Thoughts on Build Management

Siegfried Goeschl

“Do we build card houses or software?!”

Software Build Evolution

- Manual compile on a developer box
- Manual compile on a shared build box
- Automated daily compile
- Automated daily installable build
- Automated test of an installed build

Industry Best Practice

- Automated build
- Repository Manager
- Continuous Integration
- Software Quality Management

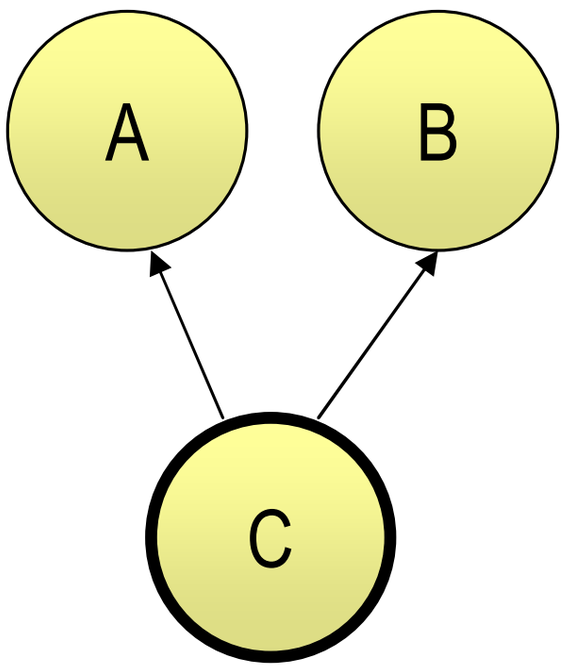
Automated Build

- Ant and Maven as de facto standard for Java
- Maven embeds build metadata in artifacts
  - Version information, dependencies
- Artifacts can be signed during release
  - Mandatory for ASF releases
- Artifacts have unique coordinates
  - o.a.commons | commons-email | 1.3 | jar

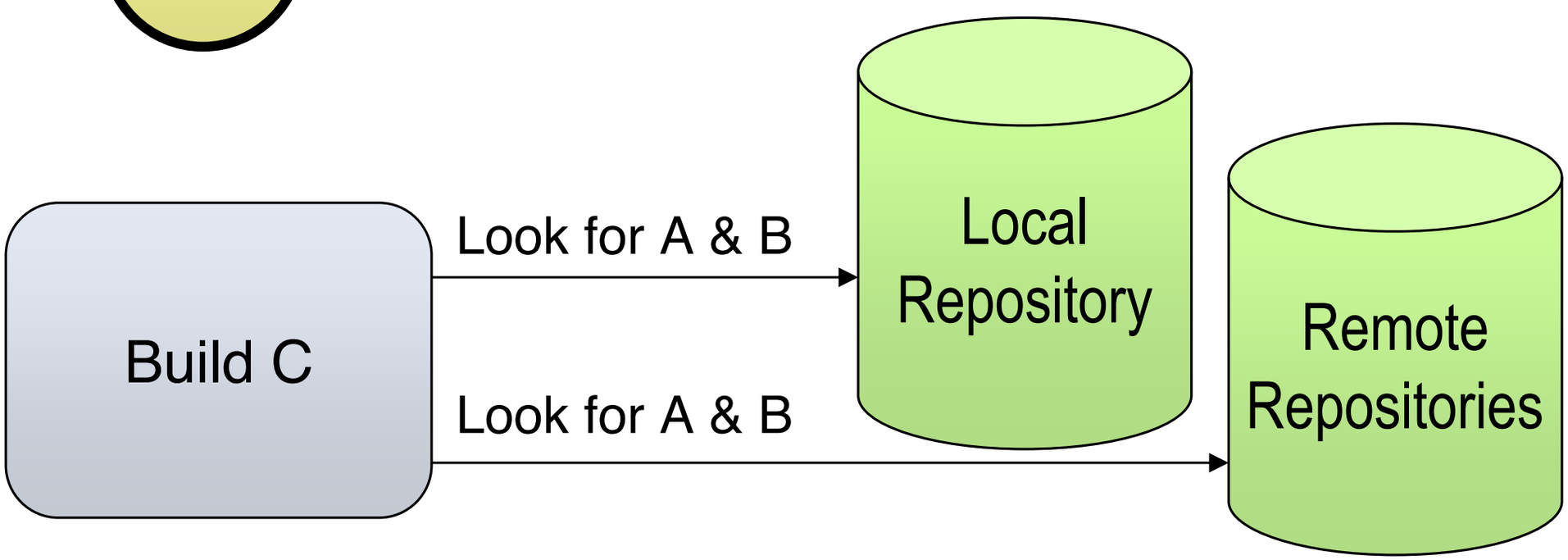
Repository Manager

- Stores third-party and in-house libraries
- “Write Once Read Many” access
- Allows access control and auditing
- Integrates nicely with Maven (and Ant?!)
- Interface between legacy and Maven builds

