


– Schneller, Billiger, Besser –

Effektiver Einsatz von Code-Reviews

Orientation in Objects GmbH
Weinheimer Str. 68
68309 Mannheim
www.oio.de
info@oio.de


Version: 1.0

Ihr Sprecher



Thorsten Maier

Trainer, Berater, Entwickler




Schwerpunkte
Prozesse
Architektur
Code-Qualität

© 2013 Orientation in Objects GmbH

Effektiver Einsatz von Code-Reviews | 2


Agenda

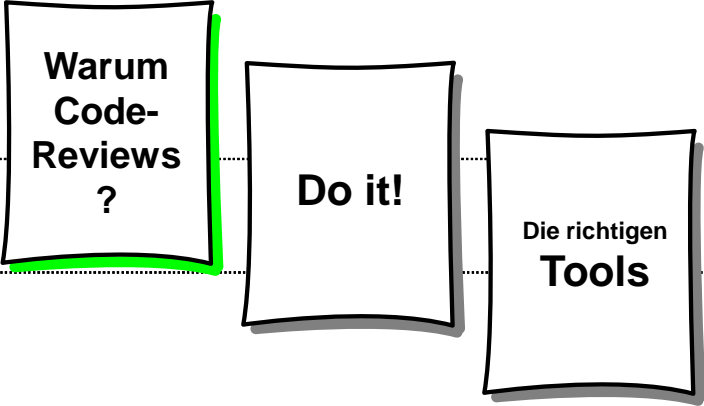


Einsteiger	<i>Code-Reviews? Noch nie gehört!</i>
Skeptiker	<i>Wir haben es ausprobiert. Es funktioniert nicht!</i>
Profis	<i>Wir setzen Code-Reviews bereits erfolgreich ein und wollen noch besser werden.</i>

© 2013 Orientation in Objects GmbH | Effektiver Einsatz von Code-Reviews | 3

Agenda



Einsteiger	
Skeptiker	
Profis	

© 2013 Orientation in Objects GmbH | Effektiver Einsatz von Code-Reviews | 4

Kennen Sie das?

„Das ist historisch gewachsen.“

Oder das?

„Das wollten wir schon lange
refactoren.“

...noch ein Klassiker


„Geh zu Thomas,
der kennt die Stelle.“

Es bleibt nie Zeit für das große
Refactoring

Code-Reviews

als kontinuierliche Lösung

Reviews nach IEEE 1028



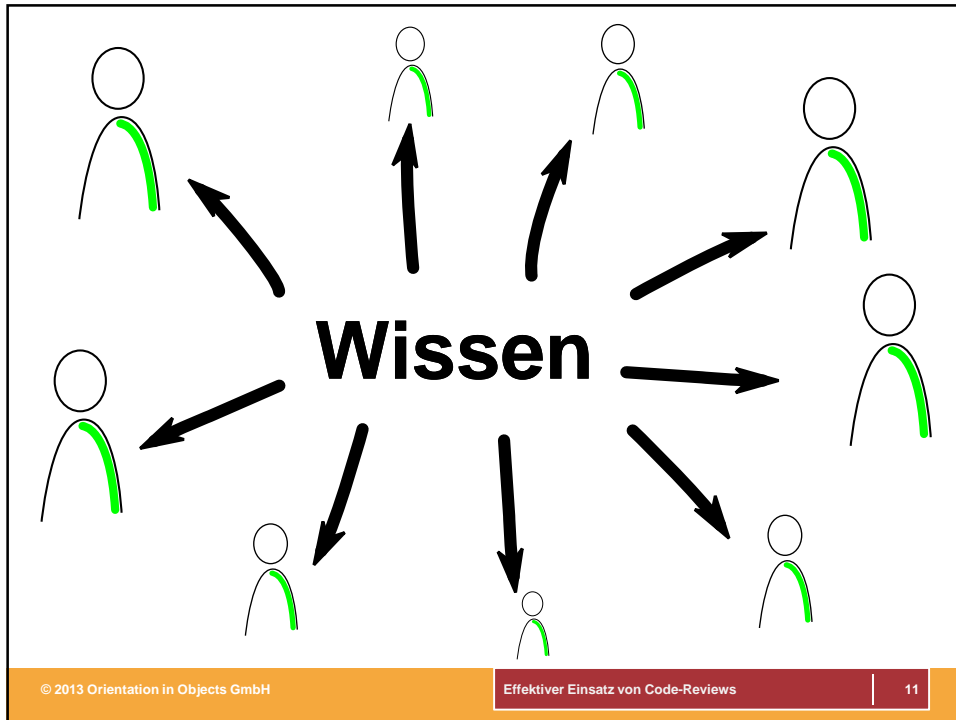
Characteristic	Management review	Technical Review	Inspection	Walk-through	Audit
Objective	Ensure progress; recommend corrective action; ensure proper allocation of resources	Evaluate conformance to specifications and plans; ensure change integrity	Find anomalies; verify resolution; verify product quality	Find anomalies; examine alternatives; improve product; forum for learning	Independently evaluate compliance with objective standards and regulations

