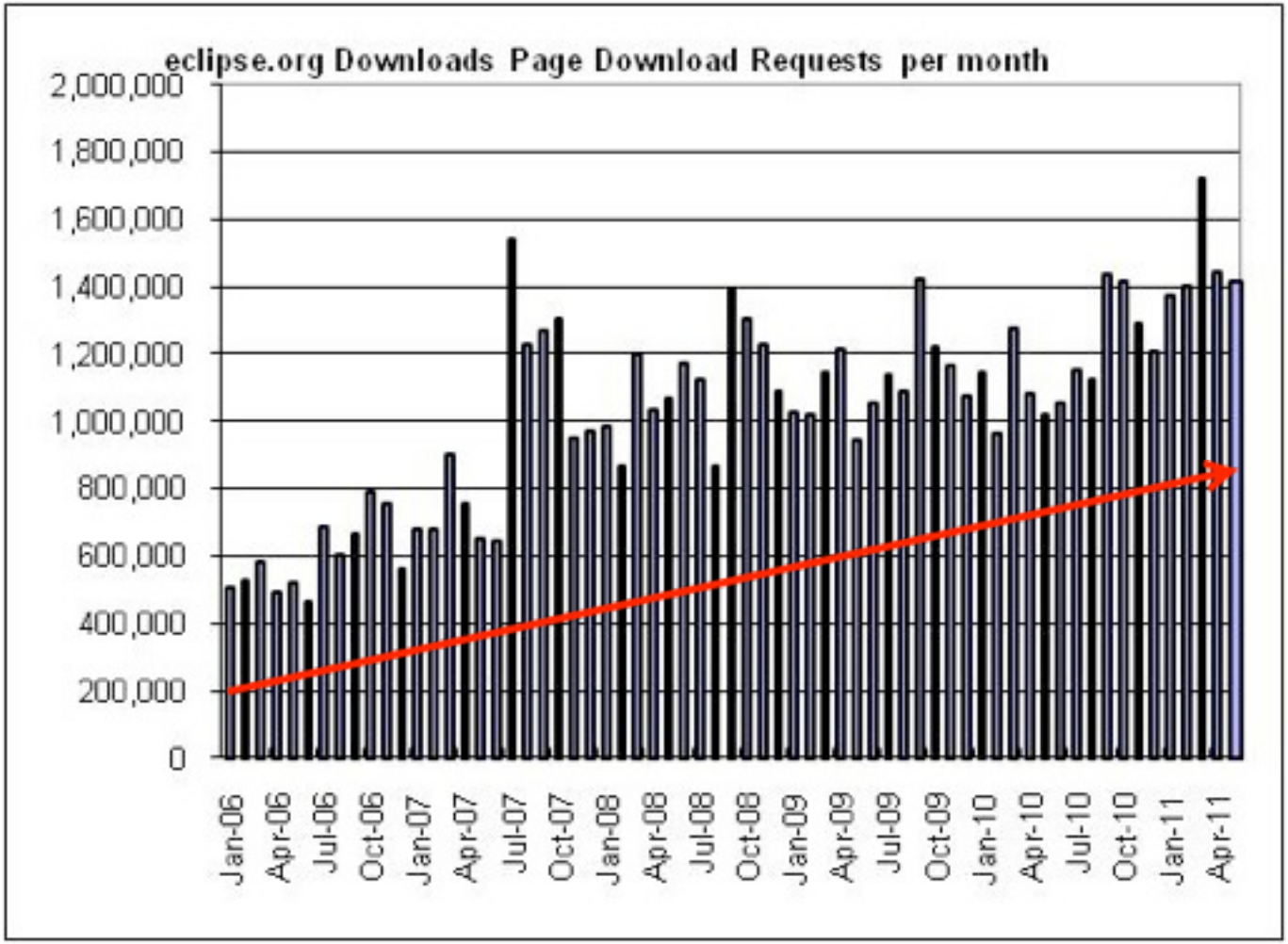
Open Source Processes: Lessons for Industry

Mike Milinkovich
Executive Director
Eclipse Foundation
@mmilinkov

BoCSE
November 16, 2011

Eclipse: The Leading Developer Community

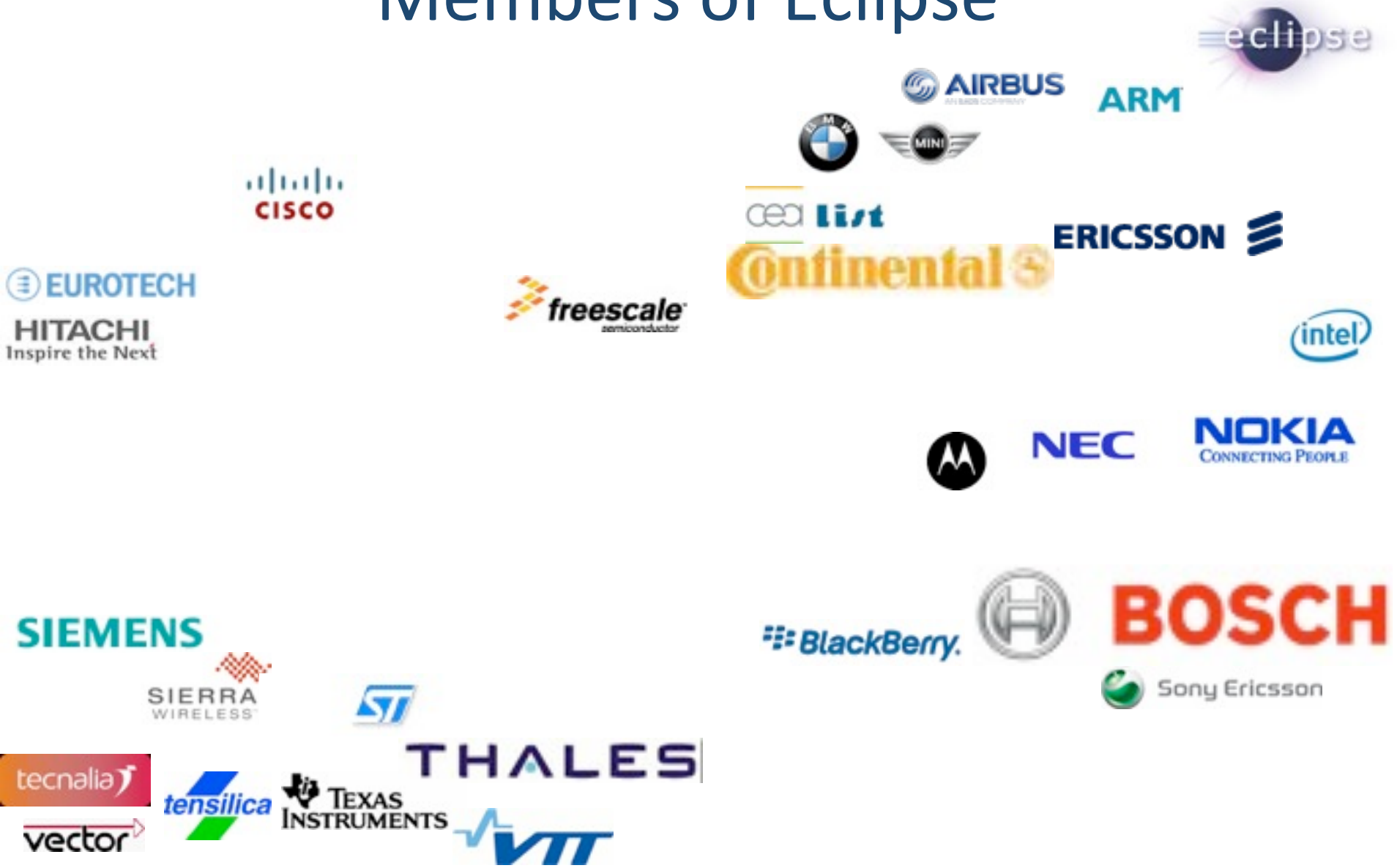


Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Members of Eclipse

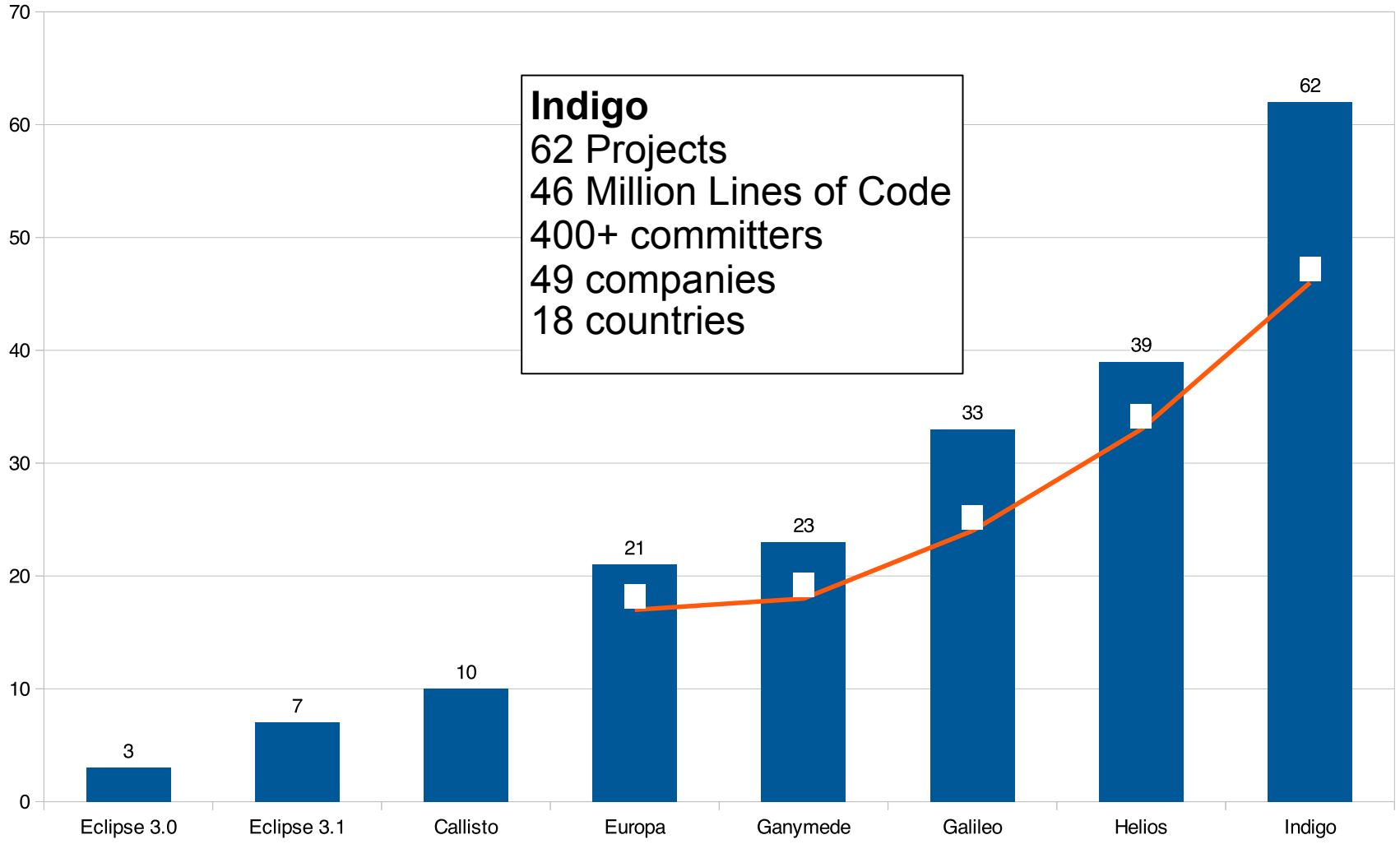
Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Members of Eclipse



Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

8 Years in a Row

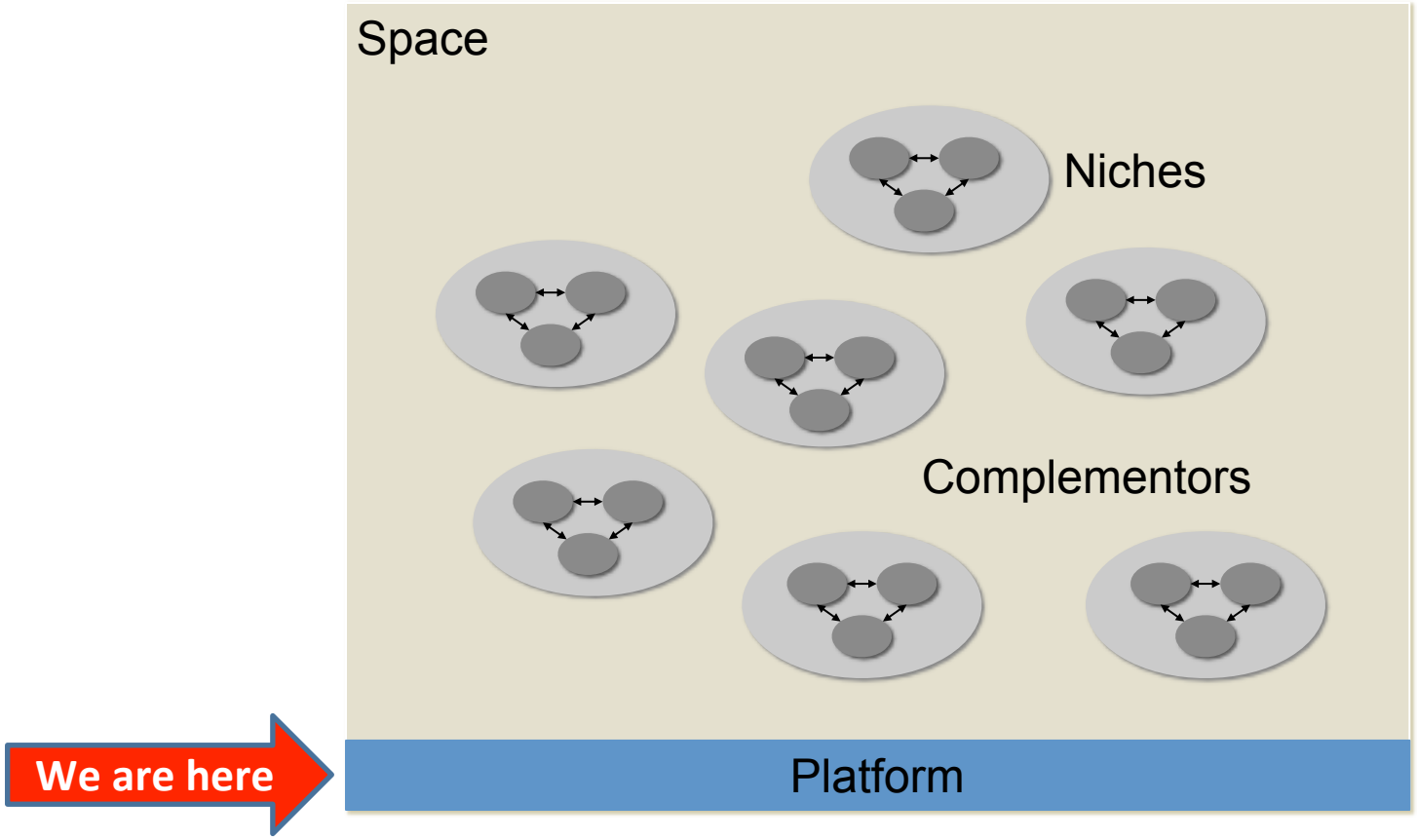


So Eclipse Has...

- Millions of users
- Thousands of products
- One thousand developers
- Hundreds of companies, hundreds of projects
- Predictable schedules
- World class intellectual property management
- Fourteen employees
- Zero product managers

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Platforms and Ecosystems



Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Why an Open Source Platform?

- Open Source development model encourages open innovation
  - Openness, Transparency, Meritocracy
  - Vendor neutrality
- Open Source licensing allows competitors to collaborate on shared platforms
  - No requirement for royalties.
  - No single control point of intellectual property
- Open Source business model encourages rapid adoption of technology
  - It is free and easy to access
- Open Source can allow companies to disrupt the business

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Our constraints

One example : AIRBUS A300

- Program began in 1972 and will stop in 2007

  2007-1972 = 35 years...

- Support will last until 2050

  2050-1972 = 78 years !!!

On board software development for very long lifecycle products

© AIRBUS FRANCE S.A.S. Tous droits réservés. Document confidentiel.

Open source business model for industrials

- **What we expect from an open source model**
  - Insure a continuity of tools with respect to industrial challenges
  - Avoid single-source dependency : knowledge is shared
  - Take advantage of innovation and risk sharing
  - Contribute to standardization effort
  - Federate the on board software development tools market and gather a significant users community in this area
  - Reduce the temptation for offshore development made on US tools by managing in Europe the creation of added value components

© AIRBUS FRANCE S.A.S. Tous droits réservés. Document confidentiel.

Open Source Questions

- Is Open Source chaotic?
- How does development *really* work?
- What about the Open Source community?
- How do you manage community contributions?
- How do you plan in Open Source?

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Meritocracy

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Transparency

Andrew Magill – flickr.com

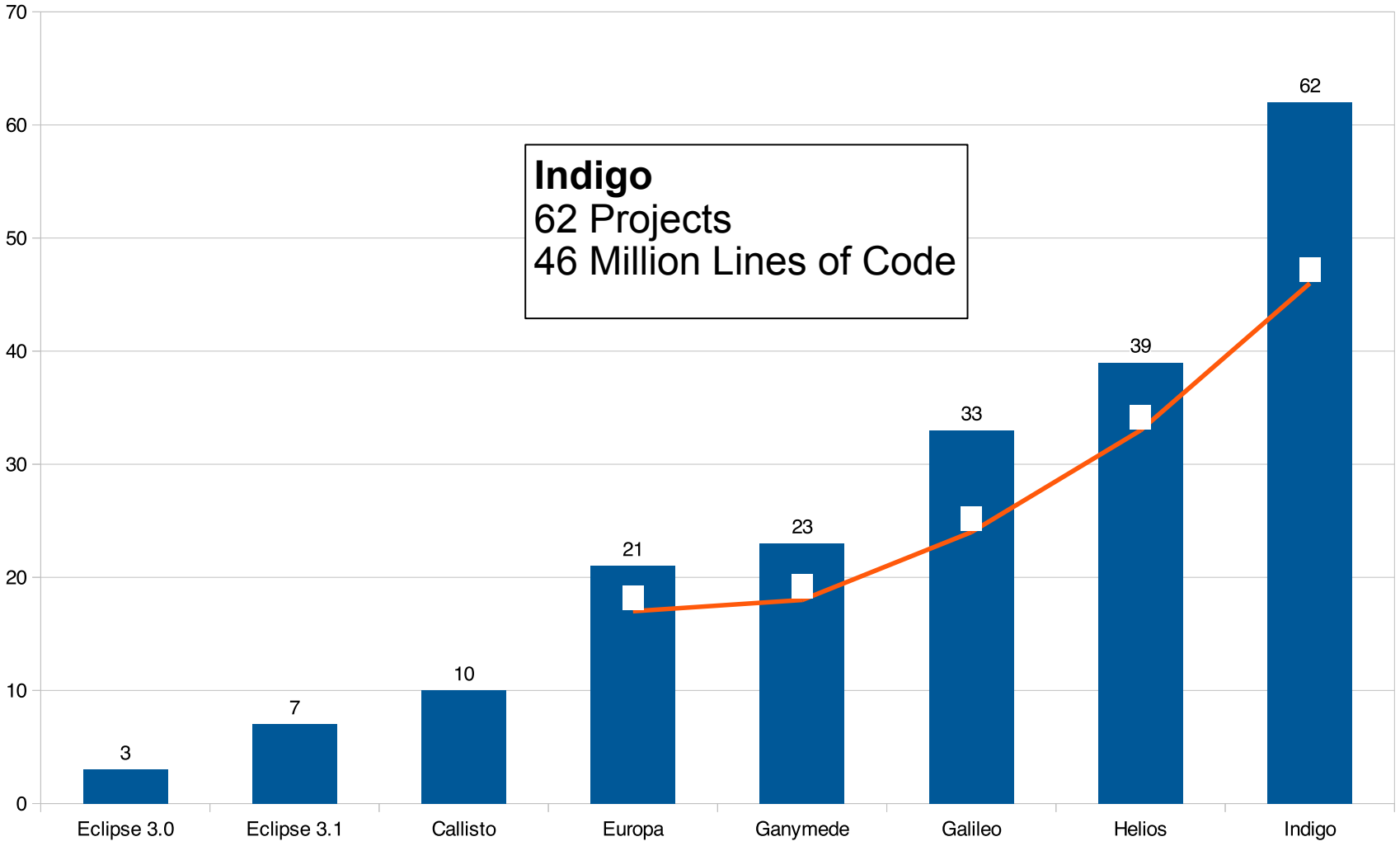
Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Openness

