

How To Apply Software Archeology To Your Development Process

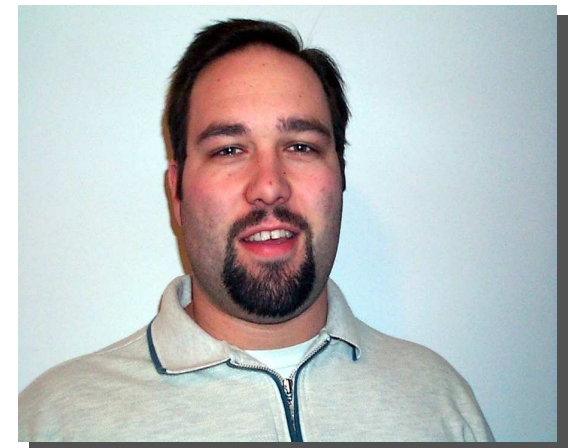
Presented for: EclipseCon 2008

Thursday, March 13, 2008

- Michael Rozlog – michael.rozlog@codegear.com
- CodeGear

Who is Mike?

- You just said that
 - PLSM (Product Line Sales Manager)
- Responsibilities
 - Helping rebuild the developer tools and databases
- Left Borland 1/2006
- Why did I come back?
- Prior
 - Chief Technical Architect – Borland
 - Creator of Mastering JBuilder – Wiley Books



michael.rozlog@codegear.com

Ground rules for presentation

- All questions are valid
- This is NOT a sales presentation
- If you have a question, ask... if you have it most likely other members of the audience has it to...

Agenda

- The process: What is Software Archeology?
- Future tools: What are Software Factories?
- CodeGear: Company background
- Q&A

The process

- What is Software Archeology?

Current state of IT in the U.S. and around the world

- The need for new or modernizing applications is not decreasing or going away
- U.S. had the lowest C/S graduation in 2007
 - Qualified individuals are not filling the market
- Outsourcing is not always a solution on the table?
 - Not allowed or applicable
 - Bigger issues around this area, latest data shows 50% for non-government organizations
- Knowledge transfer gap

What is Software Archeology?

- A process for approaching unknown software that we have become responsible for...
- An approach to unraveling the complexities of an existing application
- A lightweight process for getting understanding of existing source code
- The analysis of past applications to learn and understand why it was done that way

Software Archeology



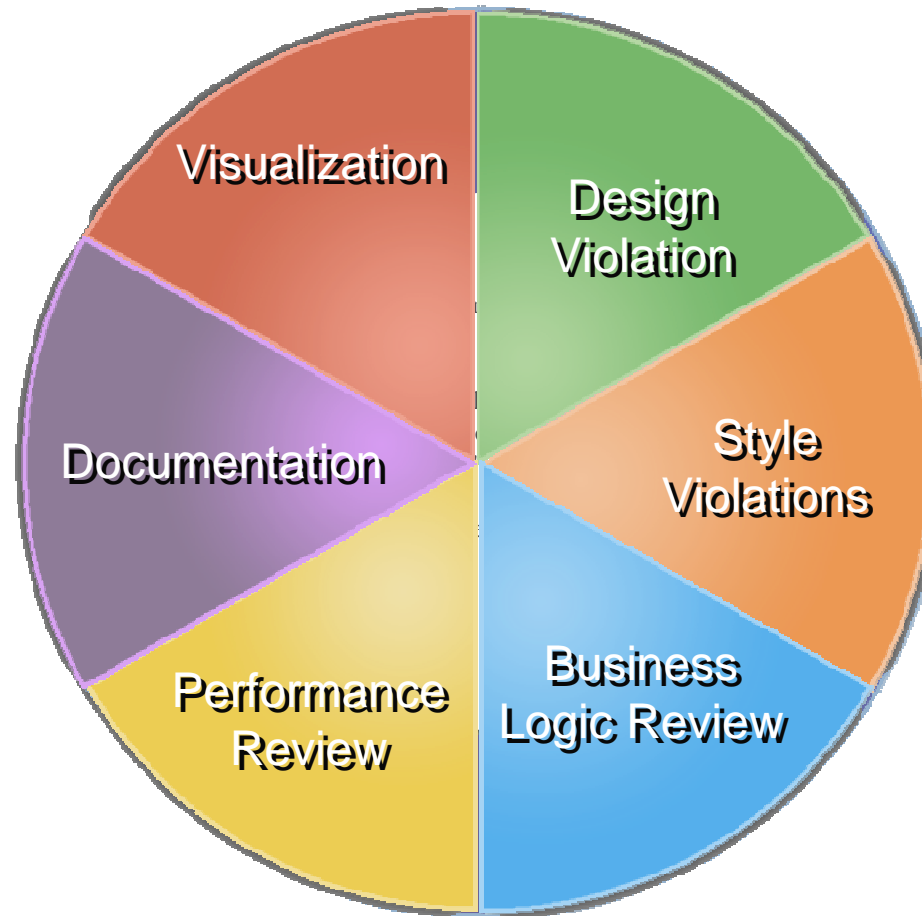
Software Archeology



Software Archeology



Software Archeology



Why is Software Archeology important?

