EECS 4313

Software Engineering Testing

Topic 15:

Software Defect Prediction

Zhen Ming (Jack) Jiang

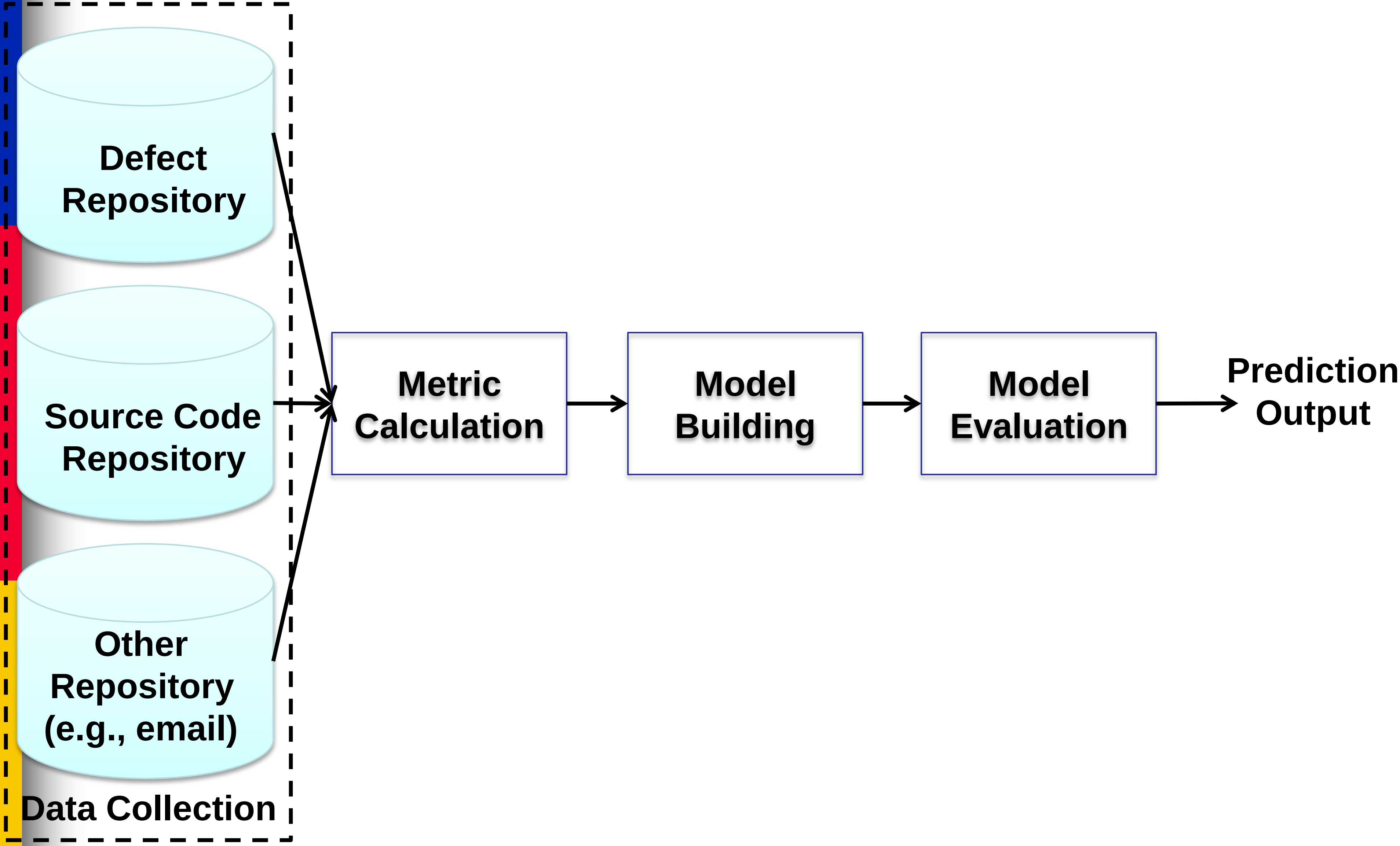
What is Software Defect Prediction?

Software Defect Prediction (SDP) is the line of research that concerned with *building prediction models, which leverage software metrics to predict defect-prone areas of a software.*

Motivation

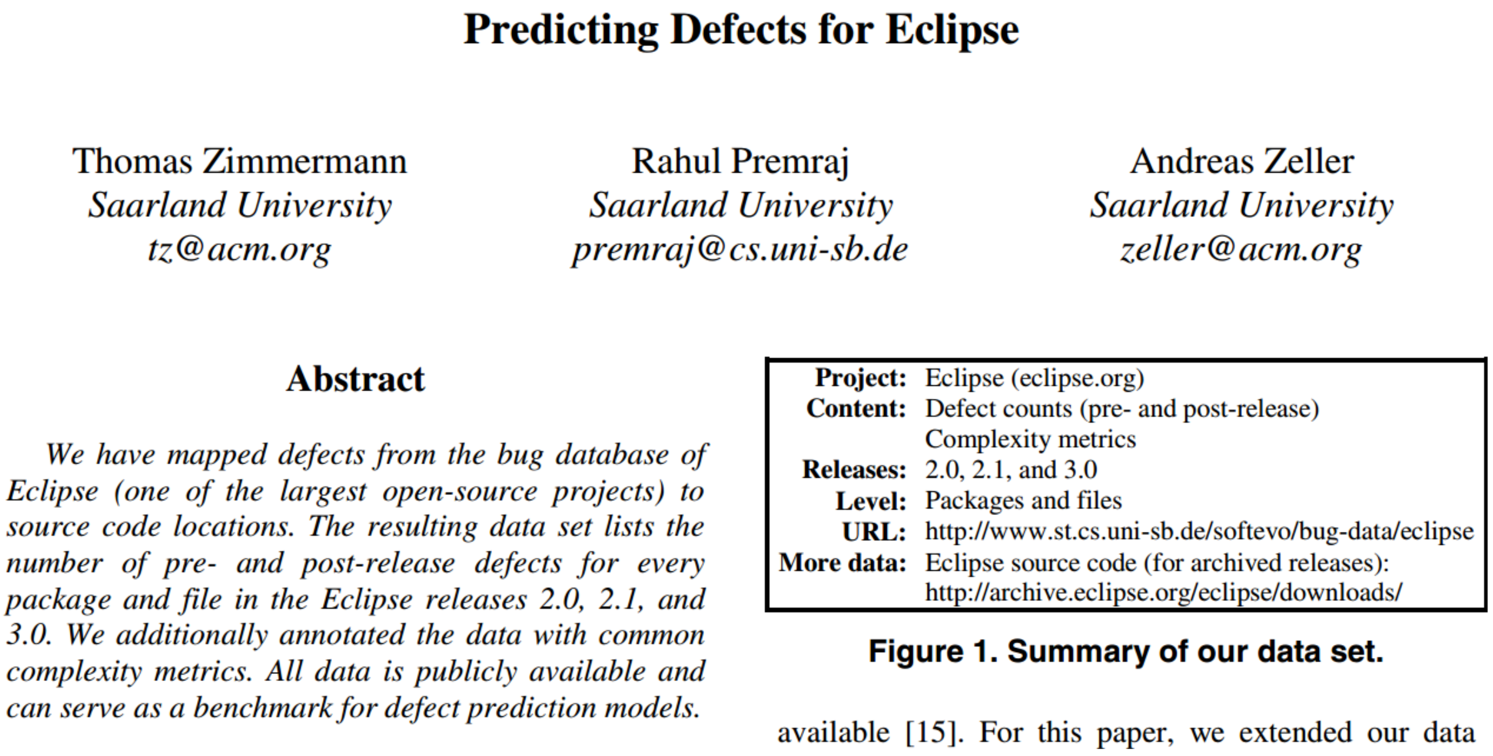
- Identify software locations (e.g., subsystems, files or functions) that quality assurance efforts should focus on. Examples are:
  - Which code changes should I review first?
  - Which module should I test first?
- Learn from the past mistakes to improve the software development process. Examples are:
  - Why subsystem A is more bug-prone than another subsystem B?
  - What can we learn from the failures of project C to improve the quality of project D?

General Process



Bug Prediction Example

Predicting Post-Release Bugs for Eclipse



Predicting Defects for Eclipse

Thomas Zimmermann
Saarland University
tz@acm.org

Rahul Premraj
Saarland University
premraj@cs.uni-sb.de

Andreas Zeller
Saarland University
zeller@acm.org

Abstract

We have mapped defects from the bug database of Eclipse (one of the largest open-source projects) to source code locations. The resulting data set lists the number of pre- and post-release defects for every package and file in the Eclipse releases 2.0, 2.1, and 3.0. We additionally annotated the data with common complexity metrics. All data is publicly available and can serve as a benchmark for defect prediction models.

Project:	Eclipse (eclipse.org)
Content:	Defect counts (pre- and post-release) Complexity metrics
Releases:	2.0, 2.1, and 3.0
Level:	Packages and files
URL:	http://www.st.cs.uni-sb.de/softevo/bug-data/eclipse
More data:	Eclipse source code (for archived releases): http://archive.eclipse.org/eclipse/downloads/

Figure 1. Summary of our data set.

available [15]. For this paper, we extended our data

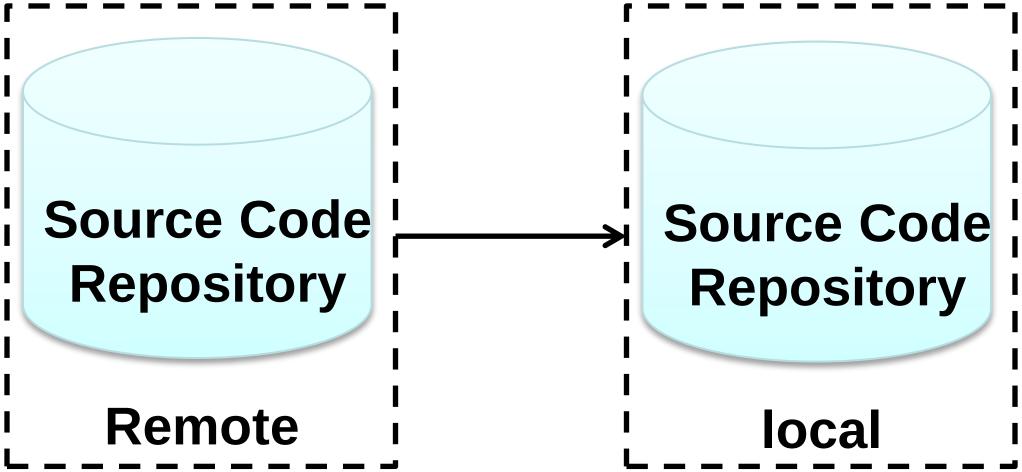
- **Data gathering and processing**
- **Statistical analysis techniques: GLM, correlations**

Collecting the Eclipse Dataset

- **Goal**: Tracking which components failed
  - We need to know the location of every defect that has been fixed. Hence, we need to analyze the source code repository data (*CVS*)
    - Mirror the source code repository
    - Identifying bug fixing changes
  - We need to know whether the bug is a pre-release or post-release bug. Hence, we need to analyze the bug tracking system (*Bugzilla*)
    - Collecting the bug reports
    - Map the bug identifiers to release numbers (Is this bug a pre- or post-release bug?)