Chris J. Fry – flickr.com

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

8 Years in a Row

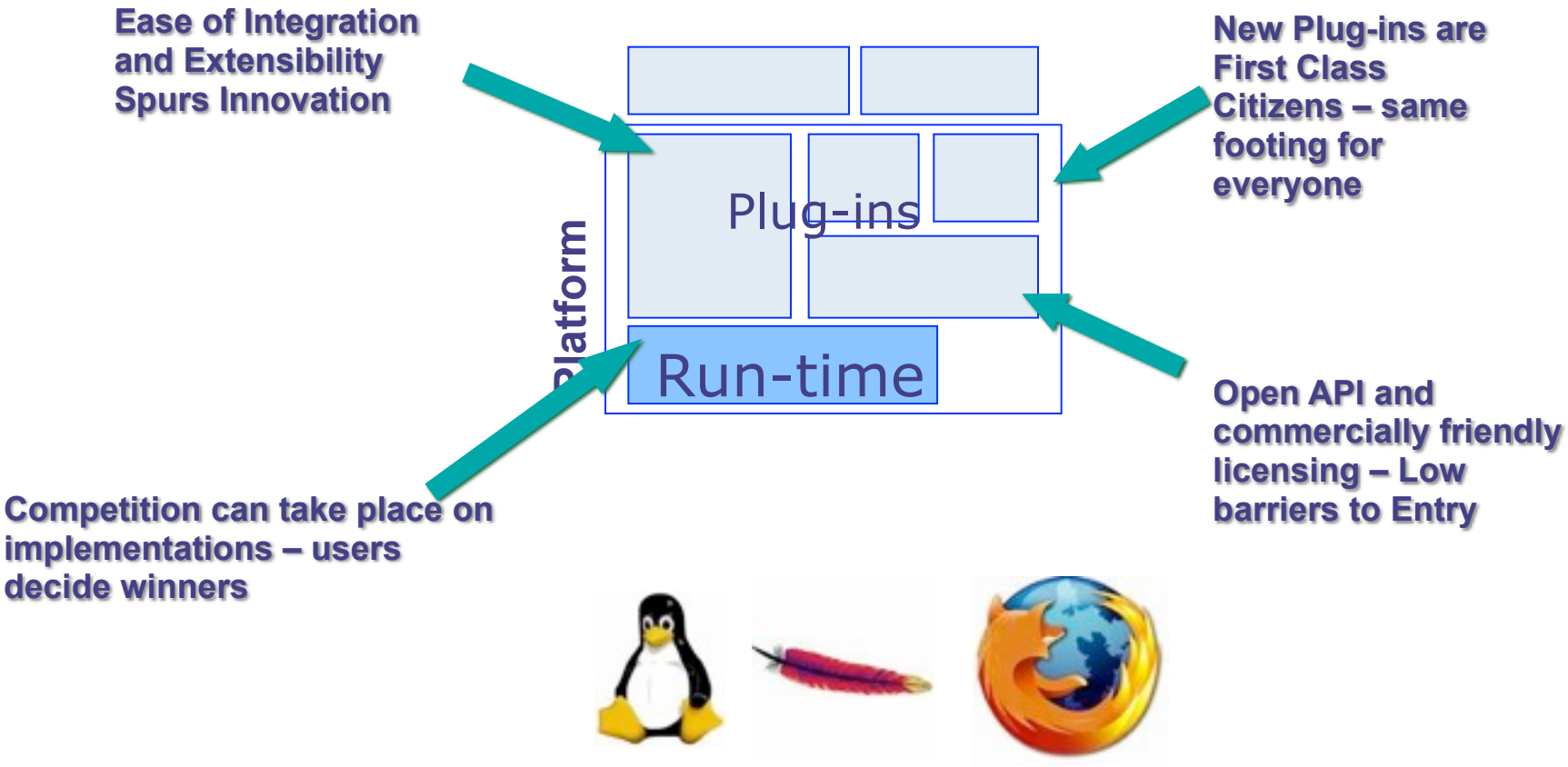


Key Success Factors

- Architecture
- Governance
- Process

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Platform Modularity: The Eclipse Experience

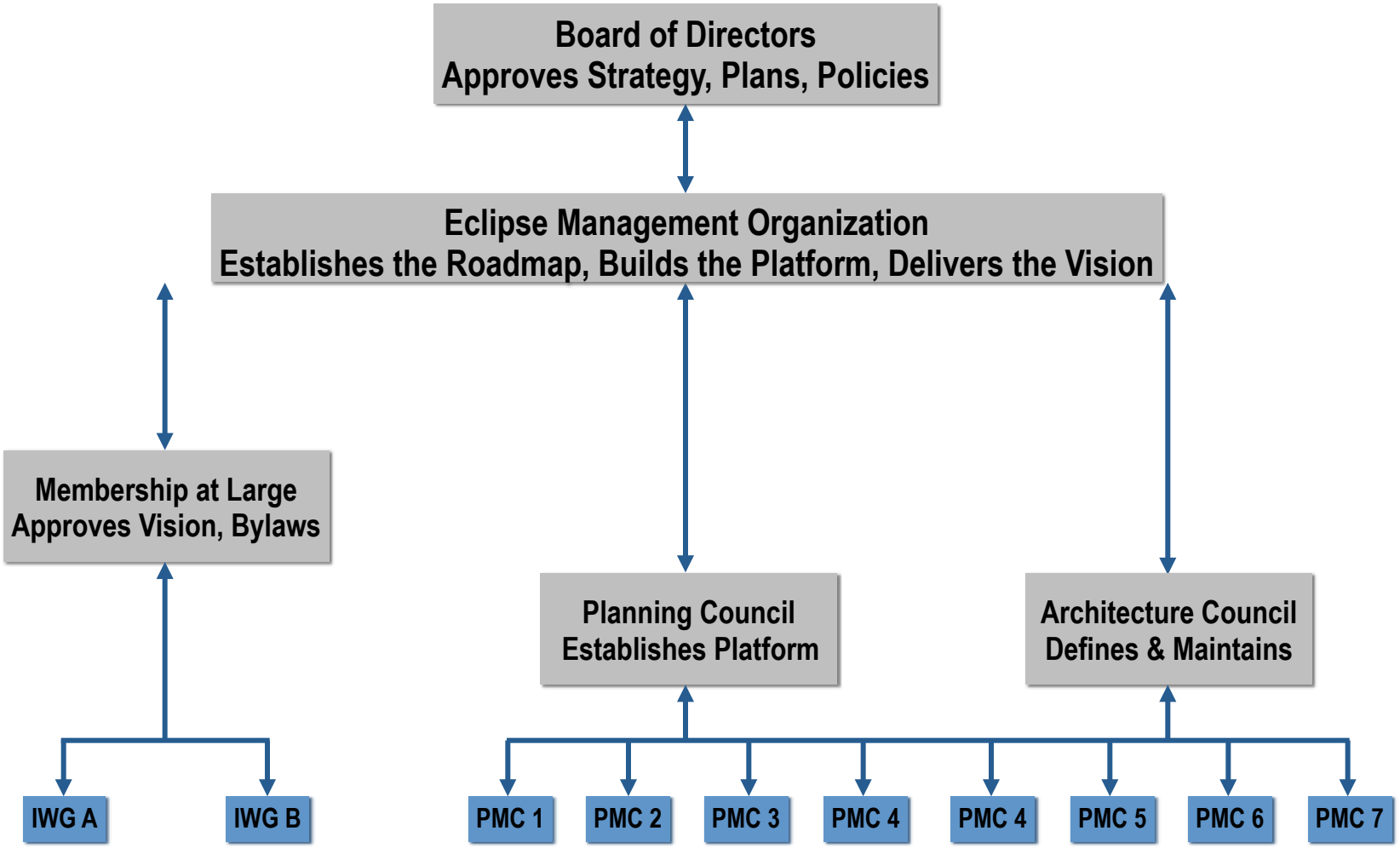


Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

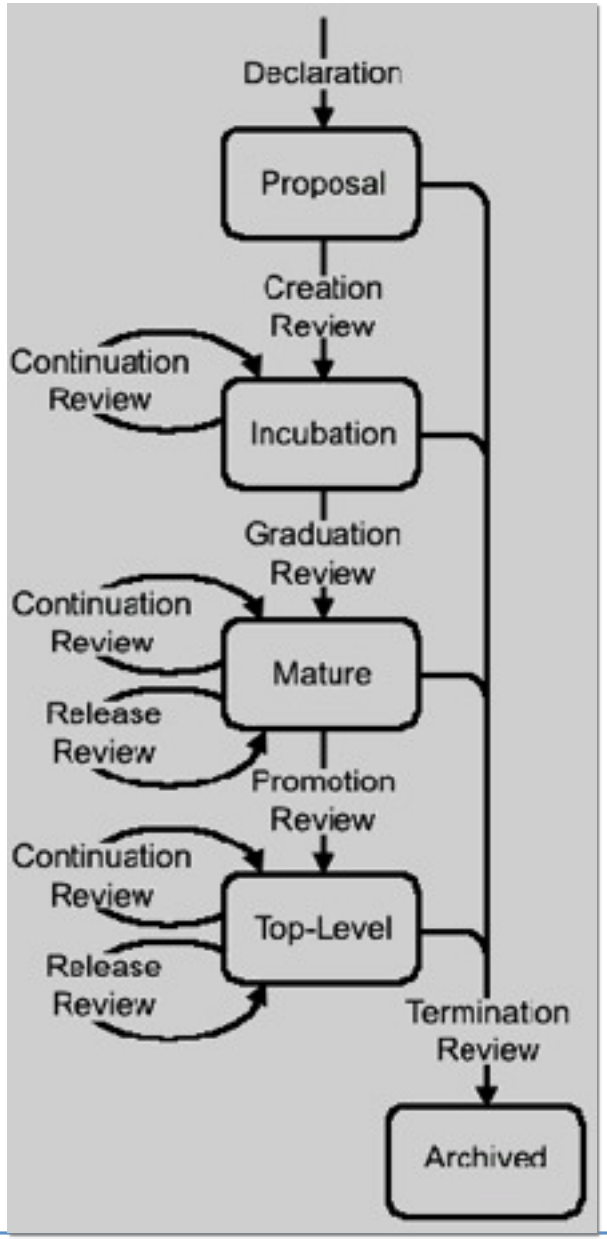
Governance ≠ Management

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Eclipse Governance Structure