- Always asked to rebuild the wheel
 - How many times have you been asked to recreate this current implementation, but better?
- Inheritance of code
- Outsourcing
- Graying of IT

Why is Software Archeology important?

- Always asked to rebuild the wheel
- Inheritance of code
 - How many times have you picked up someone else's code or project... to be enhanced?
- Outsourcing
- Graying of IT

Why is Software Archeology important?

- Always asked to rebuild the wheel
- Inheritance of code
- Outsourcing
 - Projects get outsourced, rural sourced, in-sourced, local sourced, and every other kind of sourced...
 - Do you understand the systems that were created?
 - Do the outsourcers understand the system they are inheriting?
- Graying of IT

Why is Software Archeology important?

- Always asked to rebuild the wheel
- Inheritance of code
- Outsourcing
- Graying of IT
 - By 2010... almost 37% of the software industry will consist of persons older than 50!
 - 60% of the federal workforce will be eligible to retire over the next 10 years
 - By 2010... 18.5 percent of the government workforce – about 290,000 employees – will retire
 - Do you understand the systems they have created?

What can be learned?

- High-level architectural understanding
- Weak-points in existing software
 - Maintenance nightmares
 - Performance bottlenecks
- Standards compliance or lack there of...
- The overall health and value of the application
- Architectural approaches or how not to solve a problem
- Finding and Harvesting of Algorithms and patterns

What tools are needed?

- Need to be able to get a picture
- Need to be able to do static analysis
- Need to be able to do Dynamic analysis
- Need to be able to test the system
- Need to evaluate the performance
- Can it be done by hand?
 - Yes... but very time consuming... code reviews, system tests, and primitive system performance models
- Is there a better way?
 - Use a software tool

What is the cost of code reviews



Skip AppFactories
for Now – Time Limited

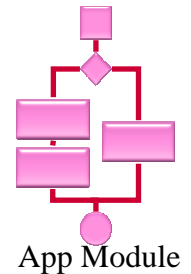
- Very controversial subject
- But for me tools can enhance the approach
- **Task**
 - Review by hand a small example set of code
 - 5 classes, 139 lines of code, 4 “experts” in 15 minutes
 - Average audit violations detected: 21 time: 0.25 hr
- **Comparison**
 - The same set of code analyzed by a machine:
 - Audit violations detected: 150 time: 2 sec
 - Humans @ 84/hr
- **Real “sample” Project**
 - 187,600 violations in 100 minutes on a project with 1233 classes and 199,146 lines of code
 - **32,443** violations using the default set
 - Actual customer case studies show **~400 man/hour savings** per code review

Future tools:

- What are Software Factories?

What is an AssetLibrary™

- A repository that can hold and manage AppModules.
- It will manage the lifecycle of AppModules