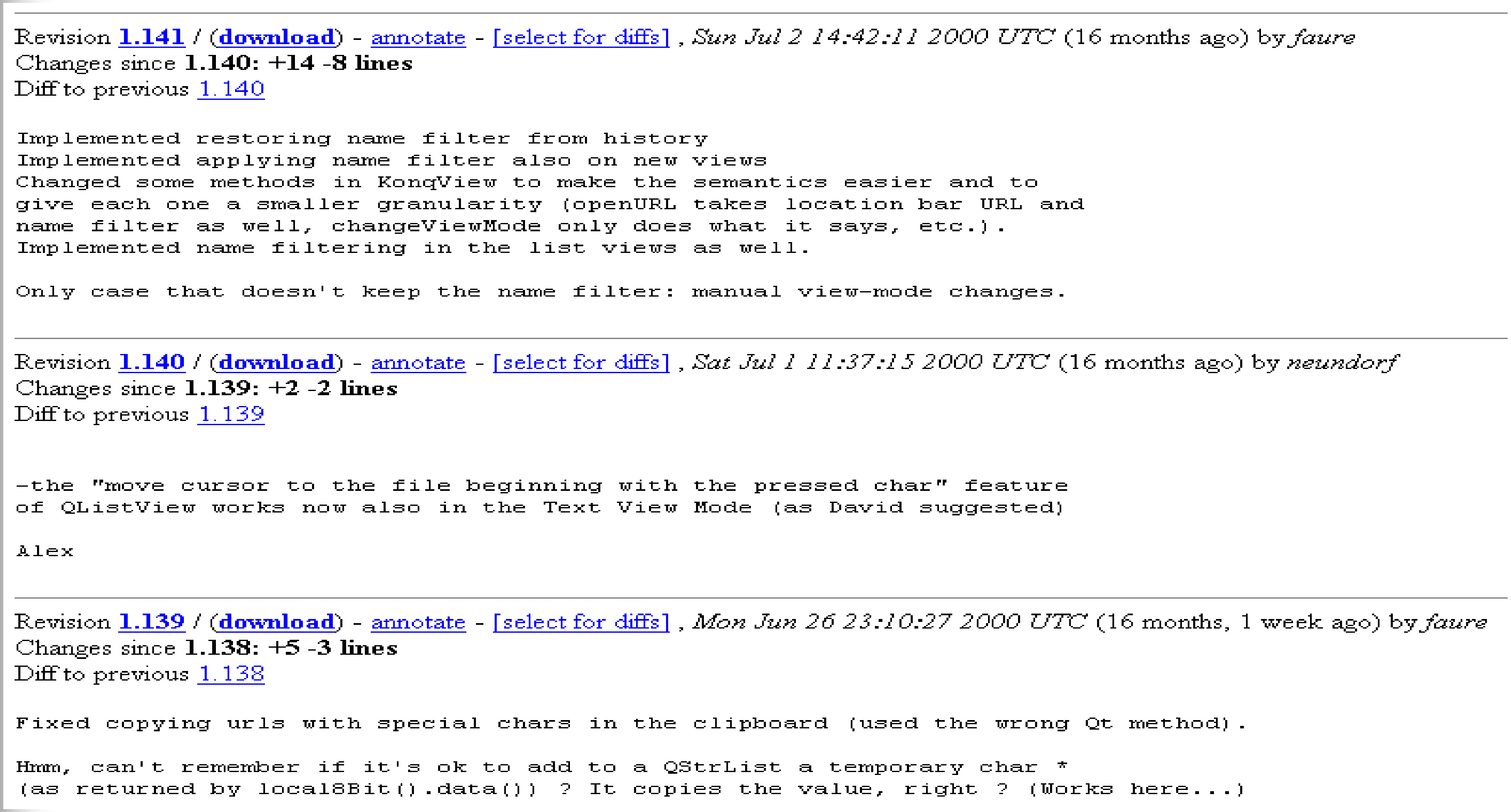
Mirror the Eclipse Source Code Repository

- There are tools (e.g., CVSup, CSVSuck, etc.) which can mirror the Eclipse CVS source code repository
  - *Note: Eclipse switched to Git as their version control system now*



Be gentle. Otherwise, you might be mistaken as a DoS attack!

An Example of the Commit Logs



Revision 1.141 / (download) - annotate - [select for diffs] , *Sun Jul 2 14:42:11 2000 UTC* (16 months ago) by *faure*
Changes since **1.140: +14 -8 lines**
Diff to previous 1.140

```
Implemented restoring name filter from history
Implemented applying name filter also on new views
Changed some methods in KonqView to make the semantics easier and to
give each one a smaller granularity (openURL takes location bar URL and
name filter as well, changeViewMode only does what it says, etc.).
Implemented name filtering in the list views as well.

Only case that doesn't keep the name filter: manual view-mode changes.
```

Revision 1.140 / (download) - annotate - [select for diffs] , *Sat Jul 1 11:37:15 2000 UTC* (16 months ago) by *neundorf*
Changes since **1.139: +2 -2 lines**
Diff to previous 1.139

```
-the "move cursor to the file beginning with the pressed char" feature
of QListView works now also in the Text View Mode (as David suggested)

Alex
```

Revision 1.139 / (download) - annotate - [select for diffs] , *Mon Jun 26 23:10:27 2000 UTC* (16 months, 1 week ago) by *faure*
Changes since **1.138: +5 -3 lines**
Diff to previous 1.138

```
Fixed copying urls with special chars in the clipboard (used the wrong Qt method).

Hmm, can't remember if it's ok to add to a QStrList a temporary char *
(as returned by local8Bit().data()) ? It copies the value, right ? (Works here...)
```

Identifying the Bug Fixing Changes

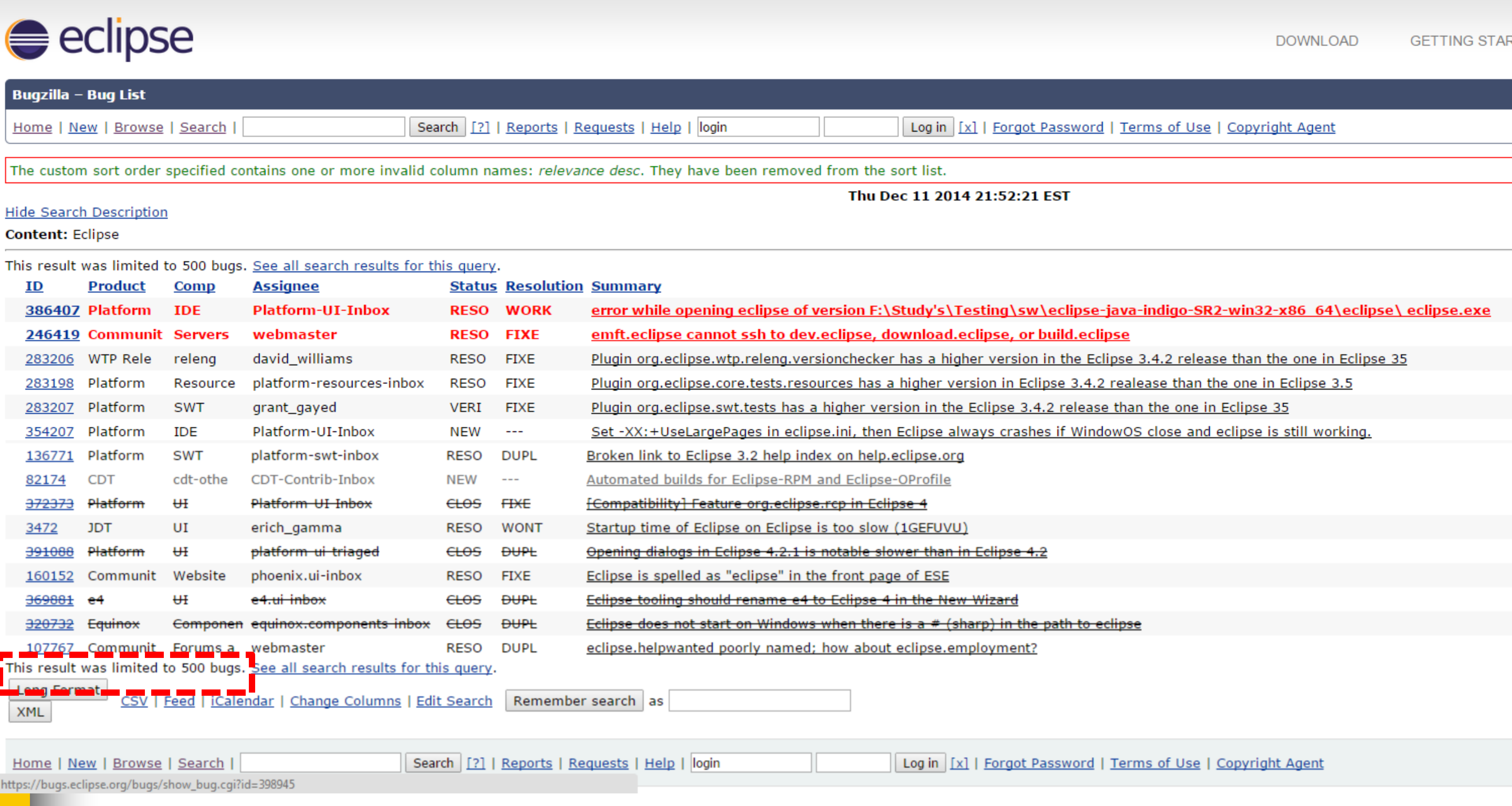
- Obtain the commit logs
- Search for references to bug reports (e.g., fixed 42233" or "bug 23444"
  - These messages have a low trust at first
- Increase the trust level when a message contains keywords like "fixed" or "bug" or matches with patterns like "# and a number"

Similarly, we can use keyword tagging to identify other types of changes:

- *Bug fixes*
- *New features*
- *License/copyright update, etc.*

Collecting the Eclipse Bug Reports

- Download the XML reports


eclipse
DOWNLOAD
GETTING STAR
Bugzilla – Bug List
Home | New | Browse | Search | Search [?] | Reports | Requests | Help | login Log in [x] | Forgot Password | Terms of Use | Copyright Agent

The custom sort order specified contains one or more invalid column names: *relevance desc*. They have been removed from the sort list.

Thu Dec 11 2014 21:52:21 EST

Hide Search Description

Content: Eclipse

This result was limited to 500 bugs. See all search results for this query.