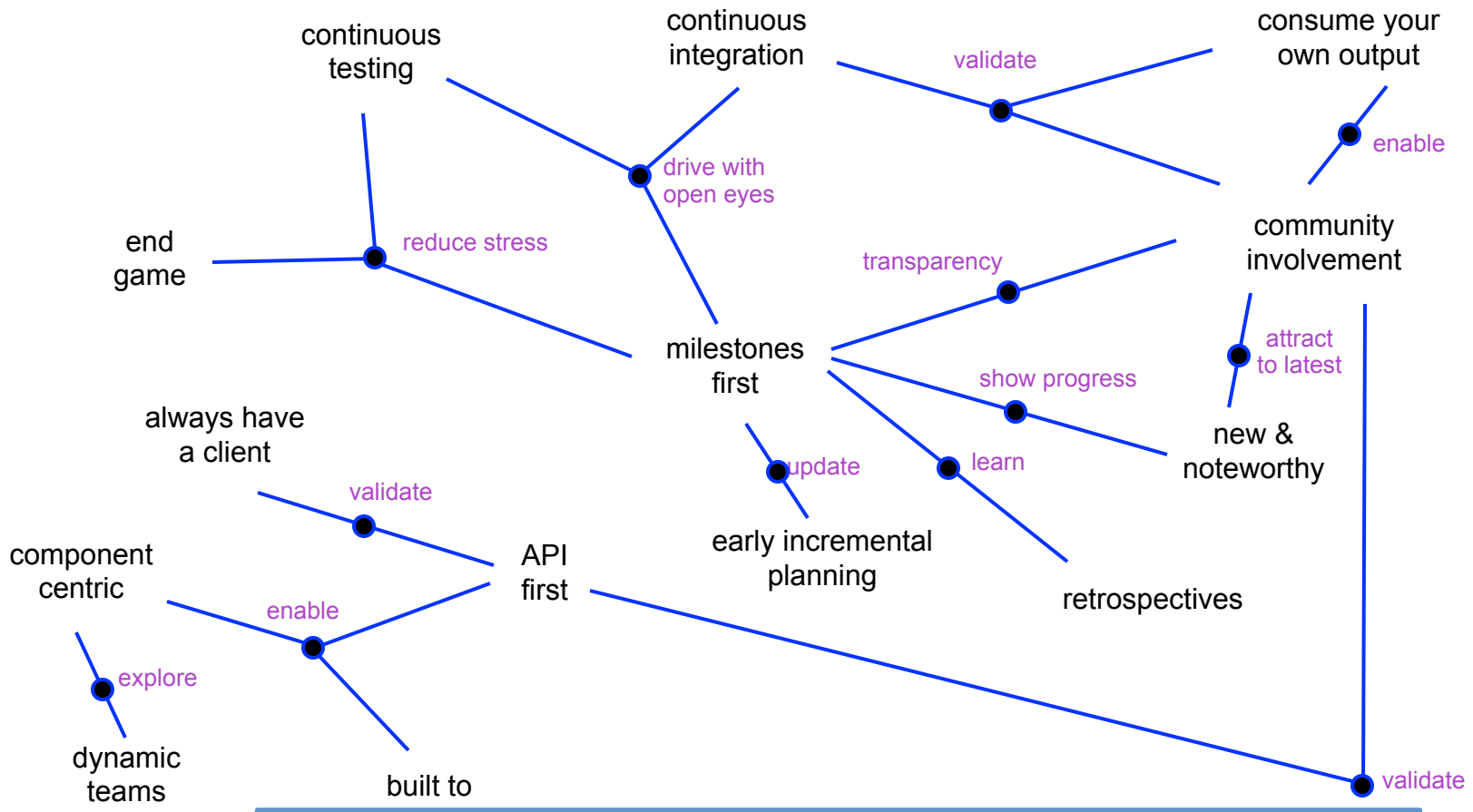
Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0



Governance:

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

How is the Development Done?



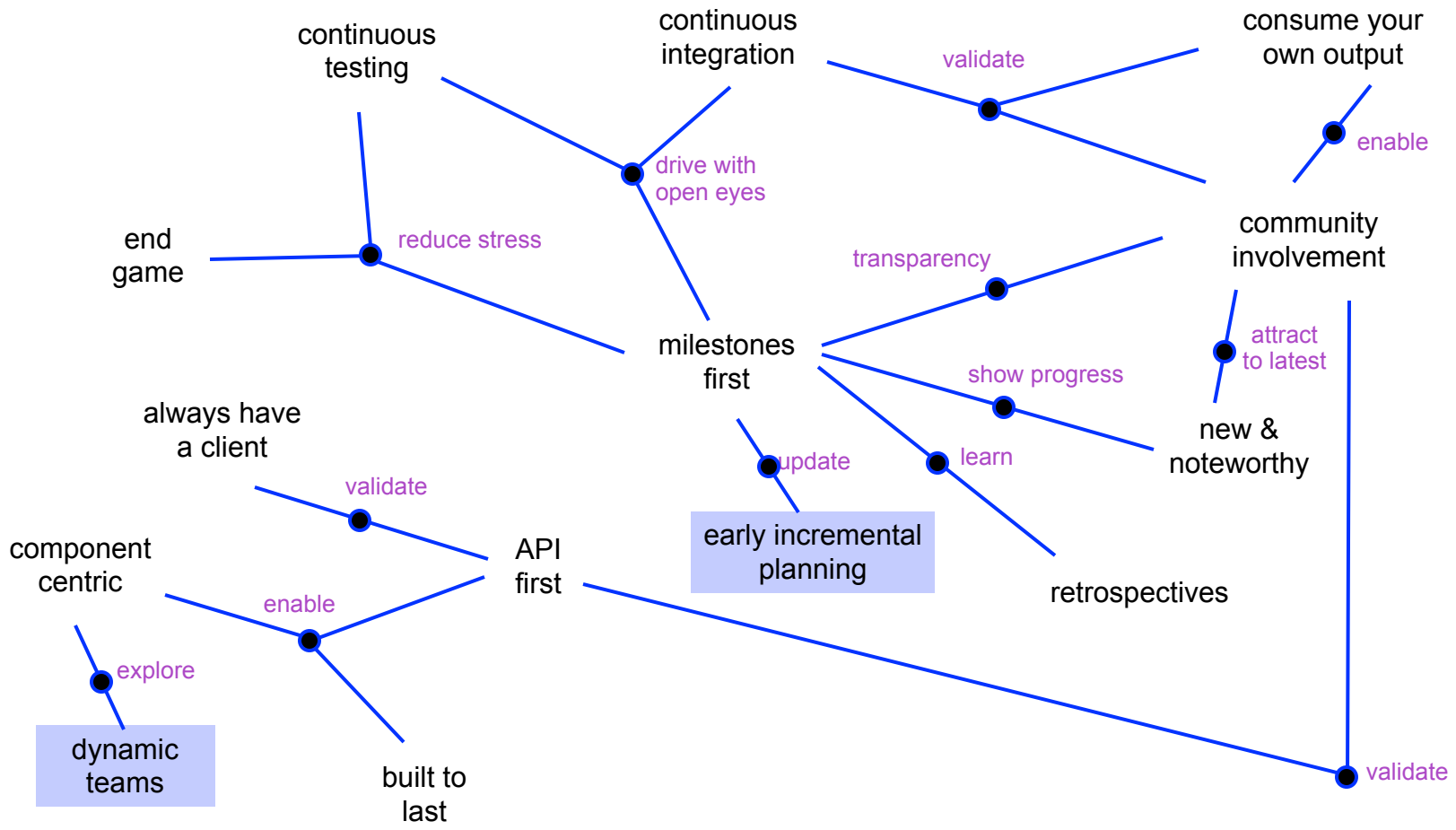
"The Eclipse Way"
Erich Gamma and John Wiegand

Open Source Rules

- OS projects are highly structured
  - explicit rules (more than in most closed source projects)
  - Who may change the source code?
  - Who is responsible for delivering?
  - Who decides about the architecture?
  - …
- Commit rights: public "meritocracy"
  - only a small number of developers can modify the source code: committers
  - key architecture defined by a small team of lead developers
  - peer pressure among committers – continuous reviewing
  - continuous review and feedback by the community
  - contributions from outside have to be reviewed by committers