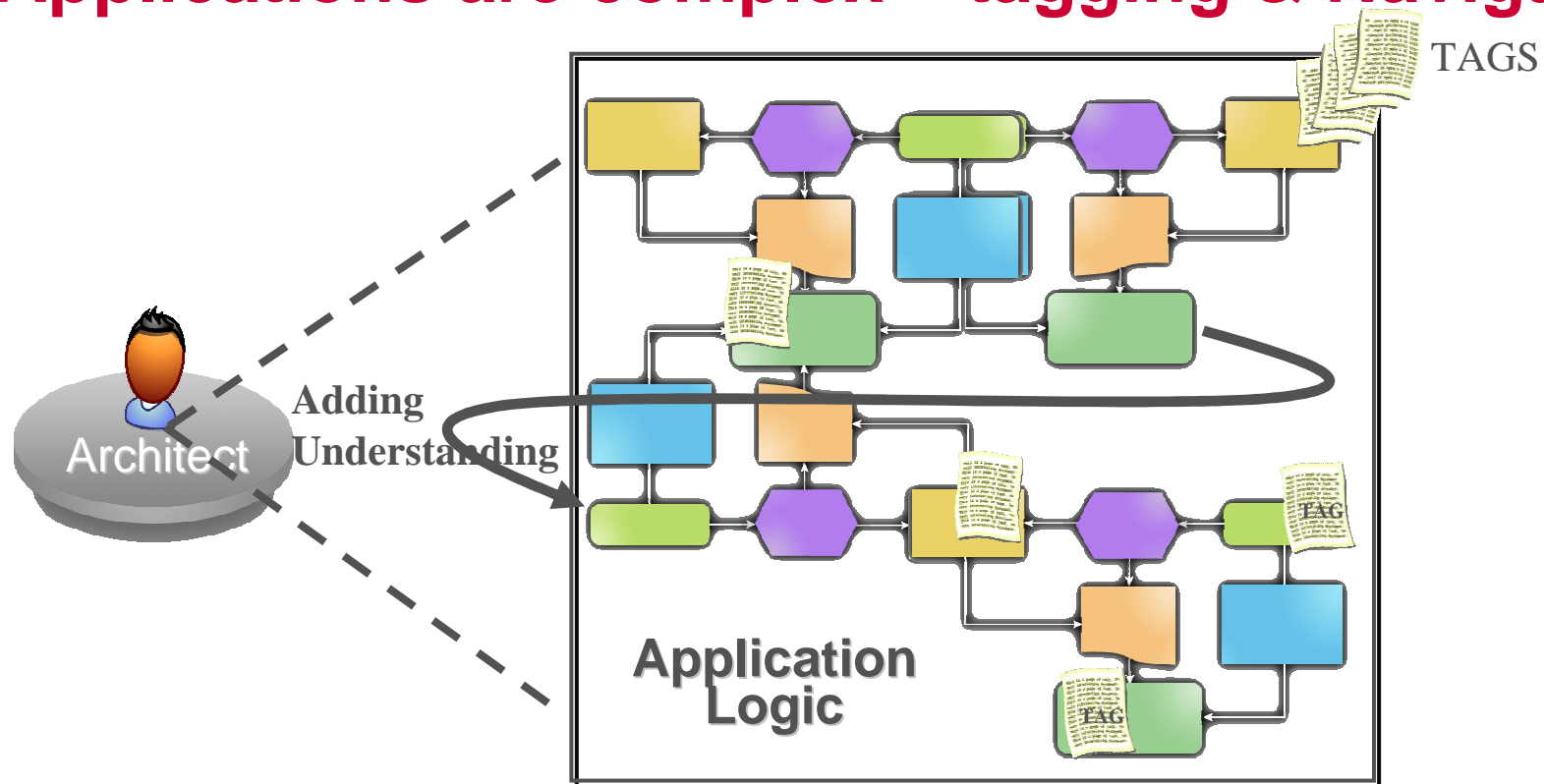
AssetLibrary™

How do Application Factories work?

- Let's walk through a simple scenario

Instant Understanding...

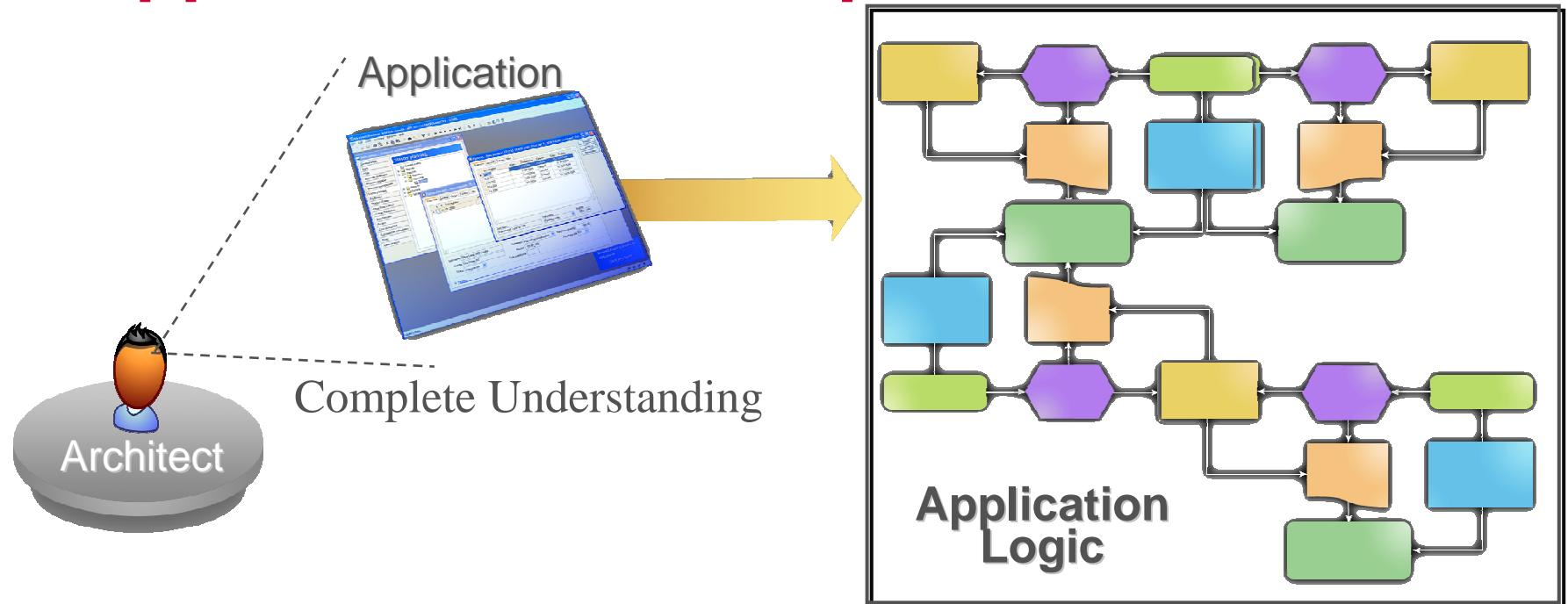
Applications are complex – tagging & Navigation