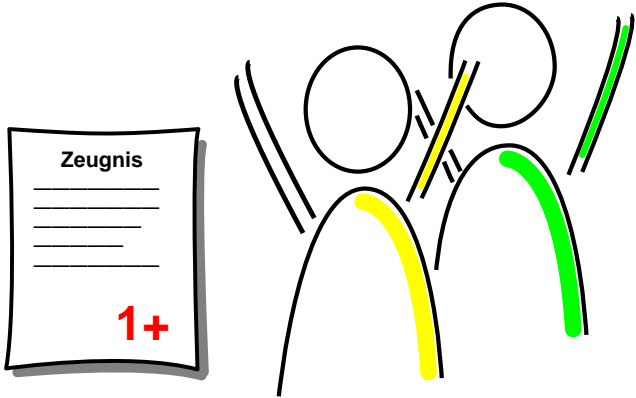
© 2013 Orientation in Objects GmbH | Effektiver Einsatz von Code-Reviews | 9



Bugs finden

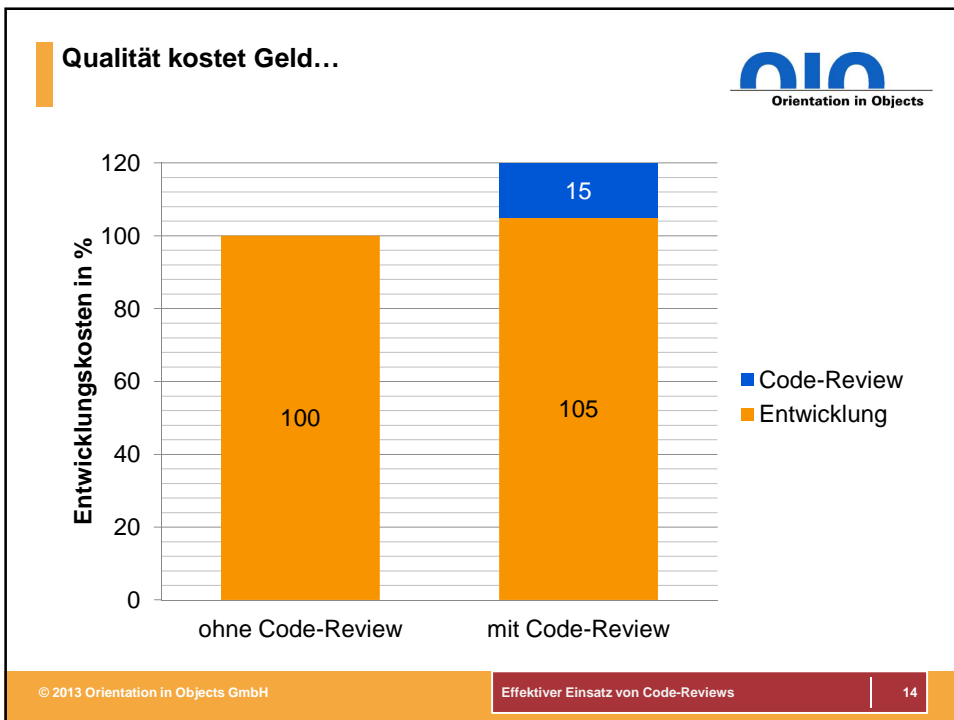
© 2013 Orientation in Objects GmbH | Effektiver Einsatz von Code-Reviews | 10