ID	Product	Comp	Assignee	Status	Resolution	Summary
386407	**Platform**	**IDE**	**Platform-UI-Inbox**	**RESO**	**WORK**	**error while opening eclipse of version F:\Study's\Testing\sw\eclipse-java-indigo-SR2-win32-x86_64\eclipse\ eclipse.exe**
246419	**Communit**	**Servers**	**webmaster**	**RESO**	**FIXE**	**emft.eclipse cannot ssh to dev.eclipse, download.eclipse, or build.eclipse**
283206	WTP Rele	releng	david_williams	RESO	FIXE	Plugin org.eclipse.wtp.releng.versionchecker has a higher version in the Eclipse 3.4.2 release than the one in Eclipse 35
283198	Platform	Resource	platform-resources-inbox	RESO	FIXE	Plugin org.eclipse.core.tests.resources has a higher version in Eclipse 3.4.2 realease than the one in Eclipse 3.5
283207	Platform	SWT	grant_gayed	VERI	FIXE	Plugin org.eclipse.swt.tests has a higher version in the Eclipse 3.4.2 release than the one in Eclipse 35
354207	Platform	IDE	Platform-UI-Inbox	NEW	---	Set -XX:+UseLargePages in eclipse.ini, then Eclipse always crashes if WindowOS close and eclipse is still working.
136771	Platform	SWT	platform-swt-inbox	RESO	DUPL	Broken link to Eclipse 3.2 help index on help.eclipse.org
82174	CDT	cdt-othe	CDT-Contrib-Inbox	NEW	---	Automated builds for Eclipse-RPM and Eclipse-OProfile
~~372373~~	~~Platform~~	~~UI~~	~~Platform-UI-Inbox~~	~~CLOS~~	~~FIXE~~	~~[Compatibility] Feature org.eclipse.rcp in Eclipse 4~~
3472	JDT	UI	erich_gamma	RESO	WONT	Startup time of Eclipse on Eclipse is too slow (1GEFUVU)
~~391088~~	~~Platform~~	~~UI~~	~~platform-ui-triaged~~	~~CLOS~~	~~DUPL~~	~~Opening dialogs in Eclipse 4.2.1 is notable slower than in Eclipse 4.2~~
160152	Communit	Website	phoenix.ui-inbox	RESO	FIXE	Eclipse is spelled as "eclipse" in the front page of ESE
~~369881~~	~~e4~~	~~UI~~	~~e4.ui-inbox~~	~~CLOS~~	~~DUPL~~	~~Eclipse tooling should rename e4 to Eclipse 4 in the New Wizard~~
~~320732~~	~~Equinox~~	~~Componen~~	~~equinox.components-inbox~~	~~CLOS~~	~~DUPL~~	~~Eclipse does not start on Windows when there is a # (sharp) in the path to eclipse~~
107767	Communit	Forums a	webmaster	RESO	DUPL	eclipse.helpwanted poorly named; how about eclipse.employment?

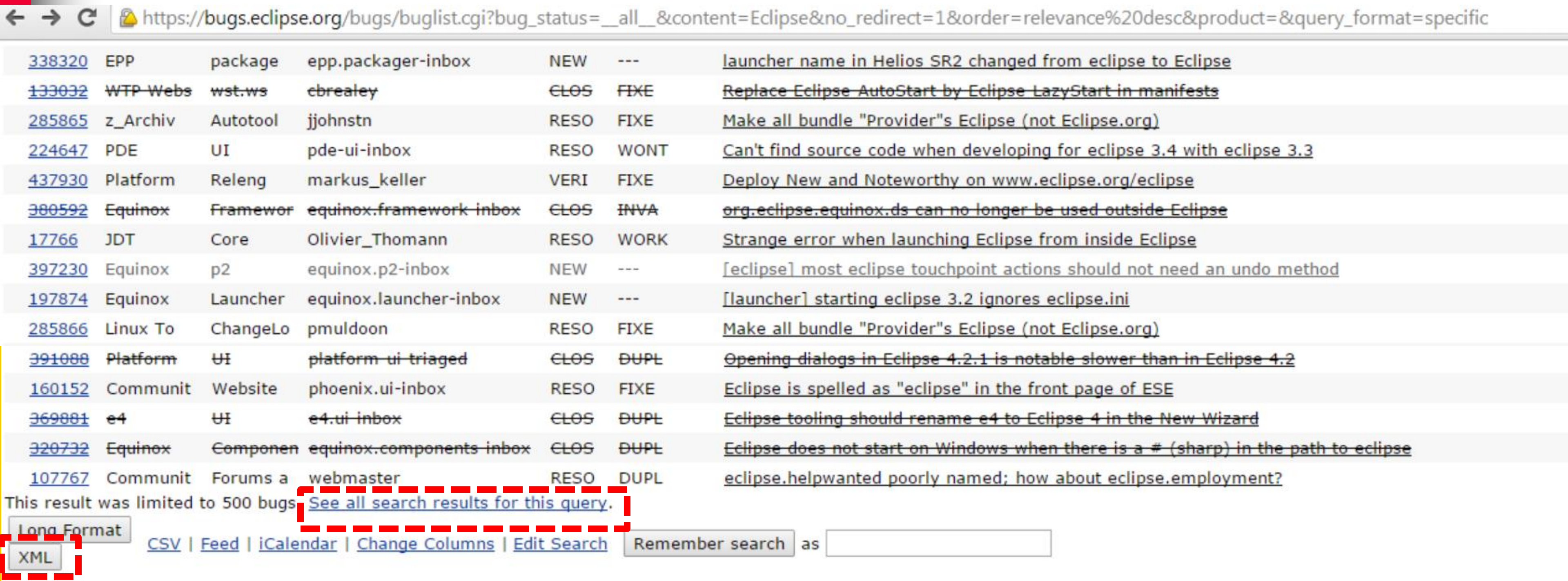
This result was limited to 500 bugs. See all search results for this query.

Long Format
XML
CSV | Feed | iCalendar | Change Columns | Edit Search
Remember search as
Home | New | Browse | Search | Search [?] | Reports | Requests | Help | login Log in [x] | Forgot Password | Terms of Use | Copyright Agent

https://bugs.eclipse.org/bugs/show_bug.cgi?id=398945

Collecting the Eclipse Bug Reports - Approach 1

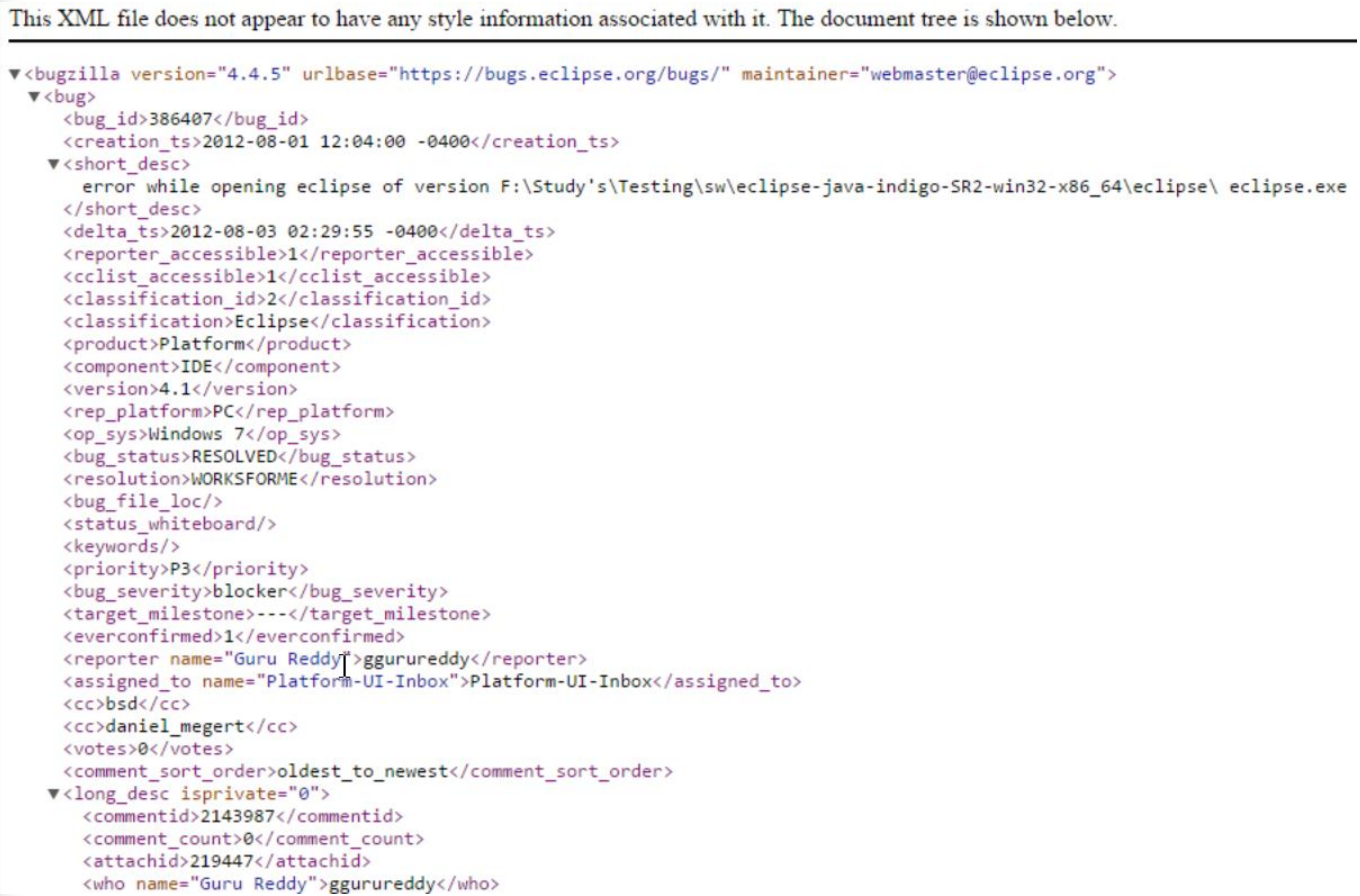
- Click “See all search results for this query” and click XML report
  - The XML data might be too big to be fitted into the browser’s memory. One work-around is to use the “save-as” feature



https://bugs.eclipse.org/bugs/buglist.cgi?bug_status=__all__&content=Eclipse&no_redirect=1&order=relevance%20desc&product=&query_format=specific

338320	EPP	package	epp.packager-inbox	NEW	---	launcher name in Helios SR2 changed from eclipse to Eclipse
~~133032~~	~~WTP Webs~~	~~wst.ws~~	~~cbrealey~~	~~CLOS~~	~~FIXE~~	~~Replace Eclipse-AutoStart by Eclipse-LazyStart in manifests~~
285865	z_Archiv	Autotool	jjohnstn	RESO	FIXE	Make all bundle "Provider"s Eclipse (not Eclipse.org)
224647	PDE	UI	pde-ui-inbox	RESO	WONT	Can't find source code when developing for eclipse 3.4 with eclipse 3.3
437930	Platform	Releng	markus_keller	VERI	FIXE	Deploy New and Noteworthy on www.eclipse.org/eclipse
~~380592~~	~~Equinox~~	~~Framewor~~	~~equinox.framework-inbox~~	~~CLOS~~	~~INVA~~	~~org.eclipse.equinox.ds can no longer be used outside Eclipse~~
17766	JDT	Core	Olivier_Thomann	RESO	WORK	Strange error when launching Eclipse from inside Eclipse
397230	Equinox	p2	equinox.p2-inbox	NEW	---	[eclipse] most eclipse touchpoint actions should not need an undo method
197874	Equinox	Launcher	equinox.launcher-inbox	NEW	---	[launcher] starting eclipse 3.2 ignores eclipse.ini
285866	Linux To	ChangeLo	pmuldoon	RESO	FIXE	Make all bundle "Provider"s Eclipse (not Eclipse.org)
~~391088~~	~~Platform~~	~~UI~~	~~platform-ui-triaged~~	~~CLOS~~	~~DUPL~~	~~Opening dialogs in Eclipse 4.2.1 is notable slower than in Eclipse 4.2~~
160152	Communit	Website	phoenix.ui-inbox	RESO	FIXE	Eclipse is spelled as "eclipse" in the front page of ESE
~~369881~~	~~e4~~	~~UI~~	~~e4.ui-inbox~~	~~CLOS~~	~~DUPL~~	~~Eclipse tooling should rename e4 to Eclipse 4 in the New Wizard~~
~~320732~~	~~Equinox~~	~~Componen~~	~~equinox.components-inbox~~	~~CLOS~~	~~DUPL~~	~~Eclipse does not start on Windows when there is a # (sharp) in the path to eclipse~~
107767	Communit	Forums a	webmaster	RESO	DUPL	eclipse.helpwanted poorly named; how about eclipse.employment?