- Almost instant productivity gains and better understanding
- Increase base understanding through the use of tagging
- Understand complex navigational structures of an application

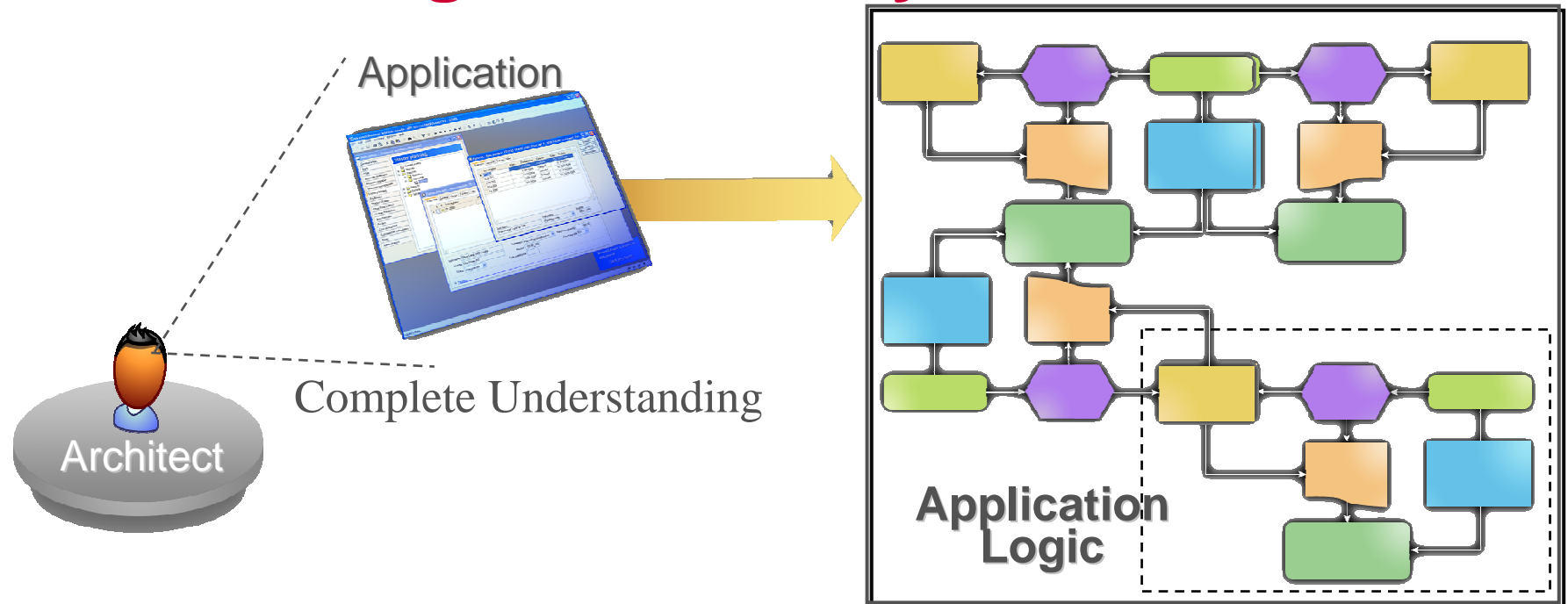
Produce process

Applications are complex



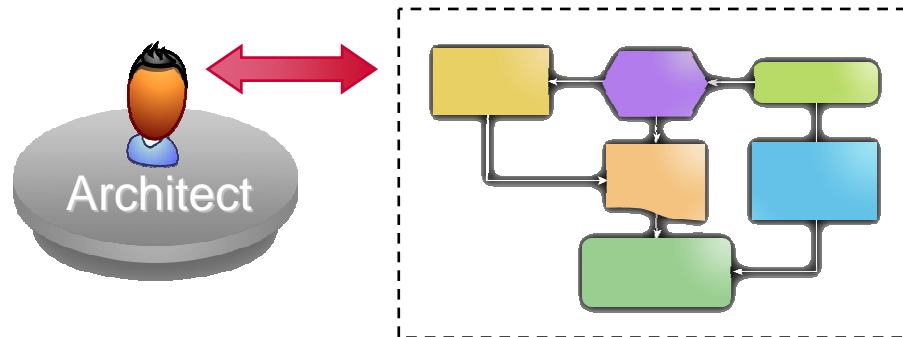
- Architects have a complete implementation and conceptual understanding of an application
- They also understand the intent and evolution of the application
- However explaining, extending, and reusing parts of the application is near impossible

