

eclipse

MAGAZIN

www.eclipse-magazin.de

Mit
CD
Alle Infos S. 3

Tools, Open Source & more

- » Eclipse Visual Editor Project
- » Apache Tomcat 6.0
- » ATLAS Transformation Language
- » Eclipse Modeling Framework (EMF)
- » Eclipse Graphical Modeling Framework (GMF)
- » Graphical Editing Framework (GEF)
- » Standard Widget Toolkit (SWT)



BONUS:

Artikelserie aus dem Java Magazin „OSGi applied“
+ Bonusartikel aus dem Eclipse Magazin Vol. 12

Professionell Testen

Best Practices: GUI-Tests für Eclipse RCP

» **Embedded Eclipse:**
Model Driven Development
bei eingebetteten Systemen

Praxis

Build-Prozess von Eclipse Plug-ins

Rich Clients

Neu seit Eclipse 3.3:
Eclipse DataBinding

Model Driven

Dokumentationen generieren

Plattformen

Eclipse und NetBeans Plattform im Vergleich



D 68864





Die Automatisierung der Softwareentwicklung als Geschäftsvision

Model to Business

» MARKO BOGER UND PETER FRIESE

Produktivitätssteigerung ist das Zauberwort der Wirtschaft in der gesamten industrialisierten Welt. Automatisierung ist dabei der Schlüssel zum Erfolg. Lediglich in der Branche, die diese Automatisierung in den meisten Fällen erst möglich macht, nämlich in der IT, war das lange kein Thema.

Softwareentwickler betrachteten sich selbst meist eher als Handwerker oder als Künstler denn als Ingenieure. Entsprechend lag der Fokus zur Produktivitätssteigerung in der IT auf der Verbesserung der Handwerkzeuge. Programmiersprachen, Compiler und Editoren sind heutzutage aber weitgehend ausgereizt, und so setzt ein Prozess des Umdenkens ein, der die Automatisierung als Mittel der Produktivitätssteigerung auch in der Softwareentwicklung zulässt und den Programmierer zum Ingenieur werden lässt. Eclipse als Plattform und den Projekten aus dem Eclipse Modeling Project kommen dabei eine Schlüsselrolle zu.

Die Gentleware AG setzt strategisch auf die Mittel des Eclipse-Ökosystems, um die Vision, die Softwareentwicklung durch die Verwendung von Modellen zu automatisieren, umzusetzen. Die Firma wurde bekannt durch „Poseidon for UML“, ein weit verbreitetes UML-Modellierungswerkzeug. Für den Einsatz von Generatoren, die aus den Modellen automatisiert Softwareartefakte erzeugen, setzt Gentleware das Eclipse-Projekt openArchitectureWare (oAW) ein. Für den erfolgreichen Einsatz von

modellgetriebener Softwareentwicklung setzen sich zunehmend domänenspezifische Modellierungssprachen (DSLs) durch, die Gentleware mit den Eclipse-Projekten EMF und GMF entwickelt. Die Kombination von DSL und oAW-Generatoren sowie der Eclipse-Plattform ermöglicht die Entwicklung von Werkzeugen im Sinne einer „Software-Produktionsstraße“. Gentleware hat sich auf die Entwicklung derartiger Werkzeuge spezialisiert.

Automatisierung durch Modelle

Für die modellgetriebene Softwareentwicklung sind zwei komplementäre Werkzeuge notwendig: das Modellierungswerkzeug und der Generator. Die Gentleware AG konzentrierte sich nach ihrer Gründung zunächst auf die Entwicklung des Ersteren und entwickelte das Produkt „Poseidon for UML“.

