

# SIP- The Software Improvement Process



Tools and Rules to Encourage Quality

Katarina Sigerud, CERN

13<sup>th</sup> October, ICALEPCS 2011

# Objectives

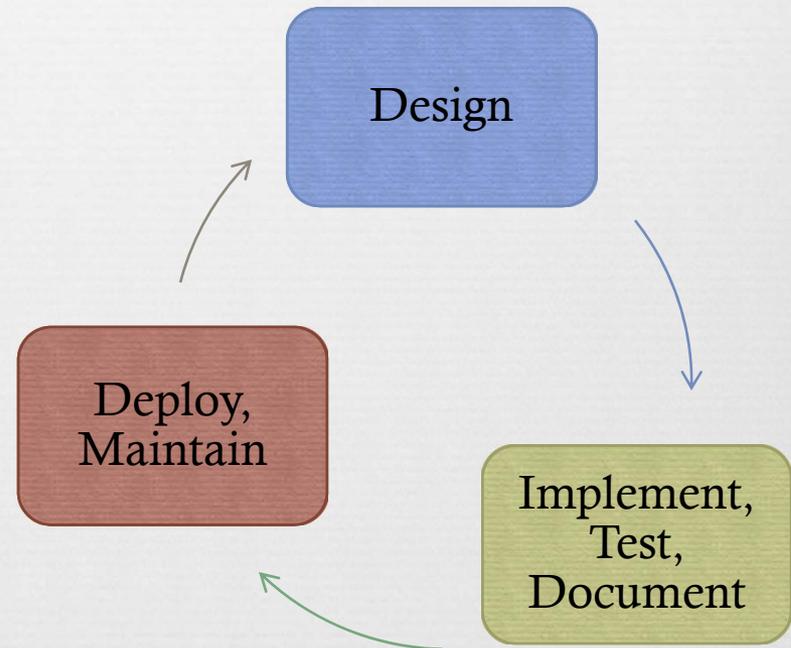


- ❧ Introduce quality improvement as an integral part of the everyday development work
- ❧ Leverage tools to automate the process as much as possible
- ❧ Establish guidelines and metrics to measure progress

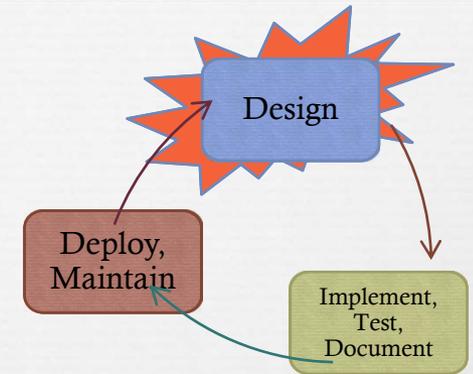
# Quality in the development cycle



- ☞ For each stage
  - ☞ Agreed mandatory activities and deliverables
  - ☞ Tools identified and integrated with IDE (Eclipse), giving immediate feedback

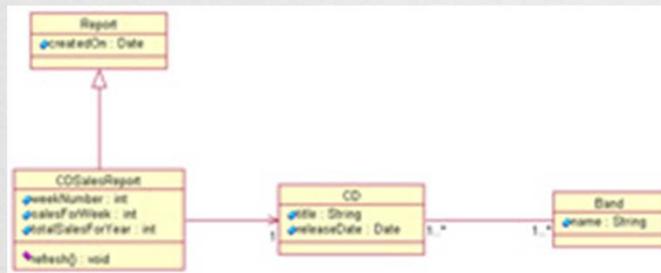


# Design reviews

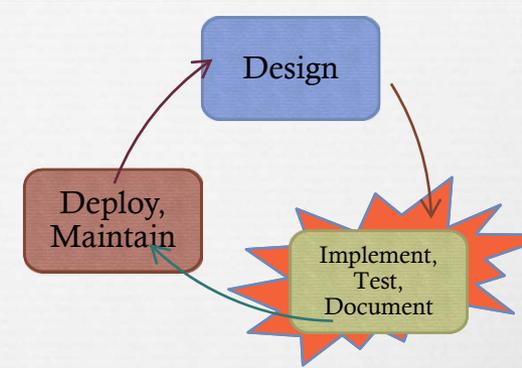


- ☞ To verify the soundness of design in an early stage
- ☞ To identify overlapping functionality

- ☞ At the level of sub-components, main classes and design patterns
- ☞ UML: class and sequence diagrams



# Code reviews



- ❧ Goal:
  - ❧ Identify defects
  - ❧ Ensure maintainable code
  - ❧ Verify conventions are followed
- ❧ Static code analysis tools identifies common mistakes and bug patterns:
  - ❧ FindBugs
  - ❧ Checkstyle
  - ❧ Eclipse warnings
- ❧ Person-to-person time consuming
  - ❧ Only for critical code
  - ❧ Mentoring of junior developers
- ❧ Light-weight person-to-person code reviews with FishEye + Crucible

FindBugs - sip-top-flop/src/java/cern/sandbox/topflop/TopFlopGen.java - Eclipse - GALILEO

File Edit Source Refactor Navigate Search Project Run Window Help

Bug Explorer

- Code contains a hard coded reference to an absolute pathname (2)
  - Hard coded reference to an absolute pathname
  - Hard coded reference to an absolute pathname
- Dead store to local variable (1)

```

154 }
155
156 private int parseInt(String numberString) {
157     return Integer.parseInt(numberString == null || "".equalsIgnoreCase(numberString) ?
158 }
159
160 private TopFlopMetrics prepareMetricsFromTest() throws ParserConfigurationException, SA
161     TopFlopMetrics metrics = new TopFlopMetrics();
162
163 File bambooBuildDir = new File
164     "C:\\Users\\ksigerud\\eclipse\\projects\\sandbox\\sip-top-flop\\src\\test-r
165 File[] listFiles = bambooBuildDir.listFiles();
166 for (int i = 0; i < listFiles.length; i++) {
167     File cloverFile = listFiles[i];
168     if (!cloverFile.isDirectory() && cloverFile.getName().contains("-coverage")) {
169         String projectName = cloverFile.getName().substring(0, cloverFile.getName()
170             metrics.addProjectMetrics(parseCoverageFile(cloverFile, projectName));
171     }
172 }
173
174 return metrics;
175 }
176 }
177
    
```