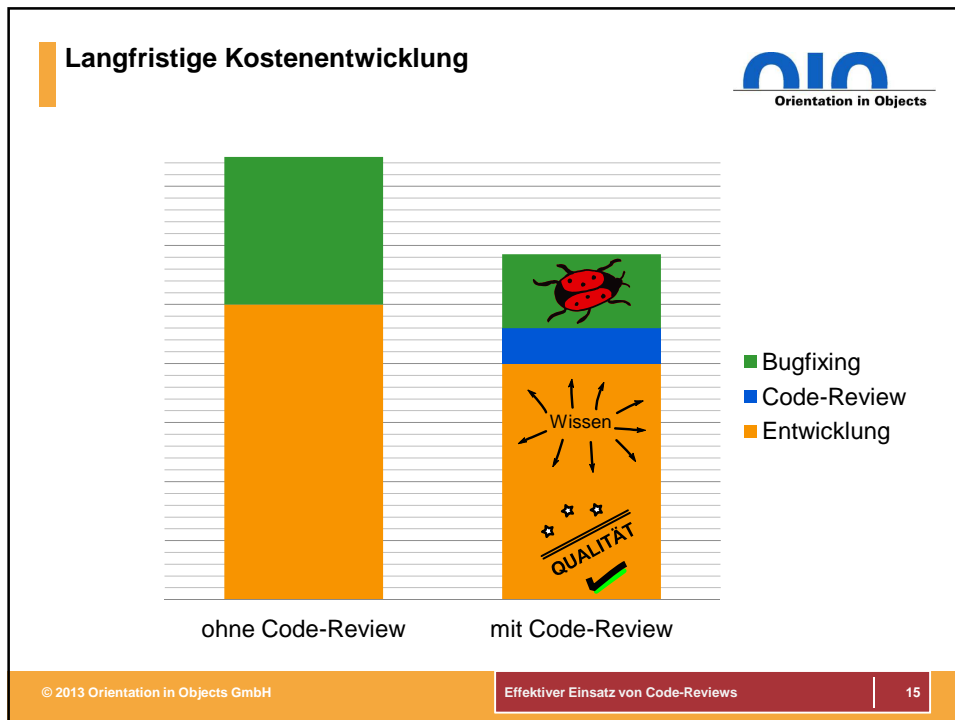




Anerkennung

© 2013 Orientation in Objects GmbH | Effektiver Einsatz von Code-Reviews | 13





Code-Reviews für alle?

Wartungsprojekte

© 2013 Orientation in Objects GmbH Effektiver Einsatz von Code-Reviews 16

Agenda

Orientation in Objects

Einsteiger	Warum Code-Reviews ?	Do it!	Die richtigen Tools
Skeptiker			
Profis			

© 2013 Orientation in Objects GmbH | Effektiver Einsatz von Code-Reviews | 17

Wann?

Orientation in Objects

develop

feature

feature-x

feature-z

Code-Review

Code-Review

„Nach jedem Feature / Bugfix“

© 2013 Orientation in Objects GmbH | Effektiver Einsatz von Code-Reviews | 18

Wer?

Orientation in Objects

Einstiegspunkte
Hinweise
Erläuterungen

Code-Autor

Reviewer

Klickt
Prüft
Hinterfragt

© 2013 Orientation in Objects GmbH | Effektiver Einsatz von Code-Reviews | 19

Ein gemeinsames Ziel

Orientation in Objects

„Wir wollen unsere Anwendung **gemeinsam** besser machen“

© 2013 Orientation in Objects GmbH | Effektiver Einsatz von Code-Reviews | 20

Positive Stimmung



*Kritik an der Sache, nicht an der Person
Loben, loben, loben...*

© 2013 Orientation in Objects GmbH | Effektiver Einsatz von Code-Reviews | 21

Stelle Fragen



Was war der Grund für dein hier gewähltes Vorgehen?

Das ist falsch

© 2013 Orientation in Objects GmbH | Effektiver Einsatz von Code-Reviews | 22

Nicht zu viel ... und nicht zu lange

nio
Orientation in Objects

200 – 400 Zeilen

1 – 2 Stunden

<http://support.smartbear.com/resources/ci/book/code-review-cisco-case-study.pdf>

© 2013 Orientation in Objects GmbH | Effektiver Einsatz von Code-Reviews | 23

Do it!!!

nio
Orientation in Objects

Aufgaben aus Code-Review "Neue Benutzerverwaltung" #34

- Potentielle NullPointerException im UserService abfangen
- Typen in der Benutzer-Dokumentation entfernen
- Unit-Test "UserServiceTest" um den neuen Anwendungsfall erweitern
- Neues Widget für die Eingabe des Benutzernamens in Bibliothek auslagern
- Type note...

🔔 | 🗑️ | 📷 | 📄 | 🗑️
17:18

© 2013 Orientation in Objects GmbH | Effektiver Einsatz von Code-Reviews | 24

Qualität kostet Geld... und deshalb messen!