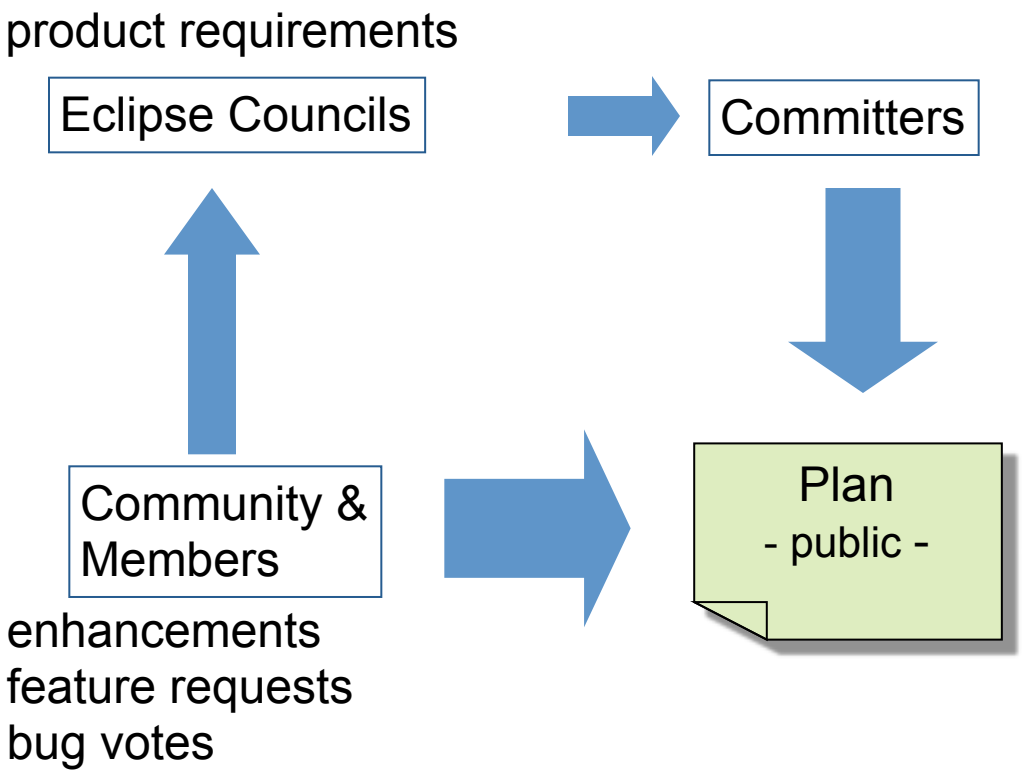
Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Planning



Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Forces of Influence



Planning Council
posts draft plan

- plans start in embryonic form and are revised throughout the release cycle
- milestones/time boxes are fixed early on

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Planning

- Release themes establish big picture
  - Community input
  - Planning council new source of input
- Component teams define component plans
- PMC collates initial project plan draft
  - Tradeoff: requirements vs. available resources
  - committed, proposed, deferred
- Plan initially spells out
  - themes
  - milestones
  - compatibility (contract, binary, source, workspace)
- Plan is alive

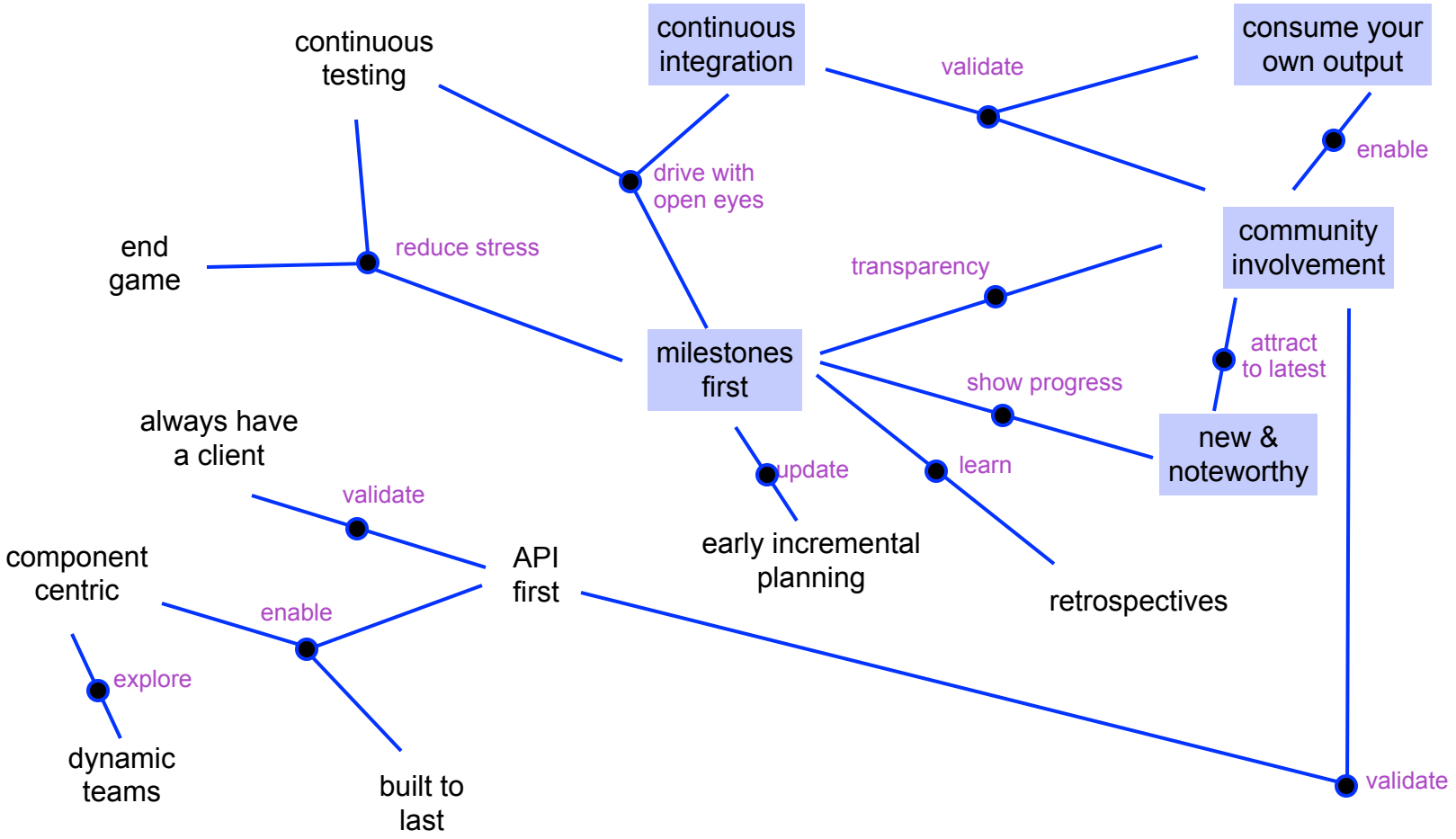
Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Ongoing Risk Assessment

- Address high risk items and items with many dependencies early
- Maintain schedule by dropping items (if necessary)
  - we will drop proposed items
  - we hate to drop committed items
  - prefer fewer completed items than more items in progress
- High risk items are sandboxed to reduce risk to other items
  - prefer to serialize highest risk items (to minimize integration pain)

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Project Rhythm



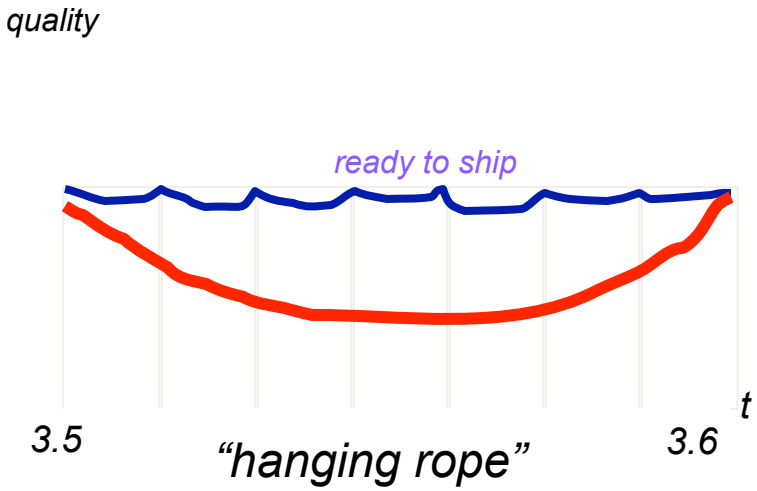
Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Milestones

- break down release cycle into milestones
  - We use 6 weeks
- milestones are a miniature development cycle
  - Plan, execute, test, retrospective
- milestone builds are good enough to be used by the community

➢ **milestones reduce stress, keep quality high**

- before/after



Continuous Integration

- Fully automated build process
- Build quality verified by automatic unit tests
- Staged builds
  - nightly builds (some projects even more frequently)
    - discover integration problems between components
  - weekly integration builds
    - all automatic unit tests must be successful
    - good enough for our own use
  - milestone builds
    - good enough for the community to use

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Practice Makes Perfect

- 7 milestones, 4 release candidates
  - 11 chances to practice releasing
- Projects denoted N_0, N_1, N_2, N_3
  - Build in order of dependencies
  - Early builds takes days, later builds take hours
- Build to shared repository, make everything available to the community for feedback and testing

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Getting on the Train

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Constant Public Status Reporting

Back to Project List
All Projects Overview Grid

Simultaneous Release Compliance Grid

This page is to summarize progress towards the yearly Simultaneous Release as the data has been provided by the projects, at the Eclipse Foundations Portal Tracking Tool. For details on the requirements see requirements for the Simultaneous Release.
If questions please see Simultaneous Release Tracker FAQ or ask the question on cross-project dev list.

	birt	datatools	eclipse	modeling	mylyn
Offset	♦	♦	♦	♦	♦
Planning	♦	♦	♦	♦	♦
IP Documentation	♦	♦	♦	♦	♦
Release Review	♦	♦	♦	♦	♦
Communication and Availability	♦	♦	♦	♦	♦
API	♦	♦	♦	♦	♦
Message Bundles	♦	♦	♦	♦	♦
Version Numbering	♦	♦	♦	♦	♦
OSGi Bundle Format	♦	♦	♦	♦	♦
Execution Environment	♦	♦	♦	♦	♦