Properties Problems

**Bug: Hard coded reference to an absolute pathname**

- > **Bug: Hard coded reference to an absolute pathname**
- > **Pattern id: DMI\_HARDCODED\_ABSOLUTE\_FILENAME, type: DMI, category: STYLE**
- > This code constructs a File object using a hard coded to an absolute pathname (e.g., new File ("/home/dannyc/workspace/j2ee/src/share/com/sun/enterprise/deployment"));

Hard coded reference to an absolute pathname | Writable | Smart Insert | 163 : 40

FindBugs - sip-top-flop/src/java/cern/sandbox/topflop/TopFlopGen.java - Eclipse - GALILEO

File Edit Source Refactor Navigate Search Project Run Window Help

Bug Explorer

- Code contains a hard coded reference to an absolute pathname (2)
- Hard coded reference to an absolute pathname
- Hard coded reference to an absolute pathname
- Dead store to local variable (1)

A list of 'bugs'

```

154 }
155
156 private int parseInteger(String numberString) {
157     return Integer.parseInt(numberString == null || "".equalsIgnoreCase(numberString) ?
158     }
159
160 private TopFlopMetrics prepareMetricsFromTest() throws ParserConfigurationException, SAX
161     TopFlopMetrics metrics = new TopFlopMetrics();
162
163     File bambooBuildDir = new File
164         "C:\\Users\\ksigerud\\eclipse\\projects\\sandbox\\sip-top-flop\\src\\test-r
165     File[] listFiles = bambooBuildDir.listFiles();
166     for (int i = 0; i < listFiles.length; i++) {
167         File cloverFile = listFiles[i];
168         if (!cloverFile.isDirectory() && cloverFile.getName().contains("-coverage")) {
169             String projectName = cloverFile.getName().substring(0, cloverFile.getName()
170             metrics.addProjectMetrics(parseCoverageFile(cloverFile, projectName));
171         }
172     }
173
174     return metrics;
175 }
176 }
177
  
```

Properties Problems

**Bug: Hard coded reference to an absolute pathname**

**Bug: Hard coded reference to an absolute pathname**  
**Pattern id: DMI\_HARDCODED\_ABSOLUTE\_FILENAME, type: DMI, category: STYLE**

This code constructs a File object using a hard coded to an absolute pathname (e.g., new File ("/home/dannyc/workspace/j2ee/src/share/com/sun/enterprise/deployment"));

Hard coded reference to an absolute pathname | Writable | Smart Insert | 163 : 40

FindBugs - sip-top-flop/src/java/cern/sandbox/topflop/TopFlopGen.java - Eclipse - GALILEO

File Edit Source Refactor Navigate Search Project Run Window Help

Bug Explorer

- Code contains a hard coded reference to an absolute pathname (2)
- Hard coded reference to an absolute pathname
- Hard coded reference to an absolute pathname
- Dead store to local variable (1)

The 'bug' line indicated

```

156 ...numberString) {
157     ...parseInt(numberString == null || "".equalsIgnoreCase(numberString) ?
158
159
160 private TopFlopMetrics prepareMetricsFromTest() throws ParserConfigurationException, SA
161     TopFlopMetrics metrics = new TopFlopMetrics();
162
163     File bambooBuildDir = new File(
164         "C:\\Users\\ksigerud\\eclipse\\projects\\sandbox\\sip-top-flop\\src\\test-r
165     File[] listFiles = bambooBuildDir.listFiles();
166     for (int i = 0; i < listFiles.length; i++) {
167         File cloverFile = listFiles[i];
168         if (!cloverFile.isDirectory() && cloverFile.getName().contains("-coverage")) {
169             String projectName = cloverFile.getName().substring(0, cloverFile.getName()
170             metrics.addProjectMetrics(parseCoverageFile(cloverFile, projectName));
171         }
172     }
173
174     return metrics;
175 }
176 }
177