Im Eclipse-Ökosystem sind in den letzten Jahren sehr mächtige Frameworks für die Entwicklung von Modellierungswerkzeugen entstanden, und Gentleware schwenkte auf diese neuen Technologien um. Auf den bereits gemachten Erfahrungen aufbauend, ent-

wickelte Gentleware ein Framework für die Herstellung von domänenspezifischen Modellierungssprachen (DSL) und -Editoren, basierend auf den Eclipse-Technologien EMF und GMF. Damit ist Gentleware in der Lage, eine maßgeschneiderte Modellierungsumgebung innerhalb von Wochen (statt Jahren) aufzubauen, deren Einführung sich sogar für einzelne Projekte rechnet. Die DSL kann an die Bedürfnisse der Firmenprozesse, die vorhandenen fachlichen Abstraktionen, die verwendeten Technologien und an das Vorwissen der Projektmitarbeiter angepasst werden. Die Sprache selbst kann meist sehr schlank gehalten und leicht erlernt werden. Außerdem lässt sich für eine schlanke DSL auch ein schlanker Generator entwickeln. Die einzelnen Eclipse-Technologien fügen sich nahtlos zusammen und bilden ein größeres Ganzes.

Im Rahmen des entwickelten Applikationsframeworks für DSLs ist die UML natürlich ebenfalls ein spezieller Ausprägungsfall einer DSL. Entsprechend entwickelt Gentleware auch eine neue Generation von UML-Werkzeugen unter dem Namen „Apollo for Eclipse“. Dieses Tool ist speziell für die Bedürfnisse agiler Entwicklungsmethoden gedacht und unterstützt eine naht-

Apollo for Eclipse

- native Einbindung in Eclipse
- live Roundtrip für Java 5
- hoch skalierbar
- intuitives Benutzerinterface
- besonders geeignet für agile Entwicklungsmethoden

Poseidon for UML

- UML 2-kompatibel
- Dokumentationsgenerator
- intuitive Benutzeroberfläche
- XML Export
- ECore Export
- Editionen: Community, Standard, Professional und Embedded

Model



- 1 DSL Design
- 2 Model Editor in Eclipse
- 3 Generator in oAW



Business

Abb. 1:
Typischer
Entwicklungs-
prozess
für eine
Software-
Produktions-
straße

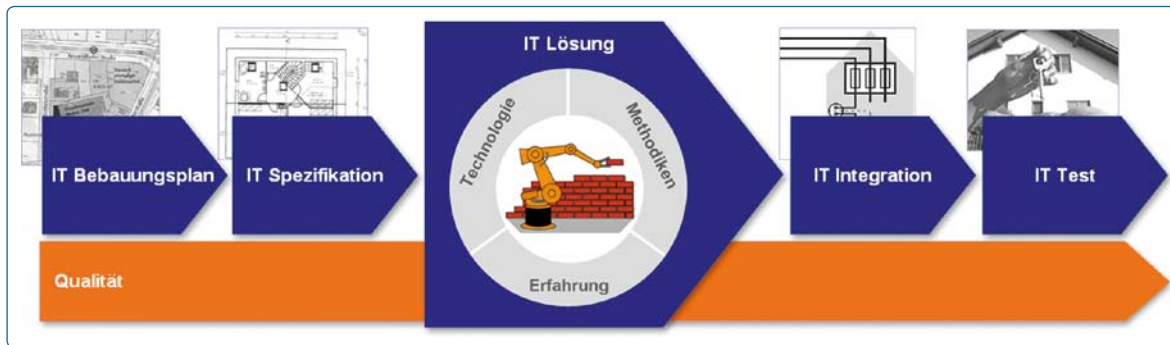


Abb. 2:
Kontext einer
Software-
Produktionsstraße

lose Integration in Eclipse mit einem „live“ synchronisierenden Roundtrip-Mechanismus. Apollo for Eclipse ist ein natives Eclipse Plug-in, steht aber auch als Eclipse-RCP-Anwendung zur Verfügung.

Produktivitätssteigerung durch Generatoren

Typischerweise werden heutzutage Frameworks in Softwareprojekten eingesetzt. Dies führt dazu, dass der Code einen hohen Anteil an schematischem Code enthält, der strukturell den Anforderungen des Frameworks gehorcht. Dieser schematische Code lässt sich meist hervorragend automatisch generieren, wobei das Modell die Generierung sozusagen parametrisiert.