Harvesting functionality

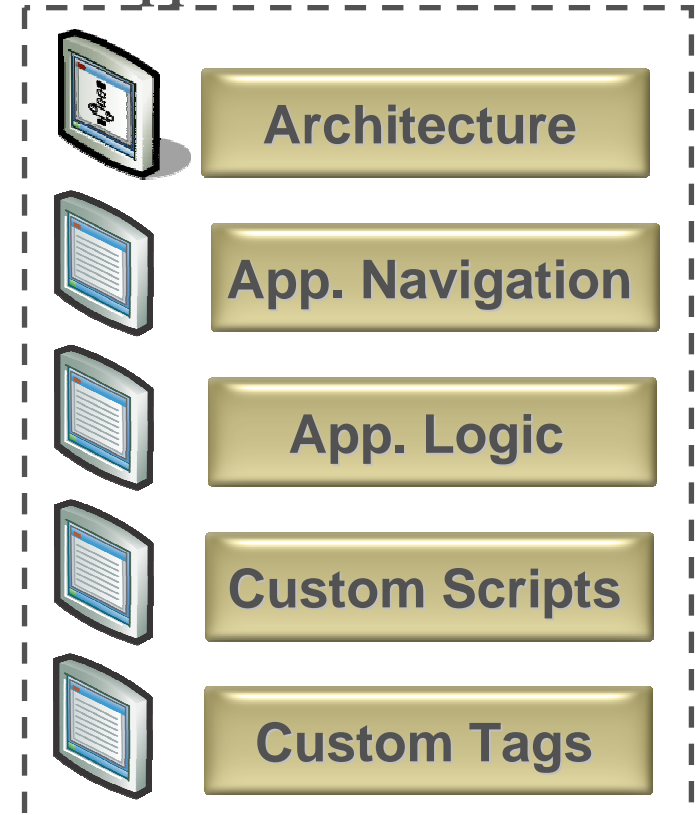


- Isolate key functionality or components of an application

Application Module

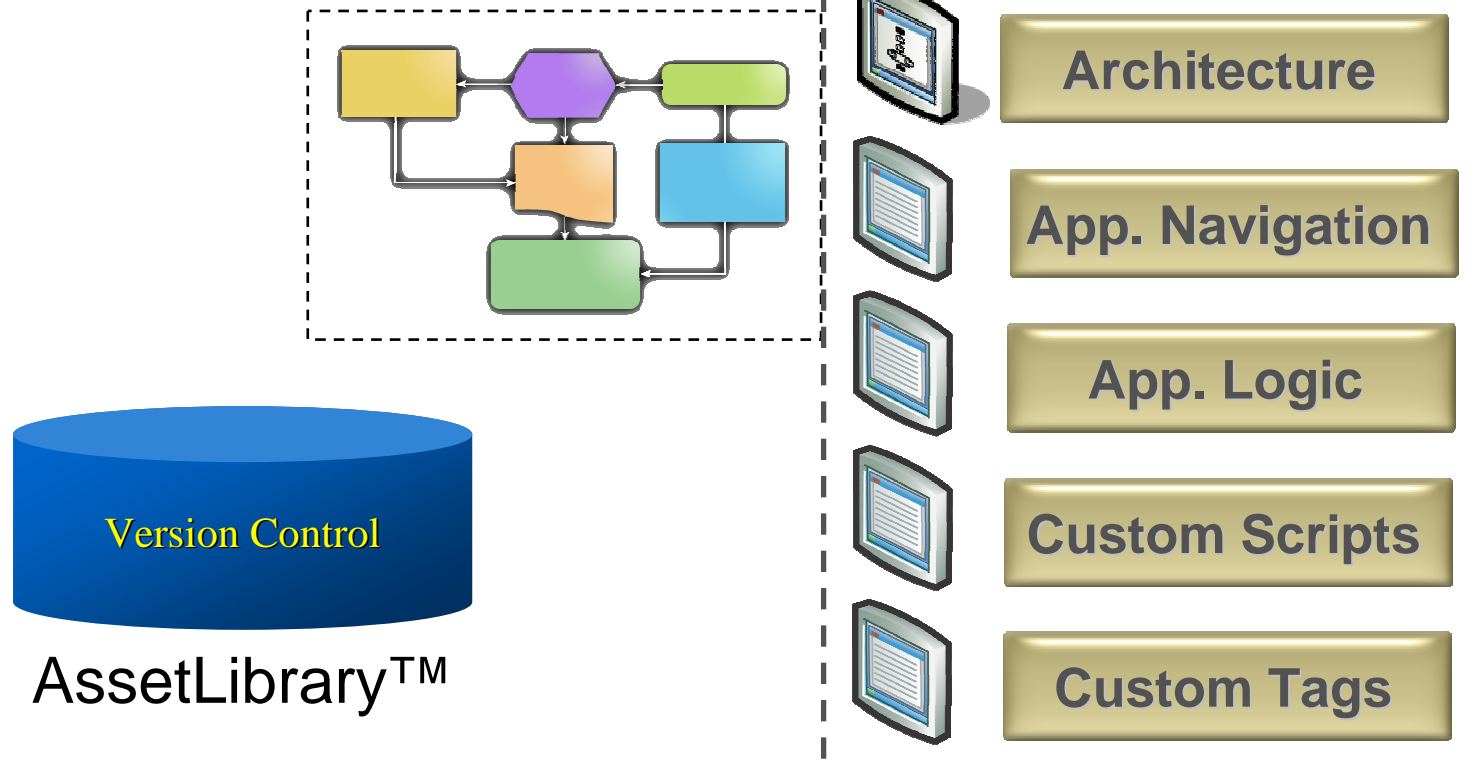


Application Metadata