```

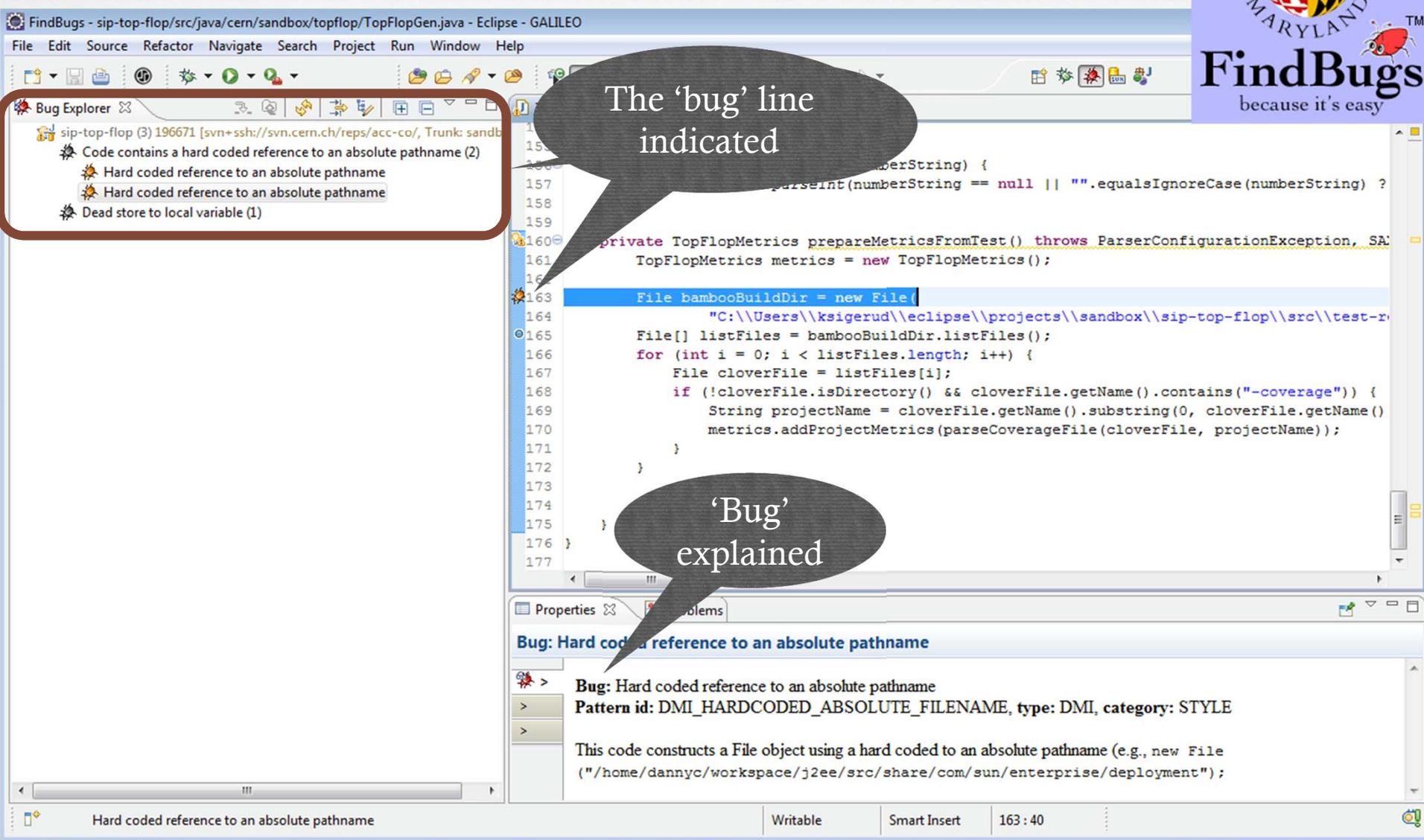
Properties Problems

**Bug: Hard coded reference to an absolute pathname**

**Bug: Hard coded reference to an absolute pathname**  
**Pattern id: DMI\_HARDCODED\_ABSOLUTE\_FILENAME, type: DMI, category: STYLE**

This code constructs a File object using a hard coded to an absolute pathname (e.g., new File ("/home/dannyc/workspace/j2ee/src/share/com/sun/enterprise/deployment"));

Hard coded reference to an absolute pathname      Writable      Smart Insert      163 : 40



FindBugs - sip-top-flop/src/java/cern/sandbox/topflop/TopFlopGen.java - Eclipse - GALILEO

File Edit Source Refactor Navigate Search Project Run Window Help

Bug Explorer

- Code contains a hard coded reference to an absolute pathname (2)
- Hard coded reference to an absolute pathname
- Hard coded reference to an absolute pathname
- Dead store to local variable (1)

The 'bug' line indicated

```
private TopFlopMetrics prepareMetricsFromTest() throws ParserConfigurationException, SAXException, IOException {
    TopFlopMetrics metrics = new TopFlopMetrics();

    File bambooBuildDir = new File(
        "C:\\Users\\ksigerud\\eclipse\\projects\\sandbox\\sip-top-flop\\src\\test-r
File[] listFiles = bambooBuildDir.listFiles();
for (int i = 0; i < listFiles.length; i++) {
    File cloverFile = listFiles[i];
    if (!cloverFile.isDirectory() && cloverFile.getName().contains("-coverage")) {
        String projectName = cloverFile.getName().substring(0, cloverFile.getName().lastIndexOf("."));
        metrics.addProjectMetrics(parseCoverageFile(cloverFile, projectName));
    }
}
}
```

'Bug' explained

**Bug: Hard coded reference to an absolute pathname**

**Bug: Hard coded reference to an absolute pathname**  
**Pattern id: DMI\_HARDCODED\_ABSOLUTE\_FILENAME, type: DMI, category: STYLE**

This code constructs a File object using a hard coded to an absolute pathname (e.g., new File ("/home/dannyc/workspace/j2ee/src/share/com/sun/enterprise/deployment"));

Hard coded reference to an absolute pathname | Writable | Smart Insert | 163 : 40

Software Improvement Project > SIP-REVIEWS-7  
**Top/flop list review**

Reopen Tools

Closed on 09 Oct

Author Reviewers

- SIP-REVIEWS-7 2
- Details
- Objectives
- Summary
- General Comments 1
- acc-co (trunk)
- trunk/sandbox/sip-top-flop/src/java/cern/sandc
  - TopFlopGen.java 1

Prefs Filter 3m

**Details**

Participant	Role	Time Spent	Comments	Latest Comment
<a href="#">Katarina Sigerud</a>	Author	3m		
<a href="#">Pablo Pera Mira</a>	Reviewer - 100% complete	18m	2	<a href="#">I suggest you use the builder pattern here. this long...</a>
<b>Total</b>		<b>21m</b>	<b>2</b>	

Files: 1

**Objectives**

Please review this class using the Eclipse Atlassian connector plugin in view of trying it out.

**Summary**

Thank you for your help! It seems that the Eclipse Atlassian connector still needs some work...

**General Comments**

[Pablo Pera Mira](#) says: 07 Oct

After a couple of hours fighting playing with several installations of Eclipse, I managed to get it all configured in my laptop... Installing/uninstalling, and especially upgrading plugins is pretty painful!

It's nice to navigate the files from Eclipse, it lets you check all changes way faster. It's not essential, though, especially for small reviews.

☆



Dashboard Source Projects People Reviews

Software Improvement Project > SIP-REVIEWS-7

Top/flop list review

Closed on 09 Oct

Author Reviewers

SIP-REVIEWS-7 2

Details Objectives Summary General Comments 1

acc-co (trunk)

trunk/sandbox/sip-top-flop/src/java/cern/sandc

TopFlopGen.java 1

Details

Participant	Role	Time Spent	Comments	Latest Comment
<a href="#">Katarina Sigerud</a>	Author	3m		
<a href="#">Pablo Pera Mira</a>	Reviewer - 100% complete	18m	2	<a href="#">I suggest you use the builder pattern here, this long...</a>
<b>Total</b>		<b>21m</b>	<b>2</b>	

Summary

Thank you for your help! It seems that the Eclipse Atlassian connector still needs some work...

General Comments

[Pablo Pera Mira](#) says: 07 Oct

After a couple of hours fighting playing with several installations of Eclipse, I managed to get it all configured in my laptop... Installing/uninstalling, and especially upgrading plugins is pretty painful!

It's nice to navigate the files from Eclipse, it lets you check all changes way faster. It's not essential, though, especially for small reviews.

Files being reviewed

Dashboard Source Projects People Reviews

Software Improvement Project > SIP-REVIEWS-7

### Top/flop list review

Closed on 09 Oct

Author Reviewers

SIP-REVIEWS-7 2

- Details
- Objectives
- Summary
- General Comments 1

acc-co (trunk)

- trunk/sandbox/sip-top-flop/src/java/cern/sandt
  - TopFlopGen.java 1

Participant	Role	Comments	Latest Comment
<a href="#">Katarina Sigerud</a>	Author		
<a href="#">Pablo Pera Mira</a>	Reviewer - 100% complete	2	<a href="#">I suggest you use the builder pattern here, this long...</a>
<b>Total</b>		<b>21m</b>	<b>2</b>

**Details**

**Summary**

Thank you for your help! It seems that the Eclipse Atlassian connector still needs some work...

