Software Engineering HS'15

Software Evolution

Thomas Fritz & Martin Glinz

Many thanks to H. Gall, K. Kevic, M. Allen, V. Razmov, M. Lanza, D. Shepherd, M. Felici, G. Murphy

Learning Goals

By the end of this unit, you will be able to:

- Explain why evolution is difficult but inherent
- Describe three categories of reasons why useful software must evolve and identify what kind of reason motivates a given software change
- Reason about and argue whether a given change retains binary/contract compatibility
- Describe the role of issue tracking systems in the software development process
- Discuss the life cycle of a bug report, criteria for writing a good one and general steps for working on one

Importance of Maintenance/Evolution

Maintenance Examples

- Microsoft Windows XP:
  - Released 12/31/2001
  - Service packs ended 8/30/2005
  - Support ended 4/14/2009 (7.5 years)
  - Extended support still possible
- Red Hat Enterprise Linux 3
  - Released 10/23/2003
  - End of life cycle 10/31/2010
  - End of extended support 01/30/2014

Software Maintenance/Evolution

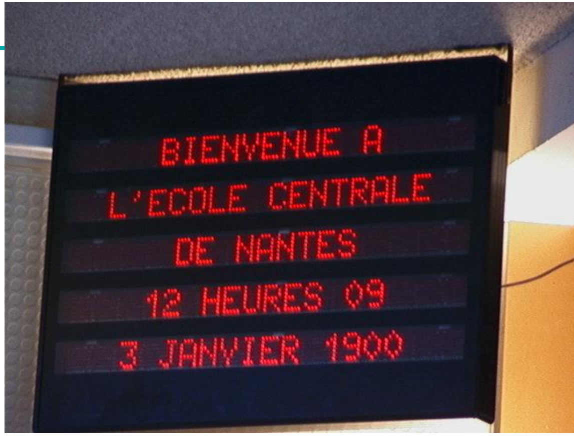
- Producing new (versions of) software under the constraints of existing software
  - Backwards compatibility is often assumed / required
  - aka "Brownfield development"
  - Legacy Software: "software which is vital to our organization, but we don't know what to do with it" (Bennett and Rajlich)
- Can comprise all phases of the lifecycle, starting with requirements gathering
  - Another turn of the spiral

Similar to development, but HARDER!

Software change

- Software change is inevitable
  - New requirements emerge when the software is used
  - The business environment changes
  - Errors must be repaired
  - New computers and equipment is added to the system
  - The performance or reliability of the system may have to be improved
- A key problem for organisations is implementing and managing change to their *existing* software systems.

Y2K Example



- Memory space used to be a problem, so, store 2 digit years
  - Rollover problem: ascending numbering assumption becomes invalid
- Mitigation efforts cost ~$300 billion worldwide
- Valuable surge in IT modernization
  - major issue for business: enterprise architecture

Computerworld: http://goo.gl/1ABH2

Software Evolution

- Organizations have huge investments in their software systems - they are critical business assets.
- To maintain the value of these assets to the business, they must be changed and updated.
- The majority of the software budget in large companies is devoted to evolving existing software rather than developing new software.

Lehman’s laws

Observations (~laws) on large systems developed by large organizations

Continuing change

A program that is used in a real-world environment necessarily must change or become progressively less useful in that environment

Increasing complexity

As an evolving program changes, its structure tends to become more complex. Extra resources must be devoted to preserving and simplifying structure.

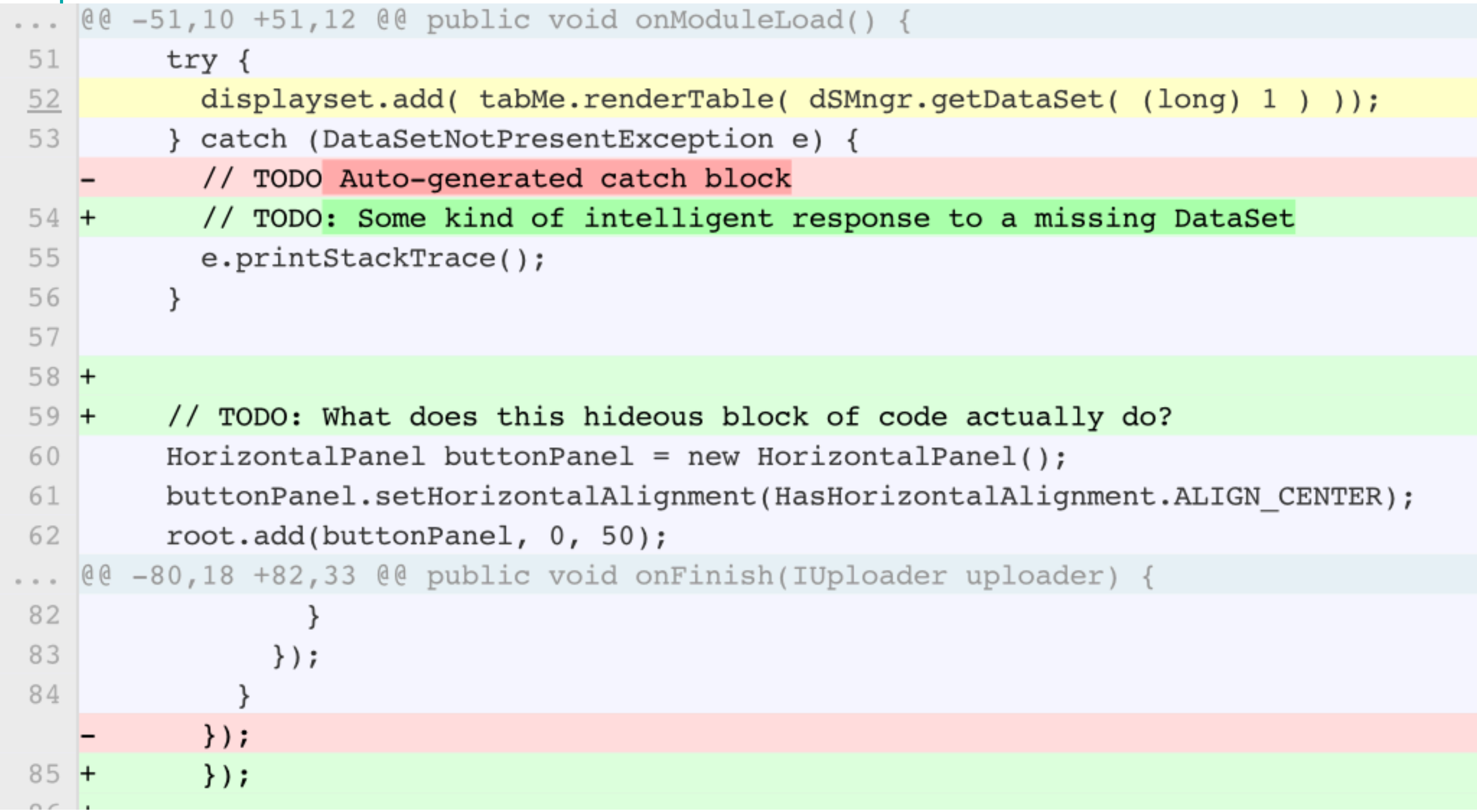
Declining quality

The quality of systems will appear to be declining unless they are adapted to changes in their operational environment

….

Evolution is hard

- Systems not robust under change.
- Lack of traceability (e.g. between requirements and code)
- Poor documentation of code, of design process and rationale and of system evolution.
- “Stupid” code features may not be so stupid.
  - work-arounds of artificial constraints may no longer be documented (e.g. OS bugs, memory limits, etc.)
- Poor management attitudes and culture:
  - Maintenance is not high-profile.
  - It is just patching code.
  - Easier/less important than design.



```
@@ -51,10 +51,12 @@ public void onModuleLoad() {
     try {
       displayset.add( tabMe.renderTable( dSMngr.getDataSet( (long) 1 ) ));
     } catch (DataSetNotPresentException e) {
-      // TODO Auto-generated catch block
+      // TODO: Some kind of intelligent response to a missing DataSet
       e.printStackTrace();
     }

+
+    // TODO: What does this hideous block of code actually do?
     HorizontalPanel buttonPanel = new HorizontalPanel();
     buttonPanel.setHorizontalAlignment(HasHorizontalAlignment.ALIGN_CENTER);
     root.add(buttonPanel, 0, 50);
@@ -80,18 +82,33 @@ public void onFinish(IUploader uploader) {
             }
           });
         }
-      });
+      });
```

Discussion Question

If we build a game like Halo 5, do we ever need to change the application’s source code?

When? Why?

