Understanding Continuous Design in F/OSS Systems

Les Gasser

Graduate School of Library and Information Science
University of Illinois at Urbana-Champaign
gasser@uiuc.edu

Joint work with Dave Dubin, Gabriel Ripoche, Bob Sandusky and Bryan Penne
Collaboration with LIMSI-CNRS / ParisXI and U.C Irvine Institute for Software Research

Supported by NSF/ITR (Digital Society and Technologies) Program
**Initial Question:**

How do large online communities manage continuous streams of software errors, mistakes, problems?
Examples cross domains:

- Mozilla Web Browser
- OpenOffice.org/StarOffice Office Suite
- Chandra X-Ray Observatory Science Software
- MMPORG game “mods”
- Linux Kernel
Why study bugs?
Fundamentally big deals:

- Right vs Wrong, Good vs Evil
- Relationships of Knowledge and Action
- Paradise model vs. Reincarnation model
  \[ \downarrow \]
  \[
  \begin{align*}
  \text{Ultimate Ideal} & \quad \text{“Continuous Improvement”}
  \end{align*}
  \]
- “Healing” vs. “Chronic Pain Management”
Simple model of software refinement

Delivery

Users
Application

Developers
Tools
Simple model of software refinement
Simple model of software refinement
What to do about deviance?
- Make it explicit
- Represent it

Developers

Tools

Application

Oops!

Delivery

Feedback

Oh No!

Pow!

!?!?
Large-scale software process demands complexity reduction strategies
Summary of SoD Issues

1. How are design and design process complexity managed now? How can they best be managed?
Large-scale software process demands complexity reduction strategies
Unstructured:

Hypermail archives (Chandra X-ray software)  
Newsgroups (www.apple.com)

Structured:

Bugzilla, Scarab, and many others:

<table>
<thead>
<tr>
<th>Structured Formal information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unstructured Informal Information</td>
</tr>
<tr>
<td>Structured database</td>
</tr>
</tbody>
</table>
Repository Structure

Repository

Issue-report

Issue-report

Issue-report

Issue-report

Issue-report

Issue-report

N = ~200,000+

Time (hours to years)

Extended text + picture + code discourse

Duplicate

resolved

open

Dependent

open

open

resolved

open
Reports have structured parts
### Reports include multimedia and code

<table>
<thead>
<tr>
<th>Attachment</th>
<th>Type</th>
<th>Created</th>
<th>Flags</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Patch for docshell and webshell</td>
<td>patch</td>
<td>2001-10-29 16:37:50</td>
<td>none</td>
<td>Edit</td>
</tr>
<tr>
<td>Error page</td>
<td>text/plain</td>
<td>2001-10-29 16:38:49</td>
<td>none</td>
<td>Edit</td>
</tr>
<tr>
<td>Error page javascript</td>
<td>text/plain</td>
<td>2001-10-29 16:39:19</td>
<td>none</td>
<td>Edit</td>
</tr>
<tr>
<td>Does Netscape marketing_not see an opportunity here? Would you rather be redirected to a helpful netscape.com/mozilla.org instead of being forced to click OK?</td>
<td>image/gif</td>
<td>2001-10-30 10:42:11</td>
<td>none</td>
<td>Edit</td>
</tr>
<tr>
<td>Patch for docshell and webshell</td>
<td>patch</td>
<td>2002-02-17 14:32:59</td>
<td>none</td>
<td>Edit</td>
</tr>
<tr>
<td>Error page</td>
<td>text/plain</td>
<td>2002-02-17 14:38:50</td>
<td>none</td>
<td>Edit</td>
</tr>
<tr>
<td>Error page javascript</td>
<td>text/plain</td>
<td>2002-02-17 14:43:04</td>
<td>none</td>
<td>Edit</td>
</tr>
<tr>
<td>Patch for docshell and webshell</td>
<td>patch</td>
<td>2002-02-27 15:45:34</td>
<td>none</td>
<td>Edit</td>
</tr>
<tr>
<td>A page that IE generates for not finding a site.</td>
<td>text/html</td>
<td>2002-03-12 01:05:11</td>
<td>none</td>
<td>Edit</td>
</tr>
<tr>
<td>Work in progress</td>
<td>patch</td>
<td>2002-05-28 14:04:30</td>
<td>none</td>
<td>Edit</td>
</tr>
<tr>
<td>More work in progress</td>
<td>patch</td>
<td>2002-05-29 15:23:01</td>
<td>none</td>
<td>Edit</td>
</tr>
</tbody>
</table>
Reports include up to hundreds of comments

----- Additional Comment #248 From Jason Bassford 2003-02-07 05:15 -----

> It seems the History of the tab is often unavailable when a network error occurs and these error messages are displayed.