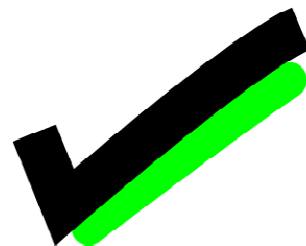


Story-Link	Autor	Reviewer	Zeit Impl	Zeit Review	Nachbearbeitung	~LOC	Anzahl Verbesserungen	CR war sinnvoll	Anzahl Bugs
EJFYLOLAINTERN-5010	SOS	TM	8h	1h	0h	100	0	ja	0
EJFYLOLAINTERN-5025	RK	TM	5h	1h	0h	50	1	ja	1
EJFYLOLAINTERN-5122	TM	FS	2h	30m	0h	10	0	ja	0
EJFYLOLAINTERN-4764	RK	TM	3h	1h	1h	20	1	ja	0
EJFYLOLAINTERN-5094	TM	SOS	4h 45m	30m	0h	50	0	ja	0
EJFYLOLAINTERN-5178	SOS	TM	3d5h	1h 30m	30m	300	1	ja	0
EJFYLOLAINTERN-5212	RK	FS	2h 45m	1h 15m	1h 15m	15	1	ja	1
EJFYLOLAINTERN-5174	SOS	TM	3h 45m	1h	15m	100	1	ja	0
EJFYLOLAINTERN-5227	TM	SOS	2h	30m	45m	30	1	ja	0
EJFYLOLAINTERN-5253	TM	SOS	1h	30m	0	20	0	ja	0
EJFYLOLAINTERN-5245	TM	SOS	1h 15m	30m	0	20	0	nein	0


Checkliste



- ✓ Sinnvolle Namen?
- ✓ Klare Struktur?
- ✓ Korrekte Verwendung von Datentypen?
- ✓ Kommentare?
- ✓ Code smells?
- ✓ Tests vorhanden?
- ✓ Wohldefinierte Schnittstellen?
- ✓ Exception-Handling?
- ✓ I18N?
- ✓ Typische Programmier-Fehler?



Guidelines




- Sun Java Code Conventions
- Effective Java
- Clean Code
- Tool-Standard-Konfiguration
 - Eclipse: Errors, Warnings und Formatter
 - Findbugs
 - PMD
 - Checkstyle

→ Objektivierung der Code-Reviews
→ Standard für Projekte

© 2013 Orientation in Objects GmbH | Effektiver Einsatz von Code-Reviews | 27

Agenda



Einsteiger	<div style="border: 1px solid black; padding: 10px; width: fit-content; margin: 0 auto;"> Warum Code- Reviews ? </div>	<div style="border: 1px solid black; padding: 10px; width: fit-content; margin: 0 auto;"> Do it! </div>	<div style="border: 1px solid black; padding: 10px; width: fit-content; margin: 0 auto;"> Die richtigen Tools </div>
Skeptiker			
Profis			

© 2013 Orientation in Objects GmbH | Effektiver Einsatz von Code-Reviews | 28

Versionsverwaltung



Revision	Actions	Author	Date	Message
4138		oio_maier	15:12:46, Dienstag, 22. Oktober 2013	KPSADAPTER-532 Test-Klassen für Lasttest in Trunk übernommen
4136		oio_maier	12:29:01, Dienstag, 22. Oktober 2013	KPSADAPTER-532 Cluster-Modus für Infrispas-Cluster angepasst
4134		bbarth	10:20:15, Dienstag, 22. Oktober 2013	KPSADAPTER-532 Fallover-Szenario - Fehlermeldungen wenn nicht mit dem System
4132		bbarth	17:18:25, Montag, 21. Oktober 2013	KPSADAPTER-532 Fallover-Szenario - Anpassung
4131		oio_maier	16:26:47, Montag, 21. Oktober 2013	KPSADAPTER-532 KPS-Sitzungen im Infrispas-Cache speichern - Erweiterung - Serial
4130		bbarth	14:38:53, Montag, 21. Oktober 2013	KPSADAPTER-532 Fallover-Szenario - Anpassung
4129		bbarth	14:21:53, Montag, 21. Oktober 2013	KPSADAPTER-532 Fallover-Szenario - Anpassung
4127		swinkler	13:18:55, Montag, 21. Oktober 2013	KPSADAPTER-1472: Entfernen der Validerung für Verwendung von freien Geräteset
4126		swinkler	13:18:24, Montag, 21. Oktober 2013	KPSADAPTER-1475: Formatierung der Position unterhalb des Kopfsatzes
4124		bbarth	15:59:52, Freitag, 18. Oktober 2013	KPSADAPTER-1471 Fallover-Knoten-Flag in KPSAdapterPrincipal hinterlegt
4123		bbarth	15:45:51, Freitag, 18. Oktober 2013	KPSADAPTER-1471 Fallover-Knoten-Flag in KPSAdapterPrincipal hinterlegt
4122		bbarth	15:33:29, Freitag, 18. Oktober 2013	KPSADAPTER-1471 Fallover-Knoten-Flag in KPSAdapterPrincipal hinterlegt
4121		bbarth	15:23:41, Freitag, 18. Oktober 2013	KPSADAPTER-1471 Fallover-Knoten-Flag in KPSAdapterPrincipal hinterlegt
4120		swinkler	13:44:09, Freitag, 18. Oktober 2013	KPSADAPTER-1471: Domain Objekte serialisierbar
4119		bbarth	17:51:04, Donnerstag, 17. Oktober 2013	KPSADAPTER-532 Default Cluster-Konfiguration angepasst
4118		bbarth	17:43:17, Donnerstag, 17. Oktober 2013	KPSADAPTER-532 Fallover-Szenario - Fehlermeldungen wenn nicht mit dem System
4117		bbarth	14:02:11, Donnerstag, 17. Oktober 2013	KPSADAPTER-532 Bugfix - Einblendung eines Hinweises bei Verwenden des Ersatzsys
4116		bbarth	11:02:46, Donnerstag, 17. Oktober 2013	KPSADAPTER-1399 Katalogaktualisierung angepasst - Beim Starten werden nun alle u

