All About Platforms

Lessons Learned from Eclipse

Mike Milinkovich
Executive Director
Eclipse Foundation

October 13, 2006
Four Key Elements of a Platform

1. Great technology
2. Broad Adoption
3. “Architecture of Participation”
4. Hijacked by a fanatical community
Four Key Elements to Platform

1. Great technology
2. Broad Adoption
3. “Architecture of Participation”
4. Hijacked by a fanatical community
Eclipse Tools Platform

- Extensibility through OSGi implementation
  - Plug-ins make Eclipse whatever you need it to be
- Focus on developing a universal platform of frameworks and exemplary tools
- Tools extend the Eclipse platform using plug-ins
  - Business Intelligence and Reporting Tools (BIRT)
  - Eclipse Communications Framework (ECF)
  - Web Tools Project (WTP)
  - Eclipse Modelling Framework (EMF)
  - Graphical Editing Framework (GEF)
  - Test and Performance Tooling Project (TPTP)
Eclipse Java IDE

- Widely regarded as the Java development environment
- With all the bells and whistles…
  - Language-aware editors, views, …
  - Refactoring support
  - Integrated unit testing and debugging
  - Incremental compilation and build
  - Team development support
    - Out of the box support for CVS
  - …
Eclipse IDE Framework

- Eclipse + JDT = Java IDE
  - First class framework for Java
  - Language aware editor
  - Incremental build
  - Integrated debugging
- Eclipse + CDT = C/C++ IDE
  - First class framework for C/C++
  - Language aware editor
  - Refactoring, search
- Eclipse + PHP = PHP IDE
- Eclipse + JDT + CDT + PHP = Java, C/C++, PHP IDE
- ...
Eclipse RCP Application Framework

- Remove the IDE elements, Java language support, team development support, … and you’re left with a pretty comprehensive general application framework
  - Support for multiple platforms
    - Linux, Windows, Mac OSX, UNIX, embedded
  - Rich widget set, graphics
  - Native-OS integration (drag and drop, OLE/XPCOM integration)
Four Key Elements to Platform

1. Great technology
2. Broad Adoption
3. “Architecture of Participation”
4. Hijacked by a fanatical community
Eclipse Today

- 150+ members, including major Java, Linux and Embedded vendors (Actuate, HP, IBM, Sybase, BEA, Borland, JBoss, SAP, RedHat, Novell, Nokia, Monta Vista, Wind River, Mentor, ENEA, QNX)
  - *Hundreds of shipping products based on Eclipse technology*

- 60+ open source projects

- 55-65% market share of Java IDE

- Rich Client Platform adoption growing rapidly

- Downloads: 70 million++; 1 million in 47 days for Eclipse Callisto release

- Web site visits:
  - 36,000 unique visitors per day
  - 1,140,000 unique visitors per month (IP addresses)

- Bugs: 150K so far…. (typically ~15K-20K per major release)

- Newsgroup postings: 9K/month
IDE Usage

Java Use and Awareness Study
© BZ Research December 2005

Copyright © 2006 Eclipse Foundation, Inc., all right reserved. Made available under the Eclipse Public License v1.0
Four Key Elements to Platform

1. Great technology
2. Broad Adoption
3. “Architecture of Participation”
4. Hijacked by a fanatical community
Defining the Term

“what really distinguishes open source is not just source, but an "architecture of participation" that includes low barriers to entry by newcomers, and some mechanism for isolating the cathedral from the bazaar. This architecture of participation allows for a real free market of ideas, in which anyone can put forward a proposed solution to a problem; it becomes adopted, if at all, by acclamation and the organic spread of its usefulness.”

Tim O’Reilly
Key Points

- You need a cathedral to enable the bazaar
  - This is not about joining the project itself *per se*, but about making it easy for individuals to add their own ideas to the ecosystem

- Low barriers to entry

- New add-ons are first class citizens, not second class

- Empowering individuals and small groups is key
  - Innovation comes from committed fanatics in small groups, not corporate committees

- *Laissez faire* --- never ever ever pretend you can pick a winner

- Provide the seeds for a broad ecosystem
Keys to Success

- Almost every successful open source community has used an open-ended architecture extensible by others
  - Linux Kernel
  - Apache HTTP (mod_php, mod_perl, …)
  - Mozilla Firefox extensions

- And those who haven’t are moving there
  - OpenOffice
The Eclipse Experience

- Eclipse (Rich Client) Platform
  - It has an open, extensible architecture
  - Built out of layers of plug-ins
  - Everything is a plug-in…both from the cathedral and from the bazaar
Distinctively Eclipse

- Predictability spurs adoption
  - June 28, 2004: Eclipse 3.0 ships
  - June 28, 2005: Eclipse 3.1 ships
  - June ???, 2006: Eclipse Callisto ships

- Quality matters
  - APIs: If you build it, they will come
  - Platform Quality APIs are formally part of the Eclipse project culture
Predictable quality

- **Eclipse 3.0**
  - June 28 2004
  - TPTP, CDT

- **Eclipse 3.1**
  - June 28 2005
  - WTP, BIRT, TPTP, EMF, VE, CDT

- **Callisto**
  - June 30 2006
  - BIRT 2.1, CDT 3.1, DTP 1.0, EMF 2.2, GEF 3.2, GMF 1.0, Eclipse 3.2, TPTP 4.2, WTP 1.5, VE 1.2
How do we make it happen?