Repository Manager




```
<dependencies>
  <dependency>
    <groupId>com.acme</groupId>
    <artifactId>B</artifactId>
    <version>2.1</version>
    <scope>compile</scope>
  </dependency>
</dependencies>
```



Continuous Integration

"Continuous Integration is a software development practice where members of a team integrate their work frequently, usually each person integrates at least daily - leading to multiple integrations per day. Each integration is verified by an automated build (including test) to detect integration errors as quickly as possible"

Hudson

Hudson

ENABLE AUTO REFRESH

New Job

Configure

Reload Config

Build Queue

hudson	
jaxb-ri	

Build Executor Status

No.	Status	
1	Idle	
2	Idle	
3	Building javanet-maven-repository-daemon #826	
4	Building jaxb-ri #3181	
5	Building glassfish #105	
6	Idle	

All | JAX-WS | JAXB | Tango | java.net | +

Job	Last Success	Last Failure	Last Duration	
Common annotations	4 days (#16)	9 months (#3)	39 seconds	
bsh	6 months (#11)	10 months (#2)	59 seconds	
dtd-parser	6 months (#8)	N/A	1 minute	
fi	28 days (#586)	1 month (#567)	7 minutes	
fi (weekly)	6 days (#53)	13 days (#52)	5 minutes	
glassfish	4 hours (#104)	1 day (#88)	1 hour	
hudson	4 minutes (#201)	N/A	1 minute	
istack-commons	12 days (#19)	16 days (#5)	14 seconds	
japex	3 days (#55)	9 hours (#64)	1 minute	
java-ws-xml community discussion updater	4 minutes (#16146)	10 hours (#16125)	1 minute	
java.net acl processor	18 hours (#162)	N/A	0 seconds	

Internet

Software Quality Management

- Build tools have extensions to measure software quality data for a project/module
- This data reflects a “tiny snapshot in time”
- We need historical software quality data

What is Sonar?

- An open-source software quality platform
- Uses various static source analysis tools
- Software quality metrics stored in database
- Web-based front-end

Home | Search | CUSI

- Dashboard
- Components
- Violations drilldown
- Time machine
- Clouds
- Design
- Hotspots
- Libraries

sonar

Version iter-27 - Tue, 22 Jun 2010 00:01 - profile Sonar way

Lines of code
133,983
174,175 lines
52,147 statements
1,747 files

Classes
1,955
173 packages
12,111 methods
+887 accessors

Rules compliance
87.7%

Violations
11,127

Blocker	0
Critical	33
Major	2,633
Minor	8,401
Info	60

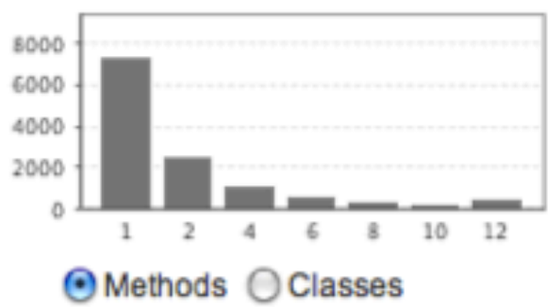
Comments
3.1%
4,292 lines
3.3% docu. API
7,198 undocu. API
443 commented LOCs

Duplications
6.0%
10,504 lines
292 blocks
140 files

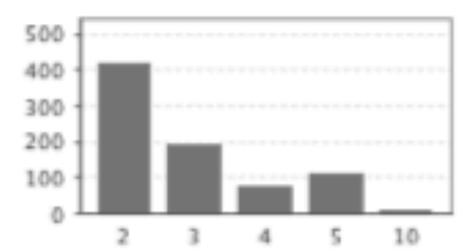
Package tangle index
5.7%
> 466 cycles

Dependencies to cut
89 between packages
201 between files

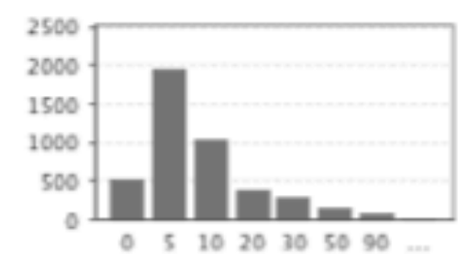
Complexity
2.2 / method
13.4 / class
15.0 / file
Total: 26,189



LCOM4
1.9 /class
38.8% files having LCOM4>1



RFC
25 /class



Code coverage
-
4,368 tests
1:18 min

Test success
100.0%
0 failures
0 errors

Events All

2010-06-22	Version	iter-27
2010-05-11	Version	iter-26

Conclusion

- No more card houses
- Tooling is open-source and mature
- Climbing up the “Software Build Evolution Ladder” is mostly an organizational problem

About me

- Independent Consultant
  - Server-side Java Development
  - Build Management
  - Software Quality Assurance
  - Performance Testing
- Doing a lot of open source development