

ModelIntegra: Modellgetriebene Fachkonzepte

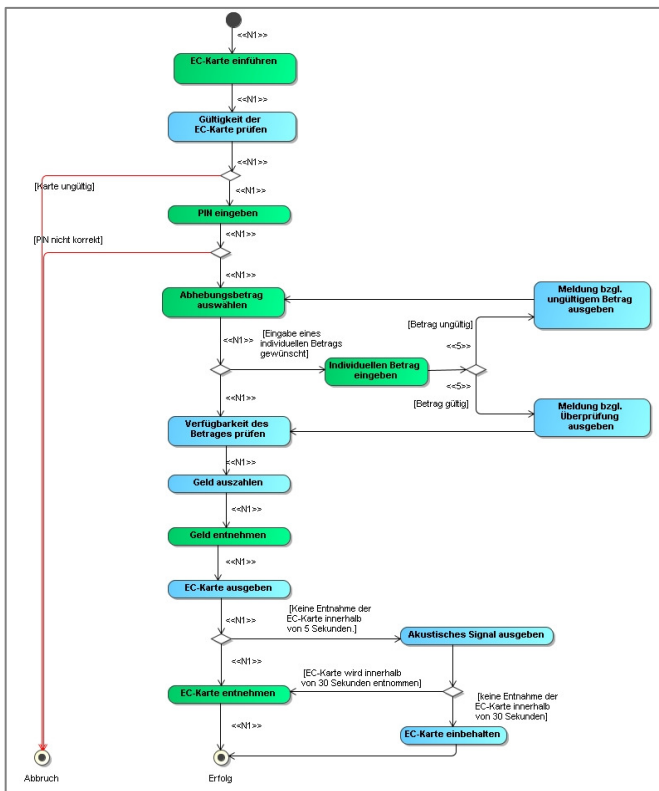
4Soft-Whitepaper – 4. Juni 2007

Model Integra

Bei der Entwicklung von IT-Systemen spielen Anforderungsanalysten eine zentrale Rolle: Sie erfassen die Anforderungen der Beteiligten und Anwender und bereiten sie so auf, dass sie als Ausgangspunkt für die weitere Entwicklung dienen können. Die entstehenden Lasten- und Pflichtenhefte müssen dazu sowohl lesbar und verständlich als auch präzise und eindeutig sein. ModelIntegra bietet Anforderungsanalysten eine Methode mit zugehörigem Werkzeug, um beides zu erreichen.

ModelIntegra: Methode mit Werkzeugunterstützung

Die ModelIntegra-Methode nutzt Anwendungsfälle und Fachklassen, um Funktionalität und Daten von Systemen zu beschreiben. Anforderungsanalysten erstellen dazu Modelle mit Hilfe von Modellierungswerkzeugen wie RSA oder MagicDraw. Dabei folgen sie den ModelIntegra-Vorgaben und verwenden das ModelIntegra-Plugin zur Qualitätssicherung und zur Generierung einheitlicher, hochqualitativer Lasten- und Pflichtenhefte.



Modellierter Ablauf

aus dem Modell. Zur besseren Lesbarkeit enthalten sie redundante Textpassagen und Verweise. Unterschiedliche Ausgabeformate sorgen für eine zielgruppengerechte und zweckmäßige Darstellung der Informationen. Eine manuelle Nachbearbeitung der generierten Anteile kann deshalb entfallen.

Strukturiertes Vorgehen
durch standardisierte Methodik

Automatische Qualitätssicherung durch Modell-Validierung

Methodische Testfall-Ableitung durch formalisierte Szenarien

Einheitliche Dokumentation durch Generierung

Modelle als Basis

Modellbasierte Anforderungen auf Basis standardisierter, formaler Beschreibungstechniken bieten große Vorteile: Sie haben eine klare Bedeutung und lassen sich damit unmissverständlich formulieren und an die Entwickler weitergeben. Zudem haben sie wenig Redundanz – Analysten müssen Informationen also nur einmal eingeben und bei Änderungen nicht aufwändig konsistent halten.

Generierung verständlicher Fachkonzepte

Für sich alleine sind formale Modelle allerdings nicht geeignet, da End-Anwender sie im Normalfall nicht ohne weiteres verstehen können.

Die ModelIntegra-Methode setzt deshalb auf das Zusammenspiel formaler Modelle und natürlichsprachiger Texte, um die Brücke zwischen Anforderungstellern und Entwicklern zu bauen. Die Anforderungsanalysten geben diese Informationen in das Modellierungswerkzeug ein und können daraus mit ModelIntegra auf Knopfdruck hochwertige und verständliche Fachkonzepte generieren.