**General Comments**

[Pablo Pera Mira](#) says: 07 Oct

After a couple of hours fighting playing with several installations of Eclipse, I managed to get it all configured in my laptop... Installing/uninstalling, and especially upgrading plugins is pretty painful!

It's nice to navigate the files from Eclipse, it lets you check all changes way faster. It's not essential, though, especially for small reviews.

Annotations:

- Author and reviewers (points to participant table)
- Files being reviewed (points to file list)

```

129     int nbOfMethods = parseInt(metrics.getAttribute("methods"));
130     int nbOfClasses = parseInt(metrics.getAttribute("classes"));
131     int nbOfFiles = parseInt(metrics.getAttribute("files"));
132     int nbOfPackages = parseInt(metrics.getAttribute("packages"));
133     int complexity = parseInt(metrics.getAttribute("complexity"));
134     int nbOfLoc = parseInt(metrics.getAttribute("loc"));
135     int nbOfNcLoc = parseInt(metrics.getAttribute("ncloc"));
136     int nbOfStatements = parseInt(metrics.getAttribute("statements"));
137     int nbOfCoveredElements = parseInt(metrics.getAttribute("coveredelements"));
138     int nbOfElements = parseInt(metrics.getAttribute("elements"));
139

```

**Pablo Pera Mira** 07-Oct-2011 19:24 DRAFT [Line: 139; Line: 140]

I'd suggest you use the builder pattern here, this long list of arguments is very error-prone

Press 'F2' for focus.

```

xHtml.getPath(), nbOfMethods, nbOfClasses, nbOfFiles,
Loc, nbOfStatements, nbOfCoveredElements, nbOfElements);

```



```

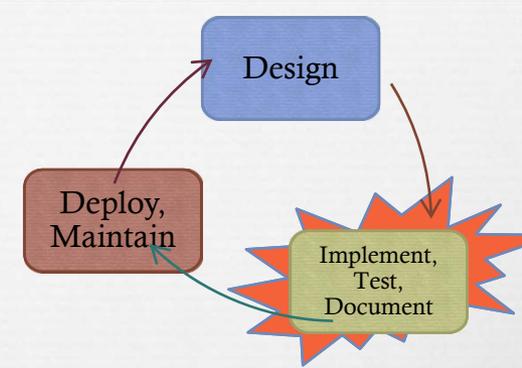
145     File cloverIndexHtml = new File(parentDir.getPath() + "/index.html");
146     if (!cloverIndexHtml.exists()) {
147         cloverIndexHtml = new File(parentDir.getPath() + "/html/index.html");
148     }
149     return cloverIndexHtml;
150 }
151
152 private int parseInt(String numberString) {
153     return Integer.parseInt(numberString == null || "".equalsIgnoreCase(numberString) ? "0" : numberString);
154 }
155
156 private TopFlopMetrics prepareMetricsFromTest() throws ParserConfigurationException, SAXException, IOException {
157     TopFlopMetrics metrics = new TopFlopMetrics();
158
159     File bambooBuildDir = new File(
160         "C:\\Users\\ksigerud\\eclipse\\projects\\sandbox\\sip-top-flop\\src\\test-resources");
161     File[] listFiles = bambooBuildDir.listFiles();
162     for (int i = 0; i < listFiles.length; i++) {
163         File cloverFile = listFiles[i];

```

Workspace Log  
type filter text

Message	Plug-in	Date
⚠ There's no mapping for Crucible repository acc-co. Review item will be downl...	com.atlassian.connector.eclipse.crucible.ui	07/10/2011 19:30
⚠ There's no mapping for Crucible repository acc-co. Review item will be downl...	com.atlassian.connector.eclipse.crucible.ui	07/10/2011 19:30
⚠	com.atlassian.connector.eclipse.crucible.ui	07/10/2011 19:15
⚠	com.atlassian.connector.eclipse.crucible.ui	07/10/2011 19:14
⚠	com.atlassian.connector.eclipse.crucible.ui	07/10/2011 19:13
⚠	com.atlassian.connector.eclipse.crucible.ui	07/10/2011 19:13
⚠	com.atlassian.connector.eclipse.crucible.ui	07/10/2011 19:13

# Testing



- ❧ General agreement on benefits of unit tests
- ❧ To increase level of testing, unit tests mandatory deliverable of project
  - ❧ <30% test coverage for non-trivial classes, measured with Clover
- ❧ To discover inter-project issues early, continuous builds with Bamboo
  - ❧ Compiles and runs unit tests

```

Java - isa-dbaccess/src/java/cern/isa/dbaccess/DaoHelper.java - Eclipse
File Edit Run Source Navigate Search Project Refactor Window Help

SupportClasses.java Magnet.java Clover Coverage Treemap Report Clover Coverage Cloud Report DaoHelper.java DataSourceSingleton.java

722 String[] result = new String[originalValues.length];
723 for (int i = 0; i < originalValues.length; i++) {
724     int n = this.annotations.indexOf(originalValues[i]);
725     if (n >= 0) {
726         result[i] = (String) this.targets.get(n);
727     }
728 }
729 return result;
730 }
731 }
732
733 /**
734  * Base class for all prepared statement callbacks which execute queries in a batch mode. Derived classes are
735  * required to compute expected batch size and return actual updated row count.
736  *
737  * @author Wojciech Sliwinski
738  */
739 public static abstract class BatchUpdateCallback implements PreparedStatementCallback {
740
741     // Define fixed batch size, Oracle recommends values in a range (5,30) -->
742     // Oracle JDBC Developers Guide and Reference
743     private static final int BATCH_SIZE = 30;
744
745     public final Object doInPreparedStatement(PreparedStatement ps) throws SQLException, DataAccessException {
746         final int dataSize = computeDataEntriesCount();
747         if (dataSize == 0) {
748             // nothing to do, return
749             return new Integer(0);
750         }
751         final OraclePreparedStatement ops = DaoHelper.convertToOraclePreparedStatement(ps);
752         ops.setExecuteBatch(BATCH_SIZE);
753         // call derived abstract method
754         int updatedRowCount = doInPreparedStatementImpl(ops);
755         // final flush of batch buffer
756         int batchRowCount = ops.sendBatch();
757         if (dataSize != updatedRowCount + batchRowCount) {
758             if (LOGGER.isInfoEnabled()) {
759                 LOGGER.info("Computed data_entries_size [" + dataSize + "] is different from updated_row_count ["
760                     + (updatedRowCount + batchRowCount) + "]);
761             }
762         }
763         return new Integer(updatedRowCount + batchRowCount);
764     }
765
766     protected abstract int computeDataEntriesCount();
767
768     protected abstract int doInPreparedStatementImpl(OraclePreparedStatement ops) throws SQLException,
769         DataAccessException;
770 }
771 }