Reasons for evolutionary changes

Corrective

- correct faults in system behaviour
- caused by errors in coding, design or requirements

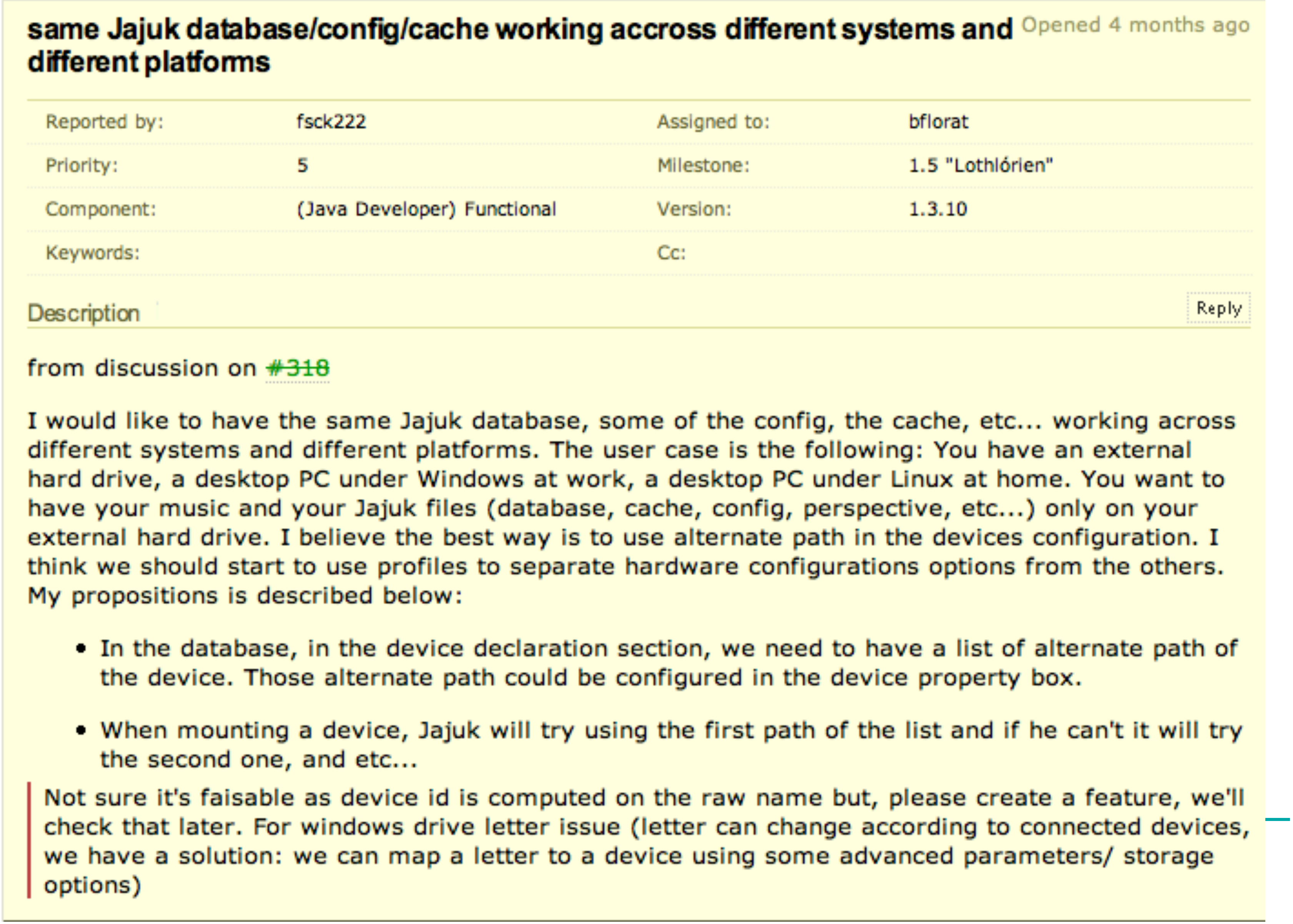
Adaptive

- due to changes in operating environment
- e.g., different hardware or operating system

Perfective

- due to changes in requirements
- often triggered by organizational, business or user learning

What is the reason for this change?



same Jajuk database/config/cache working accross different systems and different platforms Opened 4 months ago

Reported by:	fsck222	Assigned to:	bflorat
Priority:	5	Milestone:	1.5 "Lothlórien"
Component:	(Java Developer) Functional	Version:	1.3.10
Keywords:		Cc:	

Description

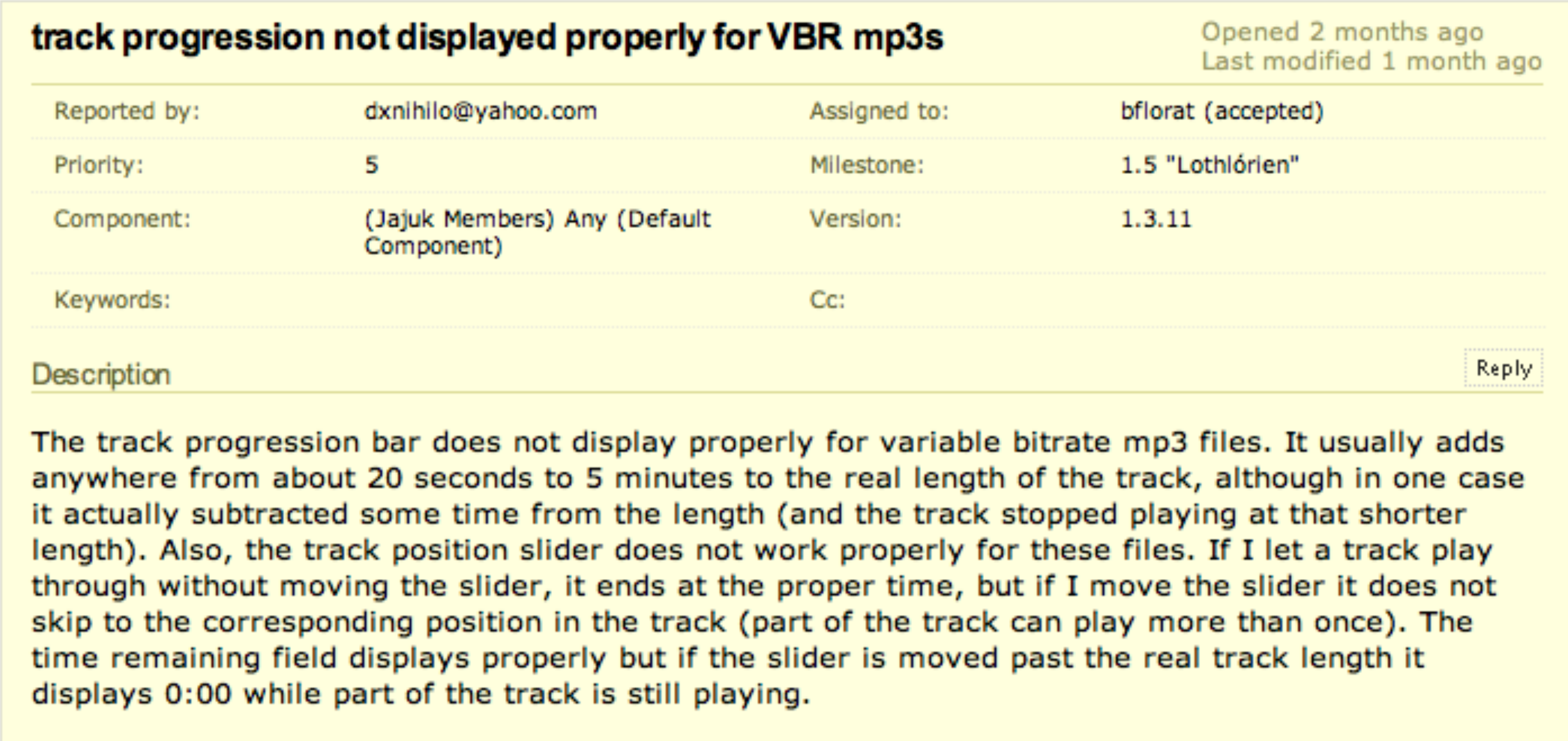
from discussion on ~~#318~~

I would like to have the same Jajuk database, some of the config, the cache, etc... working across different systems and different platforms. The user case is the following: You have an external hard drive, a desktop PC under Windows at work, a desktop PC under Linux at home. You want to have your music and your Jajuk files (database, cache, config, perspective, etc...) only on your external hard drive. I believe the best way is to use alternate path in the devices configuration. I think we should start to use profiles to separate hardware configurations options from the others. My propositions is described below:

- In the database, in the device declaration section, we need to have a list of alternate path of the device. Those alternate path could be configured in the device property box.
- When mounting a device, Jajuk will try using the first path of the list and if he can't it will try the second one, and etc...

