

Methodischer Einsatz im Projekt

Nach der Einführung in die Kernkonzepte und die Architektur von Jazz in der letzten Ausgabe beschreibt der zweite Beitrag den systematischen Einsatz von Jazz in Softwareentwicklungsprojekten. Zunächst werden die Schritte erläutert, über die ein neues Jazz-Projekt angelegt und bedarfsgemäß konfiguriert wird. Darauf aufbauend wird gezeigt, wie Jazz die dynamische Alltagsarbeit im Projekt unterstützt. Den Aufgabenbereich des Sourcecode-Managements fokussiert der letzte Teil dieser Artikelserie.

von Michael Müller, Prof. Dr. Veronika Thurner und Martin Wassermann

Die Entwicklung von großen, komplexen Softwaresystemen beinhaltet eine Menge von verschiedensten Aufgabenbereichen. Entsprechend vielfältig und unterschiedlich gestalten sich folglich die Anforderungen an die Fähigkeiten und Tätigkeiten der beteiligten Personen. Viele der zentralen Entwicklungsaufgaben eines Softwareprojekts sind dabei überwiegend kreativer Natur und somit meist nur schwer in ein starres Ablaufkorsett zu zwingen. Auf der anderen Seite erfordern umfangreiche Mehrpersonenprojekte ein gewisses Maß an koordinativen Vorgaben, damit das Projekt mit hoher Zuverlässigkeit zum Erfolg geführt werden kann. Diese Gratwanderung zwischen klar organisierender

Prozessdefinition und kreativem Laissez-faire ist das zentrale Dilemma aller Bemühungen, Entwicklungsprojekte mit standardisierten Vorgehensmodellen bzw. Prozessdefinitionen zu hinterlegen.

Jazz verfolgt die Philosophie, „gerade genug“ Prozessreglements vorzugeben, wie nötig sind, um die Zusammenarbeit im Team angemessen zu unterstützen und zu koordinieren. Gleichzeitig sollen die Prozessvorgaben so schlank gehalten sein, dass der kreative Entfaltungsspielraum der einzelnen Projektbeteiligten möglichst wenig eingeschränkt und das Maß an administrativen Tätigkeiten auf das erforderliche Minimum beschränkt wird. Da jedes Entwicklungsprojekt einzigartig ist und entsprechend spezifische Prozessanforderungen und Rahmenbedingungen mit sich bringt, bedingt diese Nutzungsphilosophie, dass mit der Architektur von Jazz eine Vielzahl von verschiedenen Prozessen realisiert und abgewickelt werden kann. So ist es

beispielsweise möglich, die völlig unterschiedlichen Anforderungen von kleinen, überschaubaren Projekten einerseits und andererseits von komplexen Großprojekten, die strengen behördlichen Regelungen unterliegen, gleichermaßen zielgerichtet zu unterstützen [1].

Vordefinierte Prozessmuster

Als Grundlage für die Definition eines projektspezifischen Prozesses stehen verschiedene vordefinierte Prozessmuster bereit, die einige heute gängige Vorgehensmodelle abbilden. In einem solchen Prozessmuster sind unter anderem Rollen und Rechte der Prozessbeteiligten, initial angelegte Entwicklungslinien und Iterationen, die Work-Item-Typen sowie die zugehörigen Workflows festgelegt. Bei Bedarf kann ein solches Prozessmuster an die Bedürfnisse eines konkreten Projekts angepasst oder auch ein völlig neues Prozessmuster angelegt werden. Tabelle 1 vermittelt einen Überblick über die Prozessmuster, die derzeit

Artikelserie

Teil 1: Kernkonzepte und Architektur

Teil 2: Methodischer Einsatz im Projekt

Teil 3: Sourcecode-Management mit Jazz

in Rational Team Concert vordefiniert sind [2].

Die in den Prozessmustern vordefinierten Rollen, Work-Item-Typen und zugehörigen Workflows stellt Tabelle 2 gegenüber. Zu beachten ist dabei, dass sich Prozesselemente des gleichen Namens bei den verschiedenen Prozessmustern in ihren Basisdefinitionen unterscheiden können. Beispielsweise verbirgt sich hinter dem Workflow „Retrospective“ des Scrum-Prozessmusters ein anderes Zustandsmodell als beim gleichnamigen Workflow des agilen Prozessmusters.