```

Java - isa-dbaccess/src/java/cern/isa/dbaccess/DaoHelper.java - Eclipse

File Edit Run Source Navigate Search Project Refactor Window Help

SupportClasses.java Magnet.java Clover Coverage Treemap Report Clover Coverage Cloud Report DaoHelper.java DataSourceSingleton.java

```

722 String[] result = new String[originalValues.length];
723 for (int i = 0; i < originalValues.length; i++) {
724     int n = this.annotations.indexOf(originalValues[i]);
725     if (n >= 0) {
726         result[i] = (String) this.targets.get(n);
727     }
728 }
729 return result;
730 }
731 }
732
733 /**
734  * Base class for all prepared statement callbacks which execute queries in a batch mode. Derived classes are
735  * required to compute expected batch size and return actual updated row count.
736  *
737  * @author Wojciech Sliwinski
738  */
739 public static abstract class BatchUpdateCallback implements PreparedStatementCallback {
740
741     // Define fixed batch size, Oracle recommends values in a range {5,30} -->
742     // Oracle JDBC Developers Guide and Reference
743     private static final int BATCH_SIZE = 30;
744
745     public final Object doInPreparedStatement(PreparedStatement ps) throws SQLException, DataAccessException {
746         final int dataSize = computeDataEntriesCount();
747         if (dataSize == 0) {
748             // nothing to do, return
749             return new Integer(0);
750         }
751         final OraclePreparedStatement ops = DaoHelper.convertToOraclePreparedStatement(ps);
752         ops.setExecuteBatch(BATCH_SIZE);
753         // call derived abstract method
754         int updatedRowCount = doInPreparedStatementImpl(ops);
755         // final flush of batch buffer
756         int batchRowCount = ops.sendBatch();
757         if (dataSize != updatedRowCount + batchRowCount) {
758             if (LOGGER.isInfoEnabled()) {
759                 LOGGER.info("Computed data_entries_size [" + dataSize + "] is different from updated_row_count ["
760                     + (updatedRowCount + batchRowCount) + "]);
761             }
762         }
763         return new Integer(updatedRowCount + batchRowCount);
764     }
765
766     protected abstract int computeDataEntriesCount();
767
768     protected abstract int doInPreparedStatementImpl(OraclePreparedStatement ops) throws SQLException,
769         DataAccessException;
770 }
771 }

```

Red = not covered

Green = covered

Writable Smart Insert 770 : 6

accosoft-commons-util

Default Stage

Default Job

## Plan Summary

### Current Activity

No builds are currently running.

### Recent History

✔ #264	Dependant of ACCCOM-CORE-139	1 week ago	131 passed
✔ #263	Manual build by Niall Stapley	1 week ago	131 passed
❗ #262	Manual build by Niall Stapley	1 week ago	131 passed
✔ #261	Updated by Kajetan Fuchsberger	3 weeks ago	131 passed
✔ #260	Updated by Huy Jeremy Nguyen Xuan	3 weeks ago	124 passed
✔ #259	Manual build by Pavel Tarasenko	1 month ago	124 passed
❗ #258	Manual build by Pavel Tarasenko	1 month ago	No tests found
❗ #257	Manual build by Pavel Tarasenko	1 month ago	No tests found
✔ #256	Updated by rgorbono	4 months ago	124 passed
✔ #255	Updated by rgorbono	4 months ago	124 passed

↔ Latest build   ↔ Last successful build

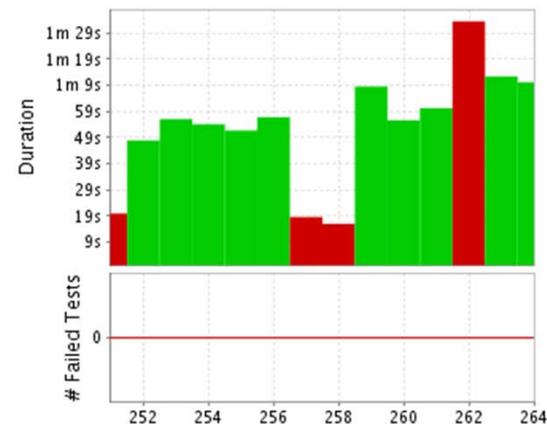
### Plan Statistics

**71%**  
Successful

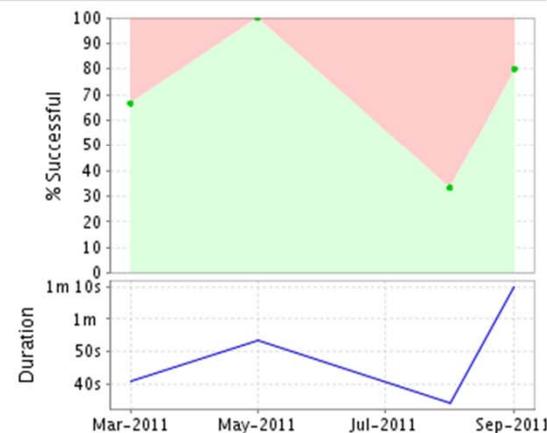
Successful Builds: 10 / 14

Average Duration: 59 seconds

### Build Duration



### Successful Builds



Group by:  MONTH

Plan Navigator

accosoft-commons-util

Default Stage

Default Job

Plan Summary Recent Failures History Issues Configuration

Showing Last 25 builds Actions

## Plan Summary

### Current Activity

No builds are currently running.