Der ModelIntegra-Generator erzeugt konsistente Dokumente aus dem Modell. Zur besseren Lesbarkeit enthalten sie redundante Textpassagen und Verweise. Unterschiedliche Ausgabeformate sorgen für eine zielgruppengerechte und zweckmäßige Darstellung der Informationen. Eine manuelle Nachbearbeitung der generierten Anteile kann deshalb entfallen.

Normalablauf

- 1: **EC-Karte einführen:** Der Kunde schiebt seine EC-Karte in den Kartenleser des ATM.
- 2: **Gültigkeit der EC-Karte prüfen:** Das System prüft die EC-Karte auf Gültigkeit.
- 3: **PIN eingeben:** Der Kunde wird zur Eingabe seiner PIN aufgefordert und gibt diese ein. Falls eine falsche PIN eingegeben wurde werden.
- 4: **Abhebungsbetrag auswählen:** Der Kunde bekommt Beträge zur Abhebung angeboten und wählt einen Betrag aus. Alternativ eingeben.
- 5: **Verfügbarkeit des Betrages prüfen:** Das System prüft die Verfügbarkeit des Betrages.
- 6: **Geld auszahlen:** Das System gibt das Bargeld an der Bargeld-Ausgabe aus.
- 7: **Geld entnehmen:** Der Kunde entnimmt das Bargeld.
- 8: **EC-Karte ausgeben:** Die EC-Karte wird am Kartenleser ausgegeben und der Kunde wird darüber wie folgt benachrichtigt: 1. C Durch einen Hinweis an der GUI des ATM.
- 9: **EC-Karte entnehmen:** Der Kunde entnimmt die EC-Karte.

Varianten zu Normalablauf

Varianten zu: 3 PIN eingeben

Variante 1.3.1: [Karte ungültig]

- 3: Die EC-Karte ist nicht gültig oder nicht lesbar.
- 4: **Use-Case endet hier**

Varianten zu: 4 Abhebungsbetrag auswählen

Variante 1.4.1: [PIN nicht korrekt]

- 4: Der PIN wurde falsch eingegeben.
- 5: **Use-Case endet hier**

Varianten zu: 5 Verfügbarkeit des Betrages prüfen

Variante 1.5.1: [Eingabe eines individuellen Betrags gewünscht] + [Betrag gültig]

- 5: Der Kunde möchte die Höhe des Betrages frei bestimmen.
- 6: Der Kunde gibt einen individuellen Betrag über die GUI ein.
- 7: Der eingegebene Betrag ist gültig.
- 8: Es wird an der GUI eine Meldung ausgegeben, dass die Verfügbarkeit geprüft wird.
- 9: weiter zu: **5 Verfügbarkeit des Betrages prüfen**

Variante 1.5.2: [Eingabe eines individuellen Betrags gewünscht] + [Betrag ungültig]

- 5: Der Kunde möchte die Höhe des Betrages frei bestimmen.
- 6: Der Kunde gibt einen individuellen Betrag über die GUI ein.
- 7: Der Betrag ist ungültig, da er zu hoch oder zu niedrig ist.
- 8: An der GUI wird eine Meldung ausgegeben, dass der Betrag nicht gültig ist.
- 9: weiter zu: **4 Abhebungsbetrag auswählen**

Generierter Ausschnitt eines Fachkonzepts
(Anwendungsfallbeschreibung)

Methodische Ableitung von Systemtestfällen

Die formale Spezifikation der Anwendungsfall-Abläufe mit Hilfe von UML-Aktivitäten erlaubt es, methodisch und automatisiert Testfälle abzuleiten. ModelIntegra ermittelt dazu die möglichen Pfade durch den fachlichen Anwendungsfall-Ablauf und generiert jeweils eine auf den konkreten Testpfad zugeschnittene Testfallbeschreibung.