This result was limited to 500 bugs. See all search results for this query.

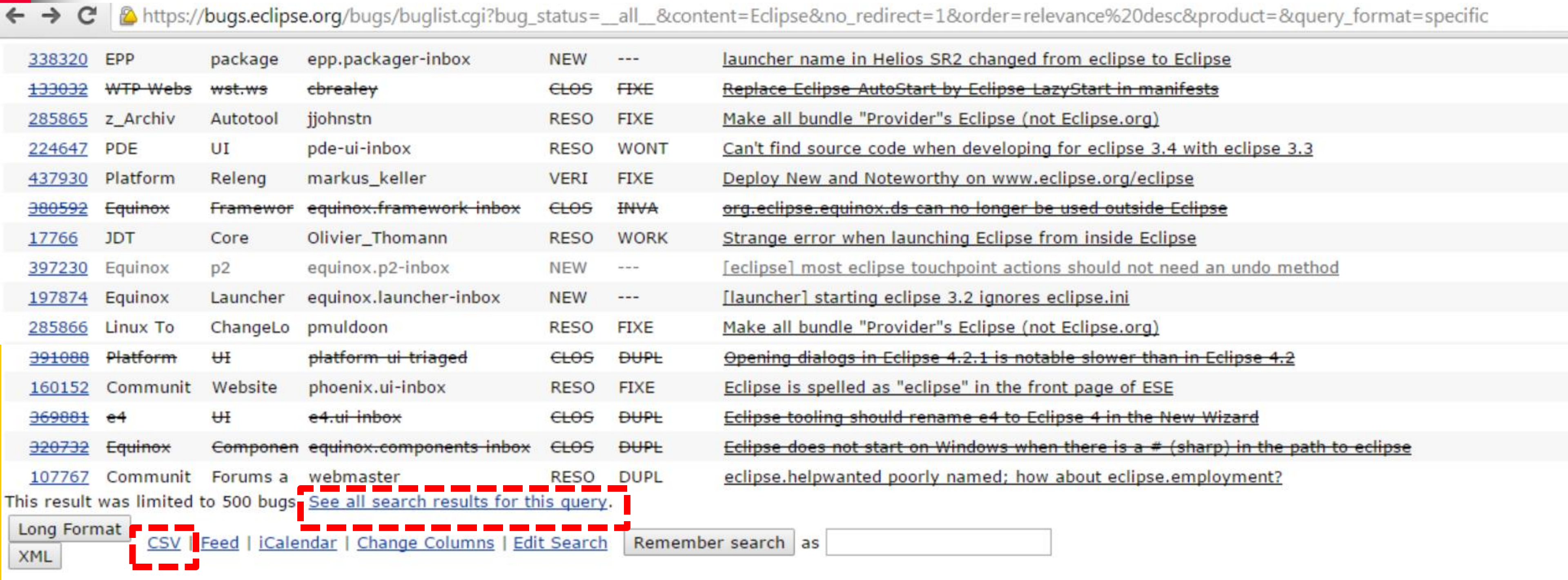


This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<bugzilla version="4.4.5" urlbase="https://bugs.eclipse.org/bugs/" maintainer="webmaster@eclipse.org">
  ▼<bug>
     <bug_id>386407</bug_id>
     <creation_ts>2012-08-01 12:04:00 -0400</creation_ts>
   ▼<short_desc>
       error while opening eclipse of version F:\Study's\Testing\sw\eclipse-java-indigo-SR2-win32-x86_64\eclipse\ eclipse.exe
     </short_desc>
     <delta_ts>2012-08-03 02:29:55 -0400</delta_ts>
     <reporter_accessible>1</reporter_accessible>
     <cclist_accessible>1</cclist_accessible>
     <classification_id>2</classification_id>
     <classification>Eclipse</classification>
     <product>Platform</product>
     <component>IDE</component>
     <version>4.1</version>
     <rep_platform>PC</rep_platform>
     <op_sys>Windows 7</op_sys>
     <bug_status>RESOLVED</bug_status>
     <resolution>WORKSFORME</resolution>
     <bug_file_loc/>
     <status_whiteboard/>
     <keywords/>
     <priority>P3</priority>
     <bug_severity>blocker</bug_severity>
     <target_milestone>---</target_milestone>
     <everconfirmed>1</everconfirmed>
     <reporter name="Guru Reddy">ggurureddy</reporter>
     <assigned_to name="Platform-UI-Inbox">Platform-UI-Inbox</assigned_to>
     <cc>bsd</cc>
     <cc>daniel_megert</cc>
     <votes>0</votes>
     <comment_sort_order>oldest_to_newest</comment_sort_order>
   ▼<long_desc isprivate="0">
       <commentid>2143987</commentid>
       <comment_count>0</comment_count>
       <attachid>219447</attachid>
       <who name="Guru Reddy">ggurureddy</who>
```

Collecting the Eclipse Bug Reports - Approach 2

- Click “See all search results for this query” and save the data using csv format
  - The CSV file will contain the Bug ID, Product name and other types of information
  - Parse the CSV file and download each bug report



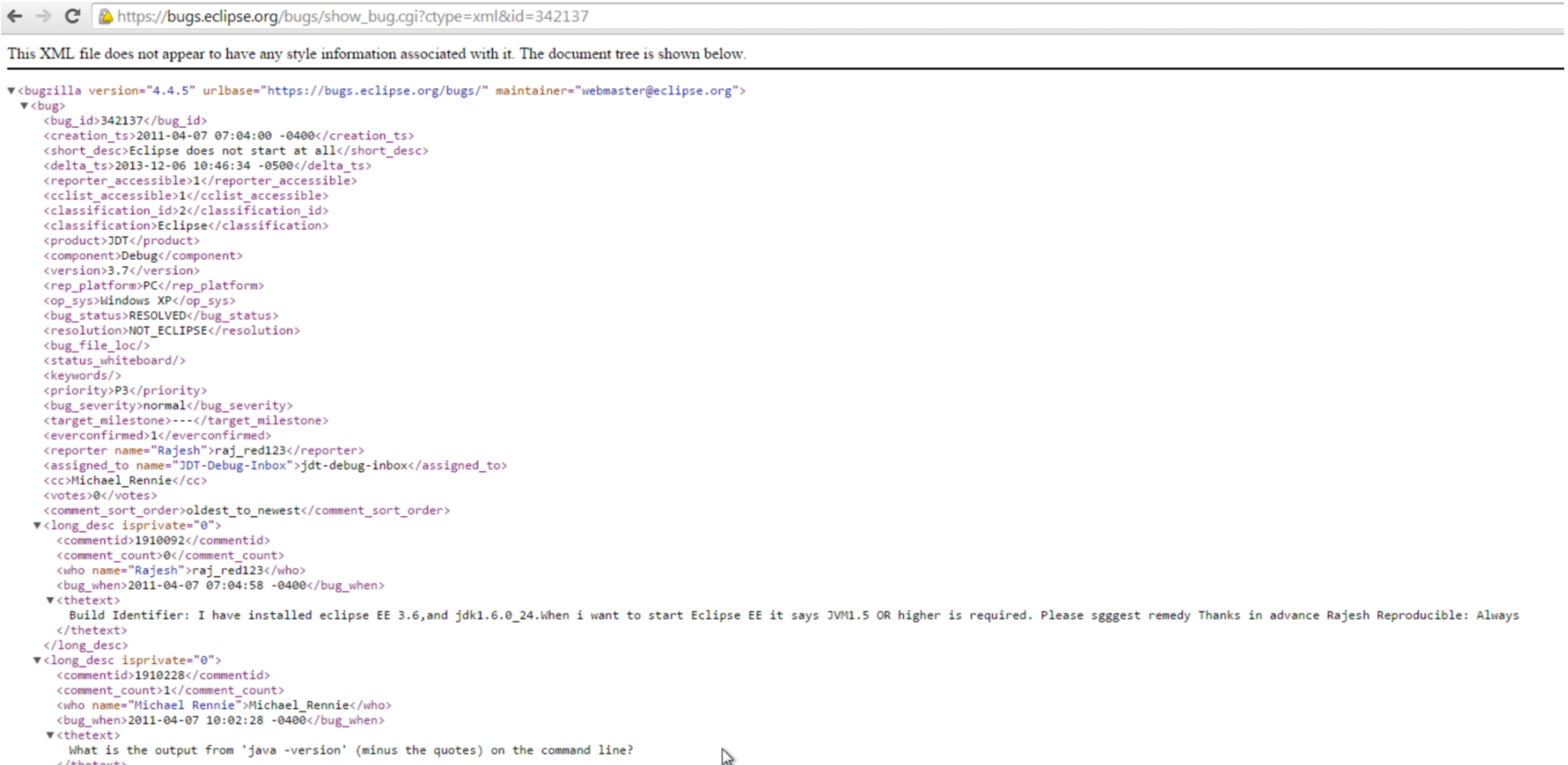
https://bugs.eclipse.org/bugs/buglist.cgi?bug_status=__all__&content=Eclipse&no_redirect=1&order=relevance%20desc&product=&query_format=specific

338320	EPP	package	epp.packager-inbox	NEW	---	launcher name in Helios SR2 changed from eclipse to Eclipse
~~133032~~	~~WTP Webs~~	~~wst.ws~~	~~cbrealey~~	~~CLOS~~	~~FIXE~~	~~Replace Eclipse-AutoStart by Eclipse-LazyStart in manifests~~
285865	z_Archiv	Autotool	jjohnstn	RESO	FIXE	Make all bundle "Provider"s Eclipse (not Eclipse.org)
224647	PDE	UI	pde-ui-inbox	RESO	WONT	Can't find source code when developing for eclipse 3.4 with eclipse 3.3
437930	Platform	Releng	markus_keller	VERI	FIXE	Deploy New and Noteworthy on www.eclipse.org/eclipse
~~380592~~	~~Equinox~~	~~Framewor~~	~~equinox.framework-inbox~~	~~CLOS~~	~~INVA~~	~~org.eclipse.equinox.ds can no longer be used outside Eclipse~~
17766	JDT	Core	Olivier_Thomann	RESO	WORK	Strange error when launching Eclipse from inside Eclipse
397230	Equinox	p2	equinox.p2-inbox	NEW	---	[eclipse] most eclipse touchpoint actions should not need an undo method
197874	Equinox	Launcher	equinox.launcher-inbox	NEW	---	[launcher] starting eclipse 3.2 ignores eclipse.ini
285866	Linux To	ChangeLo	pmuldoon	RESO	FIXE	Make all bundle "Provider"s Eclipse (not Eclipse.org)
~~391088~~	~~Platform~~	~~UI~~	~~platform-ui-triaged~~	~~CLOS~~	~~DUPL~~	~~Opening dialogs in Eclipse 4.2.1 is notable slower than in Eclipse 4.2~~
160152	Communit	Website	phoenix.ui-inbox	RESO	FIXE	Eclipse is spelled as "eclipse" in the front page of ESE
~~369881~~	~~e4~~	~~UI~~	~~e4.ui-inbox~~	~~CLOS~~	~~DUPL~~	~~Eclipse tooling should rename e4 to Eclipse 4 in the New Wizard~~
~~320732~~	~~Equinox~~	~~Componen~~	~~equinox.components-inbox~~	~~CLOS~~	~~DUPL~~	~~Eclipse does not start on Windows when there is a # (sharp) in the path to eclipse~~
107767	Communit	Forums a	webmaster	RESO	DUPL	eclipse.helpwanted poorly named; how about eclipse.employment?

This result was limited to 500 bugs. See all search results for this query.

Collection the Eclipse Bug Reports - Approach 2 (Continued)

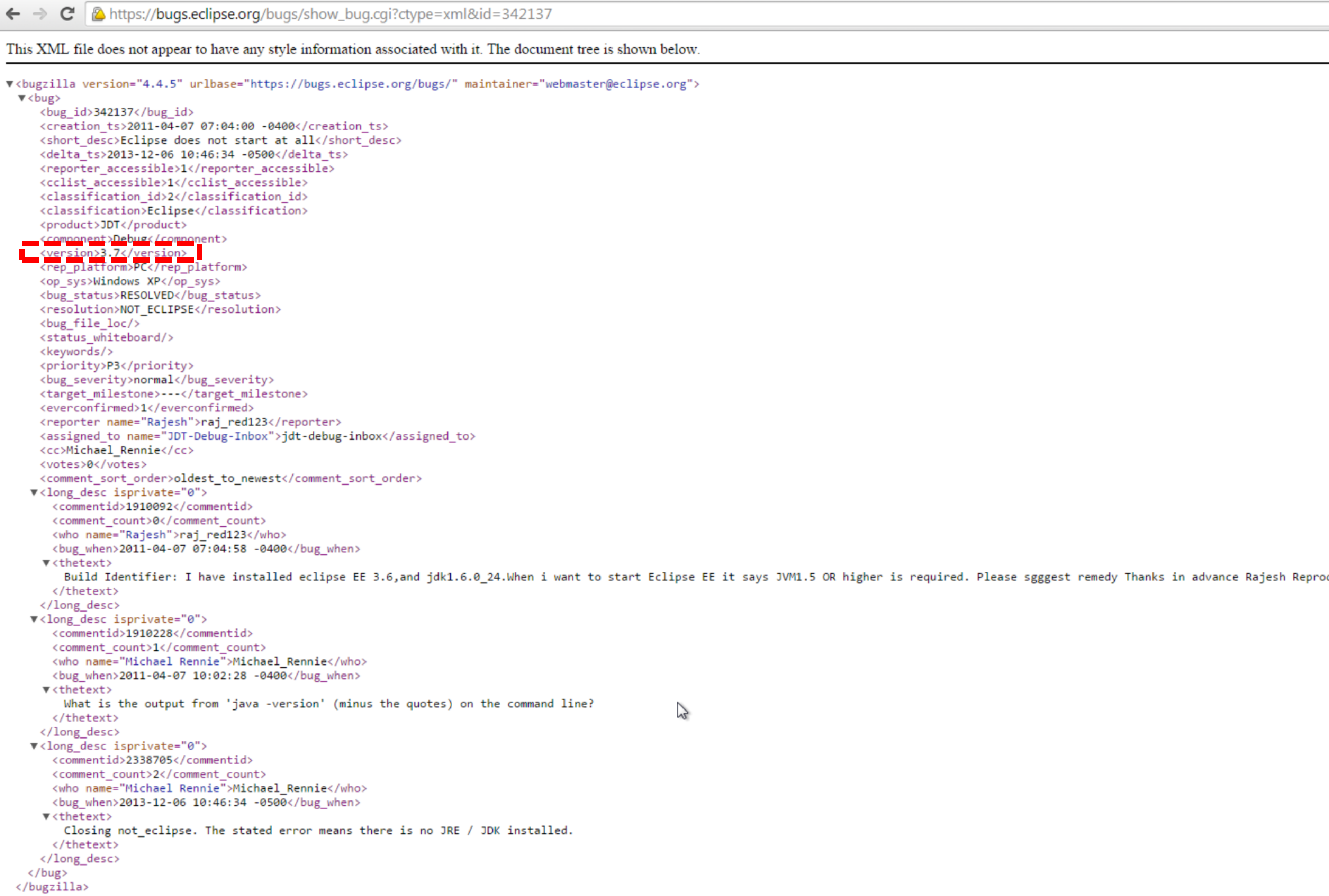
- For each of the bug ID, save the individual bug report in XML format using this link format:
  - https://bugs.eclipse.org/bugs/show_bug.cgi?ctype=xml&id=**BUGID**



https://bugs.eclipse.org/bugs/show_bug.cgi?ctype=xml&id=342137

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<bugzilla version="4.4.5" urlbase="https://bugs.eclipse.org/bugs/" maintainer="webmaster@eclipse.org">
  ▼<bug>
      <bug_id>342137</bug_id>