### Recent History

✔ #264	Dependant of <a href="#">ACCCOM-CORE-139</a>	1 week ago	131 passed
✔ #263	Manual build by <a href="#">Niall Stapley</a>	1 week ago	
❗ #262	Manual build by <a href="#">Niall Stapley</a>		
✔ #261	Updated by <a href="#">Kajetan Fuchsberger</a>		
✔ #260	Updated by <a href="#">Huy Jeremy Nguyen Xuan</a>		
✔ #259	Manual build by <a href="#">Pavel Tarasenko</a>	1 month ago	
❗ #258	Manual build by <a href="#">Pavel Tarasenko</a>	1 month ago	No tests found
❗ #257	Manual build by <a href="#">Pavel Tarasenko</a>	1 month ago	No tests found
✔ #256	Updated by <a href="#">rgorbono</a>	4 months ago	124 passed
✔ #255	Updated by <a href="#">rgorbono</a>	4 months ago	124 passed

↔ Latest build ↔ Last successful build

Triggered by changes in a dependency

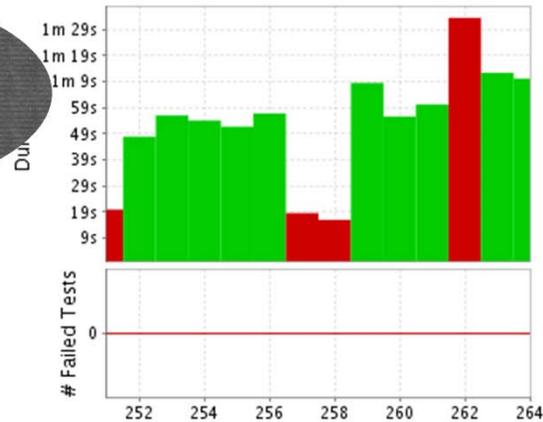
### Plan Statistics

**71%**  
Successful

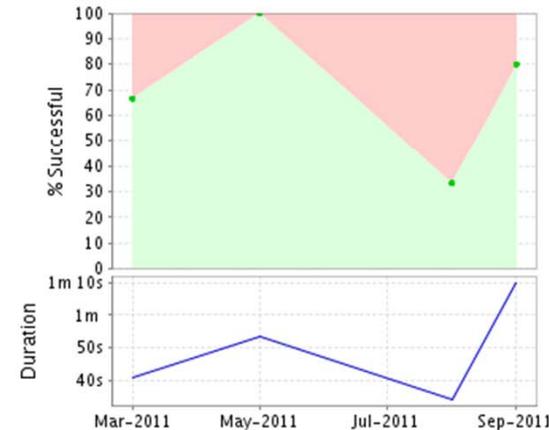
Successful Builds: 10 / 14

Average Duration: 59 seconds

### Build Duration



### Successful Builds



Group by:  MONTH

accsoft-commons-util

Default Stage

Default Job

## Plan Summary

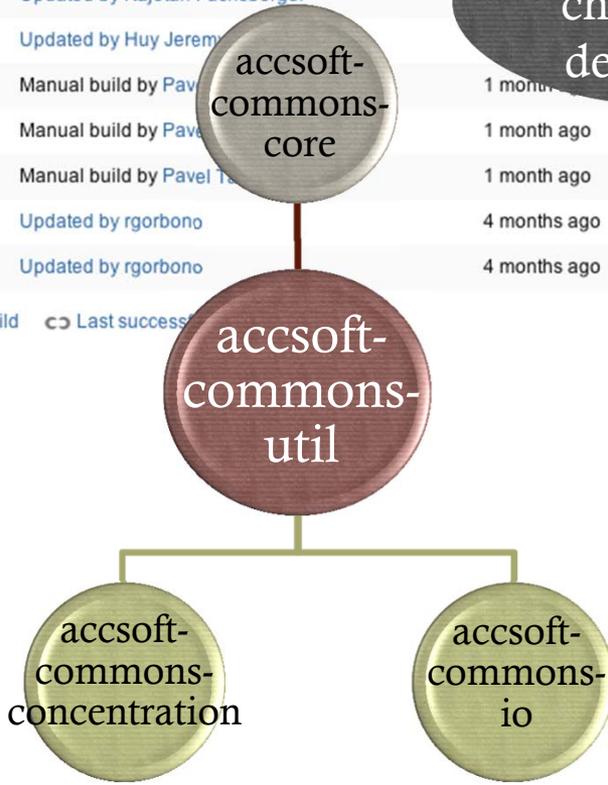
### Current Activity

No builds are currently running.

### Recent History

✓ #264	Dependant of ACCCOM-CORE-139	1 week ago	131 passed
✓ #263	Manual build by Niall Stapley	1 week ago	
✗ #262	Manual build by Niall Stapley		
✓ #261	Updated by Kajetan Fuchsberger		
✓ #260	Updated by Huy Jeremy		
✓ #259	Manual build by Pavel T	1 month ago	
✗ #258	Manual build by Pavel T	1 month ago	No tests found
✗ #257	Manual build by Pavel T	1 month ago	No tests found
✓ #256	Updated by rgorbano	4 months ago	124 passed
✓ #255	Updated by rgorbano	4 months ago	124 passed