> Not sure it's faisable as device id is computed on the raw name but, please create a feature, we'll check that later. For windows drive letter issue (letter can change according to connected devices, we have a solution: we can map a letter to a device using some advanced parameters/ storage options)

What is the reason for this change?



track progression not displayed properly for VBR mp3s

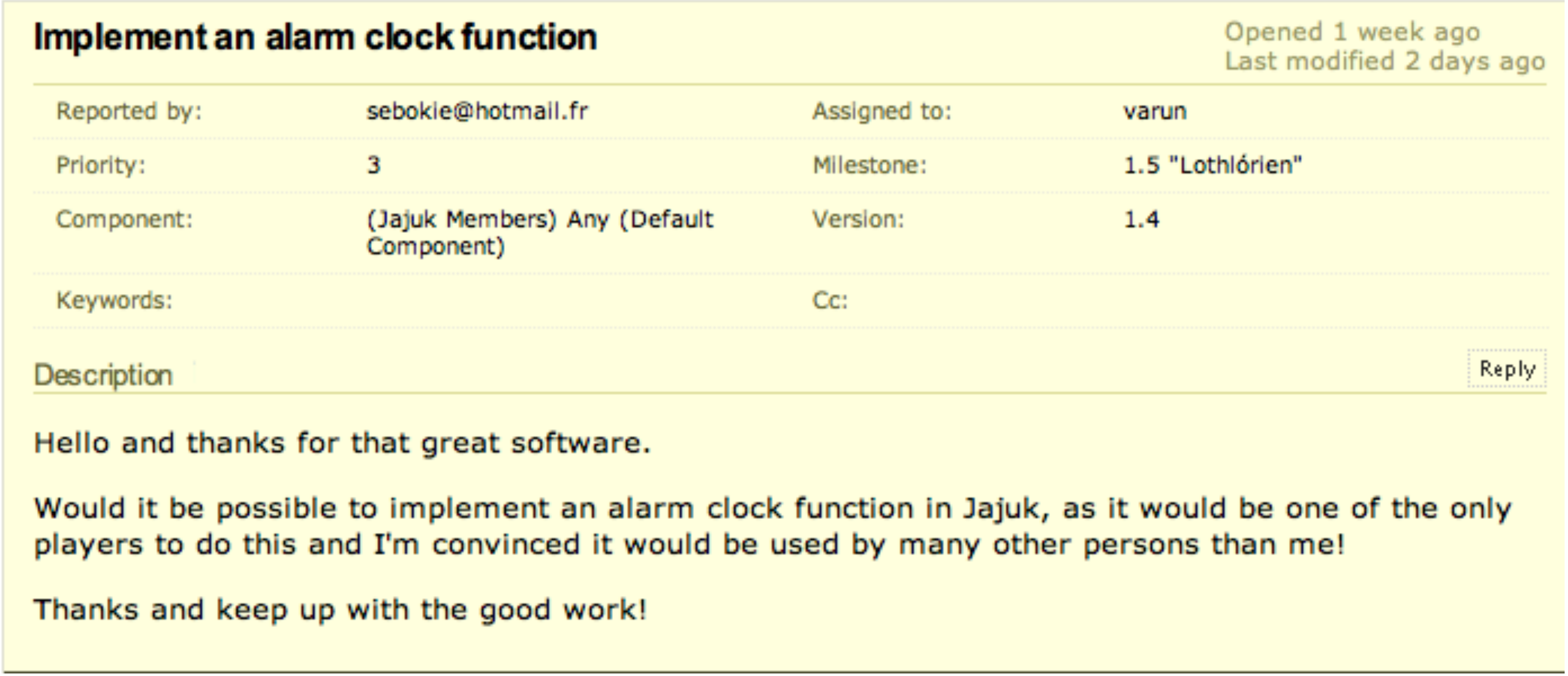
Opened 2 months ago
Last modified 1 month ago

Reported by:	dxnihilo@yahoo.com	Assigned to:	bflorat (accepted)
Priority:	5	Milestone:	1.5 "Lothlórien"
Component:	(Jajuk Members) Any (Default Component)	Version:	1.3.11
Keywords:		Cc:	

Description

The track progression bar does not display properly for variable bitrate mp3 files. It usually adds anywhere from about 20 seconds to 5 minutes to the real length of the track, although in one case it actually subtracted some time from the length (and the track stopped playing at that shorter length). Also, the track position slider does not work properly for these files. If I let a track play through without moving the slider, it ends at the proper time, but if I move the slider it does not skip to the corresponding position in the track (part of the track can play more than once). The time remaining field displays properly but if the slider is moved past the real track length it displays 0:00 while part of the track is still playing.

What is the reason for this change?



Implement an alarm clock function

Opened 1 week ago
Last modified 2 days ago

Reported by:	sebokie@hotmail.fr	Assigned to:	varun
Priority:	3	Milestone:	1.5 "Lothlórien"
Component:	(Jajuk Members) Any (Default Component)	Version:	1.4
Keywords:		Cc:	

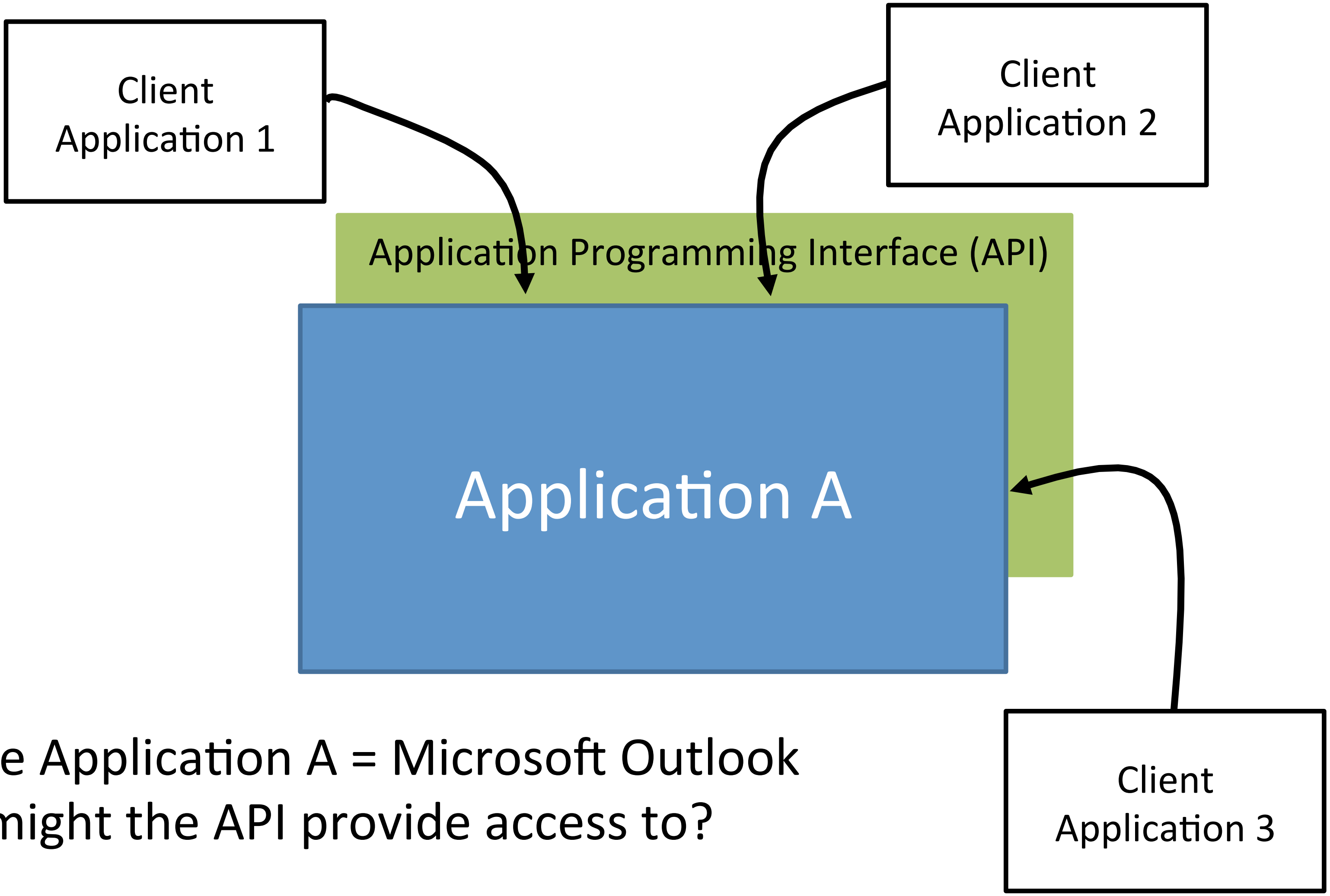
Description

Hello and thanks for that great software.