Gentleware entwickelte früh Codergeneratoren für UML mit Generatorframeworks wie Velocity oder JET (Java Emitter Templates). Damit war die direkte Abbildung von Klassendiagrammen auf Klassen gut möglich. Es stellte sich allerdings heraus, dass die UML als General Purpose Language zu komplex und nicht präzise genug war. Die DSLs bieten hier einen großen Vorteil. Außerdem führten die Template-Sprachen der ersten Generation zu einer schlechten Wartbarkeit. Moderne Generatorframeworks der zweiten Generation – wie etwa openArchitectureWare – verstehen sich als Transformatoren von Modellen zu Text und bieten hierfür maßgeschneiderte Mechanismen. Die DSL und die Generatoren können gut aufeinander abgestimmt werden, sodass der Generator einfach und gut wartbar bleibt. Außerdem ist oAW gut in Eclipse eingebettet, sodass eine saubere Integration von DSL und Generator in die Eclipse-Plattform möglich ist. oAW ist seit kurzem in das Eclipse Modeling Project integriert und damit integraler Bestandteil der Eclipse-Infrastruktur.

Um das eigene Know-how in Bezug auf DSLs und oAW-Generatoren effizienter in Projekte einbringen zu können, hat sich die Gentleware AG Anfang des Jahres der b+m-Gruppe angeschlossen. b+m ist Initiator und zugleich erfahrester oAW-Nutzer. Gemeinsam wird an dieser für Gentleware zentralen Technologie im Sinne der Community weitergearbeitet.

Umsetzung in Projekten

Ein typisches Projekt der Firma Gentleware läuft folgendermaßen ab:

1. In einem ersten Schritt wird eine ganz individuelle Modellierungssprache (DSL) entwickelt. Die Notation kann beliebig nah an bereits im Anwenderteam bekannten Modellierungskonzepten sein. Wichtig ist, dass die Sprache schlank, klar und leicht zu lernen ist und sie die intendierte Fachlichkeit auf angemessenem Abstraktionsniveau ausdrücken kann.
2. Wenn diese Sprachdefinition weit genug gefestigt ist, wird im nächsten Schritt ein Modellierungswerkzeug als Erweiterung von Eclipse entwickelt. Dies geschieht auf Basis von EMF und GMF und ist selbst ein modellgetriebener, generativer Prozess.
3. Im dritten Schritt wird ein maßgeschneiderter Generator entwickelt. Hierfür werden die sich wiederholenden Code Segmente identifiziert, in Templates überführt und mit den entsprechenden Elementen aus dem Modell verknüpft. Dies geschieht auf Basis einer vorhandenen Implementierung, einer exemplarischen Referenzimplementierung oder mehrerer technischer Durchstiche.

Diese Schritte können iterativ wiederholt werden. Typischerweise vertiefen die Anwender bereits während der Entwick-

lung die Erkenntnisse über das Potenzial dieses Ansatzes und können ihre Erfahrungen direkt in den laufenden Prozess einbringen.

Fazit

Im Ergebnis kann aus einem Modell je nach verwendeter Technologie 60–80 Prozent des notwendigen Codes generiert werden. Dadurch steigt die Produktivität der Entwicklung. Gleichzeitig erhöht sich die Qualität des Codes, da er der festgelegten Architektur genau folgt. Die Behebung eines Fehlers löst nicht mehr nur ein lokales Problem, sondern merkt es prinzipiell aus. Das Modell stellt eine zuverlässige Dokumentation dar, da der Code ja daraus hervorgeht. Last but not least gewinnt man eine große Flexibilität für die Migration auf eine andere Technologie, Bibliothek oder ein alternatives Framework. Der Generator wird einfach geändert und der Code neu generiert. Damit bringt Gentleware den Mehrwert der Modellierung in die Geschäftsprozesse seiner Kunden ein: Model to Business.



Dr. Marko Boger ist Gründer und Leiter der Gentleware AG. Er hat an der Universität Hamburg promoviert, ist Buchautor und hält regelmäßig Vorträge auf internationalen Konferenzen. Er befasst sich mit UML, MDSD, Projektmanagement, Entwicklungsprozessen und Entrepreneurship. Kontakt: marko.boger@gentleware.com.



Peter Friese ist Senior-Softwarearchitekt bei Gentleware. Er ist spezialisiert auf Spring, Eclipse-Technologien und modellgetriebene Softwareentwicklung und ist Committer für FindBugs und openArchitectureWare. Kontakt: peter.friese@gentleware.com.