Ecstasy of the Commons

Donald Smith
Director of Ecosystem Development
Agenda

- Tragedy of the Commons
- Anatomy of a Community
- “It’s about the conversation, stupid”
  - Start the conversation
  - Make it easy to join the conversation
  - Feed the conversation
  - Invite others to converse
  - Remember to thank the contributors
  - Be part of the conversation
  - Pitfalls & Role of “traditional” marketing
- Summary
Agenda

- Tragedy of the Commons
- Anatomy of a Community
- “It’s about the conversation, stupid”
  - Start the conversation
  - Make it easy to join the conversation
  - Feed the conversation
  - Invite others to converse
  - Remember to thank the contributors
  - Be part of the conversation
  - Pitfalls & Role of “traditional” marketing
- Summary
Tragedy of the Commons

- Concept popularized by ecologist Garrett Hardin in a 1968 article of the same name

- Simply put - shared resources are doomed to over-exploitation
  - Oceans – over fished
  - Communal fields – over grazed
  - Noise Pollution – Jet noise in residential areas

- Causes are deeply rooted in solid economic, psychological and game theory
Ecstasy of the Commons

- Open Source is very much a shared resource

- But the *more you use open source, the better it gets!*
  - Better quality and more secure.
  - More eyeballs, more extensions and adapted - the better it gets

- Lawrence Lessig calls this “Comedy of the Commons” – example he uses is “language”
  - The more people who speak a language, the more portable it is, the more prose and information that is generated, etc.
  - Basically true for any “Network”

http://www.flickr.com/photos/dennisflax/185734854/
Agenda

- Tragedy of the Commons
- Anatomy of a Community
- “It’s about the conversation, stupid”
  - Start the conversation
  - Make it easy to join the conversation
  - Feed the conversation
  - Invite others to converse
  - Remember to thank the contributors
  - Be part of the conversation
  - Pitfalls & Role of “traditional” marketing
- Summary
The Anatomy of a Community
Community Roles

- Typical open source projects have two communities: committers and users
  - Some projects have one community because the committers are the only users
  - Some projects have three communities: committers, non-committer contributors, and users

- Eclipse projects have three communities: (Mozilla and others too!)
  - Committers and contributors
  - Users – those who use the exemplary tools
  - Plug-in developers – those who use the extensible frameworks

Mozilla and others have three communities
Users

- An active and engaged user community is proof-positive that the project's exemplary tools are useful and needed.
- Furthermore, a large user community is one of the key factors in creating a viable ecosystem around an Eclipse project, thus bringing additional open source and commercial organizations on board.
- Like all good things, a user community takes time and effort to bring to fruition, but once established is nicely self-sustaining.
Users

- To discover projects
  - Read the project home pages
  - Download releases and milestones
  - Monitor the blogs (http://www.planeteclipse.org/)
  - Subscribe to the project newsgroup
  - Watch the screencasts
  - Try the tutorials
  - Use a distribution

- About the future
  - The Eclipse Roadmap (linked from home page)
Community Roles
Committers

- An active, open, transparent, and inclusive community of Committers, developers, and other non-Committer contributors is essential for the health of the project.
- Attracting new contributors and committers to an open source project is time consuming and requires active recruiting, not just passive "openness"
  - The project leadership must go out of its way to encourage and nurture new contributors.
- A Committer community comprised entirely, or even in the majority, from a single company is contra-indicated for a project's long-term success as an open source project.
- The long term viability of an open source project is a function of its developer community, not of the code base.
Eclipse projects are meritocracies … translates to you have to prove your abilities before being accepted.

- Start with well-formed bug reports and feature requests.
- Build a good reputation with the existing committers on a project.
- Propose code enhancements and volunteer time to the project.
- After demonstrating your skills, ask a current project committer to sponsor you as a committer. The election process is defined in the project’s charter.
Community Roles

- Users
- Committers
- Plug-In Developers
Plug-in Developers

- An active and engaged plug-in developer community is only way to prove that an Eclipse project is providing extensible frameworks and extensible tools accessible via documented APIs.
- Reuse of the frameworks within the companies that are contributing to the project is necessary, but not sufficient to demonstrate a plug-in community.
- Creating, encouraging, and nurturing a plug-in developer community outside of the project's developers takes time, energy, and creativity by the project leadership, but is essential to the project's long-term open source success.
Plug-in Developers

- Your friends are:
  - Developer mailing lists
  - Project newsgroups
  - Weekly integration builds
  - API documentation
    - Your clear and detailed feedback on the APIs
  - Bugzilla
  - Patch files
The Eclipse Example

- The purpose of the Eclipse projects is "a vendor-neutral, open development platform supplying frameworks and exemplary, extensible tools... [the] tools are extensible in that their functionality is accessible via documented programmatic interfaces."
  - Eclipse Foundation Bylaws

- Essential to that purpose is the development of three inter-related communities around each project: committers, plug-in developers, and users

- The absence of any one or more of these communities as proof that the project is not sufficiently open, transparent and inviting, and/or that it has emphasized tools at the expense of extensible frameworks or vice versa

*Much of the same can be said of Mozilla*
Technology-based Communities (Eclipse Example)
The Big picture
Metcalfe’s Law Works for Communities as Well
The Fourth Dimension….