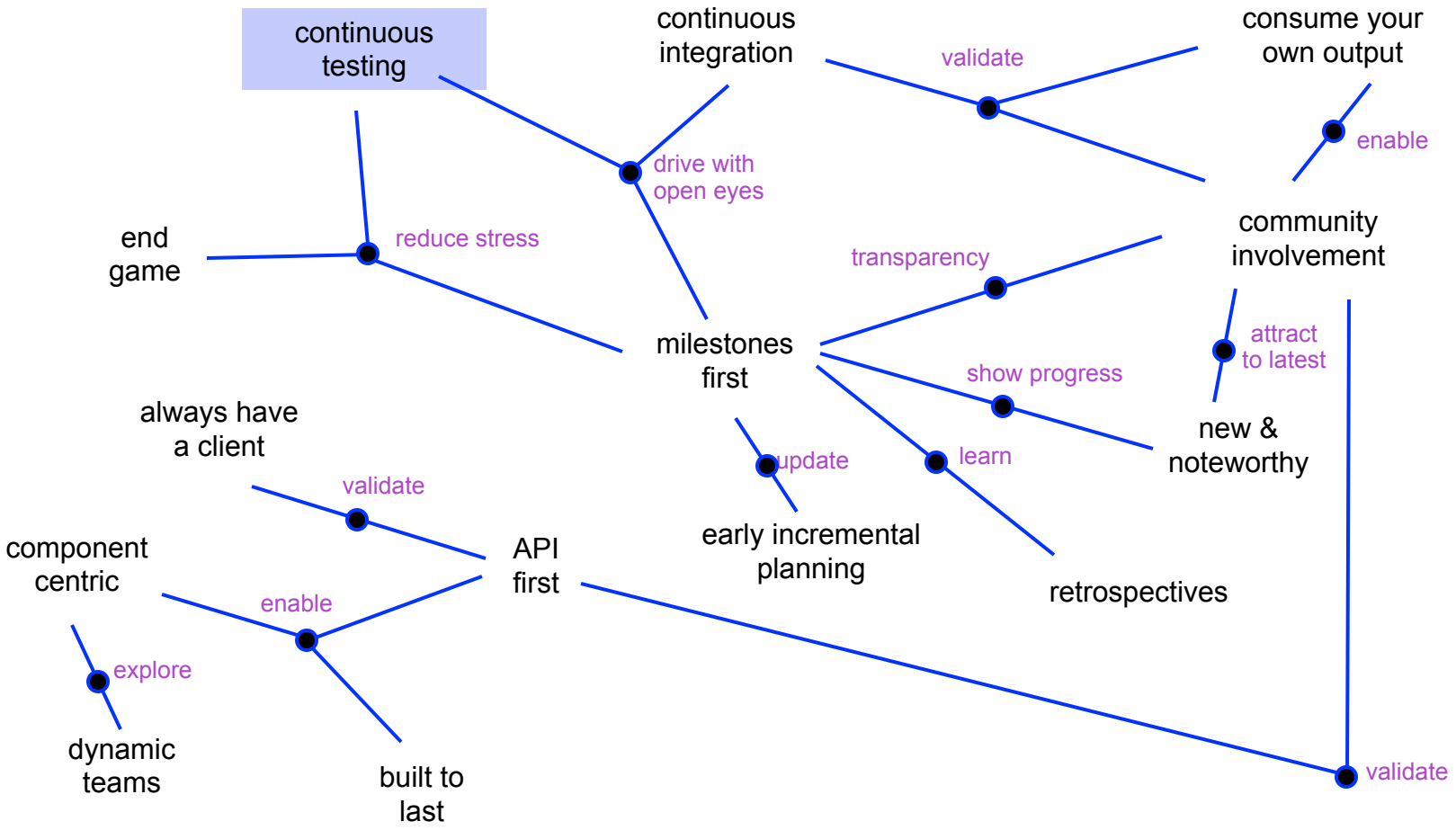
Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Community Involvement

- An active community is the major asset of an OSS project
- OSS project gives and takes:
  - OSS developer gives:
    - listen to feedback and react
    - demonstrate continuous progress
    - transparent development
  - OSS developer takes:
    - answer user questions so that developers do not have to do it
    - report defects and feature requests
    - validate technology by writing plug-ins
    - submit patches and enhancements
- Give and take isn't always balanced
  - community isn't shy and is demanding

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

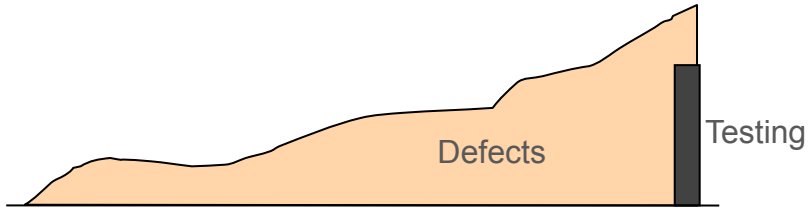
Testing

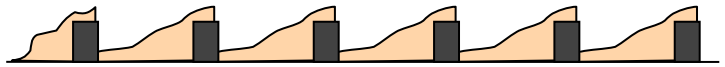


Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Testing

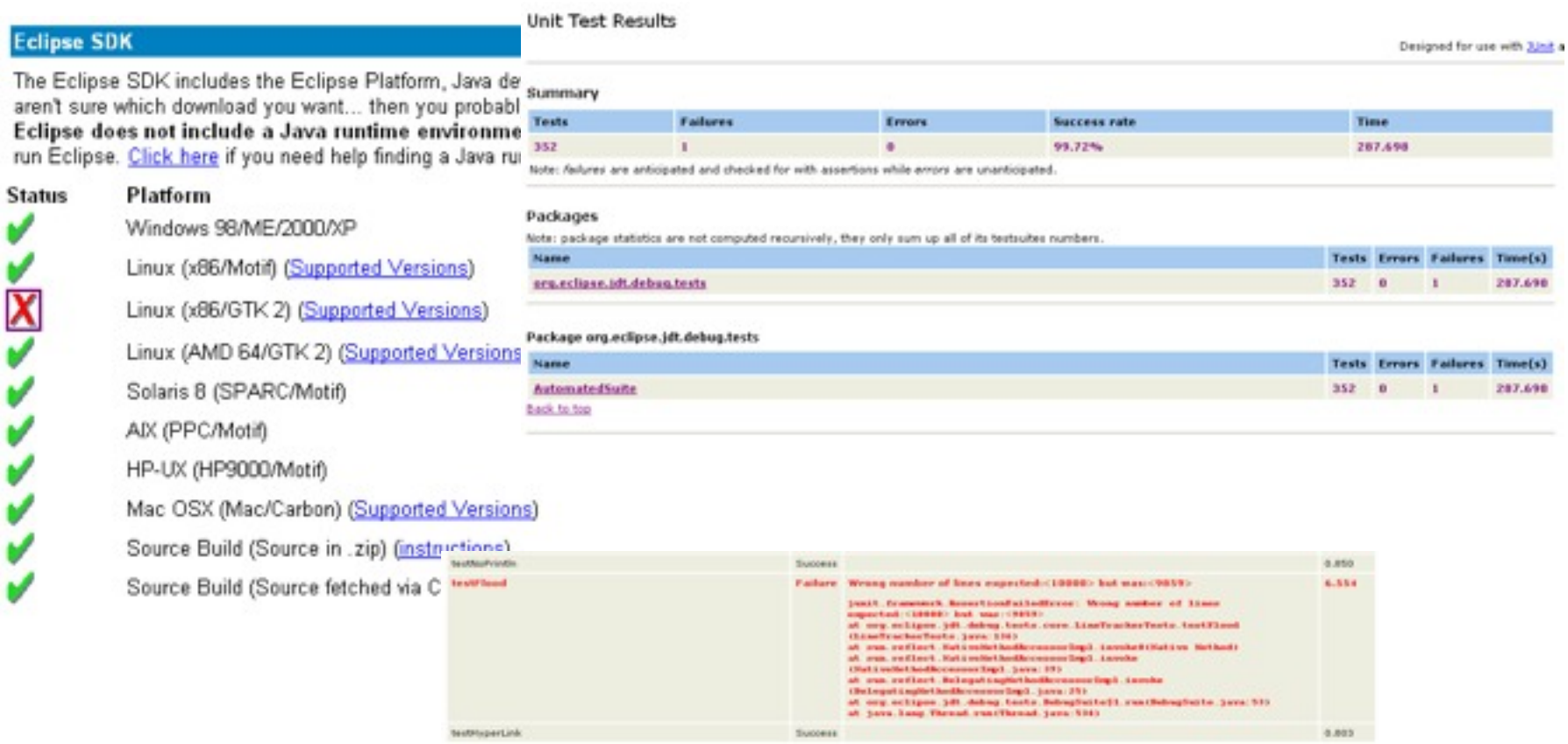
- Innovate with confidence
- Tests run after each build
- Test kinds
  - **correctness** tests
    - assert correct behavior
  - **performance** tests
    - assert no performance regressions
      - based on a database of previous test run measurements
  - **resource** tests, leak tests
    - assert no resource consumption regressions





Kent Beck – JUnit handbook

Unit Test Report



Eclipse SDK

The Eclipse SDK includes the Eclipse Platform, Java de
aren't sure which download you want... then you probabl
Eclipse does not include a Java runtime environme
run Eclipse. Click here if you need help finding a Java ru

Status	Platform
✔	Windows 98/ME/2000/XP
✔	Linux (x86/Motif) (Supported Versions)
✘	Linux (x86/GTK 2) (Supported Versions)
✔	Linux (AMD 64/GTK 2) (Supported Versions
✔	Solaris 8 (SPARC/Motif)
✔	AIX (PPC/Motif)
✔	HP-UX (HP9000/Motif)
✔	Mac OSX (Mac/Carbon) (Supported Versions)
✔	Source Build (Source in .zip) (instructions)
✔	Source Build (Source fetched via C

Unit Test Results

Designed for use with JUnit a

Summary

Tests	Failures	Errors	Success rate	Time
352	1	0	99.72%	287.698

Note: failures are anticipated and checked for with assertions while errors are unanticipated.

Packages

Note: package statistics are not computed recursively, they only sum up all of its testsuites numbers.