Aufsetzen eines eigenen Projekts

Um ein Projekt mit Jazz zu unterstützen, wird zunächst ein zugehöriger Projektbereich angelegt. Dabei wird das Prozessmuster ausgewählt, mit dem das Projekt hinterlegt werden soll. Alle im Prozessmuster vordefinierten Strukturen werden dabei automatisch mit dem neuen Projektbereich angelegt. Für einen schnellen, einfachen Einstieg in Jazz empfiehlt es sich, eines der vordefinierten Prozessmuster zu verwenden. Bei Bedarf können diese dynamisch erweitert bzw. auch komplett eigene Prozessmuster erstellt werden.

Im nächsten Schritt werden die Teams festgelegt, die im laufenden Projektbetrieb die eigentliche Projektarbeit ausführen. Dazu wird für jedes Team ein entsprechender Arbeitsbereich im Repository erzeugt und das Team mit konkreten Personen assoziiert, die anschließend die ihnen zugewiesenen Rollen ausführen. Parallel werden Iterationen definiert, indem die wichtigsten Meilensteine des Projekts auf grobgranularer Ebene festgelegt werden (Abb. 1).

Dynamische Alltagsarbeit im Projekt

Nach diesen initialen Schritten ist das Jazz-Projekt bereit für den produktiven Einsatz im Arbeitsalltag des Entwicklungsteams. Nun werden für die durchzuführenden Arbeitsschritte sukzessive, korrespondierende Work Items angelegt. Abhängig vom zugrunde liegenden Work-Item-Typ ist jeder Arbeitsschritt mit bestimmten Attributen versehen, beispielsweise Dringlichkeit, geschätztem Aufwand oder bei Fehlern dem Schweregrad. Unmittelbar mit dem Work Item verknüpft sind fachliche Diskussionen zu diesem Work Item. Kausale Abhängigkeiten, die eine bestimmte Bearbeitungsreihenfolge der Arbeits-

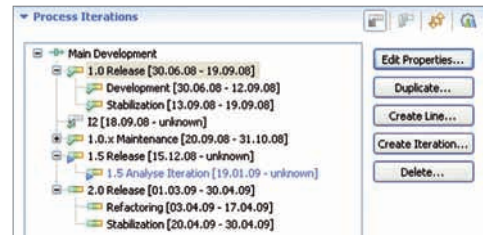


Abb. 1: Übersicht über Iterationen und Meilensteine

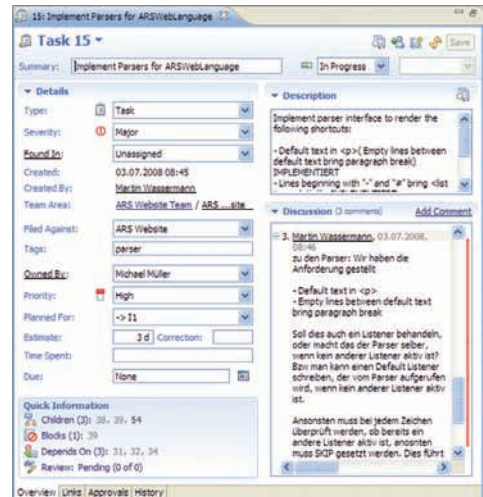


Abb. 2: Work Item

schritte nahe legen, können ebenfalls repräsentiert werden (Abb. 2).

Damit ein Arbeitsschritt tatsächlich ausgeführt wird, wird er einer bestimmten Iteration zugeordnet, in deren

Prozess	Beschreibung	Initiale Entwicklungslinien	Initiale Iterationen
Agil	Mischung aus bewährten agilen Praktiken	Main Development	Eine Iteration mit zwei Phasen: <ul style="list-style-type: none"> Entwicklung Stabilisierung
Eclipse Way	Iterationsbasierter agiler Prozess, ursprünglich vom Eclipse-Entwicklerteam entwickelt	Main Development	Zwei Iterationen: <ul style="list-style-type: none"> Entwicklung Stabilisierung
OpenUp	Anpassung des RUP, definiert vom Eclipse-Process-Framework-Projekt	Development	Vier Phaseniterationen: <ul style="list-style-type: none"> Inception Elaboration Construction Deployment Jede Phase gegliedert in weitere Iterationen
Cloudburst Sample	Prozessmuster des für Jazz verfügbaren Beispielprojekts	Zwei Entwicklungslinien, jeweils mit unterschiedlichen Meilensteinen und Prozessstrukturen: <ul style="list-style-type: none"> Development Maintenance 	
Scrum	Agile Methode	Main Development	Eine Iteration, unterteilt in mehrere Sprints fester Länge
Simple Team Process	Einfacher Basisprozess mit minimalen Vorgaben	Development	Eine Iteration