Would it be possible to implement an alarm clock function in Jajuk, as it would be one of the only players to do this and I'm convinced it would be used by many other persons than me!

Thanks and keep up with the good work!

Evolution and Compatibility



Imagine Application A = Microsoft Outlook
What might the API provide access to?
Why?

API (Java perspective)

e.g., Moyosoft Java Outlook Connector

```
public class SendMail
{
   public static void main(String[] args)
   {
      try
      {
         // Outlook application
         Outlook outlookApplication = new Outlook();

         // Get the Outbox folder
         OutlookFolder outbox = outlookApplication.getDefaultFolder(
                                   FolderType.OUTBOX);

         // Create a new mail in the outbox folder
         OutlookMail mail = (OutlookMail) outbox.createItem(ItemType.MAIL);

         // Set the subject, destination and contents of the mail
         mail.setSubject("Hello world !");
         mail.setTo("your_email@test.com");
         mail.setBody("This is a test message.");

         // Send the mail
         mail.send();
```

Evolving a Java-based API

What kind of changes to the Java API code can we make to maintain **binary compatibility** so that existing (already compiled) applications using the API do not break?

Add a new (Java) package to the API?

Change name of a public method in a public class?

Change the name of a parameter to a method?

Evolving a Java-based API

What kind of changes to the Java API code can we make to maintain binary compatibility so that existing applications using the API do not break?

Add a new (Java) package to the API? (binary compatible)

Change name of a public method in a public class?

(binary incompatible)

Change the name of a parameter to a method?
(binary compatible)

Try these ones…

1. Re-order methods in a class declaration?
2. Add an unchecked exception thrown to an API method?
3. Change an API method from public access to protected access?
4. Add API field?

See http://wiki.eclipse.org/Evolving_Java-based_APIs_2
(don’t memorize these and just understand the very basic kinds of changes)

API Contract Compatibility

API changes must not invalidate formerly legal Client code.

Consider the following API method specification.

```
/** Returns the list of children of this widget.
  * @return a non-empty list of widgets
  */
Widget[] getChildren();
```

API Contract Compatibility…

What if that specification was changed in a revised API to allow an empty list of widgets to be returned?

```
/** Returns the list of children of this  widget.
  * @return a list of widgets
  */
Widget[] getChildren();
```

Would this change break a client who calls `getChildren()`? Why or why not?

API Contract Compatibility…

What if that specification was changed in a revised API to allow an empty list of widgets to be returned?

```
/** Returns the list of children of this  widget.
  * @return a list of widgets
  */
Widget[] getChildren();
```

Would this change break a client who calls `getChildren()`? Why or why not?

```
Consider:
Widget[] children = widget.getChildren():
Widget firstChild = children[0];
```

API Contract Compatibility…

What about this one?

```
/** Removes the given widgets from this widget’s
  * list of children.
  * @param widgets a non-empty list of widgets
  */
void remove(Widget[] widgets);
```

changes to

```
/** Removes the given widgets from this widget’s
  * list of children.
  * @param widgets a list of widgets
  */
void remove(Widget[] widgets);
```

API Contract Compatibility..

Method pre-conditions	Strengthen	Breaks compatibility for callers
	Weaken	Contract compatible for callers
Method post-conditions	Strengthen	Contract compatible for callers
	Weaken	Breaks compatibility for callers

For more, see: http://wiki.eclipse.org/index.php/Evolving_Java-based_APIs

This is just part of the story of contract compatibility as the compatibility depends on the role of the client code.

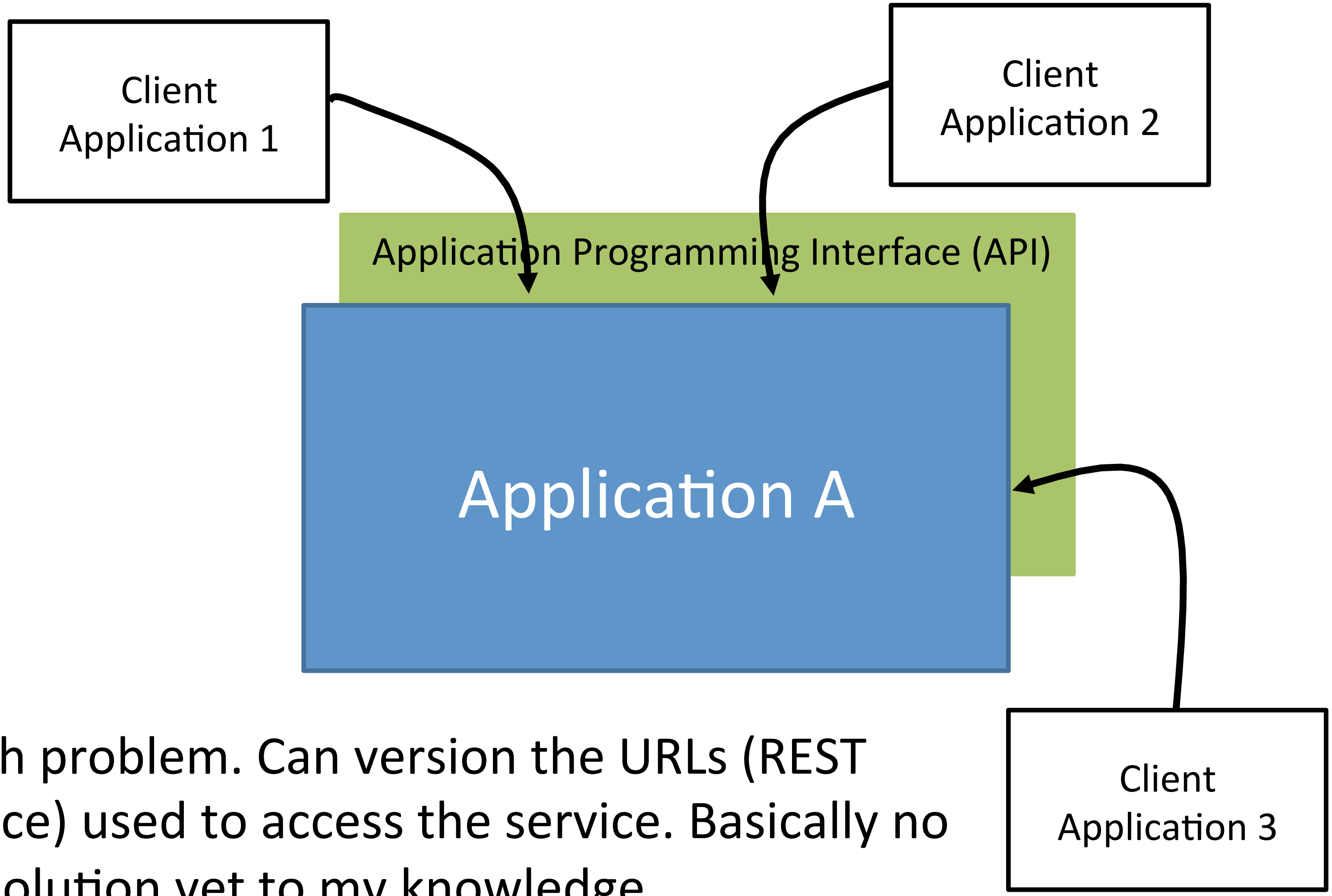
Coping with API evolution

What if you need to upgrade part of an API and it breaks compatibility?

a) add new API elements alongside the old
 e.g., search() and search2()

b) deprecate the old “version” of the API

What if the API is to a web service?



A tough problem. Can version the URLs (REST interface) used to access the service. Basically no great solution yet to my knowledge.