Latest build Last success



Triggered by changes in a dependency

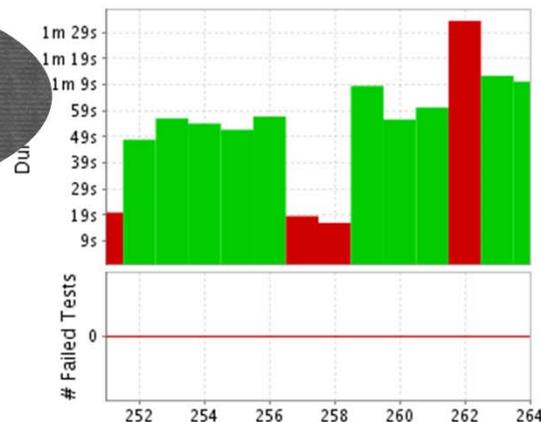
### Plan Statistics

**71%**  
Successful

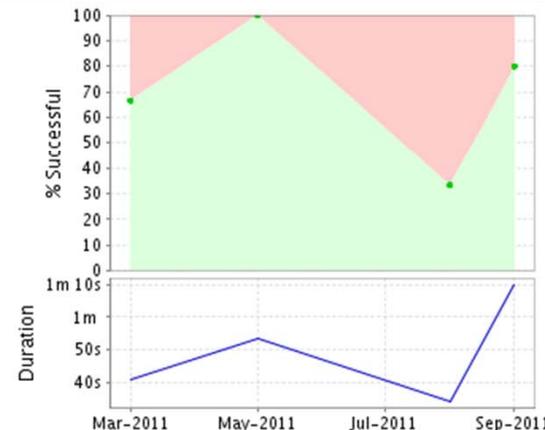
Successful Builds: 10 / 14

Average Duration: 59 seconds

### Build Duration



### Successful Builds



Group by: AUTO MONTH

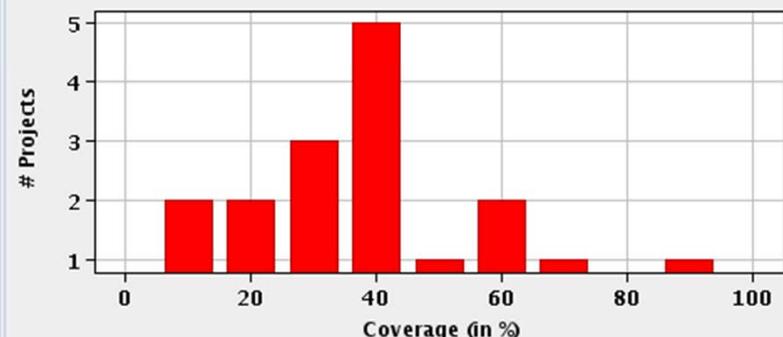
Projects :	17	LoC :	153024
Stmts :	33228	NCLoC :	79227
Methods :	8343	Files :	1298
Classes :	1499	Packages :	260

Total cmp :	16197	Stmts/Method :	3.98
Cmp density :	0.49	Method/Class :	5.57
Avg method cmp...	5.57	Class/Package :	5.77

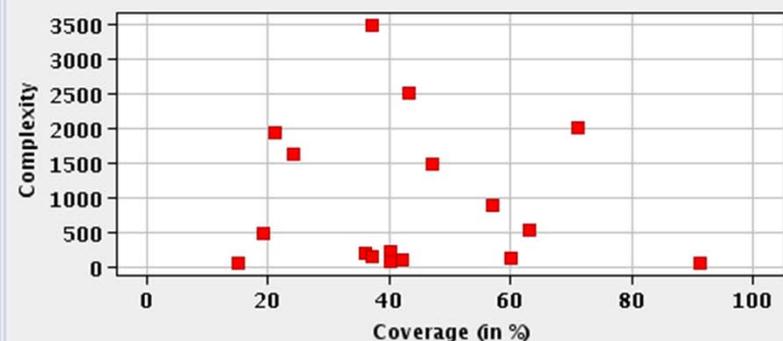
Coverage (17 projects, 22968/54079 elements)

42%

## Project Coverage Distribution



## Project Complexity



## Top 15 Complex Projects (ordered by complexity)

1.	37.7%	japc
2.	43.3%	japc-value
3.	71.7%	pm-core
4.	21%	inca-server-core
5.	24.7%	accsoft-commons-util
6.	47.9%	japc-ext-remote
7.	57.4%	japc-ext-cmwrda
8.	63.4%	pm-domain
9.	20%	inca-server-valcomp
10.	40.8%	accsoft-commons-cache
11.	36.9%	pm-db-journals
12.	37.7%	inca-common-utils
13.	60.2%	pm-server

## Top 10 Risk Projects (high complexity, low coverage)

pm-converters  
 pm-powering-server  
 pm-server  
 pm-viewers  
 inca-common-utils  
 japc-directory-client  
 accsoft-commons-cache  
 pm-db-journals  
 pm-domain  
 japc-ext-cmwrda

## 15 Best Covered Projects

91.3%	pm-converters
71.7%	pm-core
63.4%	pm-domain
60.2%	pm-server
57.4%	japc-ext-cmwrda
47.9%	japc-ext-remote
43.3%	japc-value
43%	pm-viewers
40.8%	accsoft-commons-cache
40.2%	pm-powering-server
37.7%	japc
37.7%	inca-common-utils

## 15 Least Covered Projects

15.1%	japc-directory-client
20%	inca-server-valcomp
21%	inca-server-core
24.7%	accsoft-commons-util
36.9%	pm-db-journals
37.7%	inca-common-utils
37.7%	japc
40.2%	pm-powering-server
40.8%	accsoft-commons-cache
43%	pm-viewers
43.3%	japc-value
47.9%	japc-ext-remote

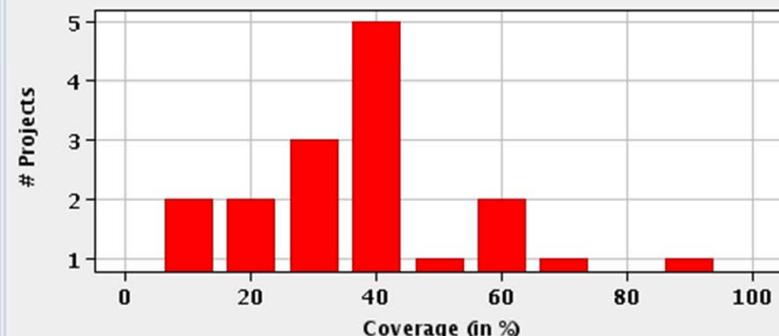
Projects : 17 LoC : 3.98  
 Stmts : 33228 NCLoC : 5.57  
 Methods : 8343 Files :  
 Classes : 1499 Packages : 200 Package : 5.77

Top/flop list generated  
 (Work in progress)

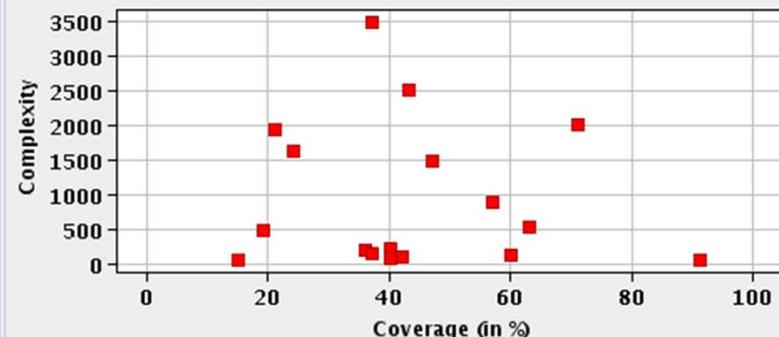
Coverage (17 projects, 22968/54079 elements)

42%

Project Coverage Distribution



Project Complexity



Top 15 Complex Projects (ordered by complexity)

Rank	Coverage	Project Name
1.	37.7%	japc
2.	43.3%	japc-value
3.	71.7%	pm-core
4.	21%	inca-server-core
5.	24.7%	accsoft-commons-util
6.	47.9%	japc-ext-remote
7.	57.4%	japc-ext-cmwrda
8.	63.4%	pm-domain
9.	20%	inca-server-valcomp
10.	40.8%	accsoft-commons-cache
11.	36.9%	pm-db-journals
12.	37.7%	inca-common-utils
13.	60.2%	pm-server