Tabelle 1: Vordefinierte Prozessmuster in Rational Team Concert

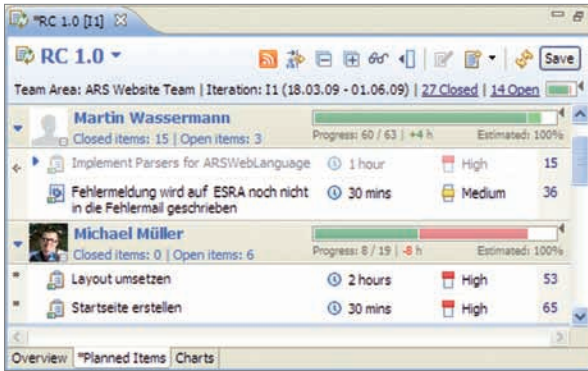


Abb. 3: Iterationsplan

Entwicklungszeitraum er zu erledigen ist. Des Weiteren werden die auszuführenden Arbeitsschritte dem jeweils zu-

ständigen Teammitglied zugewiesen. Die Zuordnung von Work Items zu einer bestimmten Iteration wird im Iterations-

plan dokumentiert, ebenso wie der Bearbeitungsstand der Iteration (Abb. 3). Gleichzeitig lassen sich die anstehenden Aufgaben und die damit verbundenen Aufwände für die einzelnen Mitarbeiter visualisieren.

Jedem Teammitglied wird in seinem Jazz-Client angezeigt, welche Work Items aktuell zu erledigen sind und wie hoch der dafür jeweils erforderliche Zeitaufwand geschätzt wurde. Hat ein Teammitglied einen Work Item bearbeitet, so dokumentiert es diese Änderung über den Bearbeitungszustand.

		Agile	Eclipse Way	OpenUp	Cloudburst Sample	Scrum	Simple Team Process
Rollen	Default	X	X	X	X	X	X
	Team Member	X					X
	Team Lead	X	X		X	X	
	Analyst			X			
	Architect			X			
	Developer			X			
	Project Manager			X			
	Stakeholder			X		X	
	Tester			X			
	Contributor		X			X	
	Product Owner					X	
	Scrum Master	X	X	X	X	X	X
Work Item Types	Defect		X	X	X		X
	Enhancement		X				
	Plan Item		X				
	Retrospective	X	X			X	
	Story	X	X			X	
	Task	X	X	X	X	X	X
	Track Build Item		X				
	Risk			X			
	Use Case			X			
	Impediment					X	
Workflows	Default	X	X	X	X	X	X
	Story	X				X	
	Plan Item		X				
	User Story		X				
	Testing	X					
	Retrospective	X	X			X	
	Impediment					X	
	Build Tracking		X				
	Risk			X			
	Use Case			X			

Tabelle 2: Vordefinierte Rollen, Work-Item-Typen und Workflows

Anpassbarkeit von Jazz

Da die vordefinierten Prozessmuster nicht jede denkbare Projektsituation optimal unterstützen, bietet Jazz vielfältige Möglichkeiten zur flexiblen Anpassung. Beispielsweise lassen sich neue Rollen definieren und mit entsprechenden Rechten hinterlegen. Für iterative Prozessmodelle lassen sich ferner Iterationstypen erstellen und mit Regeln verknüpfen, die die in einer solchen Iteration möglichen Aktionen einschränken. Unterscheidet man beispielsweise in einem Projekt die Iterationstypen „Entwicklung“ und „Stabilisierung“, so kann über die Regeln festgelegt werden, dass einer Stabilisierungsiteration keine neuen Arbeitsaufträge und damit Work Items zugewiesen werden dürfen, ausgenommen solche vom Typ *Defect* zur Fehlerkorrektur. Auch die verfügbaren Work-Item-Typen lassen sich bedarfs-gesteuert anpassen. Zum einen ist es möglich, einen vordefinierten Work-Item-Typ um neue Attribute zu erweitern. Zum anderen können aber auch völlig neue Work-Item-Typen erstellt werden. Neben den Standardtypen *Task* für die Entwicklung eines neuen Artefakts und *Defect* für die Fehlerkorrektur ist beispielsweise ein Typ *Test* hilfreich, um Aktivitäten zur Qualitätssicherung zu kennzeichnen. Mit jedem Work-Item-Typ lässt sich ergänzend ein so genannter *Workflow* assoziieren. Dieser beschreibt ein Zustandsmodell, das die möglichen Bearbeitungsstände eines konkreten Work Items dieses Typs sowie mögliche Übergänge zwischen diesen Zuständen festlegt (Abb. 4).