Issue Trackers

Are critical to teams for tracking and managing bugs and feature requests

Some examples

- Bugzilla
- Jazz/RTC (work items)
- Jira
- …

Issue Tracking Systems

- Manage development requests (also called bug reports, work items, change requests, change tasks)
  - Defects, e.g. “Using Ctrl-+ to enlarge display fails to add scroll bars when content exceeds window size”
  - Enhancements (new features), e.g. “Add an option to render page immediately without waiting CSS to load”
  - boundary between defects and enhancements can be fuzzy
- Maintain a list of bugs in the software
  - In a database called the bug repository
- Assign responsibility for each bug, feature, or task
- Organize the work to be done
  - Life cycle of a bug report
- Break up the work into “releases” or other deadlines

Issues / Bug Reports / Work Items

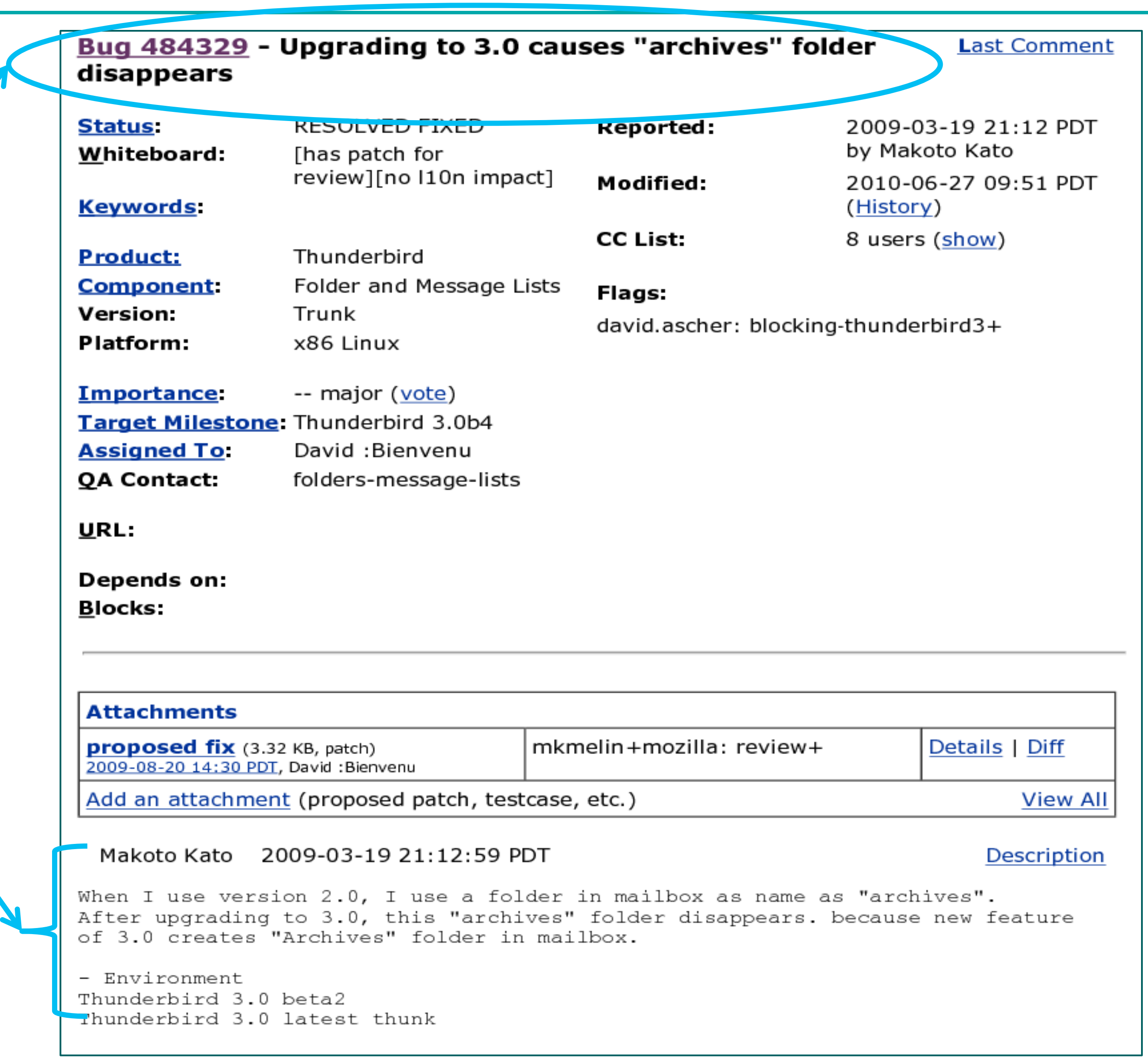
Are often the major piece of information in software development teams

- Contain the rationale for changes (provide documentation)
- Contain the links to related documents
- Contain information on the people who are involved
- ….

Anatomy of a bug report

A bug report has the following fields:

- Title (also called Summary)
- Description
- Status
- Assignee
- Priority
- Target Milestone
- …
- Comments



Summary (Title)

Description

Bug 484329 - Upgrading to 3.0 causes "archives" folder disappears

Last Comment

Status:	RESOLVED FIXED
Whiteboard:	[has patch for review][no l10n impact]
Keywords:	
Product:	Thunderbird
Component:	Folder and Message Lists
Version:	Trunk
Platform:	x86 Linux
Importance:	-- major (vote)
Target Milestone:	Thunderbird 3.0b4
Assigned To:	David :Bienvenu
QA Contact:	folders-message-lists
URL:	
Depends on:	
Blocks:	

Reported:	2009-03-19 21:12 PDT by Makoto Kato
Modified:	2010-06-27 09:51 PDT (History)
CC List:	8 users (show)
Flags:	

david.ascher: blocking-thunderbird3+

Attachments		
proposed fix (3.32 KB, patch) 2009-08-20 14:30 PDT, David :Bienvenu	mkmelin+mozilla: review+	Details | Diff
Add an attachment (proposed patch, testcase, etc.)		View All

Makoto Kato 2009-03-19 21:12:59 PDT Description

```
When I use version 2.0, I use a folder in mailbox as name as "archives".
After upgrading to 3.0, this "archives" folder disappears. because new feature
of 3.0 creates "Archives" folder in mailbox.

- Environment
Thunderbird 3.0 beta2
Thunderbird 3.0 latest thunk
```

https://bugzilla.mozilla.org/page.cgi?id=fields.html

Filing and triaging a bug report

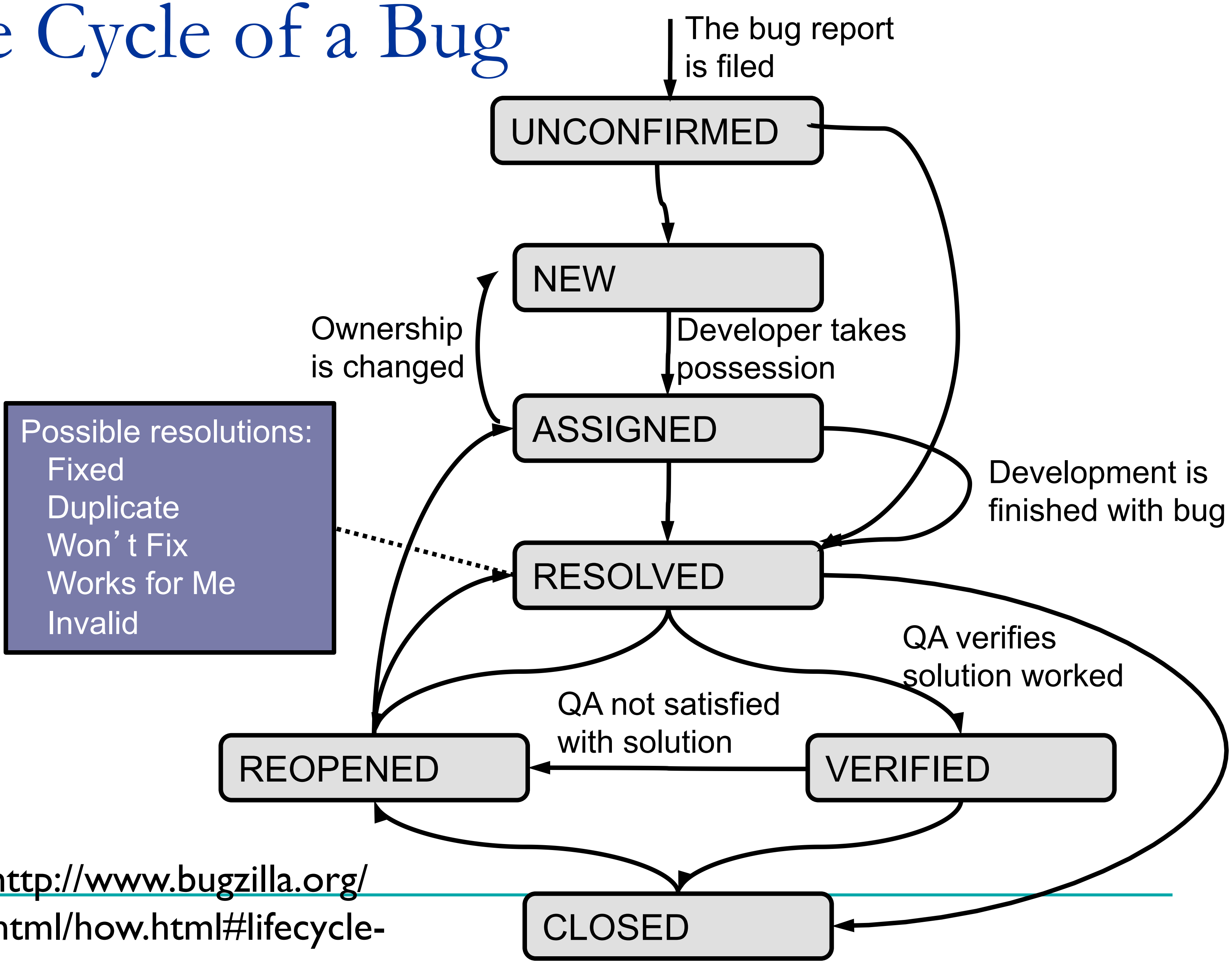
Who can file a bug report?