- We’ve already talked about:
  - Users
  - Committers
  - Plug in Developers
- We could segment these further – QA teams, build teams, Engineering managers, architects
- But there is a fourth dimension….

Marketing!
Marketing and Open Source

- “Marketing” has a bad name in many developer circles
  - “Flacksters and Hucksters”
  - In fact, we are all in Marketing

- Marketing is about communication
  - Specifically – conversations (to be discussed further)

- One reason Open Source works so well is developers are more empowered to promote and nurture their products and communities directly
Agenda

- Tragedy of the Commons
- Anatomy of a Community
- “It’s about the conversation, stupid”
  - Start the conversation
  - Make it easy to join the conversation
  - Feed the conversation
  - Invite others to converse
  - Remember to thank the contributors
  - Be part of the conversation
  - Pitfalls & Role of “traditional” marketing
- Summary
Agenda

- Tragedy of the Commons
- Anatomy of a Community
- “It’s about the conversation, stupid”
  - Start the conversation
  - Make it easy to join the conversation
  - Feed the conversation
  - Invite others to converse
  - Remember to thank the contributors
  - Be part of the conversation
  - Pitfalls & Role of “traditional” marketing
- Summary
Communities are about conversations
Cluetrain Manifesto – 95 Theses

1. Markets are conversations


3. Conversations among human beings sound human. They are conducted in a human voice.

9. These networked conversations are enabling powerful new forms of social organization and knowledge exchange to emerge.

62 Markets do not want to talk to flacks and hucksters. They want to participate in the conversations going on behind the corporate firewall.

90 Even at its worst, our newfound conversation is more interesting than most trade shows, more entertaining than any TV sitcom, and certainly more true-to-life than the corporate web sites we've been seeing.
Agenda

- Tragedy of the Commons
- Anatomy of a Community
- “It’s about the conversation, stupid”
  - Start the conversation
  - Make it easy to join the conversation
  - Feed the conversation
  - Invite others to converse
  - Remember to thank the contributors
  - Be part of the conversation
- Pitfalls & Role of “traditional” marketing
- Summary
Starting the conversation

Developers want to talk to developers

Fire the Marketers!
“My development manager thinks we are being out marketed, so I would like to do a press release….”

- Developers and Marketers need to share the responsibility of creating the community.
- We are all marketers!
- BTW, press releases do not create communities 😊
Starting the conversation

The Conversation Starts with Code

18 month release cycles don’t cut it;
Try 6 weeks
Case Study

- Release due June 2006 but in January 2006 already 64,100 Google pages on “Eclipse 3.2”. August 2006 = 371,000. Oct 8 = 480,000
The Eclipse 3.2 conversation started last September, 2005
9 months before the release!
It continues in many languages
The community decides what is cool; not the vendor
Joining the conversation

Make it easy to join
the conversation

- Need to have low barriers to entry
- Tutorials, white papers, books and CODE!
- Have experts monitor newsgroups and bug database
BIRT Tutorials

- Basic Tutorial
  - Flash Movie of "My First Report"
  - Other Screencasts

Basic Tutorial

The Basic Tutorial walks you through the steps to build a basic listing report. Introduces the basic BIRT concepts. This is the place to start.

Flash Movie of "My First Report"

This Flash movie shows the creation of "My First Report" from the tutorial in BIRT's online help.

Other Screencasts

In the following screencasts, Scott Rosenbaum (BIRT PMC) show how to perform some specific tasks in BIRT:

- Using data set parameters and report parameters.
- Creating a subreport.
- Creating a sculpted data set.
Eclipse Rich Client Platform started as a community initiative
Not planned by a vendor
Joining the conversation

Create an Architecture of Participation

- Enable users and other parties to extend and modify your technology
- Eclipse, Apache, Firefox
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
Eclipse makes it easy for third parties to create plug-ins

The result has been one of the secrets to our success
Agenda

- Tragedy of the Commons
- Anatomy of a Community
- “It’s about the conversation, stupid”
  - Start the conversation
  - Make it easy to join the conversation
  - Feed the conversation
  - Invite others to converse
  - Remember to thank the contributors
  - Be part of the conversation
  - Pitfalls & Role of “traditional” marketing
- Summary
Feeding the conversation

Create and publish technical content

- Technical content feeds the conversation
- People love to generate links
An article creates the seed for the conversation
Article linked to from different newsgroups and blogs
Eclipse Web Tools now starts showing up in all sorts of places

Extending the conversation to new people
People also save it as a resource
Feeding the conversation

Nurture and promote your tribal leaders

- Persona of the community
- Speak at conferences; write books
- Often define the culture of the community
Feeding the conversation

Be as transparent as possible

- Public project plans
- Open bug tracking databases
17 people watching!