Impressum

Verlag:
Software & Support Verlag GmbH

Anschrift der Redaktion:
Java Magazin
Software & Support Verlag GmbH
Geleitsstraße 14
D-60599 Frankfurt am Main
Tel. +49 (0) 69 6300890
Fax. +49 (0) 69 63008989
redaktion@javamagazin.de
www.javamagazin.de

Chefredakteur: Sebastian Meyen
Redaktion: Claudia Fröhling, Hartmut Schlosser, Mirko Schrempf
Chefin vom Dienst: Nicole Bechtel
Schlussredaktion: Nicole Bechtel, Katharina Klassen, Frauke Pesch
Leitung Grafik & Produktion: Jens Mainz
Layout, Titel: Daniela Albert, Kristin Brockmann, Pöbpor Fischer, Karolina Gaspar, Melanie Hahn, Katharina Ochsenhirt, Maria Rudi, Patricia Schwesinger
CD/DVD-Erstellung: Daniel Zuzek

Autoren dieser Ausgabe:
Pavlo Baron, Markus Demolsky, Dr. Lofi Dewanto, Kristopher Eriksson, Christian Grobmeier, Thorsten Kamann, Michael Kurz, Bernhard Löwenstein, Berthold Maier, Martin Marinschek, Florian Müller, Michael Müller, Hajo Normann, Mirko Novakovic, Gerhard Petracek, Daniel Pieper, Alois Reitbauer, Roman Roelofsen, Arne Roock, Lars Röwekamp, Berthold Scheuringer, Bernd Schiffer, Christian Schneider, Heiko Seeberger, Markus Stäuble, Veronika Thurner, Dalibor Topic, Bernd Trops, Clemens Utschig-Utschig, Markus Völter, Martin Wassermann, Matthias Weßendorf, Torsten Winterberg, Henning Wolf, Stefan Zörner

Anzeigenverkauf:
Software & Support Verlag GmbH
Patrik Baumann
Tel. +49 (0) 69 63008 90
Fax. +49 (0) 69 630089 89
pbaumann@javamagazin.de

Es gilt die Anzeigenpreisliste Nummer 12

Pressevertrieb:
DPV Network
Tel.+49 (0) 40 378456261,
www.dpv-network.de

Druck: PVA Landau
ISSN: 1619-795X

Abo-Service:
Software & Support Verlag GmbH
Tel. +49 (0) 69 6300890
Fax +49 (0) 69 63008989
www.javamagazin.de/service/

Abonnementpreise der Zeitschrift:

Inland:	12 Ausgaben	€ 79,-
Europ. Ausland:	12 Ausgaben	€ 89,-
Studentenpreis (Inland):	12 Ausgaben	€ 69,-
Studentenpreis (Ausland):	12 Ausgaben	€ 79,-

Einzelverkaufspreis:

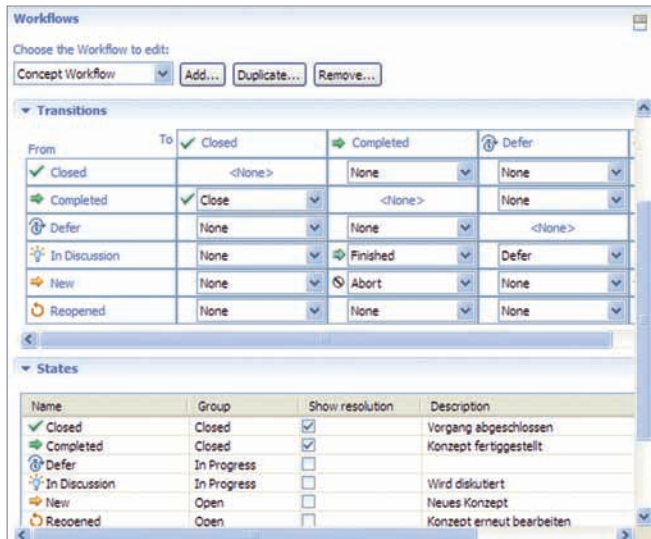
Deutschland:	€ 7,50
Österreich:	€ 8,60
Schweiz:	sFr 15,80