Diff-Tool




```

D:\tmp\GreetingService.java
package de.oio.sample.gwt.client;
import com.google.gwt.user.client.rpc.RemoteService;
import com.google.gwt.user.client.rpc.RemoteServiceRelati
/**
 * The client side stub for the RPC service.
 */
@RemoteServiceRelativePath("greet")
public interface GreetingService extends RemoteService {
    String greetServer(String name) throws IllegalArgumentException;
}

D:\tmp\GreetingService_new.java
package de.oio.sample.gwt.client;
import com.google.gwt.user.client.rpc.RemoteService;
import com.google.gwt.user.client.rpc.RemoteServiceRelati
/**
 * The client side stub for the RPC service.
 */
@RemoteServiceRelativePath("greet")
public interface GreetingService extends RemoteService {
    String greetServer(String name, String email) throws I
}
    
```

Wünsche an ein Code-Review-Tool



Orientation in Objects

- Compile erfolgreich
- 356 Tests erfolgreich
- Checkstyle erfolgreich

```

D:\tmp\GreetingService.java
package de.oio.sample.gwt.client;

import com.google.gwt.user.client.rpc.RemoteService;
import com.google.gwt.user.client.rpc.RemoteServiceRelati

/**
 * The client side stub for the RPC service.
 */
@RemoteServiceRelativePath("greet")
public interface GreetingService extends RemoteService {
    String greetServer(String name) throws IllegalArgumentException;
}
                
```

```

D:\tmp\GreetingService_new.java
package de.oio.sample.gwt.client;

import com.google.gwt.user.client.rpc.RemoteService;
import com.google.gwt.user.client.rpc.RemoteServiceRelati

/**
 * The client side stub for the RPC service.
 */
@RemoteServiceRelativePath("greet")
public interface GreetingService extends RemoteService {
    String greetServer(String name, String email) throws I
}
                
```

Neue Aufgabe erstellen


Änderung auf Abnahme-System bereitstellen

© 2013 Orientation in Objects GmbH

Effektiver Einsatz von Code-Reviews

31

Zusammenspiel der Tools



Orientation in Objects

```

graph TD
    SCM[SCM] -- "1. Änderung" --> CI[CI-Server]
    CI -- "2. Build erfolgreich" --> IT[Issue-Tracker]
    CI -- "3. Review erstellen" --> CRT[Code-Review-Tool]
    CRT -- "4a. Neue Aufgabe erstellen" --> IT
    CRT -- "4b. Änderung auf Abnahme-System bereitstellen" --> CI
    
```