- Open bug repository: Any user (usual case in OSS)
- Closed bug repository: Developers, Testers, QA (usual case in proprietary software)

Triaging: go through a newly filed bug report to **validate** and **assign** the bug to a developer.

- Over the first six months of 2010, 19839 reports were filed for Eclipse: 110 reports/day; ~5 min per bug report: ~9 person-hours/day spent on triaging
- Validate: confirm it is a non-duplicate bug report; of 20K bug reports for Eclipse, 3.4K (17%) were marked invalid, duplicate, could not be replicated or won't fix

Life Cycle of a Bug


The bug report is filed
UNCONFIRMED
NEW
Ownership is changed
Developer takes possession
ASSIGNED
Development is finished with bug
RESOLVED
QA verifies solution worked
QA not satisfied with solution
REOPENED
VERIFIED
CLOSED

Possible resolutions:
- Fixed
- Duplicate
- Won’t Fix
- Works for Me
- Invalid

Based on: http://www.bugzilla.org/docs/2.16/html/how.html#lifecycle-image

Role of bug report comments

- **Discussion** among reporters and developers
- Might be the only way of communication for a distributed development team (different locations, time zones)
- Even when the developers have a mean of communicating face-to-face, it provides a way of **documenting** the **history** of the development process and the **rationale**.
    - e.g. why it was decided to fix a bug in a certain way.

Summary of Bug Report Writing Guidelines (Mozilla)

- Be precise
- Be clear - explain it so others can reproduce the bug
- One bug per report
- No bug is too trivial to report - small bugs may hide big bugs
- Clearly separate fact from speculation

Based on:

https://developer.mozilla.org/en/Bug_writing_guidelines

Guidelines cont’d

A good *summary (title)* should

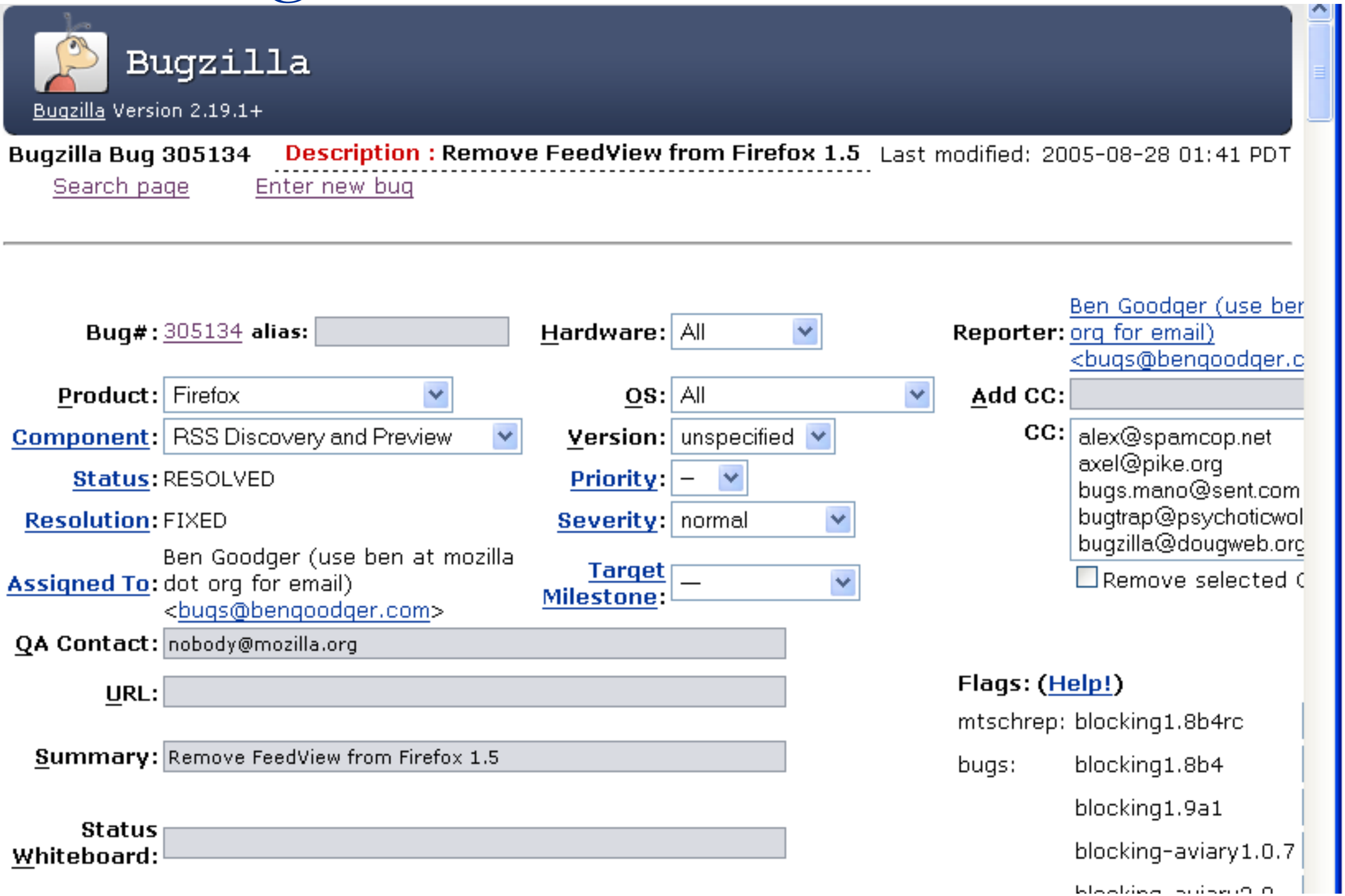
- quickly and uniquely identify a bug report
- explain the problem, not your suggested solution
- Good: "Cancelling a File Copy dialog crashes File Manager”
- Bad: "Software crashes”
- Bad: "Browser should work with my web site“

A good *description* should include

- Provide enough context
- Overview, steps to reproduce, actual results, expected results
- Survey study found most helpful information to be:Steps to reproduce, stack traces, test cases

T.Zimmerman et al., “*What makes a good bug report*”, IEEE Transactions on Software Engineering, vol. 36, No. 5, Sep./Oct. 2010