Erscheinungsweise: monatlich

© Software & Support Verlag GmbH

Alle Rechte, auch für Übersetzungen, sind vorbehalten. Reproduktionen jeglicher Art (Fotokopie, Nachdruck, Mikrofilm oder Erfassung auf elektronischen Datenträgern) nur mit schriftlicher Genehmigung des Verlages. Jegliche Software auf der Begleit-CD zum *Java Magazin* unterliegt den Bestimmungen des jeweiligen Herstellers. Eine Haftung für die Richtigkeit der Veröffentlichungen kann trotz Prüfung durch die Redaktion vom Herausgeber nicht übernommen werden. Honorierter Artikel gehen in das Verfügungsrecht des Verlages über. Mit der Übergabe der Manuskripte und Abbildungen an den Verlag erteilt der Verfasser dem Herausgeber das Exklusivitätsrecht zur Veröffentlichung. Für unverlangt eingeschickte Manuskripte, Fotos und Abbildungen keine Gewähr. Java™ ist ein eingetragenes Warenzeichen der Sun Microsystems Inc.

Abb. 4:
Workflow



Falls erforderlich, kann der für das Gesamtprojekt definierte Prozess für jedes Team individuell angepasst werden. So ist es möglich, die Vorgaben des Prozessmusters flexibel auf die spezifischen Bedürfnisse des Projekts und der einzelnen Teams zuzuschneiden. Die grafische Oberfläche von Rational Team Concert bietet bereits eine Fülle von Möglichkeiten, um aus dem Gesamtpool der vordefinierten Prozessfunktionalität auf sehr feingranularer Ebene neue Prozessmuster zu konfigurieren. Prinzipiell ist es möglich, auch komplett neue Prozessfunktionalität in Form von Java-Klassen zu realisieren und diese dann über die in XML vorliegenden Prozesseinstellungen in die Prozessdefinition zu integrieren.

Zusammenfassung und Ausblick

Ein wesentlicher Hauptnutzen von Jazz ist der verbesserte Überblick über den aktuellen Stand des Projekts und über die Tätigkeiten, die als Nächstes zu erledigen sind. Durch die Integration von technischen und planerischen Artefakten in ein einziges Werkzeug und durch die Verzahnung von Planung und technischem Vorgehensmodell wird eine weitgehend automatische Steuerung der zugrunde liegenden Workflows möglich. Dadurch vereinfacht sich sowohl die Aufgabenplanung als auch die Fortschrittskontrolle im Projekt. Nicht zuletzt unterstützt der konsequente Einsatz von Jazz die aufgabenorientierte Kommunikation im Gesamtprojekt

und verbessert so die Team Awareness erheblich. Nach der Einführung in die Kernkonzepte und die Architektur und einem Überblick über den methodischen Einsatz von Jazz im Projekt fokussiert der letzte Beitrag dieser Artikelseerie die Konzepte und Mechanismen, die Jazz für das Sourcecode-Management bereitstellt. ■



Michael Müller und Martin Wassermann arbeiten als Berater bei der ARS Computer und Consulting GmbH in München im Umfeld von Enterprise Java, Web Services und Softwarequalität.



Prof. Dr. Veronika Thurner forscht und lehrt an der Hochschule München in den Themenbereichen Software Engineering und Geschäftsprozesse.



Links & Literatur

- [1] Barnett, Liz: „The IBM Rational Jazz Strategy for Collaborative Application Life Cycle Management“, EZ Insight, Inc., Juli 2008
- [2] Jazz Community Site: „Prozessschablonen“: https://jazz.net/help/rational-team-concert/1.0.1/index.jsp?topic=/com.ibm.team.apt.doc/topics/c_iterations_workitems.html (kostenfreie Registrierung erforderlich), Mai 2009