➢ 81 comments
Agenda

- Tragedy of the Commons
- Anatomy of a Community
- “It’s about the conversation, stupid”
  - Start the conversation
  - Make it easy to join the conversation
  - Feed the conversation
  - Invite others to converse
  - Remember to thank the contributors
  - Be part of the conversation
  - Pitfalls & Role of “traditional” marketing
- Summary
Expanding the community

Get your competitors involved

- Allows you to link into other established communities
Eclipse Web Tools Project

Overview of the Eclipse Web Tools Platform - Mozilla Firefox

Building dynamic Web applications
JavaKarta-Tomcat. Learn how to install complete web application using JST

Introduction
Eclipse is a rich platform for developing dynamic web applications and is well suited to
JavaKarta-Tomcat. The persistent data layer

 vielen different JSPs, JavaServer Pages Standard Tag Lib

In order to get the most out of this article we have some knowledge of Eclipse. Some of
Tip: Learn more on these topics please refer

IBM Cloudscape is the commercial release
generally, unless referenced to a specific

Eclipse WTP Project

OVERVIEW OF THE ECLIPSE WEB TOOLS PLATFORM
by Tom Wagner, Ted D'Eon, Paul Meyer, and Pieter Humphrey
6/22/2005

Abstract
The Web Tools Platform (WTP) project extends Eclipse into the domain of Web and
to enterprise J2EE application development. This open source project is dedicated to
providing a generic, extensible, standard-compliant tool suite for managing

eXtensible Markup Language (XML) extensions, including a customizable project model and application server support.
This article provides an overview of WTP's sub-projects—Web Standard Tools
(WST) and J2EE Platform Standard Tools (JST)—and explores the major themes of
the 0.7 release:

- Extend the Eclipse platform for J2EEWeb application development
- Define servers, runtime, and modules
- Provide platform APIs and suitable models for tool developers
- Provide core tools for application developers
  - A set of high-quality tools in their own right, plus exemplars for tool developers
  - A collection of high-quality editing services for Web and J2EE-related languages
Agenda

- Tragedy of the Commons
- Anatomy of a Community
- “It’s about the conversation, stupid”
  - Start the conversation
  - Make it easy to join the conversation
  - Feed the conversation
  - Invite others to converse
  - Remember to thank the contributors
- Be part of the conversation
- Pitfalls & Role of “traditional” marketing
- Summary
Giving back to the community

Thank the contributors

- Reference blog postings; bloggers love links
- Acknowledge community leaders
- This also help Google results
Giving back to the community

Promote companies that use or contribute to the community
Agenda

- Tragedy of the Commons
- Anatomy of a Community
- “It’s about the conversation, stupid”
  - Start the conversation
  - Make it easy to join the conversation
  - Feed the conversation
  - Invite others to converse
  - Remember to thank the contributors
  - Be part of the conversation
  - Pitfalls & Role of “traditional” marketing
- Summary
Be Part of the Community

If you love something, set it free

➢ Win by letting go
Agenda

- Tragedy of the Commons
- Anatomy of a Community
- “It’s about the conversation, stupid”
  - Start the conversation
  - Make it easy to join the conversation
  - Feed the conversation
  - Invite others to converse
  - Remember to thank the contributors
  - Be part of the conversation
- Pitfalls & Role of “traditional” marketing
- Summary
Pitfalls

- Not all developers are good communicators; or want to be
- Marketers still want to be in control
  - Remember you win by giving up control
- Too easy to fall back into the old way of doing things
- How do you know when you are successful?
What is the marketing role

- Still need to do top-down marketing
  - High level messaging for non-technical decision makers
  - Provide support for developers trying to sell up the chain

- Be part of the community
  - Developers want to their community to be successful
  - Make sure you speak their language
  - Mentor the developers!

- Create a partnership between marketing & development
Agenda

- Tragedy of the Commons
- Anatomy of a Community
- “It’s about the conversation, stupid”
  - Start the conversation
  - Make it easy to join the conversation
  - Feed the conversation
  - Invite others to converse
  - Remember to thank the contributors
  - Be part of the conversation
  - Pitfalls & Role of “traditional” marketing
- Summary
Summary

- Start the conversation
- Need to make it easy to join the conversation
- Feed the conversation
- Invite others to be part of the community
- Remember to thank the contributors
- Be part of the community
A Final Thought…
The conversation escapes IT…
...again and again...
...further feeding the conversation!
Q&A