



# Eclipse Process Framework Project

September 27, 2006



# Introduction



- Major project milestones
  - Proposed October 2005
  - Created December 2005
  - IBM contribution January 2006
  - Infrastructure up and running February 2006
  - Milestone releases – Every 6 weeks
- Continuing to expand community
  - 27 committers from 11 organizations
  - 6 face-to-face meetings with on average 15-20 people
    - 2 in Europe
    - 4 in North America

# EPF 1.0 Features



- EPF Composer 1.0 (Code and supporting documentation)
  - Plan available at:  
<http://www.eclipse.org/epf/>, under Tool / Meta-model component
- OpenUP 0.9 (Method and Process Content)
  - Plan available at  
<http://www.eclipse.org/epf/>, under OpenUP component

# EPF 1.0 Features – EPF Composer 1.0



- Process authoring capabilities for non-technical audience
  - Method authoring: create and manage method content as building blocks for process (roles, tasks, work products definitions, guidance such as examples, templates, and guidelines)
  - Process authoring: create and manage development processes in different presentations
- Editors for multiple presentations of the same process
  - Support for process patterns
  - Breakdown structure editors (work breakdown, work product breakdown, team breakdown)
  - Graphical workflow editor
- Library and Configuration Management Support
  - Manage content in libraries, reusable plug-ins and packages
  - Integration with team plug-in to support CVS, Rational ClearCase, and Telelogic Synergy
  - Support export and deployment of configurations (user-defined subsets)

# EPF 1.0 Features – EPF Composer 1.0



- HTML Publishing
  - Publish configurations and processes into HTML format for practitioners to reference.
- XML import and export
  - The ability to export EPF Composer library into a readable XML format for other 3<sup>rd</sup> party tool to consume and the ability to create the XML file from 3<sup>rd</sup> party process content repository and then import it into EPF Composer.
- Export to Microsoft Project
  - Ability to export EPF Composer delivery process into Microsoft Project
- Extensible Process Engineering Framework and Tool
  - Extensibility through multi-layered API to support UI extensions as well as middle-tier extensions

# EPF 1.0 Features – OpenUP 0.9



- OpenUP/Basic
  - An iterative software development process that is minimal, complete, and extensible.
  - Minimal - only fundamental content is included
  - Complete - can be manifested as an entire process to build a system
  - Extensible - can be used as a foundation on which process content can be added or tailored as needed
- OpenUP/Basic data:
  - Less than 200 pages of text
  - 6 roles, 18 tasks, 20 work products, and supporting guidance
  - 1 delivery processes
  - Organized into 4 sub-processes: Collaboration, Intent Management, Management, Solutions Development

# EPF 1.0 Features – OpenUP 0.9



- Key principles of OpenUP 0.9
  - Collaborate to align interests and share understanding
  - Evolve to continuously obtain feedback and improve
  - Balance competing priorities to maximize stakeholder value
  - Focus on articulating the architecture
- Supported delivery format of OpenUP 0.9
  - EPF Composer Plug-in format
  - A generated Web site (that is, a couple of hundred html pages)
  - A MSP project plan, with links to generated Web site
  - All content and associated structure in XML format



## Non-code Aspects

- User documentation and tutorials, see <http://www.eclipse.org/epf/> under Documentation component and under Tool/meta-model component
  - Using EPF Composer with a Version Control System
  - Change Request Management FAQ
  - Plug-in development guideline for EPF Composer
  - EPF Composer Development Guide
  - EPF Composer API Documentation
  - EPF XML Schema
  - Release notes with each milestone build and release build
- Design documentation
  - EPF Architecture Overview document is available at <http://www.eclipse.org/epf/>, under Tool / Meta-model component
  - Meta-Model design documentation is available at <http://www.eclipse.org/epf/>, under Tool / Meta-model component



