without the bite of skill ramp
  - Lower risk and increase project success rate.
  - Lower costs of training.
  - Lower Cost of development.
- Deliver Projects faster



For More information

External RBD Web Site (www.ibm.com).

http://www-306.ibm.com/software/awdtools/developer/business/

EGL Café

http://www.ibm.com/rational/eglcafe

RBD/EGL zone on developerWorks

http://www-128.ibm.com/developerworks/rational/products/rbde/


Thank
You