I can't remember a time when that HASN'T been the case for me. All history is lost and the back button (naturally) is greyed out.

----- Additional Comment #249 From Jason Bassford 2003-02-11 05:57 -----

> All history is lost and the back button (naturally) is greyed out.

Correction. You CAN still go back by clicking on the Go menu and selecting the entry at the bottom for the site you were at before. So, technically, history is not lost. However, neither the back button nor Go -> Back work - despite the fact that previous URL information obviously does exit.

----- Additional Comment #250 From amutch@dl.lib.mi.us 2003-02-11 07:02 -----

Please remember that this feature is also used by embedders and features that are available in Mozilla may not be available or implemented in the same way that Mozilla does. Any solution to problems needs to take that into account. I know that Adam is aware of that but its a reminder to others working on the bug.

----- Additional Comment #251 From Matti (Matthias "hating marquee" Versen) 2003-02-16 08:17 -----

*** Bug 133559 has been marked as a duplicate of this bug. ***
Slowly dawning insights:

1. This process continues for very long periods and sometimes doesn’t end.

2. This process spreads out across many interwoven instances, people, events, representations.

3. Specifications, needs, and criteria are being invented community-wide and on-the-fly.

Hence, the “continuous re-design” concept and the burning question: How does it work?
Summary of SoD Issues

1. How are design and design process complexity managed now? How can they best be managed?

2. For what classes of systems/environments is pre-specification impossible or undesirable?
Some reports are for party planning

Bugzilla Bug 100309  mozilla1.0 party!

Bug#: 100309
Product: Browser
Component: Browser-General
Status: NEW
Resolution:
Assigned To: asa@mozilla.org (Asa Dotzler)
Summary: mozilla1.0 party!
Whiteboard: kw: nsJunkFood, beerwanted

Platform: All
OS: All
Version: other
Priority: --
Severity: critical
Target Milestone: ---

Description:

we need preparation as well as a good place to have the biggest & coolest party ever! suggestions welcome ....

------ Additional Comments From Jake 2001-09-18 08:58 ------
Let’s have it in my back yard... then I wouldn’t have to fly to California ;)

------ Additional Comments From Chase Tingley 2001-09-18 11:43 ------
I can’t think of a more appropriate place to have this party than in boston -- the, umm, birthplace of mozilla.

------ Additional Comments From Jesse Ruderman 2001-10-09 15:08 ------
I think we should mark this as blocking bug 103705, since you can’t release a major product without having a party. Nobody should complain.

------ Additional Comments From Gervase Markham 2001-10-16 15:43 ------
Wrong way round - you can't have the party without releasing 1.0 ;)

Opened: 2001-09-18 08:55
Some reports are for tracking & coordination

<table>
<thead>
<tr>
<th>Attachment</th>
<th>Type</th>
<th>Created</th>
<th>Flags</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create a New Attachment</td>
<td>(proposed patch, testcase, etc.)</td>
<td>View All</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Bug 122274 depends on:
- 100836, 101403, 101836, 101985, 102790, 102804, 103044, 103083, 103170, 103368, 103732, 103730, 104048, 104237, 104687, 105160, 105258, 105443, 106049, 106327, 106049, 106539, 106712, 106734, 106857, 106858, 107134, 107447, 107878, 108055, 108115, 108209, 108376, 108650, 108906, 109151, 109486, 109995, 112162
- 113334, 113851, 114054, 114211, 114346, 114570, 114695, 114952, 114955, 115091, 115160, 115316, 115599, 115841, 115894, 115906, 115908, 116024, 116702, 117136, 117225, 117452, 117636, 117840, 117852, 118294, 118270, 118817, 118878, 118882, 118887, 119001, 119000, 119062, 119360, 119559, 120273, 120278, 120644, 121660, 122329, 122692, 122716, 123102, 123421, 123687, 124449, 124460, 124750, 125783
- 125823, 127109, 128744, 129374, 138109, 155954