## Non-code Aspects (cont.)



- Whitepapers/Publications/Books (sample set)
  - DeveloperWorks: The Eclipse Process Framework Project, Kroll, [http://www.ibm.com/developerworks/rational/library/05/1011\\_kroll/](http://www.ibm.com/developerworks/rational/library/05/1011_kroll/)
  - Embedded Computing Design: Building embedded software with the Eclipse Process Framework, Sibbald and Sand, <http://www.embedded-computing.com/departments/eclipse/2006/07/>
  - Eclipse Review: A Development Library at your Fingertips, Kroll and Sand, [http://www.eclipsereview.com/retrieve/er\\_200609.htm](http://www.eclipsereview.com/retrieve/er_200609.htm)
  - Rational Edge: Eclipse Process Framework Composer - Part 1: Key Concepts, Haumer, <http://www.eclipse.org/epf/general/EPFComposerOverviewPart1.pdf>
  - Rational Edge: Eclipse Process Framework Composer - Part 2: Authoring Method Content and Processes, Haumer, <http://www.eclipse.org/epf/general/EPFComposerOverviewPart2.pdf>
  - Addison-Wesley: Agility and Discipline Made Easy—Practices from OpenUP and RUP, Kroll & MacIsaac, (2006)
  - Eclipse Review: Increasing Development Knowledge with EPF Composer, Haumer, [http://www.eclipsereview.com/issues/eclipsereview\\_200606.pdf](http://www.eclipsereview.com/issues/eclipsereview_200606.pdf)

## Non-code Aspects (cont.)



- Presentations (sample set)
  - EclipseCon2006: Introducing the Eclipse Process Framework, Haumer & Balduino, [http://www.eclipse.org/epf/community/Intro\\_EPF-ph.pdf](http://www.eclipse.org/epf/community/Intro_EPF-ph.pdf)
  - EclipseCon2006: What is the Eclipse Process Framework?, Kroll
  - RSDC 2006: "Iterative Program Management Using Practices from the Eclipse Process Framework", Vaidya
  - Agile 2006: EPF and Agile Development, Kroll
  - SD Best Practices 2006: Eclipse Process Framework – An Open Source Initiative, Kroll
  - SD Best Practices 2006: Open Unified Process (OpenUP) Distilled, Kroll & Lyons
  - Verify 2006: "How to Adopt Agile, Iterative Testing with OpenUP", Oster
- Press
  - *+100 press mentions of EPF*

# Community Feedback



- Composer tool used by many organization to model their processes
  - French Scrum, PMBOK, ITIL processes, DSDM, many home-brewed
- Active discussion on epf-dev, news group, and RUP forums on how to make the UI even more approachable
- A lot of requests on additional publication and reporting capabilities
- In general positive response and feedback, see e.g. Newsgroup.
  - “I'd also like to take this opportunity to thank the development team - this is an incredible product and we truly appreciate the effort that's going into making it!” – Gage Renzi

# API: 1.0 Status



- Implementation
  - The EPF 1.0 API is 100% complete.
  - More refactoring will be done in the next release to improve their usability.
  
- Documentation
  - The EPF 1.0 JavaDoc API has been published to [http://www.eclipse.org/epf/composer\\_javadoc/](http://www.eclipse.org/epf/composer_javadoc/)
  
- Test Cases
  - Some basic test cases have been written and are being run with each EPF build.
  
- Compatibility
  - No compatibility issues exists since this is the first release.

# Tool Usability



- No technical or process modeling knowledge required
  - Uptake from many non-technical users
  - Proven that process modeling does not require complicated diagrams, but can be done via guided forms and wizards
- Intuitive, form-based interface
  - Observed ease of learning with contributors which became productive very quickly
- Identified issues around the parallel development usage model
- Model for using the processes in a development environment needs to evolve to become more embedded and 'fun'
- Desire to have more community based processes by leveraging e.g. Wiki technology

# Architectural Issues



- Linux Support
  - Ensuring that the rich text editor functions properly on a wide variety of Linux operating systems, default and user-installed HTML browsers has become a challenge for both development and testing. The list of supported Linux operating systems and HTML browsers need to be scaled down.
- Use of internal Eclipse APIs
  - The existing code uses some internal Eclipse APIs. They will be removed in the next release.
- API & Extension Points
  - More APIs and extension points need to be made available for 3<sup>rd</sup> party vendors to extend the EPF Composer functionality and UI.