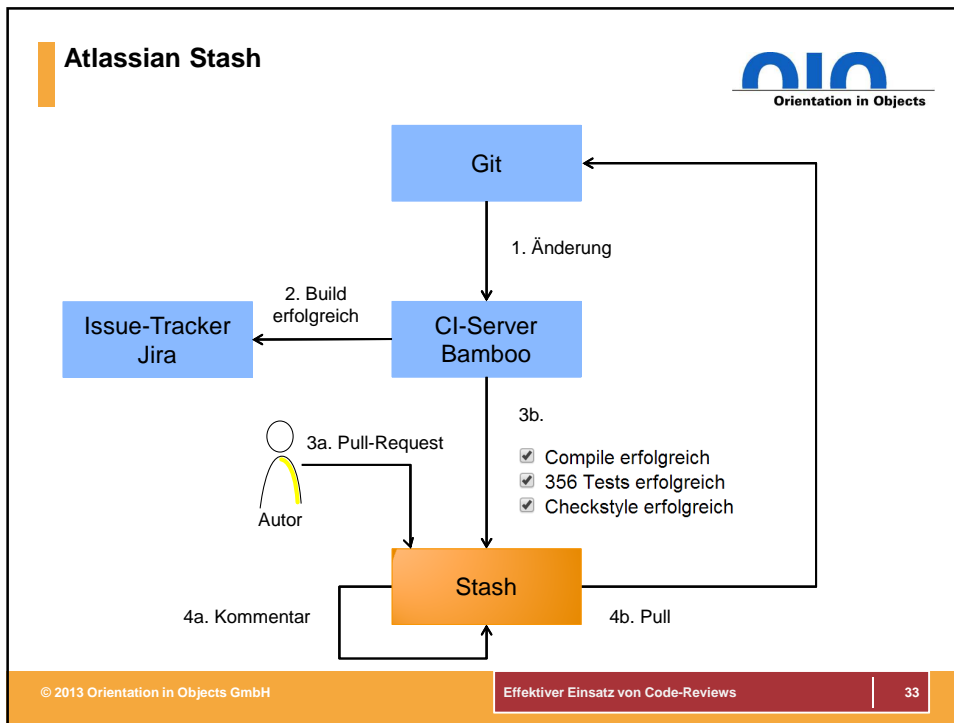
3. Review erstellen

- Compile erfolgreich
- 356 Tests erfolgreich
- Checkstyle erfolgreich

© 2013 Orientation in Objects GmbH

Effektiver Einsatz von Code-Reviews

32



Atlassian Stash

The screenshot shows the Atlassian Stash interface for a repository named "helloworld". It displays a pull request for a change in the file `GreetingService.java`. The code diff shows a modification to the `greetServer` method signature, adding an `email` parameter.

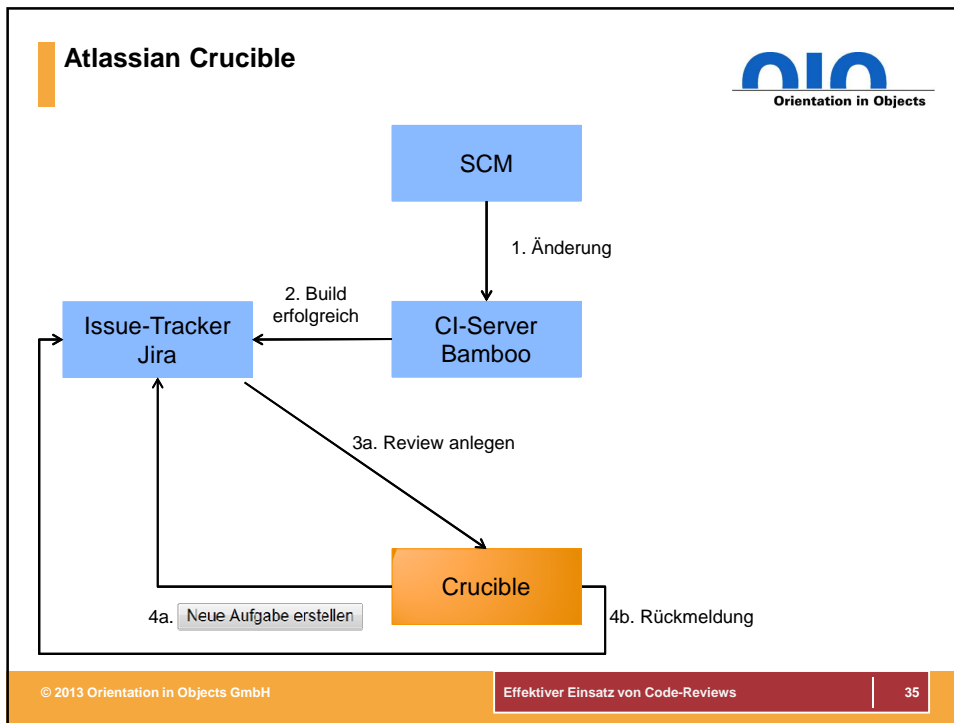
```

1 package de.oio.sample.gwt.client;
2
3 import com.google.gwt.user.client.rpc.RemoteService;
4 import com.google.gwt.user.client.rpc.RemoteServiceRelativePath;
5
6 /**
7  * The client side stub for the RPC service.
8  */
9 @RemoteServiceRelativePath("greet")
10 public interface GreetingService extends RemoteService {
- 11   String greetServer(String name) throws IllegalArgumentException;
+ 11   String greetServer(String name, String email) throws IllegalArgumentException;

```

Below the code diff, there is a comment box with the text: "Ist der zusätzliche E-Mail-Parameter an dieser Stelle wirklich notwendig?".