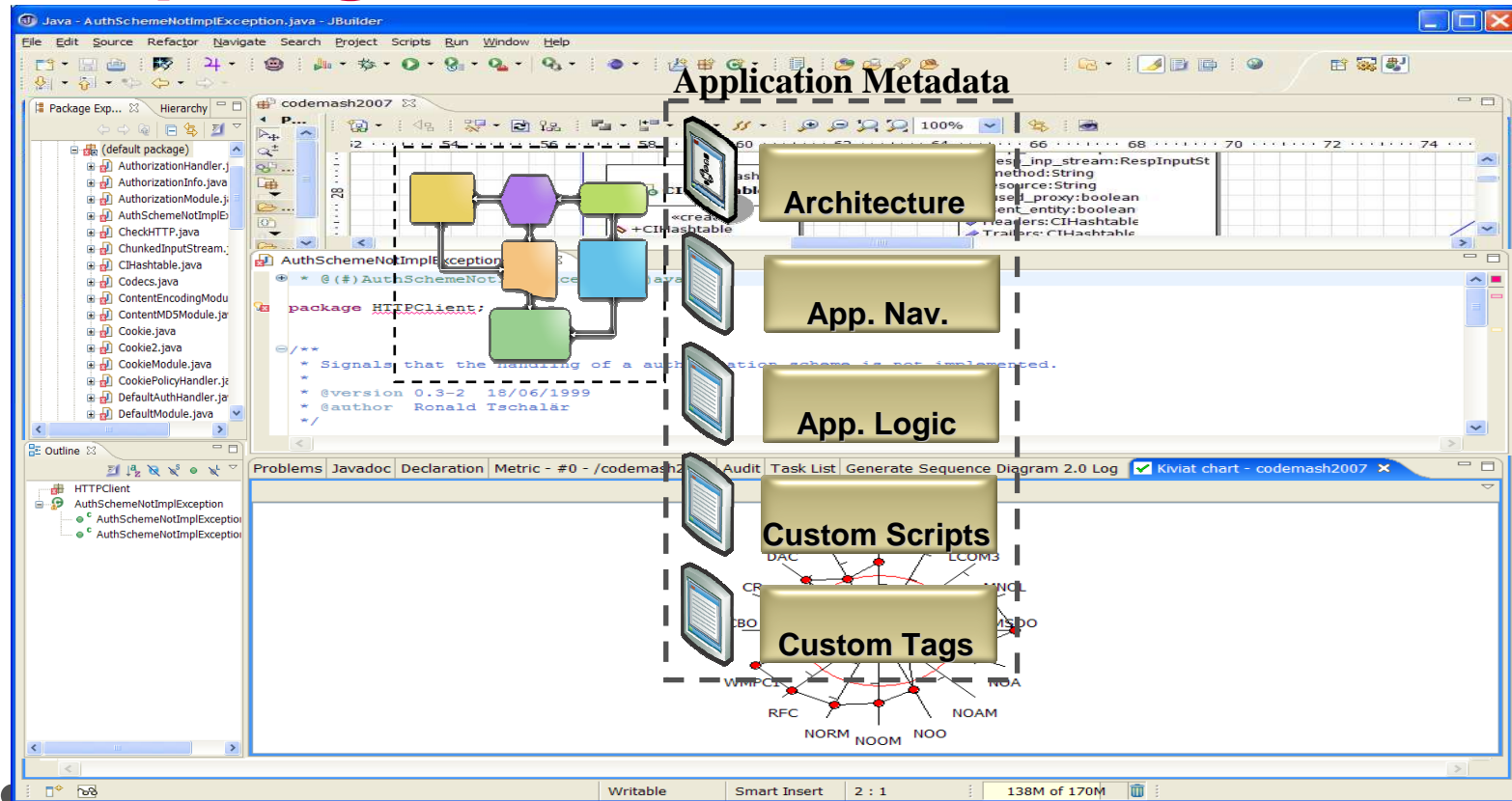
- Think of the application module as the shopping cart functionality found inside a large e-Commerce application
- Add “expert” context and understanding with intent