      <creation_ts>2011-04-07 07:04:00 -0400</creation_ts>
      <short_desc>Eclipse does not start at all</short_desc>
      <delta_ts>2013-12-06 10:46:34 -0500</delta_ts>
      <reporter_accessible>1</reporter_accessible>
      <cclist_accessible>1</cclist_accessible>
      <classification_id>2</classification_id>
      <classification>Eclipse</classification>
      <product>JDT</product>
      <component>Debug</component>
      <version>3.7</version>
      <rep_platform>PC</rep_platform>
      <op_sys>Windows XP</op_sys>
      <bug_status>RESOLVED</bug_status>
      <resolution>NOT_ECLIPSE</resolution>
      <bug_file_loc/>
      <status_whiteboard/>
      <keywords/>
      <priority>P3</priority>
      <bug_severity>normal</bug_severity>
      <target_milestone>---</target_milestone>
      <everconfirmed>1</everconfirmed>
      <reporter name="Rajesh">raj_red123</reporter>
      <assigned_to name="JDT-Debug-Inbox">jdt-debug-inbox</assigned_to>
      <cc>Michael_Rennie</cc>
      <votes>0</votes>
      <comment_sort_order>oldest_to_newest</comment_sort_order>
    ▼<long_desc isprivate="0">
        <commentid>1910092</commentid>
        <comment_count>0</comment_count>
        <who name="Rajesh">raj_red123</who>
        <bug_when>2011-04-07 07:04:58 -0400</bug_when>
      ▼<thetext>
          Build Identifier: I have installed eclipse EE 3.6,and jdk1.6.0_24.When i want to start Eclipse EE it says JVM1.5 OR higher is required. Please sgggest remedy Thanks in advance Rajesh Reproducible: Always
        </thetext>
      </long_desc>
    ▼<long_desc isprivate="0">
        <commentid>1910228</commentid>
        <comment_count>1</comment_count>
        <who name="Michael Rennie">Michael_Rennie</who>
        <bug_when>2011-04-07 10:02:28 -0400</bug_when>
      ▼<thetext>
          What is the output from 'java -version' (minus the quotes) on the command line?
        </thetext>
```



https://bugs.eclipse.org/bugs/show_bug.cgi?ctype=xml&id=342137

This XML file does not appear to have any style information associated with it. The document tree is shown below.

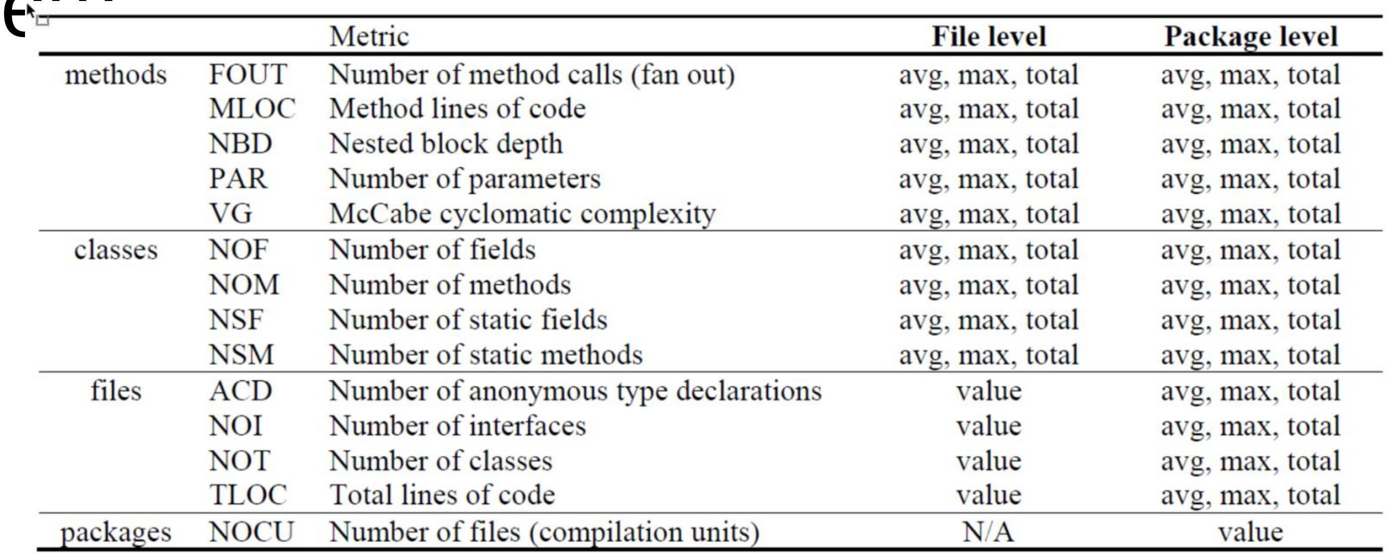
```
▼<bugzilla version="4.4.5" urlbase="https://bugs.eclipse.org/bugs/" maintainer="webmaster@eclipse.org">
  ▼<bug>
     <bug_id>342137</bug_id>
     <creation_ts>2011-04-07 07:04:00 -0400</creation_ts>
     <short_desc>Eclipse does not start at all</short_desc>
     <delta_ts>2013-12-06 10:46:34 -0500</delta_ts>
     <reporter_accessible>1</reporter_accessible>
     <cclist_accessible>1</cclist_accessible>
     <classification_id>2</classification_id>
     <classification>Eclipse</classification>
     <product>JDT</product>
     <component>Debug</component>
     <version>3.7</version>
     <rep_platform>PC</rep_platform>
     <op_sys>Windows XP</op_sys>
     <bug_status>RESOLVED</bug_status>
     <resolution>NOT_ECLIPSE</resolution>
     <bug_file_loc/>
     <status_whiteboard/>
     <keywords/>
     <priority>P3</priority>
     <bug_severity>normal</bug_severity>
     <target_milestone>---</target_milestone>
     <everconfirmed>1</everconfirmed>
     <reporter name="Rajesh">raj_red123</reporter>
     <assigned_to name="JDT-Debug-Inbox">jdt-debug-inbox</assigned_to>
     <cc>Michael_Rennie</cc>
     <votes>0</votes>
     <comment_sort_order>oldest_to_newest</comment_sort_order>
   ▼<long_desc isprivate="0">
       <commentid>1910092</commentid>
       <comment_count>0</comment_count>
       <who name="Rajesh">raj_red123</who>
       <bug_when>2011-04-07 07:04:58 -0400</bug_when>
     ▼<thetext>
         Build Identifier: I have installed eclipse EE 3.6,and jdk1.6.0_24.When i want to start Eclipse EE it says JVM1.5 OR higher is required. Please sgggest remedy Thanks in advance Rajesh Reproc
       </thetext>
     </long_desc>
   ▼<long_desc isprivate="0">
       <commentid>1910228</commentid>
       <comment_count>1</comment_count>
       <who name="Michael Rennie">Michael_Rennie</who>
       <bug_when>2011-04-07 10:02:28 -0400</bug_when>
     ▼<thetext>
         What is the output from 'java -version' (minus the quotes) on the command line?
       </thetext>
     </long_desc>
   ▼<long_desc isprivate="0">
       <commentid>2338705</commentid>
       <comment_count>2</comment_count>
       <who name="Michael Rennie">Michael_Rennie</who>
       <bug_when>2013-12-06 10:46:34 -0500</bug_when>
     ▼<thetext>
         Closing not_eclipse. The stated error means there is no JRE / JDK installed.
       </thetext>
     </long_desc>
   </bug>
 </bugzilla>
```