© 2013 Orientation in Objects GmbH | Effektiver Einsatz von Code-Reviews | 34



Atlassian Crucible

Default Project, CR-12

SCAP-5 Korrektur eines Typos

Under Review for a few seconds (due in 6 days) Author & Moderator

CR-12

Details
Objectives
General Comments

```

src/de/.../controller
  RegisterController.java 1
  
```

```

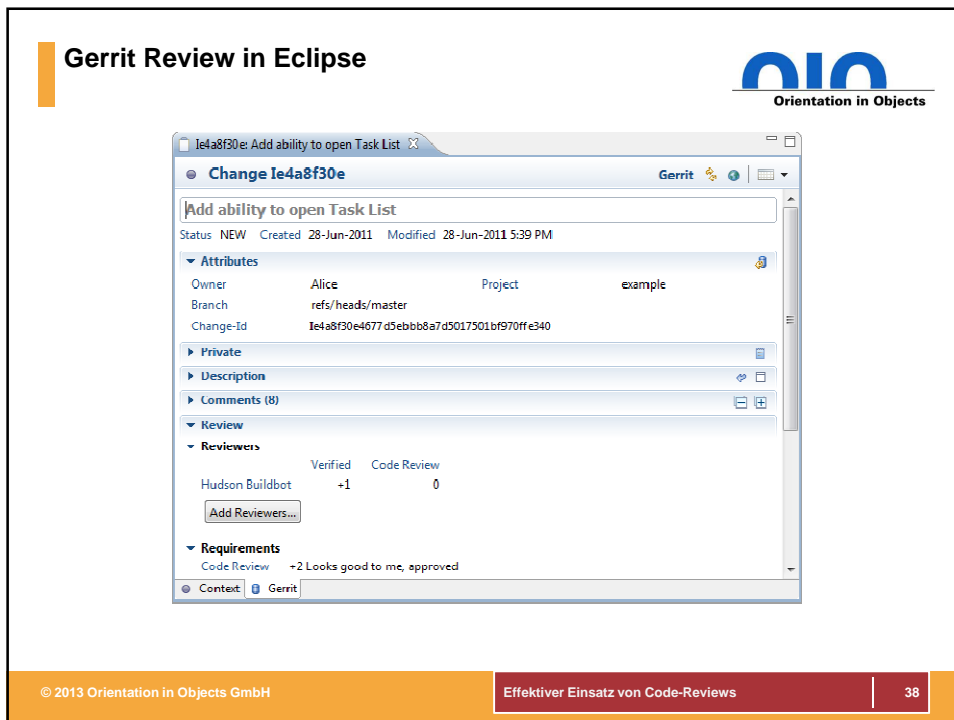
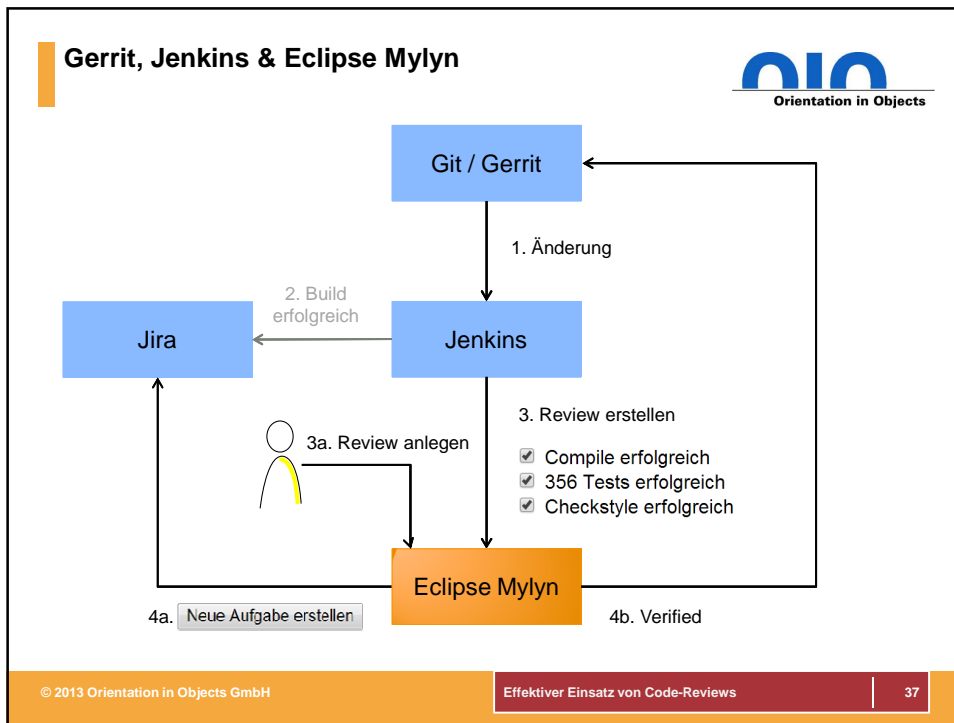
/src/de/.../controller/RegisterController.java  Changed  1
a90bf9f 6496674
Add a file comment. Click on source lines to add an inline comment.
3 3 public class RegisterController {
4 4
5 5     public String showRegisterForm() {
6 6         return "registerForm";
7 7     }
8 8 }
  