- **Architecture**
  - Modular & Extensible Architecture vs Monolithic Release

- **Governance**
  - Projects remain independent
  - Every participant engages under exactly the same rules

- **Process**
  - Open source development process
  - Frequent milestone releases
Distinctively Eclipse

- The concept of “architecture of participation” is primarily a technical statement

- Eclipse has pushed this model even further to include a business and ecosystem model which is enabling new and innovative businesses which is:
  - Applicable to all companies, regardless of their level of “open source maturity”
  - Provides a level playing field and open governance model to all comers
  - Provides quality technology on a predictable schedule to businesses to build products on
Company Behavior Appears to Follow a Maturity Model

Reference: Research performed at Carleton University
Financially supported by Nortel
Company Behavior Appears to Follow a Maturity Model

- **Proactive Value Creation**
  - Assertiveness
    - Single Product
    - Multiple Products
  - Passive Technology Use
    - 0. Denial

Copyright © 2006 Eclipse Foundation, Inc., all right reserved. Made available under the Eclipse Public License v1.0
Company Behavior Appears to Follow a Maturity Model

0. Denial

1. User

2. Collaboration

3. Champion

4. Strategist

5. Aggressive

Proactive Value Creation

Assertiveness

Passive Technology Use

Scope

Single Product

Multiple Products
Governance Matters!

Board of Directors
Approves Strategy, Plans, Policies

Eclipse Management Organization
Establishes the Roadmap, Builds the Platform, Delivers the Vision

Membership at Large
Approves Vision, Bylaws
Builds the Ecosystem

Requirements Council
Proposes Themes & Priorities

Planning Council
Establishes Platform
Release Plan

Architecture Council
Defines & Maintains
Architecture

Subcommittee A  Subcommittee B

PMC 1  PMC 2  PMC 3  PMC 4
PMC 4  PMC 5  PMC 6  PMC 7

Copyright © 2006 Eclipse Foundation, Inc., all right reserved. Made available under the Eclipse Public License v1.0
Four Key Elements to Platform

1. Great technology
2. Broad Adoption
3. “Architecture of Participation”
4. Hijacked by a fanatical community
Definition: Brand Hijack

- Let the community define what “Eclipse” means
- The antithesis of traditional marketing
  - Let the community take over
  - Be spontaneous and fun, not control-oriented
  - Embrace uncertainty
- Embrace community mavens and leaders
- Win by letting go

*Brand Hijack: Marketing Without Marketing*
By Alex Wipperfurth
The Anatomy of a Community

The Eclipse Community
Role-based Communities

I can definitely use this for my next project...

Build a great platform!

Extend the Platform to do cool stuff!

Users

Committers

Plug-In Developers

Build a great platform!

Extend the Platform to do cool stuff!
Technology-based Communities

- Rich Client
- Java Developers
- Modeling
- Embedded Developers
- Report Developers
- PHP Developers
The real picture
“Mike’s Observation”: The value of a community is proportional to the number of functioning relationships between the people and groups within the community. In other words, the greater the communication, the greater the community’s value to its participants.

Metcalfe’s Law Works for Communities as Well
Eclipse Community by the Numbers

- 75+ project leaders, PMC members
- 758 committers
- 24,426 registered Bugzilla users
- Estimated 2 million ++ users
But is it fanatical?

- Eclipse 3.1 metrics
  - Number of newsgroup posts: 32,223
  - Number of mailing list posts: 7,792
  - Number of Eclipse-specific blogs: 27
  - Number of eclipse.org page views: 500,000,000
  - Average Bugzilla queries per day: 600,000
  - Bugzilla bug or enhancement reports entered: 21,831
  - Bugzilla reports fixed: 9,871
  - Most comments on a single bug: 202
  - Number of bugs reported entirely in verse: 2
How to get there

- First, get rid of all the marketers
- Second, admit that we’re all marketers
- If you love something, set it free
- It’s about the human conversation
  - And developers want to talk to developers, not marketing flacks
  - And the conversation is about code
- Embrace others as leaders
- Be transparent
- Say “Thank You”

If you’re interested in creating passionate users, or keeping your job, or breathing life into a startup, or getting others to contribute to your open source project, or getting your significant other to agree to the vacation you want to go on... congratulations. You’re in marketing. Now go kill yourself.

Credit: Creating Passionate Users
Pillars of Eclipse

Enterprise Development
Embedded & Device Development
SOA
Application Lifecycle Management
Rich Internet Application
Rich Client Platform (RCP)
Application Frameworks

Eclipse Platform
JP Morgan – Internal Banking Applications

- Need an alternative to spreadsheets for building applications
- Use Eclipse RCP to building applications for bond trading, foreign currency pricing, etc.
- Benefit from update manager and native look and feel on the desktop
Adobe Version Cue

- Version Cue provides collaboration functionality for small groups using Adobe Creative Suite.
- Server component uses Eclipse Equinox to build a component-based application.
City of Stuttgart

- iMedic used Eclipse RCP to implement new IT purchasing system for City of Stuttgart
Thank You!