Pre-release vs. Post-release defects

- Pre-release defects
  - The defects were reported in the *last six months before release*
- Post-release defects
  - The defects were report in the *first six months after the release*
- For example, bug #342137 was reported on 2011/04/07 for version 3.7. Eclipse version 3.7 was release on 2011/06/22. Hence it’s a pre-release bug.

Calculating the Complexity Metrics

- For each of the releases (2.0, 2.1, 3.0), we calculate the code complexity metrics at the file and the package level. (A total of 6 files)

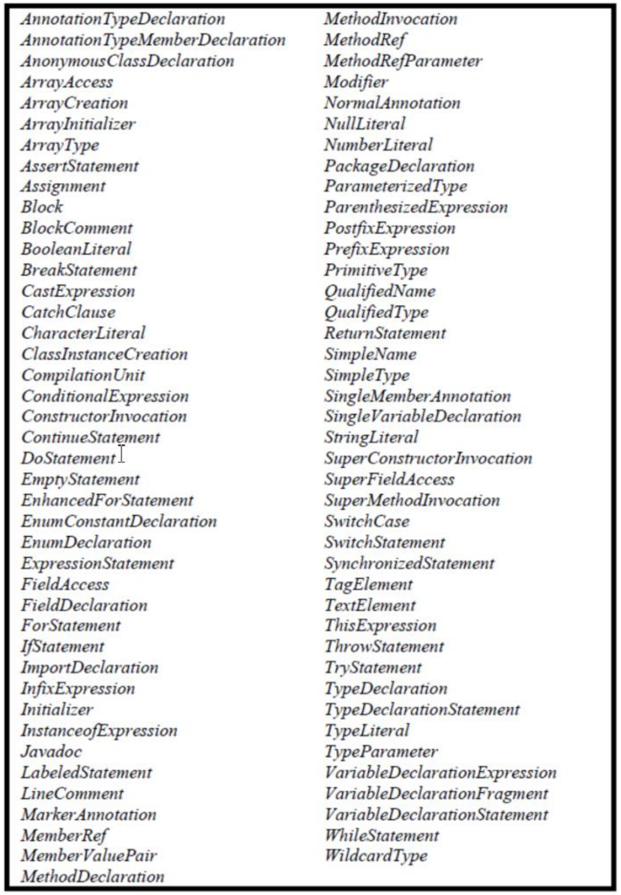


		Metric	**File level**	**Package level**
methods	FOUT	Number of method calls (fan out)	avg, max, total	avg, max, total
	MLOC	Method lines of code	avg, max, total	avg, max, total
	NBD	Nested block depth	avg, max, total	avg, max, total
	PAR	Number of parameters	avg, max, total	avg, max, total
	VG	McCabe cyclomatic complexity	avg, max, total	avg, max, total
classes	NOF	Number of fields	avg, max, total	avg, max, total
	NOM	Number of methods	avg, max, total	avg, max, total
	NSF	Number of static fields	avg, max, total	avg, max, total
	NSM	Number of static methods	avg, max, total	avg, max, total
files	ACD	Number of anonymous type declarations	value	avg, max, total
	NOI	Number of interfaces	value	avg, max, total
	NOT	Number of classes	value	avg, max, total
	TLOC	Total lines of code	value	avg, max, total
packages	NOCU	Number of files (compilation units)	N/A	value

- ***Omit the minimum values, since they are mostly zero.***
- ***File level is different from class level, as one file can have multiple classes***

Structure of the Abstract Syntax Trees (ASTs)

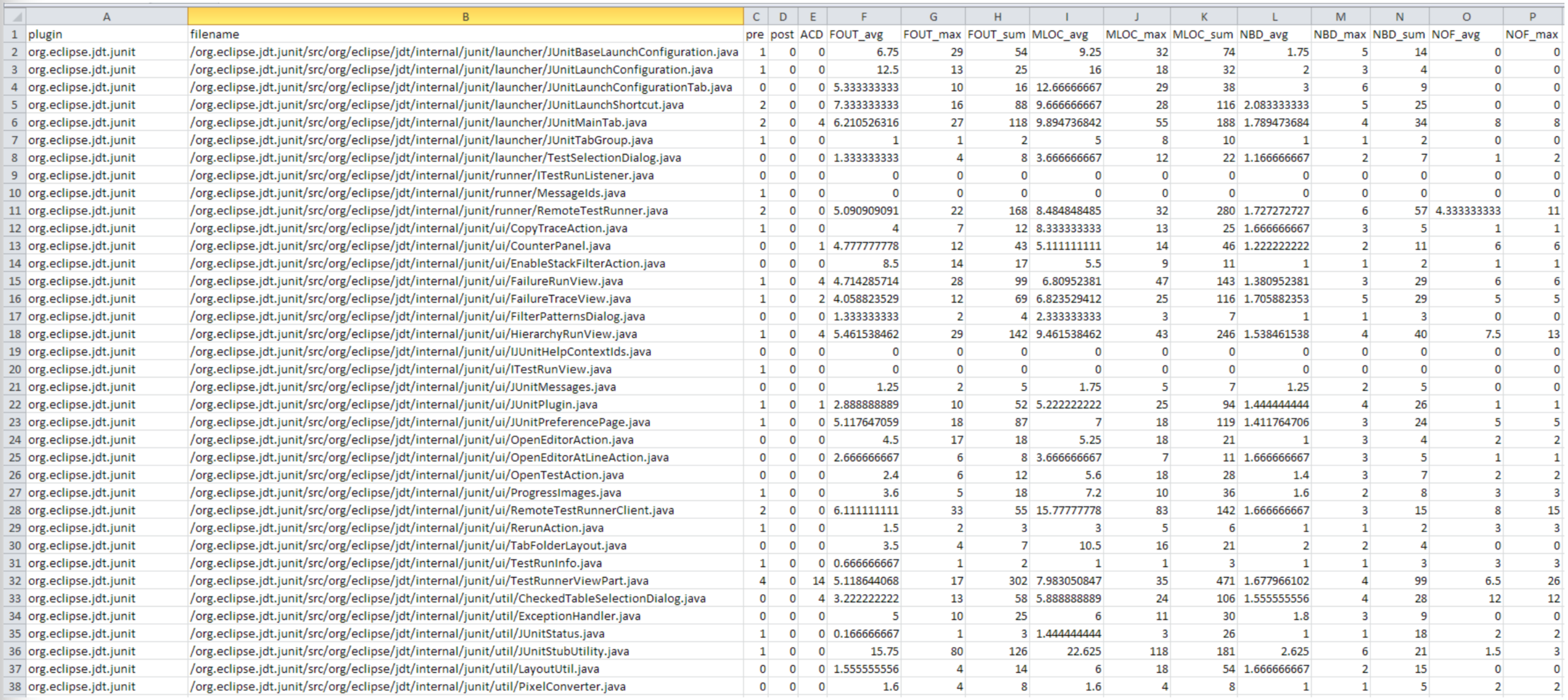
- For each case (file/package), additional data from the structure of the ASTs are also tracked
  - They can be used to calcula without processing the source
- Consult the Eclipse JDT pkg