One bug

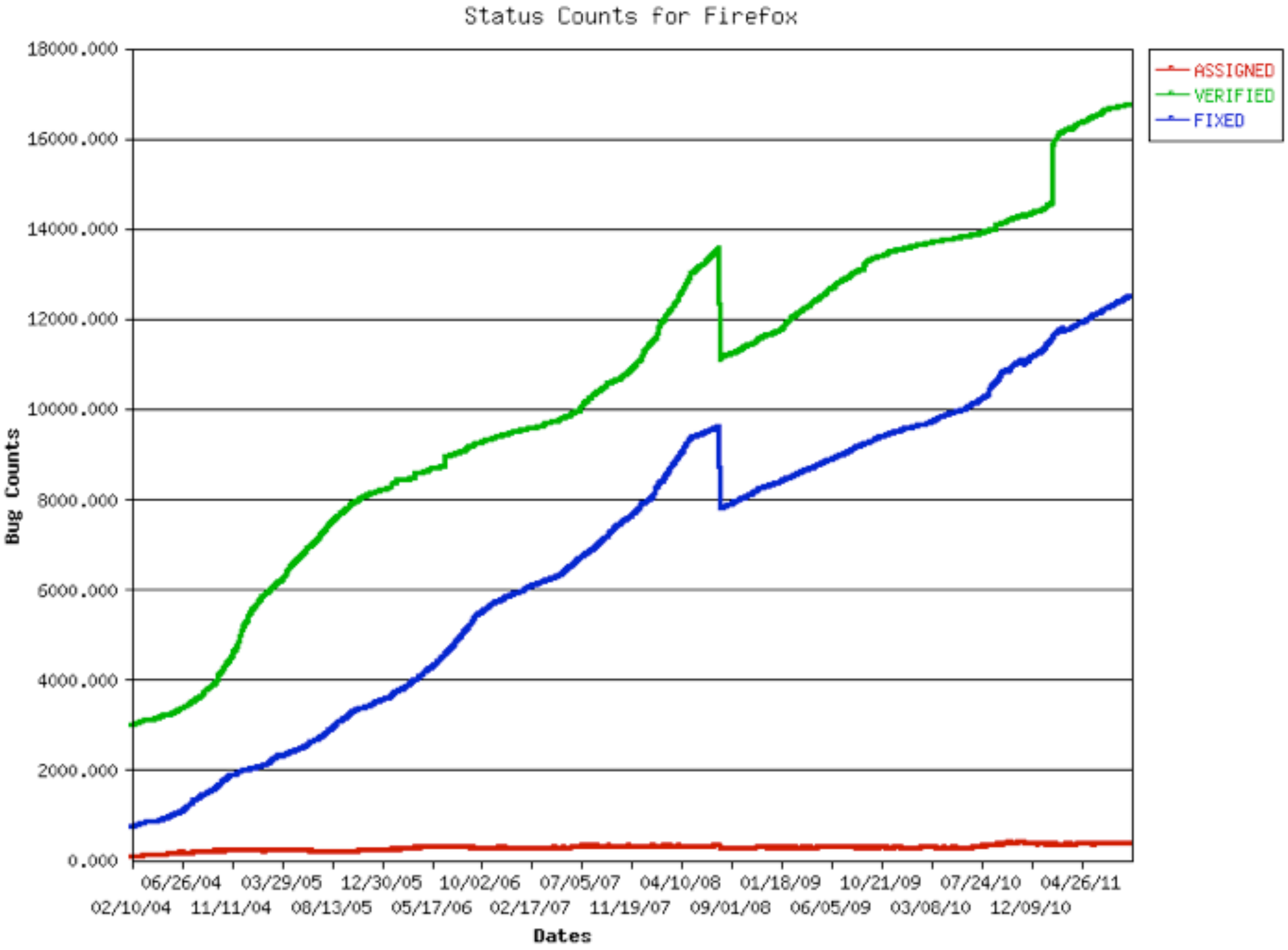


Bugzilla Bug 305134 **Description :** Remove FeedView from Firefox 1.5 Last modified: 2005-08-28 01:41 PDT

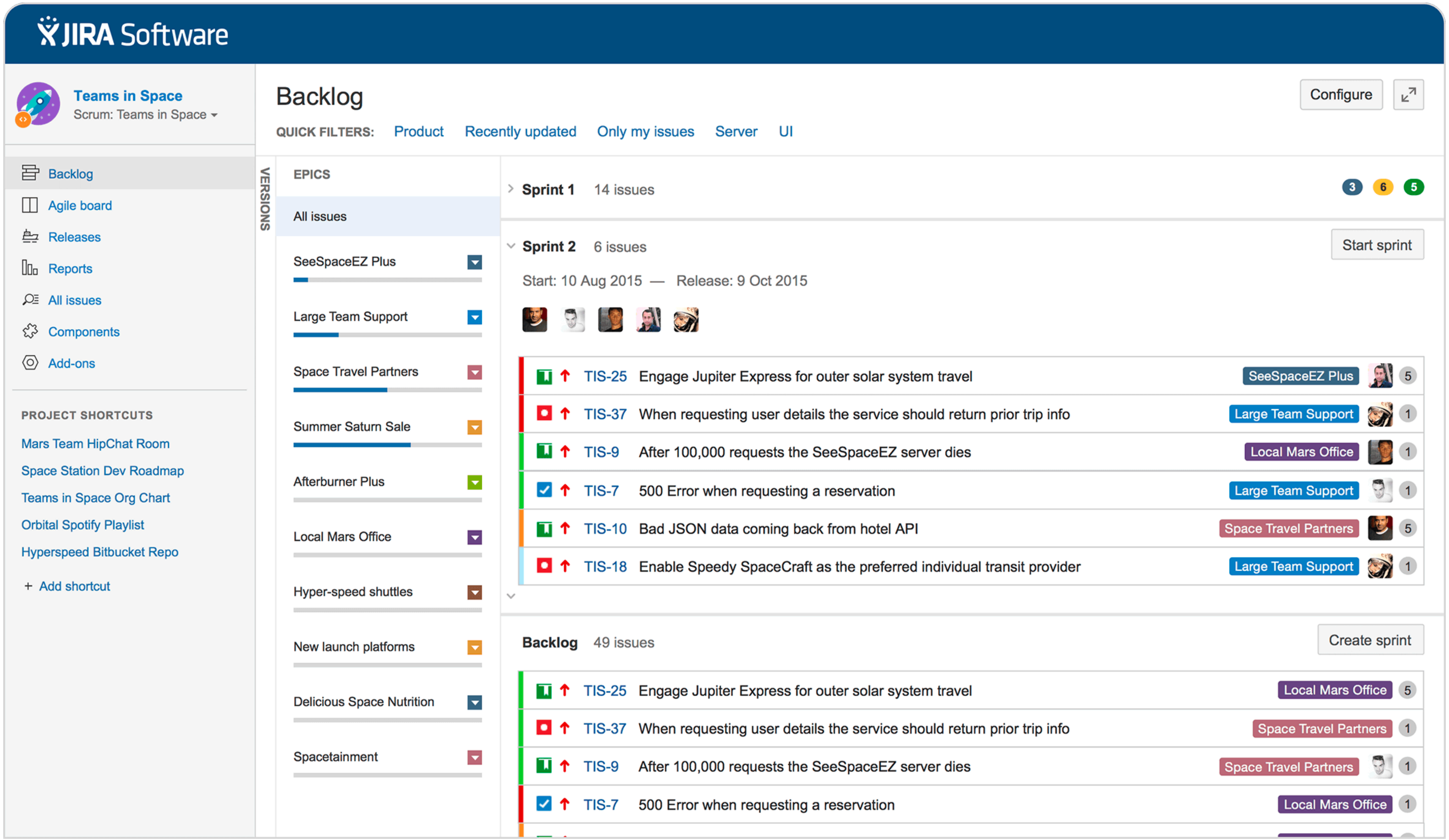
Search page Enter new bug

Bug#:	305134 **alias:**	**Hardware:**	All	**Reporter:**	Ben Goodger (use ben org for email) <bugs@bengoodger.c
Product:	Firefox	**OS:**	All	**Add CC:**	
Component:	RSS Discovery and Preview	**Version:**	unspecified	**CC:**	alex@spamcop.net axel@pike.org bugs.mano@sent.com bugtrap@psychoticwol bugzilla@dougweb.orc
Status:	RESOLVED	**Priority:**	--		
Resolution:	FIXED	**Severity:**	normal		
Assigned To:	Ben Goodger (use ben at mozilla dot org for email) <bugs@bengoodger.com>	**Target Milestone:**	---		□ Remove selected (
QA Contact:	nobody@mozilla.org				
URL:				**Flags: (Help!)**	
				mtschrep:	blocking1.8b4rc
Summary:	Remove FeedView from Firefox 1.5			bugs:	blocking1.8b4
					blocking1.9a1
Status Whiteboard:					blocking-aviary1.0.7

Many bugs


Status Counts for Firefox
18000.000
16000.000
14000.000
12000.000
10000.000
8000.000
6000.000
4000.000
2000.000
0.000
Bug Counts
06/26/04
03/29/05
12/30/05
10/02/06
07/05/07
04/10/08
01/18/09
10/21/09
07/24/10
04/26/11
02/10/04
11/11/04
08/13/05
05/17/06
02/17/07
11/19/07
09/01/08
06/05/09
03/08/10
12/09/10
Dates
ASSIGNED
VERIFIED
FIXED

Many issues/bugs



Start: 10 Aug 2015 — Release: 9 Oct 2015

TIS-25	Engage Jupiter Express for outer solar system travel	SeeSpaceEZ Plus	5
TIS-37	When requesting user details the service should return prior trip info	Large Team Support	1
TIS-9	After 100,000 requests the SeeSpaceEZ server dies	Local Mars Office	1
TIS-7	500 Error when requesting a reservation	Large Team Support	1
TIS-10	Bad JSON data coming back from hotel API	Space Travel Partners	5
TIS-18	Enable Speedy SpaceCraft as the preferred individual transit provider	Large Team Support	1

Backlog 49 issues

TIS-25	Engage Jupiter Express for outer solar system travel	Local Mars Office	5
TIS-37	When requesting user details the service should return prior trip info	Space Travel Partners	1
TIS-9	After 100,000 requests the SeeSpaceEZ server dies	Space Travel Partners	1
TIS-7	500 Error when requesting a reservation	Local Mars Office	1

Question:
How do you perform a change task?

