Who am I?

- Ross Turk, SourceForge, Inc.
  - Email: rturk@corp.sourceforge.com
  - Jabber: rossturk@gmail.com

- My job: Community Manager
  - Go to our users, wherever they are
  - Talk to SourceForge.net users
  - Learn from them
  - Drive the site’s strategic direction
What are we talking about?

• What is SourceForge.net?
  – Basic usage stats
  – Demographics

• Case Studies
  – Projects that have done things right

• Analysis
  – What did they do right?

• Best Practices
  – What should I do?
What is SourceForge.net?

“The whole is more than the sum of its parts.”
- Aristotle
SourceForge.net is...

• ...the most substantial OSS repository, founded in 1999 and operated by OSTG.
• ...a provider of tools and technology.
• ...a marketplace for OSS development and consumption.
SourceForge is big... really big

• SF.net website (in 03/2006):
  – 25 million unique visitors
  – 106 million page views
  – 250,000 searches per day
  – 1.55 million registered users
  – 142,000 registered projects
SourceForge is big... really big

  - 359 million page views (10.3 TB)
  - 67 million files downloaded (639.4 TB)
Always growing

Registered Users

2000 2001 2002 2003 2004 2005 2006 2007

0 250,000 500,000 750,000 1,000,000 1,250,000 1,500,000
Always growing

Registered Projects

- 2000: 0
- 2001: 30,000
- 2002: 60,000
- 2003: 90,000
- 2004: 120,000
- 2005: 150,000
- 2006: 180,000
- 2007: 210,000
Geographic Breakdown

- United States: 18.9%
- Other: 39.7%
- Germany: 8.4%
- Spain: 5.5%
- Italy: 5.5%
- France: 5.4%
- UK: 4.5%
- Brazil: 3.5%
- Canada: 3.2%
- Japan: 2.7%
- Poland: 2.7%
Case Studies: Projects that have done things right.

“If we knew what we were doing, it wouldn't be research.”
- Albert Einstein
Done things “right”?

• Projects have done things right when:
  – The main objective has been achieved
  – Users are happy with the solution

• Other signs:
  – Derivatives or forks are created
  – The solution has wide distribution
Introducing Audacity

• Purpose:
  – Allow users to record and edit multi-track audio files

• Audience:
  – Audio hobbyists, garage bands, and anyone else who deals with audio
Audacity: Development

• First released in May of 2000
• Developers: 6
• Releases: 21 total
• Commits: 352 (past 5 months)
  – 2-3 commits per day average
Audacity: Consumption

- **Major Release**


  Consumption (in units):
  - 2001: 0
  - 2002: 200,000
  - 2003: 375,000
  - 2004: 750,000
  - 2005: 1,125,000
  - 2006: 1,500,000

SourceForge.net
Introducing PDFCreator

• Purpose:
  – Allow creation of PDFs via the print mechanism in Windows

• Audience:
  – Windows users who want to create PDFs
  – ...but don’t need the full features of something like Adobe Acrobat or don’t want to be limited by its license.
PDFCreator: Development

- Started in July of 2002
- Developers: 2
- Commits: 146 (past 9 months)
  - 1 commit every two days average
- Releases: 9 total
- Average times...
  - to close a bug: 9 days
  - to close a patch: 15 days
  - to close a feature request: 11 days
PDFCreator: Consumption

- **Major Release**

<table>
<thead>
<tr>
<th>Year</th>
<th>Consumption</th>
</tr>
</thead>
<tbody>
<tr>
<td>2003</td>
<td>137,500</td>
</tr>
<tr>
<td>2004</td>
<td>275,000</td>
</tr>
<tr>
<td>2005</td>
<td>412,500</td>
</tr>
<tr>
<td>2006</td>
<td>550,000</td>
</tr>
</tbody>
</table>
Introducing: Pidgin (was Gaim)

• Purpose:
  – Allow users to communicate using multiple instant messaging protocols at the same time

• Audience:
  – Desktop users, people who chat (which is everybody)
Pidgin: Development

• Started in November of 1998
• Developers: 24 (~12 core)
• Commits: 1998 (past 9 months)
  – 7 commits per day average
• Releases: 107 total
• Average times...
  – to close a bug: 29 days
  – to close a patch: 16 days
  – to close a feature request: 21 days
Pidgin: Consumption

- Notable releases


Consumption: 0, 150,000, 300,000, 450,000, 600,000

SourceForge.net
Introducing: Nagios

• Purpose:
  – Host and system monitoring tool

• Audience:
  – Systems administrators, IT management, service quality analysts
Nagios: Development