AnnotationTypeDeclaration
AnnotationTypeMemberDeclaration
AnonymousClassDeclaration
ArrayAccess
ArrayCreation
ArrayInitializer
ArrayType
AssertStatement
Assignment
Block
BlockComment
BooleanLiteral
BreakStatement
CastExpression
CatchClause
CharacterLiteral
ClassInstanceCreation
CompilationUnit
ConditionalExpression
ConstructorInvocation
ContinueStatement
DoStatement
EmptyStatement
EnhancedForStatement
EnumConstantDeclaration
EnumDeclaration
ExpressionStatement
FieldAccess
FieldDeclaration
ForStatement
IfStatement
ImportDeclaration
InfixExpression
Initializer
InstanceofExpression
Javadoc
LabeledStatement
LineComment
MarkerAnnotation
MemberRef
MemberValuePair
MethodDeclaration
MethodInvocation
MethodRef
MethodRefParameter
Modifier
NormalAnnotation
NullLiteral
NumberLiteral
PackageDeclaration
ParameterizedType
ParenthesizedExpression
PostfixExpression
PrefixExpression
PrimitiveType
QualifiedName
QualifiedType
ReturnStatement
SimpleName
SimpleType
SingleMemberAnnotation
SingleVariableDeclaration
StringLiteral
SuperConstructorInvocation
SuperFieldAccess
SuperMethodInvocation
SwitchCase
SwitchStatement
SynchronizedStatement
TagElement
TextElement
ThisExpression
ThrowStatement
TryStatement
TypeDeclaration
TypeDeclarationStatement
TypeLiteral
TypeParameter
VariableDeclarationExpression
VariableDeclarationFragment
VariableDeclarationStatement
WhileStatement
WildcardType

eclipse-metrics-files-2.0.csv

Caution: the delimiter is semicolon not



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	plugin	filename	pre	post	ACD	FOUT_avg	FOUT_max	FOUT_sum	MLOC_avg	MLOC_max	MLOC_sum	NBD_avg	NBD_max	NBD_sum	NOF_avg	NOF_max
2	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/launcher/JUnitBaseLaunchConfiguration.java	1	0	0	6.75	29	54	9.25	32	74	1.75	5	14	0	0
3	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/launcher/JUnitLaunchConfiguration.java	1	0	0	12.5	13	25	16	18	32	2	3	4	0	0
4	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/launcher/JUnitLaunchConfigurationTab.java	0	0	0	5.333333333	10	16	12.66666667	29	38	3	6	9	0	0
5	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/launcher/JUnitLaunchShortcut.java	2	0	0	7.333333333	16	88	9.666666667	28	116	2.083333333	5	25	0	0
6	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/launcher/JUnitMainTab.java	2	0	4	6.210526316	27	118	9.894736842	55	188	1.789473684	4	34	8	8
7	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/launcher/JUnitTabGroup.java	1	0	0	1	1	2	5	8	10	1	1	2	0	0
8	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/launcher/TestSelectionDialog.java	0	0	0	1.333333333	4	8	3.666666667	12	22	1.166666667	2	7	1	2
9	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/runner/ITestRunListener.java	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/runner/MessageIds.java	1	0	0	0	0	0	0	0	0	0	0	0	0	0
11	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/runner/RemoteTestRunner.java	2	0	0	5.090909091	22	168	8.484848485	32	280	1.727272727	6	57	4.333333333	11
12	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/CopyTraceAction.java	1	0	0	4	7	12	8.333333333	13	25	1.666666667	3	5	1	1
13	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/CounterPanel.java	0	0	1	4.777777778	12	43	5.111111111	14	46	1.222222222	2	11	6	6
14	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/EnableStackFilterAction.java	0	0	0	8.5	14	17	5.5	9	11	1	1	2	1	1
15	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/FailureRunView.java	1	0	4	4.714285714	28	99	6.80952381	47	143	1.380952381	3	29	6	6
16	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/FailureTraceView.java	1	0	2	4.058823529	12	69	6.823529412	25	116	1.705882353	5	29	5	5
17	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/FilterPatternsDialog.java	0	0	0	1.333333333	2	4	2.333333333	3	7	1	1	3	0	0
18	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/HierarchyRunView.java	1	0	4	5.461538462	29	142	9.461538462	43	246	1.538461538	4	40	7.5	13
19	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/IJUnitHelpContextIds.java	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/ITestRunView.java	1	0	0	0	0	0	0	0	0	0	0	0	0	0
21	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/JUnitMessages.java	0	0	0	1.25	2	5	1.75	5	7	1.25	2	5	0	0
22	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/JUnitPlugin.java	1	0	1	2.888888889	10	52	5.222222222	25	94	1.444444444	4	26	1	1
23	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/JUnitPreferencePage.java	1	0	0	5.117647059	18	87	7	18	119	1.411764706	3	24	5	5
24	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/OpenEditorAction.java	0	0	0	4.5	17	18	5.25	18	21	1	3	4	2	2
25	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/OpenEditorAtLineAction.java	0	0	0	2.666666667	6	8	3.666666667	7	11	1.666666667	3	5	1	1
26	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/OpenTestAction.java	0	0	0	2.4	6	12	5.6	18	28	1.4	3	7	2	2
27	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/ProgressImages.java	1	0	0	3.6	5	18	7.2	10	36	1.6	2	8	3	3
28	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/RemoteTestRunnerClient.java	2	0	0	6.111111111	33	55	15.77777778	83	142	1.666666667	3	15	8	15
29	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/RerunAction.java	1	0	0	1.5	2	3	3	5	6	1	1	2	3	3
30	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/TabFolderLayout.java	0	0	0	3.5	4	7	10.5	16	21	2	2	4	0	0
31	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/TestRunInfo.java	1	0	0	0.666666667	1	2	1	1	3	1	1	3	3	3
32	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/ui/TestRunnerViewPart.java	4	0	14	5.118644068	17	302	7.983050847	35	471	1.677966102	4	99	6.5	26
33	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/util/CheckedTableSelectionDialog.java	0	0	4	3.222222222	13	58	5.888888889	24	106	1.555555556	4	28	12	12
34	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/util/ExceptionHandler.java	0	0	0	5	10	25	6	11	30	1.8	3	9	0	0
35	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/util/JUnitStatus.java	1	0	0	0.166666667	1	3	1.444444444	3	26	1	1	18	2	2
36	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/util/JUnitStubUtility.java	1	0	0	15.75	80	126	22.625	118	181	2.625	6	21	1.5	3
37	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/util/LayoutUtil.java	0	0	0	1.555555556	4	14	6	18	54	1.666666667	2	15	0	0
38	org.eclipse.jdt.junit	/org.eclipse.jdt.junit/src/org/eclipse/jdt/internal/junit/util/PixelConverter.java	0	0	0	1.6	4	8	1.6	4	8	1	1	5	2	2

What do we want to learn from this data?

- Finding a single indicator or predictor for the number of defects is extremely unlikely. Hence, we need to combine input features by building regression models

- Which files/packages have defects?
  - This is a ***classification*** problem

- Which files/packages have the most defects?
  - This is a ***ranking*** problem

Which files/packages have defects?

- Classify files/packages as defect-prone or not based on the code metrics
  - Defect-prone: has_defect = 1
  - Defect-free: has_defect = 0
- **Logistic regression** is useful when predicting a binary outcome (post-release bugs) from a set of continuous (e.g., FOUT_avg, MLOG_avg) and/or categorical predictor variables. Logistic regression models typically predict the likelihoods a value between [0, 1]:

$$\textit{Defect Classification} = \begin{cases} \text{defect-prone } (0.5 < \text{value} \le 1) \\ \text{defect-free } (0 \le \text{value} \le 0.5) \end{cases}$$

  - Logistic regression is a type of *glm* (generalized linear models)
- Build (train) the model using data from one version (e.g., v2.0) and test the model on another version (e.g., v2.1)

Evaluate the Performance of the Defect Classification Models

	Are defects observed?	
	True	False
Positive	True Positive (TP)	False Positive (FP)
Negative	False Negative (FN)	True Negative (TN)

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Which files/packages have the most post-release defects?

- Use **(multiple) linear regression models** to predict the number of post-release defects for each files/packages based on code metrics
- Similarly, we build (train) the model using data from one version (e.g., v2.0) and test the model on another version (e.g., v2.1)

Evaluate the Performance of the Defect Ranking Models

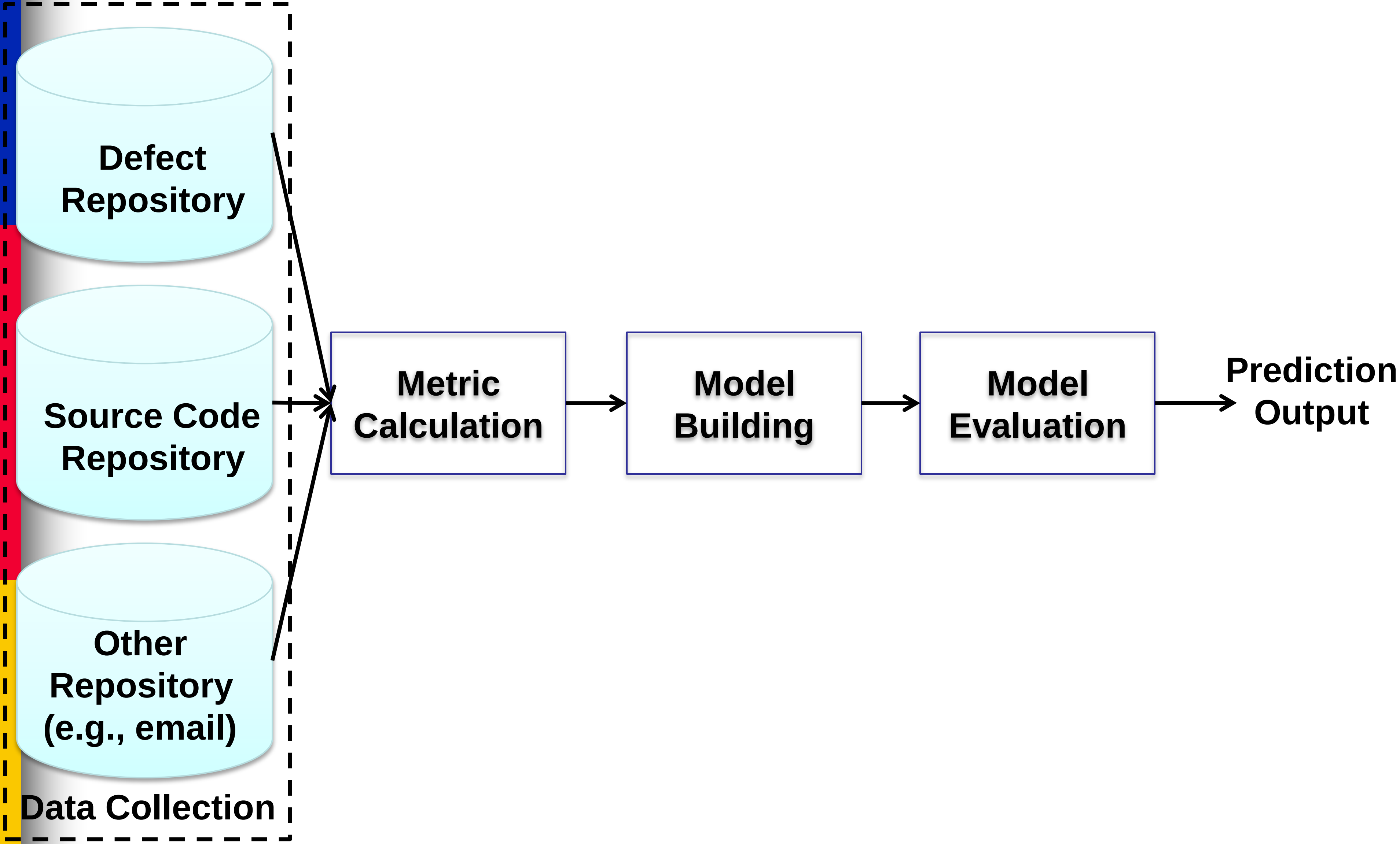
- List R^2 of the trained model
- Compared the predicted resulting ranking with the actual observed ranking
  - Spearman rank correlation
- Only for the sake of completeness, they also calculated the Pearson correlation
  - *Pearson correlation assumes a linear relationships between the correlated variables.*