Name	Tests	Errors	Failures	Time(s)
org.eclipse.jdt.debug.tests	352	0	1	287.698

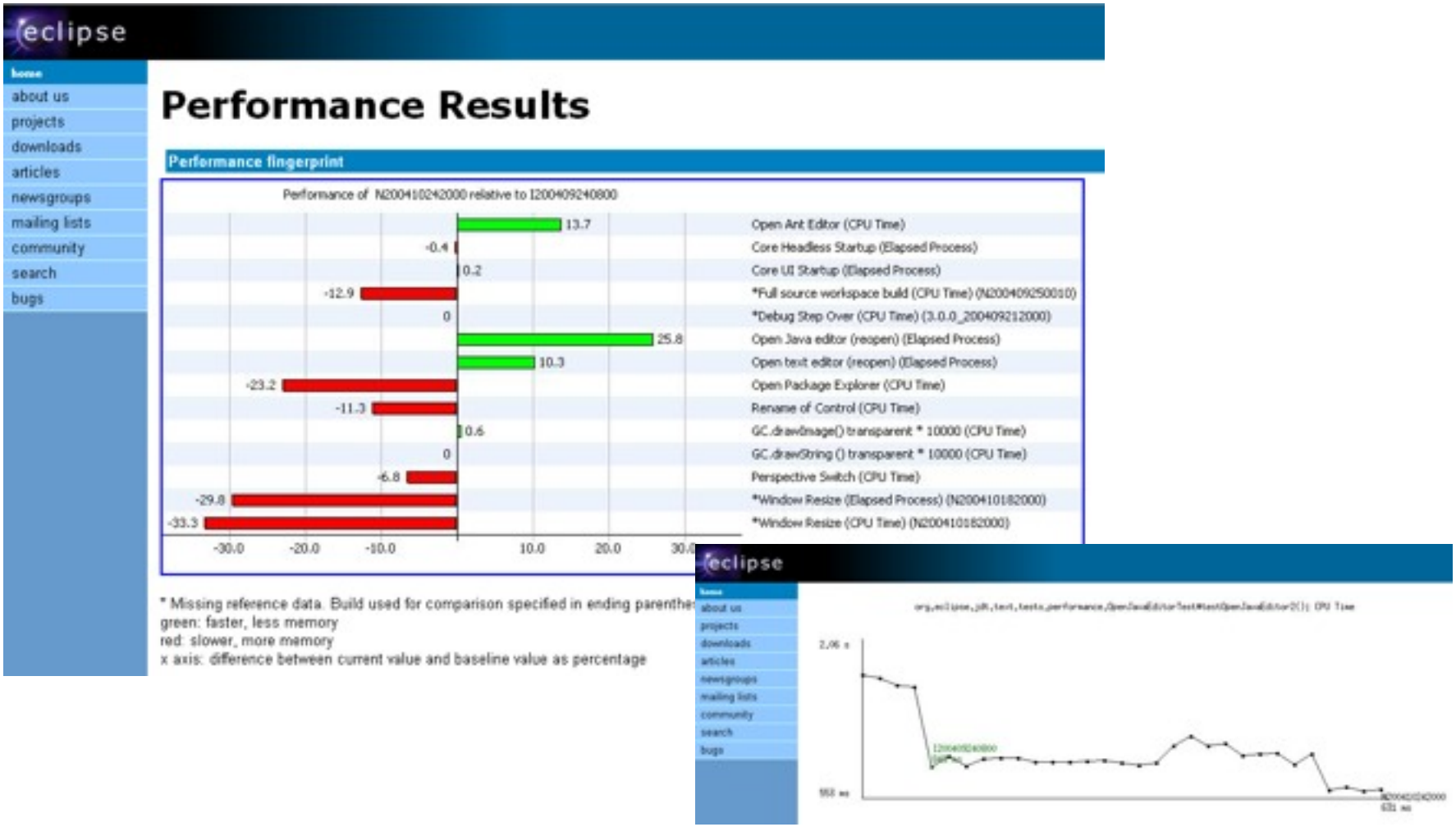
Package org.eclipse.jdt.debug.tests

Name	Tests	Errors	Failures	Time(s)
AutomatedSuite	352	0	1	287.698

Back to top

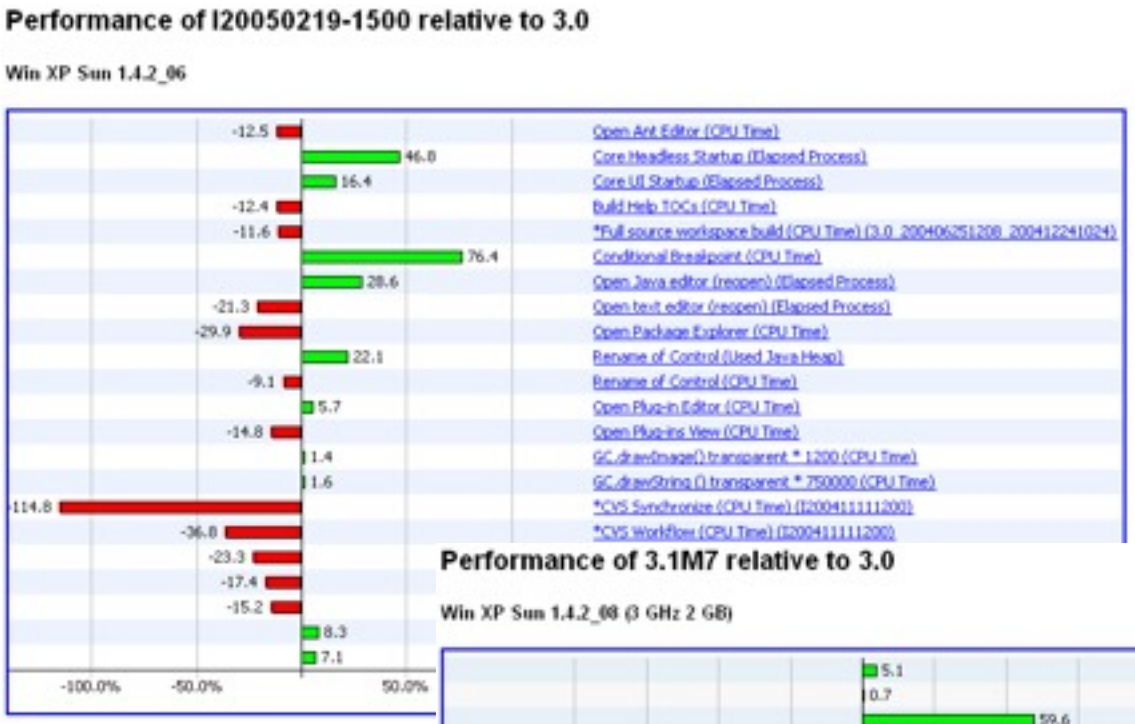
testNoPrintln	Success		0.050
testFlood	Failure	Wrong number of lines expected:<10000> but was:<9859>	6.554
testHyperLink	Success		0.003

Performance Test Report



Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Before (M5) – After (M7)



Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Code Coverage

org.eclipse.jdt.core

Element	Missed Instructions	Cov.	Missed Branches	Cov.
org.eclipse.jdt.internal.core		81%		76%
org.eclipse.jdt.internal.core.util		68%		57%
org.eclipse.jdt.internal.eval		8%		3%
org.eclipse.jdt.core.dom		84%		76%
org.eclipse.jdt.internal.compiler.lookup		86%		80%
org.eclipse.jdt.internal.compiler.parser		86%		79%
org.eclipse.jdt.internal.codeassist		83%		69%
org.eclipse.jdt.internal.compiler.ast		90%		82%
org.eclipse.jdt.internal.compiler.impl		53%		40%
org.eclipse.jdt.internal.formatter		86%		80%
org.eclipse.jdt.internal.core.jdom		41%		31%
org.eclipse.jdt.internal.compiler		83%		77%
org.eclipse.jdt.internal.compiler.codegen		81%		71%
org.eclipse.jdt.internal.core.search.matching		86%		77%
org.eclipse.jdt.internal.compiler.problem		81%		73%

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

API Conformance Testing

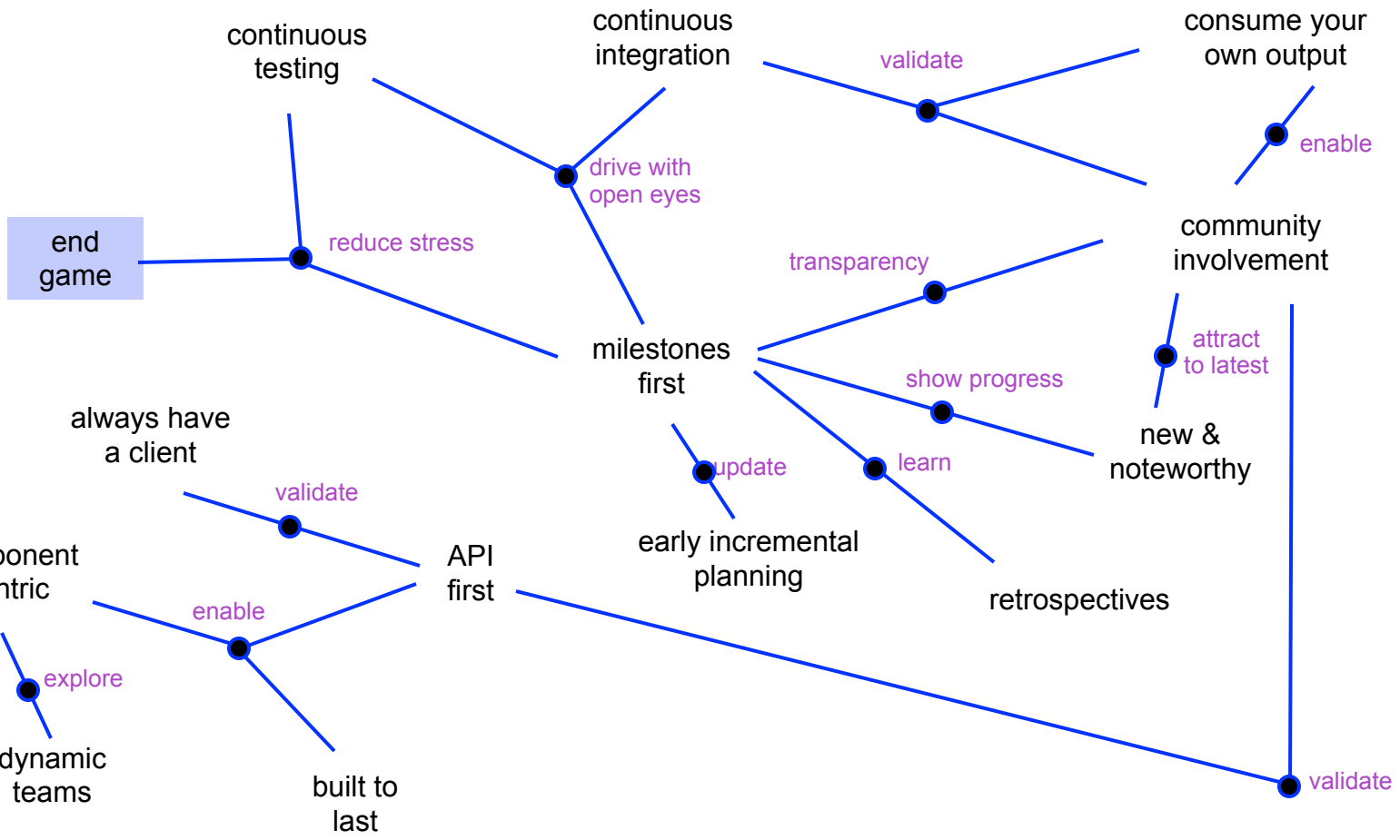
API Tools Verification Reports

List of bundles not configured for API analysis.