Storage for reuse



- Application Modules have a life of their own
- Application Modules evolve and grow in understanding and intent
- Application Modules are now ready for reuse

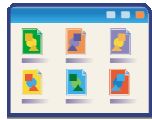
Morphing the IDE



Metadata changes the IDE

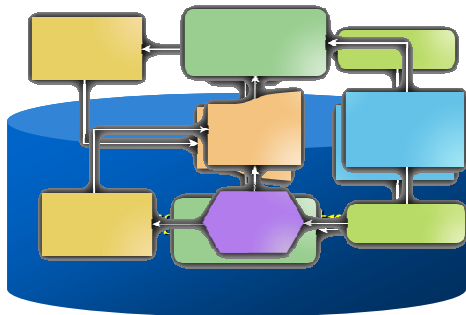
Consume

Ready to build new application



ProjectAssist™
Dashboard

App. Module 1
Shopping Cart



App. Module 2
Invoicing

AssetLibrary™

- Each module is assembled with understanding and intent
- Application modules include the underlying metadata
- Application Modules are now ready for reuse

Framework neutral approach

- Application Factories
 - Don't rely on any “one” framework – all Java frameworks are valid
 - Use existing Eclipse-ecosystem to support multiple frameworks
 - Don't build a better wizard, building a higher level of abstraction and understanding
 - Mix and Match for best solution

Application Factories Vision

- Application-driven development paradigm
- The structure, evolution, and logic behind the development of the Application belong to the application itself and are attached as metadata.
- Applications along with attached metadata are shared as reusable software assets

CodeGear background/history

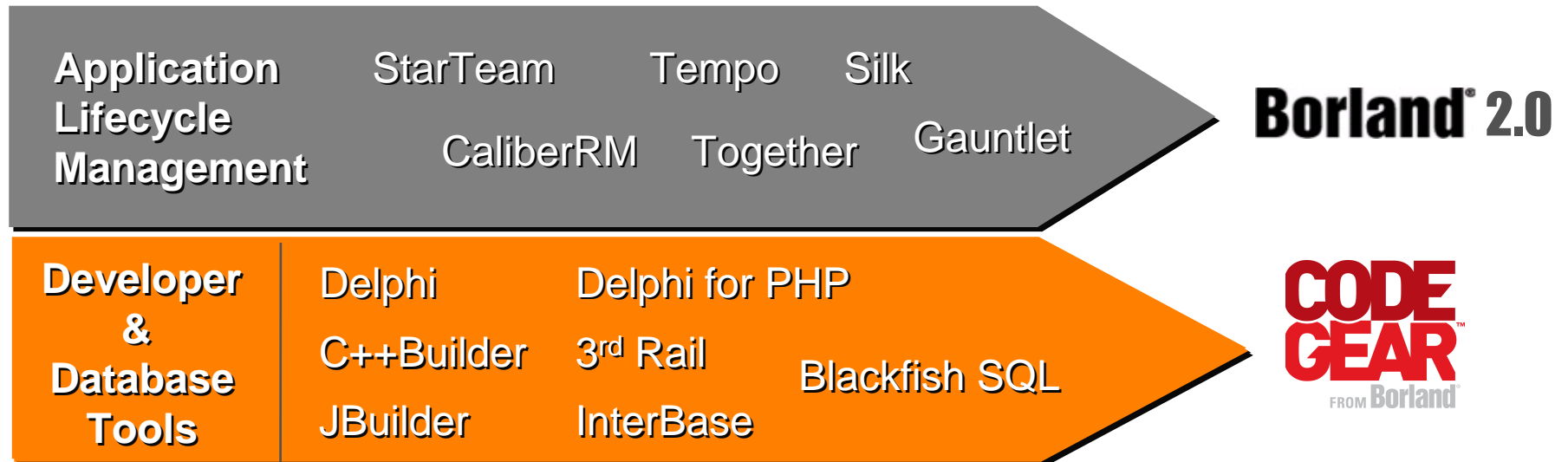
Evolution of Borland

Borland Founded in 1983 – By French Immigrant Philippe Kahn

Focused on:

- 100% Focus on Developer Productivity
- Target Market: Developers
- First Product: Turbo Pascal

Late 2006: CodeGear – the tools division created



New News...

Google CodeMash Conference - Jan... CodeGear™ JBuilder® 20...

CODE GEAR™ FROM Borland ENGLISH ▼ LOG OFF | Welcome, Michael Rozlog SEARCH:

Home Products Developer Network Support Education Downloads How to Buy About Us

Translations
CodeGear™ JBuilder®
2007がInfoWorldのベスト
Java IDEを受賞

CodeGear™ JBuilder® 2007 Is Named Best Java Integrated Development Environment of the Year by InfoWorld

SCOTTS VALLEY, Calif. - January 8, 2008 - Calling it "a very smooth, very robust IDE with many innovative features," InfoWorld today named CodeGear JBuilder 2007 "Best Java Integrated Development Environment" in its 2007 Technology of the Year awards.

JBuilder 2007 is an IDE built on the open source Eclipse framework, which makes development of Java and Web-based collaborative applications faster and more reliable. JBuilder 2007 combines the advantages of the Eclipse framework with JBuilder's innovative rapid application development (RAD) functionality, ease-of-use and productivity, and collaborative capabilities that help developers and organizations more effectively manage complex projects across locations and teams.

The InfoWorld awards recognize significant technologies of the past year that promise to make the greatest impact on enterprise IT strategies as well as the products that best exemplify the implementation of those technologies. The InfoWorld Technology of the Year awards are not open to outside nominations; only products reviewed by InfoWorld's Test Center during the previous year are eligible. The final selections are made by InfoWorld's Test Center analysts and editors.

"I found a very smooth, very robust IDE with many innovative features. It's safe to say that CodeGear decided to throw everything it had at this release - and succeeded brilliantly," wrote Andrew Binstock, senior contributing editor at InfoWorld. "...JBuilder feels solid throughout - a remarkable achievement given its status as a first release on Eclipse."

Binstock also praised CodeGear JBuilder 2007 for "a metrics package that is more extensive than any I've seen in any IDE" and "impressive collaborative features."

Said CodeGear CEO Jim Douglas, "We'd like to thank InfoWorld as well as all the supporters in the Java developer community for the fantastic reception and endorsement you've given to JBuilder 2007. It's a great honor to receive this award, and we're thrilled by the success of the product in just one year. JBuilder 2007 offers unique graphical tools, performance tuning capabilities, and collaborative development features on top of Eclipse which are translating into real productivity and costs benefits for developers and development teams."

Details on all winners of the InfoWorld 2007 Technology of the Year awards are available online at: <http://www.infoworld.com/>

About CodeGear
CodeGear from Borland Software Corporation (NASDAQ: BORL) delivers innovative, high-productivity development tools for a wide spectrum of software developers ranging from individuals to enterprise teams. CodeGear products enable developers to freely develop on their platform of choice while focusing on simplifying complex technologies and tasks so they can concentrate on application design, not infrastructure, to enable on-time project delivery. To learn more about CodeGear and its products, visit www.codegear.com.
CodeGear. Where Developers Matter.

.....

JBuilder / JGear product line

JBuilder 2007 Enterprise

JGear: Performance

JGear: LiveSource™

JGear: Team Server & Client

JBuilder 2007

Turbo JBuilder

JBuilder 2007 Enterprise:

Brings the entire JBuilder and JGear lines together to offer a complete easy to install Java development package. It includes everything in the other packages plus more.

JBuilder 2007 Enterprise is the total solution.

Questions & Answers

Demonstrations

- An overview
 - Visualizing the “code”
 - Analysis
 - Tests
 - Performance
 - Documentation

Thank You ☺

- Additional JBuilder Resources:
- JBuilder Product page:
<http://www.codegear.com/products/jbuilder>
- JBuilder 2007 Videos:
<http://dn.codegear.com/article/33880>
- Java developer information
<http://dn.codegear.com/java>
- Java developer multimedia tutorials
<http://dn.codegear.com/tv/java>

Thank You