Top 10 (high complexity, low coverage)

pm-converters  
 pm-powering-server  
 pm-server  
 pm-viewers  
 inca-common-utils  
 japc-directory-client  
 accsoft-commons-cache  
 pm-db-journals  
 pm-domain  
 japc-ext-cmwrda

15 Best Covered Projects

Coverage	Project Name
91.3%	pm-converters
71.7%	pm-core
63.4%	pm-domain
60.2%	pm-server
57.4%	japc-ext-cmwrda
47.9%	japc-ext-remote
43.3%	japc-value
43%	pm-viewers
40.8%	accsoft-commons-cache
40.2%	pm-powering-server
37.7%	japc
37.7%	inca-common-utils

15 Least Covered Projects

Coverage	Project Name
15.1%	japc-directory-client
20%	inca-server-valcomp
21%	inca-server-core
24.7%	accsoft-commons-util
36.9%	pm-db-journals
37.7%	inca-common-utils
37.7%	japc
40.2%	pm-powering-server
40.8%	accsoft-commons-cache
43%	pm-viewers
43.3%	japc-value
47.9%	japc-ext-remote

Projects :	17	LoC :	3.98
Stmts :	33228	NCLoC :	5.57
Methods :	8343	Files :	Package : 5.77
Classes :	1499	Packages :	200

Top/flop list generated  
(Work in progress)

Coverage (17 projects, 22968/54079 elements)  
42%

Top 10 (high complexity, low coverage)

pm-converters  
pm-powering-server  
pm-server

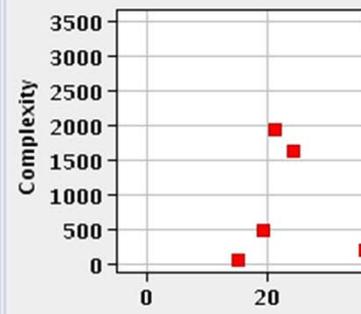
Project Coverage Distribution



15 Least Covered Projects

- 15.1% japc-directory-client
- 20% inca-server-valcomp
- 21% inca-server-core
- 24.7% accsoft-commons-util
- 36.9% pm-db-journals
- 37.7% inca-common-utils
- 37.7% japc
- 40.2% pm-powering-server
- 40.8% accsoft-commons-cache
- 43% pm-viewers
- 43.3% japc-value
- 47.9% japc-ext-remote

Project Complexity



Top 15 Complex Projects (ordered)

Rank	Coverage	Project Name
1.	37.7%	japc
2.	43.3%	japc-val
3.	71.7%	pm-core
4.	21%	inca-server-core
5.	24.7%	accsoft-commons-util
6.	47.9%	japc-ext-remote
7.	57.4%	japc-ext-cmwrda
8.	63.4%	pm-domain
9.	20%	inca-server-valcomp
10.	40.8%	accsoft-commons-cache
11.	36.9%	pm-db-journals
12.	37.7%	inca-common-utils
13.	60.2%	pm-server

20%	inca-server-valcomp
21%	inca-server-core
24.7%	accsoft-commons-util
36.9%	pm-db-journals
37.7%	inca-common-utils
37.7%	japc
40.2%	pm-powering-server
40.8%	accsoft-commons-cache
43%	pm-viewers
43.3%	japc-value
47.9%	japc-ext-remote

300

Actions

0/14

seconds

262 264

Sep-2011

Projects :	17	LoC :	3.98
Stmts :	33228	NCLoC :	5.57
Methods :	8343	Files :	Package : 5.77
Classes :	1499	Packages :	200

Top/flop list generated  
(Work in progress)

Coverage (17 projects, 22968/54079 elements)



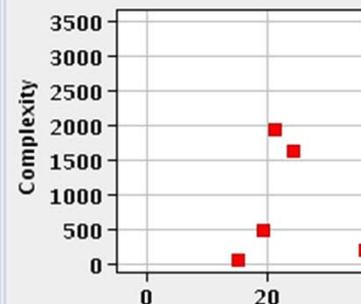
Project Coverage Distribution



15 Best Covered Projects

- 91.3% pm-converters
- 71.7% pm-core
- 63.4% pm-domain
- 60.2% pm-server
- 57.4% japc-ext-cmwrda
- 47.9% japc-ext-remote
- 43.3% japc-value
- 43% pm-viewers
- 40.8% accsoft-commons-cache
- 40.2% pm-powering-server
- 37.7% japc
- 37.7% inca-common-utils

Project Complexity



Top 15 Complex Projects (ordered)

1.	37.7%	japc
2.	43.3%	japc-val
3.	71.7%	pm-core
4.	21%	inca-server-core
5.	24.7%	accsoft-commons-util
6.	47.9%	japc-ext-remote
7.	57.4%	japc-ext-cmwrda
8.	63.4%	pm-domain
9.	20%	inca-server-valcomp
10.	40.8%	accsoft-commons-cache
11.	36.9%	pm-db-journals
12.	37.7%	inca-common-utils
13.	60.2%	pm-server

Top 10 (high complexity, low coverage)

- pm-converters
- pm-powering-server
- pm-server

21%	inca-server-core
24.7%	accsoft-commons-util
36.9%	pm-db-journals
37.7%	inca-common-utils
37.7%	japc
40.2%	pm-powering-server
40.8%	accsoft-commons-cache
43%	pm-viewers
43.3%	japc-value
47.9%	japc-ext-remote

300

Actions

0/14

seconds

262 264

Sep-2011

# Challenges



# Agree on standards and configurations



- ❧ Think of and organize us as one big team, not many small ones
- ❧ The code belongs to the group, not to a project or an individual developer
- ❧ The guidelines and standards established and agreed by everyone

# Encourage developers to invest time on quality



- ❧ Quality-related tasks part of the yearly objectives for a project
- ❧ Progress measured against the level a project adheres to guidelines – top/flop lists
- ❧ Focus the effort where the effect is highest, rely on tools as much as possible
- ❧ SIP days – common days with a common goal

# Summary



- ❧ SIP an initiative to integrate quality improvement in the development cycle
- ❧ A number of guidelines/standards and supporting tools have been agreed
- ❧ Developer motivation through
  - ❧ Immediate feedback when developing
  - ❧ Official tracking of progress and top/flop lists
- ❧ Increased awareness of benefits, application in individual projects and confidence when introducing changes