Votes: Show votes for this bug  Vote for this bug
Summary of SoD Issues

1. How are design and design process complexity managed now? How can they best be managed?

2. For what classes of systems/environments is pre-specification impossible or undesirable?

3. What information is critical for effective community design? How is community knowledge organized, managed and integrated into design practice now? How can it best be managed and integrated over long periods?
How do Communities Understand their Artifacts?

Linear/learned model:

- Design ................................. Use

  Specify…design…build…test…release…maintain

  Push costs earlier

Continuous/constructed model:

- Using initial seed, community continuously, jointly transforms its knowledge of what the target artifact is, how it works, what it should be, and how it should work: continuous collective design

- Process is catalyzed by structured collective KM tools (e.g. Bugzilla)

- Process is characterized by high multiplicity of a) viewpoints, b) representations, c) experiences, d) basic social processes.
What model of knowledge representation and organization?

Bounded, faithful reflections model:
"Issue reports represent things, events, experience"

Reports <-- Relationship --> Life-world

Massive, contentious assemblages model:

- Reports record viewpoints (under interpretation)
- Scope of reporting is unknowable (e.g., duplicates, index terms)
- Fidelity of relationship is uncertain (e.g., report vs. problem)
- Relevant life-world is unknowable (e.g., use-contexts/styles)
- Vocabularies and categories are irregular/inconsistent
Specific (Re)Design Process Issues

• Collective sensemaking: How do people collectively "translate" from issue reports to issues to resolutions?

• Design Coordination: How does a community recognize duplicate and redundant issue reports? How does a community recognize interdependencies among issues?

• Collective Action: With 50,000+ open problems, how does the community decide what to do next and how to do it? (How do issue reports shape resolution activity and vice versa?)

• Quality, Cost, Time: Why do some issues persist for long periods while others get resolved quickly? Can this “time-to-resolution” be predicted?
Sensemaking: How do people collectively "translate" from issue reports to issues to resolutions?

Collectively constructed & stabilized objects

NB: Issue reports \(\neq\) Issues!

What is the process? (see below..)
How does a community recognize duplicate or redundant issue reports?

• Duplicate identification is important:
  - Significant duplication occurs
    49,765 reports with duplication / 128,823 bug reports
    = 38% prior chance of being a duplicate
    Avg 3.03 bug reports in a duplication set
  - Unrecognized duplicates lead to redundant work
  - Unrecognized duplicates mean compartmentalized knowledge

• Definition of duplication is progressive and uncertain
  - Early bug reports can be duplicates of later reports (backwards)
  - Data shows that duplicate-marking decisions are negotiated.

  88481:
  1. *** This bug has been marked as a duplicate of 69167 ***
  2. This is not a dupe! The effects are very similar but the underlying conditions are very different. I was specifically asked to file this separately.
  3. Ok my bad reopening

• Human recognition is incomplete
  - Through transitivity analysis we have found duplicates that are not marked
  - We have found late-stage discovery of duplicates
  - Full discovery is mathematically implausible
How does a community recognize interdependencies among issues?

Dependency identification (DI) is important:

- [Numbers not yet analyzed]
- Dependencies represent / refine causal knowledge
- Dependency knowledge helps order work:
  - 88810 c14: “I see the following ways of reducing this problem: 1) Fix bug 77675 (this is something everybody seems to agree on) 2) Fix bug 28586 "use error page, not dialog for inaccessible pages". When that bug is fixed, most error messages would be displayed in an existing window and nothing will have to be raised anywhere. 3) I would imagine that these two bugs do _not_ cover all the cases where windows are raised unnecessary. I believe that we should try to document all of those (we can start in this bug and then file additional bug reports as needed). This way we can have a meaningful discussion of concrete issues instead of just "Mozilla raises windows to often" (which is something I completely agree with, but it’s not getting us anywhere).”