• Started in January of 1999
• Developers: 1 core
• Commits: 696 (past 6 months)
  – 3-4 commits per day average
• Releases: 35 total since May of 2002
  – ~1 release every two months average
Introducing: phpMyAdmin

• Purpose:
  – Web-based management of MySQL servers

• Audience:
  – Database administrators, developers using MySQL, system administrators, ISPs, education
phpMyAdmin: Development

• Started in 1998
• Developers: 8
• Commits: 383 (past 9 months)
  – 1-2 commits per day average
• Releases: 130
• Average times...
  – to close a bug: 47 days
  – to close a patch: 70 days
  – to close a feature request: 49 days
phpMyAdmin: Consumption

- Major Release

Year: 2002
- Consumption: 27

Year: 2003
- Consumption: 87,500

Year: 2004
- Consumption: 175,000

Year: 2005
- Consumption: 262,500

Year: 2006
- Consumption: 350,000

SourceForge.NET
Introducing: Gallery

• Purpose:
  – Web-based photo management

• Audience:
  – People who want to share pictures
• Started in July of 2002
• Developers: 22
• Commits: 2163 (past 9 months)
  – 7-8 commits per day average
• Releases: 101 total
  – Almost 2 per month average
• Average times...
  – to close a bug: 30 days
  – to close a patch: 11 days
  – to close a feature request: 51 days
Analysis:
What did they do right?
They have a good “problem”

• Every project is trying to solve a problem.
  – “Every good work of software starts by scratching a developer's personal itch.” - ESR

• Every solution has an audience.
  – Who wants it to be solved?

• Every solution has an impact.
  – How often is it used?
  – How important is it to the user?
Understanding the “problem”

- Nagios
- Gallery
- phpMyAdmin
- Ross’s project that nobody uses
- Firefox
- Pidgin
- Audacity
- PDFCreator
Their barrier to entry is low

- Easy installation processes
  - Gallery has a good web-based installer
  - phpMyAdmin just gets untarred :)
  - Audacity / PDFCreator have native installers

- Cross-platform support
  - Audacity works on Linux, OS X, Windows
  - Gallery and phpMyAdmin run anywhere
  - Pidgin works on Linux and Windows
    - Adium for OS X leverages libpurple
Their software works

- Extensive refactoring has improved quality
  - Pidgin: separation of libpurple
  - Gallery: complete rewrite for 2.0
  - Gallery: automated test framework
  - Audacity: separation of mezzo (in the works)

- Because consumption is high:
  - New versions are verified quickly
  - Bugs are reported in larger numbers
“Jagged binding surfaces”

• Most of these projects are hackable
  – Pidgin: Write against libpurple, write plugins
  – Gallery: Can write themes and plugins
  – Nagios: Can call external macros to do almost anything
  – Audacity: Can write new effect plugins
  – Gallery, phpMyAdmin, Audacity, Nagios, and Pidgin have wikis
They leverage standards

• These projects support tech standards that are accepted and proven.
  – PDFCreator: Leverages the popular PDF standard and GhostScript
  – Gallery: Supports WebDAV, RSS, and too many more to list
  – Audacity: Supports most audio formats, AudioUnit and VST plugins
  – Pidgin: Supports common IM protocols

• This increases the potential audience
Best Practices: What should I do?

“Success is simply a matter of luck. Ask any failure.” - Earl Nightingale
Understand the barriers

Potential users

Casual Users

Involved Users

Contributors

Should I bother downloading this?
Should I bother installing this?

Should I bother submitting this defect?
Should I bother asking for this feature?

Should I bother fixing this defect myself?
Solve a good problem

• Create something you care about
  – Someone else probably will too.

• Serve a good purpose
  – Solve a new problem
  – Solve an existing problem better

• Have a clear, basic objective
Use the community

• Keep your barrier to entry low
  – Make it easy to install
    • More users means more potential for participation
  – Keep it easy to provide feedback
• Follow up with outside contributors
  – Close out bugs, accept patches, and implement features quickly
• Accept criticism gracefully
• Be willing to give up some control
• Have “groupies”
Design for flexible use

- Have an extensible design
  - Create “jagged binding surfaces” for contributors

- Adopt designs and processes that allow you to make a quality product
  - Automated testing
  - Logic separation

- Leverage existing standards if they apply
Questions / Comments?

Ross Turk
[rturk@corp.sourceforge.com]