Discussion

- How can we improve the performance of bug prediction?
  - More data
  - New prediction models
- Did the data linking (code changes to bugs) approach manage to extract all the bug data?
  - Any bias in the bug-fix dataset?
- Was the statistical analysis performed properly?
  - Was statistical assumptions violated?
  - How can we open up (understand) this prediction mode?
- How can we make bug prediction more useful?
  - Effort-aware
  - Security/Performance/etc.,
  - Re-opened
  - Just-in-time
  - Cross-project
  - “Surprise”
  - Fine-grained (method-level)
  - etc.,

More metrics and more prediction techniques

General Process

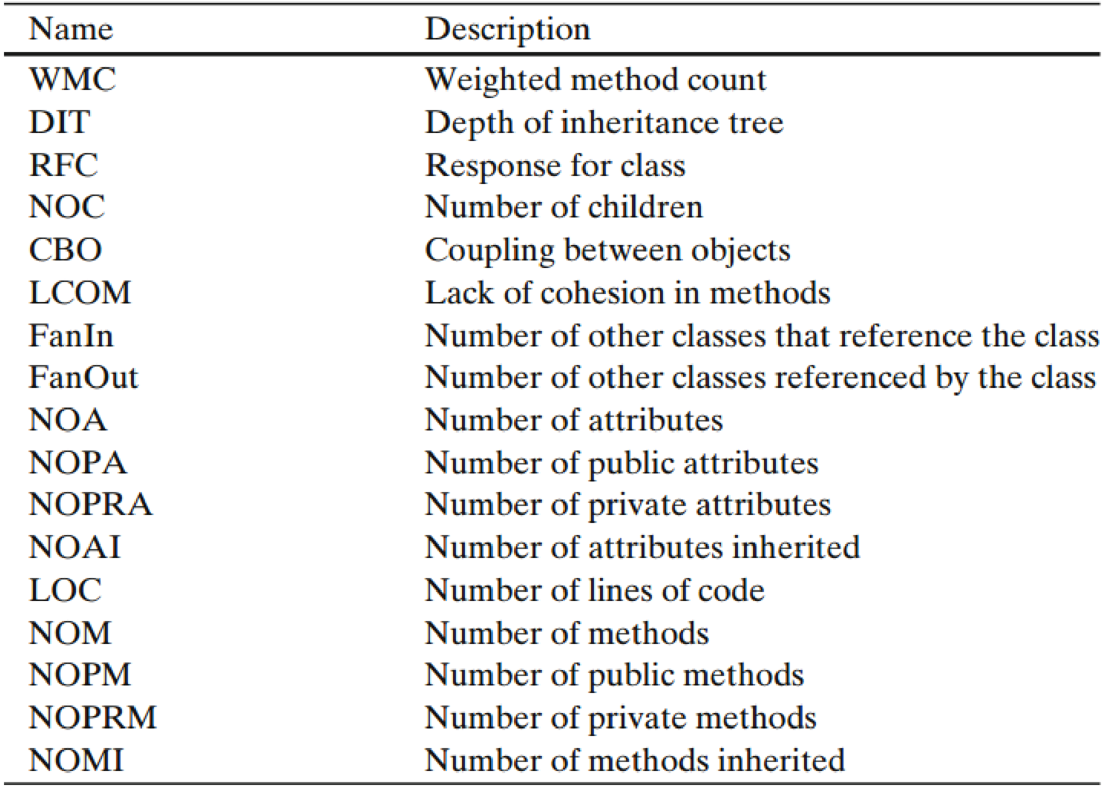


Metrics Data Used in the Bug Prediction Models

- Independent variables:
  - *Product factors* (e.g., code size) are used to predict
  - *Process factors* (e.g., churn) are used to predict
  - *Other factors*: other than product and process used to predict
- Dependent variables:
  - *Pre:* Predictions are made to predict pre-release defects
  - *Post*: Predictions are made to predict post-release defects
  - *Other*: Predictions are made for a dependent variable other than pre- and post-release defects

Product Metrics

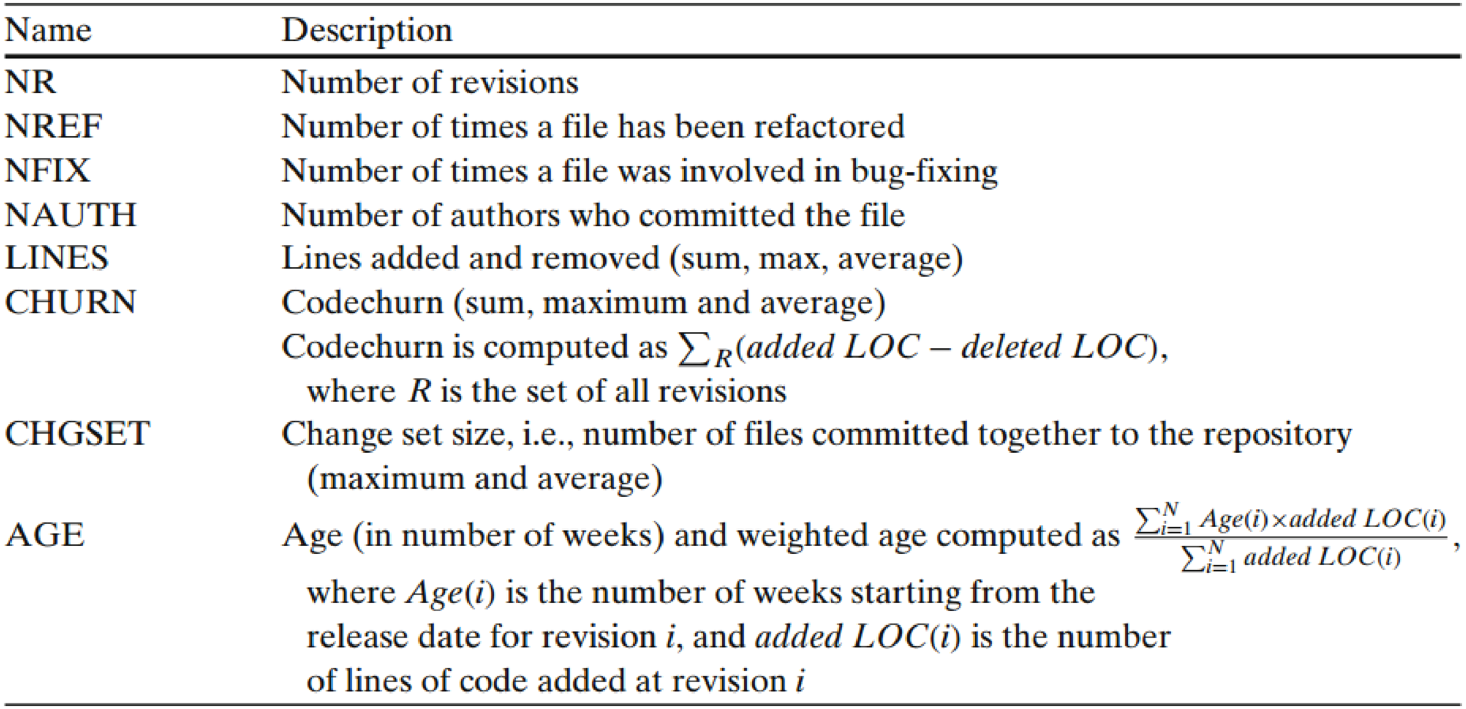
- Rationale:
  - *Complex components are harder to change. Hence, they are more error prone*
- *Product Metrics* (also called *Source Code Metrics*) are metrics that are directly derived from the source code (e.g., complexity or size).



Name	Description
WMC	Weighted method count
DIT	Depth of inheritance tree
RFC	Response for class
NOC	Number of children
CBO	Coupling between objects
LCOM	Lack of cohesion in methods
FanIn	Number of other classes that reference the class
FanOut	Number of other classes referenced by the class
NOA	Number of attributes
NOPA	Number of public attributes
NOPRA	Number of private attributes
NOAI	Number of attributes inherited
LOC	Number of lines of code
NOM	Number of methods
NOPM	Number of public methods
NOPRM	Number of private methods
NOMI	Number of methods inherited

Process Metrics

- Rationale:
  - *Bugs are caused by changes*
- E.g., a piece of code is changed many times or by many people, this may indicate that it is more likely to be defect prone.



Name	Description
NR	Number of revisions
NREF	Number of times a file has been refactored
NFIX	Number of times a file was involved in bug-fixing
NAUTH	Number of authors who committed the file
LINES	Lines added and removed (sum, max, average)
CHURN	Codechurn (sum, maximum and average) Codechurn is computed as $\sum_R (added\ LOC - deleted\ LOC)$, where R is the set of all revisions
CHGSET	Change set size, i.e., number of files committed together to the repository (maximum and average)
AGE	Age (in number of weeks) and weighted age computed as $\frac{\sum_{i=1}^{N} Age(i) \times added\ LOC(i)}{\sum_{i=1}^{N} added\ LOC(i)}$, where $Age(i)$ is the number of weeks starting from the release date for revision i, and $added\ LOC(i)$ is the number of lines of code added at revision i

Other process metrics are # of pre-release defects, relative churn, social, ownership, etc.

Other Metrics used as the Independent Variables

- **Execution**: Captures the execution characteristics of a software system. For example, execution factors can be the deployment percentage of a module and the average transaction time on a system serving a typical user.
- **Programming Language**: The programming language in which the software is written. For example, Java, C, C++ or Perl.
- **Module Knowledge**: A subjective measure which captures the team's knowledge of a module.
- **Design/UML**: Are factors that capture the design of the software system. These factors can be derived from the definition of the class interfaces at the design stage (e.g., from UML diagrams). These factors may include class factors, parameter types, class attributes and inheritance relationships.
- **Platform and Hardware Configuration**: Factors that capture the platform and HW configurations that software system runs on. For example, whether the software system runs on a Windows or Linux based platform and whether it runs on a single- or multi-core system.

Dependent Variables

- **Post-release Defects**: is the number of defects that appear after the software is released. Generally, post-release defects is the number of defects within six months of the software release date.
- **Defect Density**: is generally measured as the number of defects per LOC or KLOC.
- **Defect-introducing Change**: is a dependent variable that specifies whether a change introduced a defect.
- **Vulnerabilities**: is a dependent variable which accounts for a security vulnerability that exists in a software artifact.

Bug Prediction Models

Category	Model	Notes
Statistical	Naive Bayes	
	MARS	A multivariate adaptive regression splines model
	Logistic regression	
	Linear regression	
Tree-based	Decision trees	
	Random forests	
	CART	A classification and regression trees model
	Recursive partitioning	
	SVM	Support Vector Machine

Other Bug Prediction Studies

- How can we make bug prediction more useful?

- Effort-aware
- Security/Performance/etc.,
- Re-opened
- Just-in-time
- Cross-project
- “Surprise”
- Fine-grained (method-level)
- etc.,

References

- Emad Shihab. An Exploration of Challenges Limiting Pragmatic Software Defect Prediction. PhD Thesis. School of Computing. Queen's University, Ontario, Canada, 2012. [Chapter 2]
- Marco D'Ambros, Michele Lanza, Romain Robbes. Evaluating defect prediction approaches: a benchmark and an extensive comparison. Empirical Software Engineering (EMSE). 2012.
  - Dataset: http://bug.inf.usi.ch/