```

Thorsten Maier says: 21:48
Die entsprechende JSP-Datei muss ebenfalls umbenannt werden.

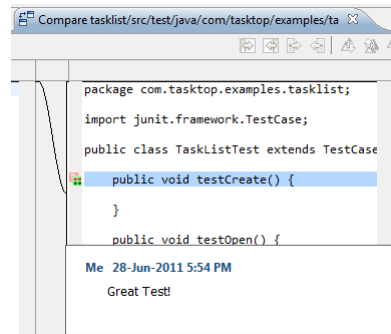
++ Reply Edit Delete Defect Ranking: Major Classification: Inconsistent Create Issue

© 2013 Orientation in Objects GmbH

Effektiver Einsatz von Code-Reviews | 36



Gerrit Diff in Eclipse

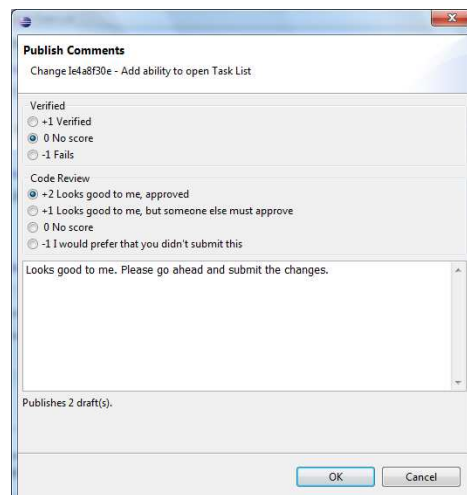


```

package com.tasktop.examples.tasklist;
import junit.framework.TestCase;
public class TaskListTest extends TestCase
{
    public void testCreate() {
    }
    public void testOpen() {
    }
}
Me 28-Jun-2011 5:54 PM
Great Test!

```

Gerrit Review in Eclipse abschließen



Publish Comments
Change Ie4a8f30e - Add ability to open Task List

Verified

+1 Verified

0 No score

-1 Fails

Code Review

+2 Looks good to me, approved

+1 Looks good to me, but someone else must approve

0 No score

-1 I would prefer that you didn't submit this

Looks good to me. Please go ahead and submit the changes.

Publishes 2 draft(s).

OK Cancel

Fazit

Einsteiger

Skeptiker

Profis

Wissen

KISS

© 2013 Orientation in Objects GmbH

Effektiver Einsatz von Code-Reviews

41

The diagram illustrates the progression of knowledge and code review practices. On the left, a box labeled 'Wissen' (Knowledge) has arrows pointing outwards. In the middle, a box shows two stick figures, one yellow and one green, with arrows pointing towards each other, representing collaboration. On the right, a box labeled 'KISS' (Keep It Simple, Stupid) is shown. The background is divided into three horizontal levels: 'Einsteiger' (Beginner) at the top, 'Skeptiker' (Skeptic) in the middle, and 'Profis' (Professionals) at the bottom, indicated by dotted lines.

Fragen ?

Orientation in Objects GmbH
Weinheimer Str. 68
68309 Mannheim
www.oio.de
info@oio.de

The slide features a large central graphic of several orange question marks of varying sizes. The text 'Fragen ?' is prominently displayed in the center. The left side of the slide has a vertical orange bar containing three small, square images of tree trunks, each with a white square marker on it. The bottom left corner contains the company name and contact information.



**Vielen Dank für ihre
Aufmerksamkeit !**

Orientation in Objects GmbH

Weinheimer Str. 68
68309 Mannheim

www.oio.de
info@oio.de