Individual report	Compatibility Warnings	API Usage Warnings
org.eclipse.ant.core	0	2
org.eclipse.ant.ui	0	8
org.eclipse.compare	0	5
org.eclipse.core.jobs	0	1
org.eclipse.core.runtime.compatibility	0	6
org.eclipse.debug.ui	0	9
org.eclipse.equinox.event	0	1
org.eclipse.equinox.http.servlet	0	1
org.eclipse.equinox.p2.artifact.repository	0	4
org.eclipse.equinox.p2.director	0	14
org.eclipse.equinox.p2.director.app	0	1

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

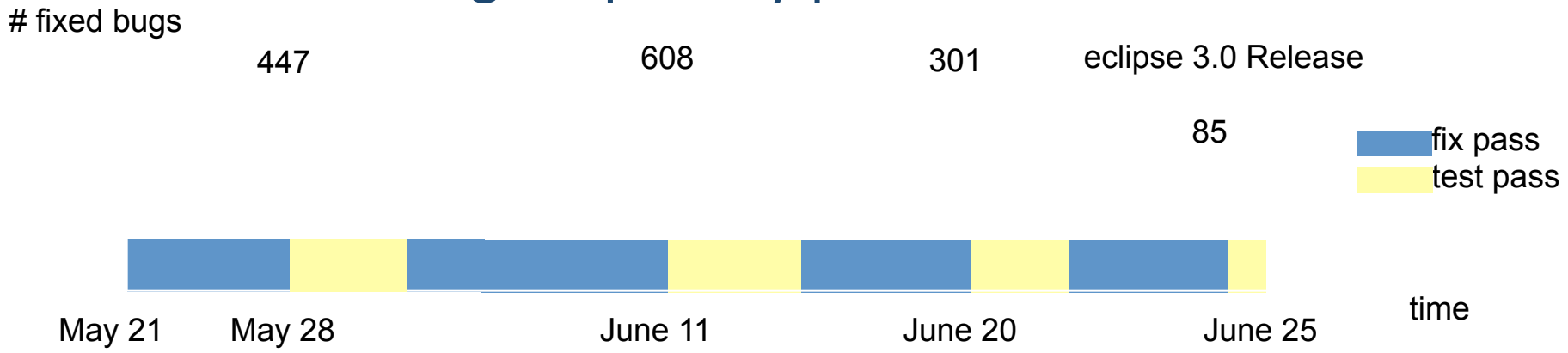
End Game



Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

End Game Convergence

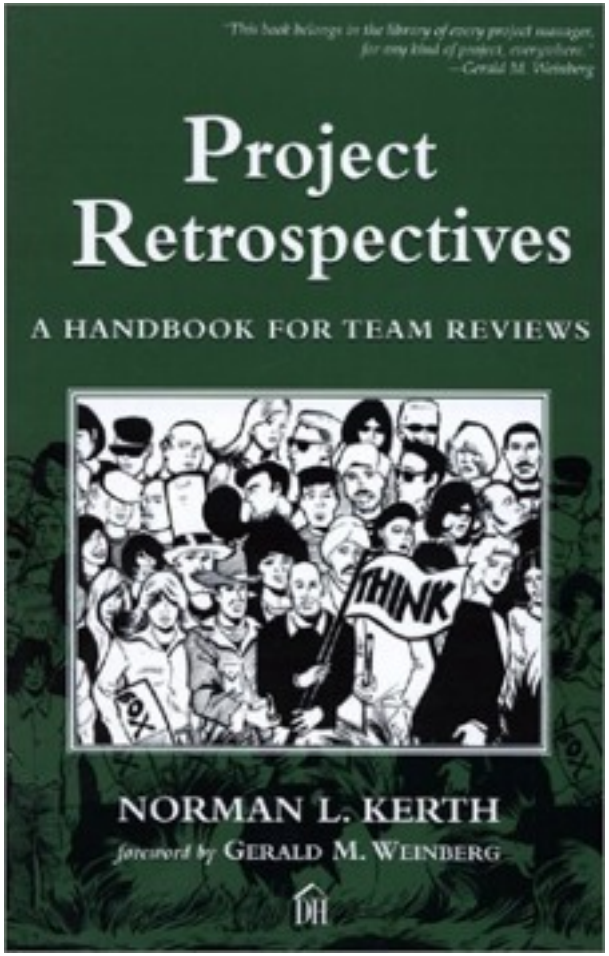
- with each pass the costs for fixing are increased
  - higher burden to work on fix for a problem
  - higher burden to release a fix for a problem
  - focus on higher priority problems



Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Decompression

- recover from release
- retrospective of the last cycle
  - learn from the last cycle
    - achievements
    - failures
  - "stay aware, adapt, change"
  - define retrospective actions

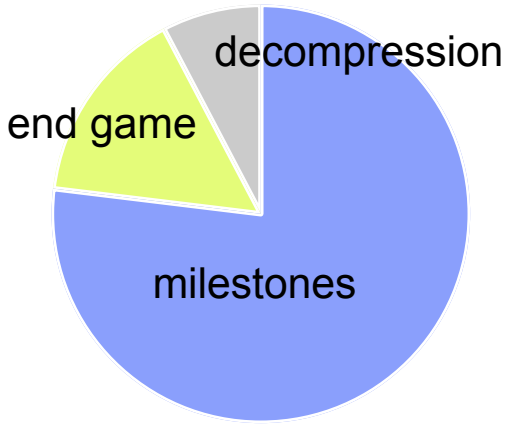


- start to plan the next release and cycle

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Where the Time Goes

- release cycle 12 months
  - milestones – 9 months
  - endgame – 2 months
  - decompression – 1 month



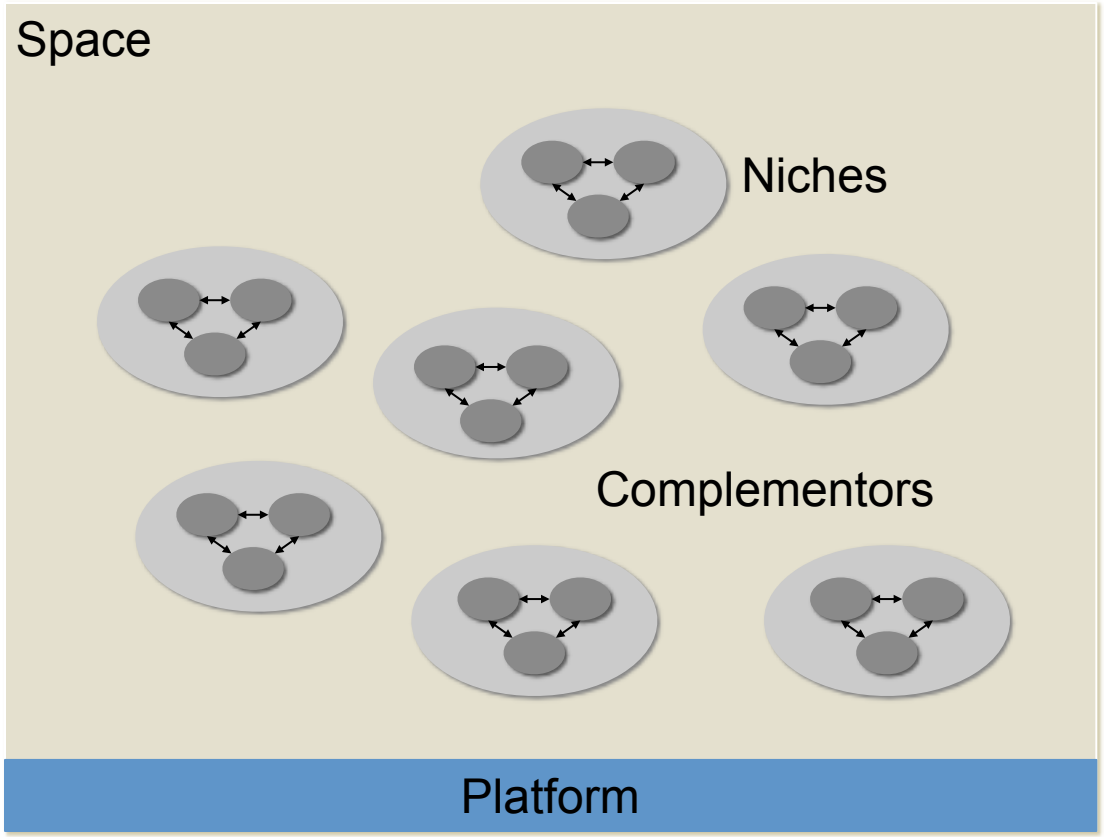
Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Conclusions

- Open source uses highly rigorous and disciplined processes
- Adopt these principles:
  - Meritocracy
  - Openness
  - Transparency

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Think in Platforms



Bosch here

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0

Thank You!

Questions?

Copyright (c) 2011, Eclipse Foundation, Inc. Made available under the Eclipse Public License 1.0