# End Of Life



- N/A – there is no release to replace or EOL

# Defect Statistics – EPF Composer



- As of Sep 18, 2006:
  - 30 low severity defects deferred
  - 0 ASSIGNED
  - 13 RESOLVED ( will be closed by release time )
  - 4 VERIFIED ( will be closed by release time )
  - 570 CLOSED
- Release exit criteria
  - 0 high severity defects
  - 100% test case attempted, 95% pass of total 3900+ test cases per platform
    - As of 8/29/06 , our pass rate is > 97%



# Defect Statistics (OpenUP/Basic content)



- As of September 19, 2006:
  - 288 TOTAL (and 21 enhancements deferred to later releases)
  - 21 NEW
  - 28 ASSIGNED
  - 1 REOPENED (will be closed by release)
  - 93 RESOLVED (will be closed by release)
  - 52 VERIFIED (will be closed by release)
  - 93 CLOSED
- Version 0.9 to be released

# Standards



- OMG standard Software Process Engineering Metamodel (SPEM) 2.0 is related to current EPF metamodel
  - Expectation is to adopt SPEM 2.0 as it becomes a standard

# Committers, Contributors and Process



- 27 committers from 11 organizations
- 13 contributors from 7 organizations
- Emphasis on growing the project
- 6 face-to-face meetings
  - 2 in Europe
  - 4 in North America
- EPF Composer
  - All design documents available at <http://www.eclipse.org/epf/>, under Tool / Meta-model component
  - Design discussions are held via mailing list and weekly phone calls
- OpenUP
  - 5 subteams each have weekly phone calls
  - Active discussions on mailing lists
- Release planning
  - Conference calls used to refine releases

# Community



- Uptake faster than expected
  - For example organizations that will either deliver services around, or incorporate EPF in commercial offerings, see next slide
  - Universities leveraging for curricula
    - Example: USC (Prof. Barry Boehm is building COCOMO estimation tools on top of EPF Composer in their fall 2006 curricula)
    - A number of research grants have been awarded by IBM for research around EPF
- Download activity
  - +500 downloads in last 30 days
- Outreach activity
  - Extensive outreach per slide 8 and 9

## Example companies intending to deliver services around, or incorporate EPF in commercial offerings



- Armstrong Process Group
- Aubry Conseil
- Capgemini
- FMISolutions
- Covansys
- European Software Institute
- GOOD Software Inc.
- IBM
- Method Park
- NumberSix
- PrimeUP
- Sigma
- Telelogic
- Xansa
- Whatever Consulting Group
- WSA Consulting

# Intellectual Property – Source Code



- All contributions to date have been made under EPL 1.0
- All plug-ins contain appropriate license files
- All committers have completed Eclipse Committer Agreements and have been approved by the PMC
- Major contributions have followed due diligence process
- Project log available at
  - [http://www.eclipse.org/epf/project-info/project\\_log.csv](http://www.eclipse.org/epf/project-info/project_log.csv)

# Intellectual Property – Method Content



- Content initially contributed went through due diligence by Eclipse organization
- All the content currently available is under EPL 1.0
- Sources of inspiration for contributed content
  - RUP for Small Projects Configuration (due diligence by Eclipse done)
  - Books, articles and presentations by industry experts
    - References are appropriately made
  - Own experience of various committers and contributors
- Method Library contains copyright page
- OpenUP Copyright information is linked at the bottom of each published page and consolidated on one Copyright page available from the “About” category.
  - Copyrights include IBM and Telelogic
  - All committers have completed Eclipse Committer Agreements and have been approved by the PMC
- Project Log
  - [http://www.eclipse.org/epf/project-info/project\\_log.csv](http://www.eclipse.org/epf/project-info/project_log.csv)

# Future Plans



- OpenUP 1.0 Release
  - Probably 3-6 months out
  - Fill in some holes, fine-tune, use on projects to get feedback, restructure to improve extensibility
- Other OpenUP work
  - MDD plug-in
  - Value-based software engineering
- EPF Composer
  - Planning for next release has not yet started
  - Providing a Wiki solution
  - Improved extensibility of content
  - Defect fixing
- Other content
  - XP (pending contribution from ObjectMentor and IBM, IP due diligence left to do)
  - Consolidated agile framework
  - Scrum
  - Agile Modeling and Agile Database Refactoring