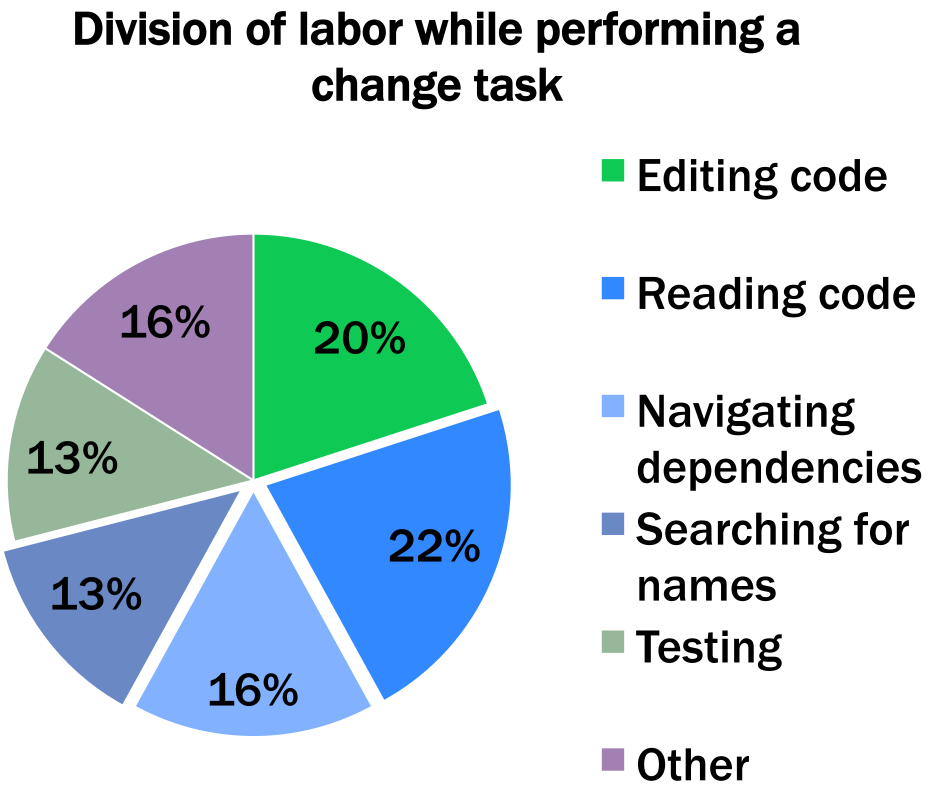
Summary:
Application crash on clicking the SAVE button while creating a new user.

Description:
Application crash on clicking the SAVE button while creating a new user, hence unable to create a new user in the application

Steps to reproduce:
1) Logon into the application
2) Navigate to the Users Menu > New User
3) Filled all the user information fields
4) Clicked on ‘Save’ Button
5) Seen an error page “ORA1090 Exception: Insert values Error…”
6) See attached logs for more information and also attached screenshot

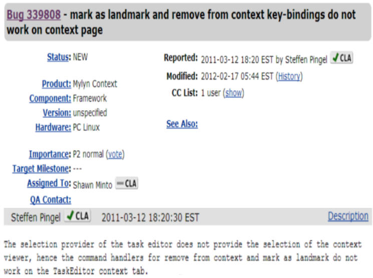
http://www.softwaretestinghelp.com/sample-bug-report/

Time spend during change tasks



Andrew J. Ko, Htet Aung, and Brad A. Myers. Eliciting design requirements for maintenance-oriented ides: a detailed study of corrective and perfective maintenance tasks. ICSE ’05.

Working on a change task…



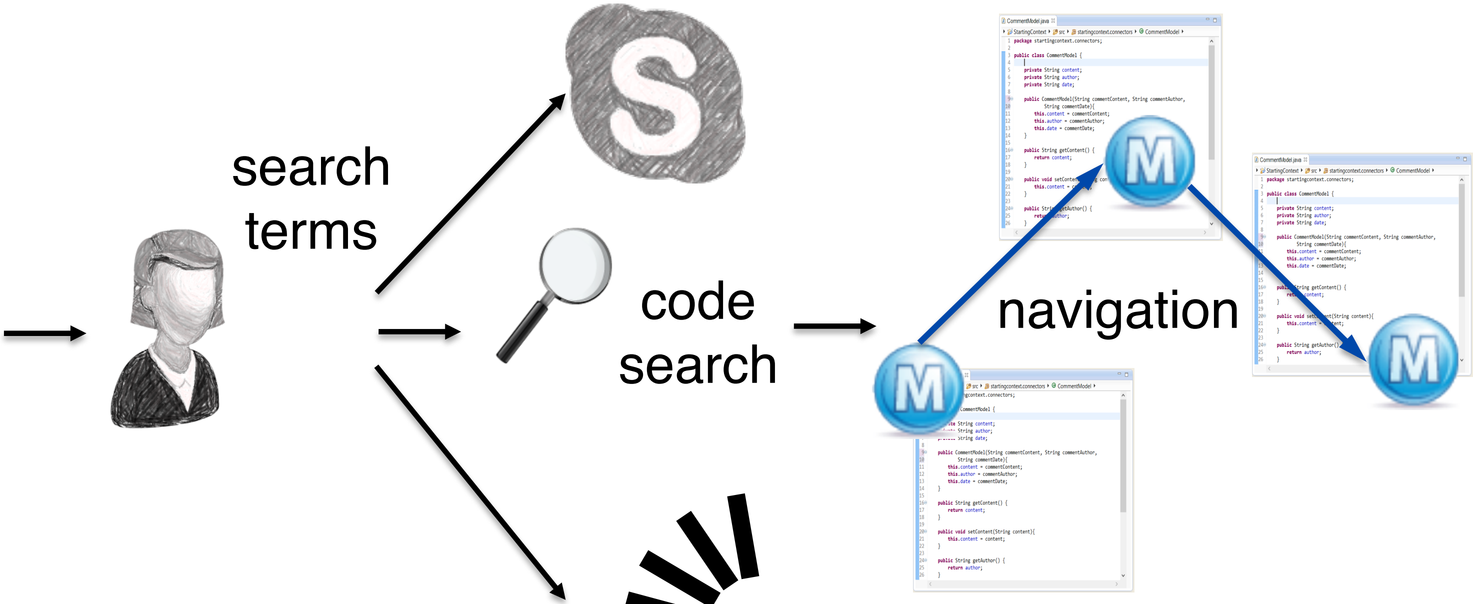
Bug 339808 - mark as landmark and remove from context key-bindings do not work on context page

Status:	NEW	Reported:	2011-03-12 18:20 EST by Steffen Pingel ✓CLA
		Modified:	2012-02-17 05:44 EST (History)
Product:	Mylyn Context	CC List:	1 user (show)
Component:	Framework		
Version:	unspecified		
Hardware:	PC Linux	See Also:	
Importance:	P2 normal (vote)		
Target Milestone:	---		
Assigned To:	Shawn Minto —CLA		
QA Contact:			

Steffen Pingel ✓CLA 2011-03-12 18:20:30 EST Description

```
The selection provider of the task editor does not provide the selection of the context
viewer, hence the command handlers for remove from context and mark as landmark do not
work on the TaskEditor context tab.
```

change task
(Jira/Bugzilla/...)



search terms

code search

navigation

CommentModel.java
StartingContext ▸ src ▸ startingcontext.connectors ▸ CommentModel ▸

```
package startingcontext.connectors;

public class CommentModel {
    |
    private String content;
    private String author;
    private String date;

    public CommentModel(String commentContent, String commentAuthor,
            String commentDate){
        this.content = commentContent;
        this.author = commentAuthor;
        this.date = commentDate;
    }

    public String getContent() {
        return content;
    }

    public void setContent(String content){
        this.content = content;
    }

    public String getAuthor() {
        return author;
    }
```

Integrated Development Environment (IDE)

…eventually editing

Working on a change task

Problems/Difficulties:

- Too much code to understand/read all
- Language mismatch: bug reports (NL) ≠ source code
- Crosscutting concerns: code is often tangled and scattered
- …

To localize and edit relevant code…

- Identify good search terms
- Take advantage of tool support, e.g. code search, structural navigation support, breakpoints (debugging)
- Take advantage of information provided in bug reports, such as stack traces, and also on news forums, such as stackoverflow

Summary

- Evolution is tough but an inherent and important part of software development
- Evolutionary changes can be corrective, adaptive and perfective
- API contract compatibility ensures that client code still runs as expected after API evolves
- Issue Tracking Systems are an integral part of most software development and writing good bug reports is important
- Much time on a change task is spend navigating and searching