Human DI is progressive and uncertain

- Formal data is incomplete
  - 28586 c38: “bug 91632 was filed as a dependant on this one in order to fix the....” ????
- DI decisions are negotiated.
  - 89939 c16: “I don’t think that this bug blocks bug 61521, because in that bug, we can use nntp: URIs with *hostnames* and msgids, which work already (I think)."
Why do some issues persist for long periods while others get resolved quickly?
Summary of SoD Issues

1. How are design and design process complexity managed now? How can they best be managed?

2. For what classes of systems/environments is pre-specification impossible or undesirable?

3. What information is critical for effective community design? How is community knowledge organized, managed and integrated into design practice now? How can it best be managed and integrated over long periods?

4. *** What is being designed?  
   System/artifact? Information/action environment?  
   User experience/developer experience? Social world / life stream?

5. What collective design practices are most effective and why? How do we measure continuous collective design practices with no end states?
Explaining differences in resolution time

- Prior work: Mis-alignment in distributed activity nets --> difficult resolution.
- Code initial comparative sample for basic social processes:
  
  59 issue reports > 1000 days; 100 issue reports = 30 days
- Preliminary data: more conflict, ambiguity, uncertainty in long reports.
- Hypothesis: duration depends on "time to consensus" on each issue dimension: values, behavior, causes, code.

1. What is the consensus process?
2. What explains "time to consensus"?
3. Through what processes do people reach consensus?
4. How do people initiate and develop "objects" to consense around?
5. How do consensus and objects of consensus co-construct each other?
6. What other "time-to" dimensions could we consider in similar ways?
7. Is consensus necessary for action? Is "time to act" the critical issue?

How to extract processes scalably? ......
### Computational process extraction / modeling

![Bugzilla Activity Log](image)

<table>
<thead>
<tr>
<th>Who</th>
<th>When</th>
<th>What</th>
<th>Removed</th>
<th>Added</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="mailto:matti@mversen.de">matti@mversen.de</a></td>
<td>2001-04-24 09:00:41</td>
<td>Severity</td>
<td>normal</td>
<td>critical</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Status</td>
<td>UNCONFIRMED</td>
<td>NEW</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ever Confirmed</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Keywords</td>
<td>crash</td>
<td></td>
</tr>
<tr>
<td><a href="mailto:pavlov@pavlov.net">pavlov@pavlov.net</a></td>
<td>2001-05-03 17:53:14</td>
<td>Status</td>
<td>NEW</td>
<td>ASSIGNED</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Target Milestone</td>
<td>---</td>
<td>mozilla0.9.1</td>
</tr>
<tr>
<td><a href="mailto:pavlov@pavlov.net">pavlov@pavlov.net</a></td>
<td>2001-05-04 00:24:13</td>
<td>Status</td>
<td>ASSIGNED</td>
<td>RESOLVED</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Resolution</td>
<td>FIXED</td>
<td></td>
</tr>
<tr>
<td><a href="mailto:tpreston@netscape.com">tpreston@netscape.com</a></td>
<td>2001-05-24 14:06:29</td>
<td>Status</td>
<td>RESOLVED</td>
<td>VERIFIED</td>
</tr>
</tbody>
</table>
Note: Max edges = 56 (64); Process Density = 37/56 = .66
How does information structure activity?

What characteristics of $S_0$ (including presence, absence, frequency, history of concepts, BSPs, contexts) differentiate choices?

(How) can these characteristics be generalized for prediction?
Note: Max edges = 1640; Process Density = 258/1640 = .15
1. How are design and design process complexity managed now? How can they best be managed?

2. For what classes of systems/environments is pre-specification impossible or undesirable?

3. What information is critical for effective community design? How is community knowledge organized, managed and integrated into design practice now? How can it best be managed and integrated over long periods?

4. What is being designed?
   - System/artifact?
   - Information/action environment?
   - User experience/developer experience?
   - Social world / life stream?

5. What collective design practices are most effective and why? How do we measure continuous collective design practices with no end states?

6. How do we best organize practices of information flow, knowledge management, and design practice for continuous, recurrent situations? (How do they work?)
End