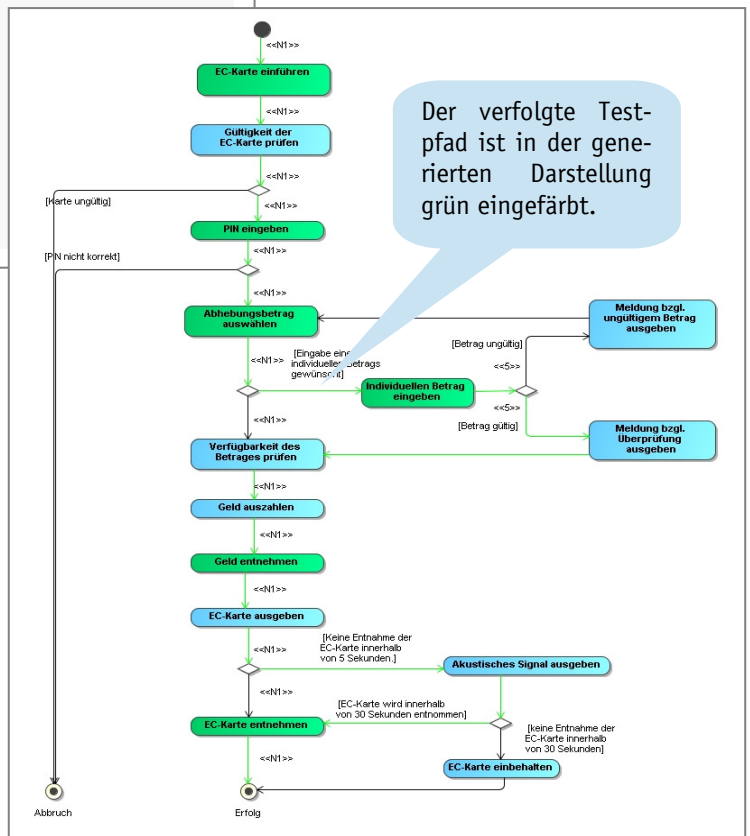
Die generierten Testfallbeschreibungen sind eine solide Ausgangsbasis zur Erstellung der funktionalen Abnahmetests.

ModelIntegra kann folgende Anteile von Fachkonzepten generieren:

- ◆ **Anwendungsfall-Spezifikationen** auf zwei Detaillierungsebenen für Grob- und Feinkonzepte
- ◆ **Fachdaten-Spezifikationen** mit den Fachklassen samt ihren Attributen und Datentypen
- ◆ **System-Testfälle** mit Abläufen für Integrations- und Abnahme-Tests

ModelIntegra unterstützt folgende Ziel-Formate:

- ◆ **MS Word** für lesbare, in sich abgeschlossene Dokumente
- ◆ **HTML-Seiten** zum Nachschlagen und zur Abstimmung im Team



Visualisierung eines generierten Testfalls

Durchgängigkeit und Traceability im Entwicklungsprozess

Als Brücke zwischen Anwendern und Entwicklern sind Anforderungsanalysten und ihre erstellten Lasten- und Pflichtenhefte stark in den restlichen Entwicklungsprozess eingebunden.

ModelIntegra bietet deswegen eine vollständige Integration:

- ◆ Anwendungsfälle lassen sich mit textuell spezifizierten Anforderungen verknüpfen, die in unterschiedlicher Form vorliegen können – von Excel-Tabellen bis hin zu spezialisierten Anforderungsmanagement-Systemen.
- ◆ Ablaufspezifikationen und Fachklassen gestatten eine unmittelbare Durchgängigkeit zu den Dienstschnittstellen der Software-Architektur.
- ◆ Testspezifikationen lassen sich in Test-Management-Systeme einspielen.

Derartige Integrationen sorgen für die methodische und technische Durchgängigkeit der Prozesse und die Vernetzung der Entwicklungsdokumente.

ModelIntegra kann damit die Traceability von den Anforderungen bis hin zu den Abnahmetests und der Implementierung gewährleisten, die moderne Vorgehensmodelle wie das V-Modell XT fordern.

Werkzeugunabhängigkeit und Integrierbarkeit

ModelIntegra nutzt zur Integration mit benachbarten Entwicklungswerkzeugen einfache Import/Export-Funktionalität zur Datenübernahme. Die Verarbeitungslogik arbeitet auf einer ModelIntegra-eigenen Domain Specific Language (DSL) zu Anforderungsanalyse und ist somit werkzeugunabhängig und wiederverwendbar.

Integrationen sind ausschließlich an den werkzeugunabhängigen ModelIntegra-Kern angebunden. Somit bestehen keine Abhängigkeiten der Werkzeuge untereinander. Jedes einmal integrierte Werkzeug ist somit mit allen anderen Integrationen kompatibel. Die Anbindung neuer Werkzeuge und die Integration in bestehende Werkzeuglandschaften sind mit begrenztem Aufwand realisierbar.

Realisierung als Plugin für

- ◆ IBM Rational Software Architect (RSA)
- ◆ NoMagic MagicDraw UML

Integrationen zu weiteren UML2-Werkzeugen möglich

Entwicklung als Open-Source-Projekt

ModelIntegra steht als Open-Source-Projekt bei Sourceforge zur Verfügung. Damit können Anwender weitere Funktionalität und zusätzliche Integrationen einbringen. Mehr zu ModelIntegra findet sich auf der 4Soft-Website.

Unser Angebot zu ModelIntegra

Das ModelIntegra-Konzept basiert auf unserer Expertise im Bereich von Vorgehensmodellen, der modellbasierten Entwicklung und von Entwicklungswerkzeugen. Unser Angebot umfasst Training, Beratung und Support rund um das modellgetriebene Anforderungsmanagement und den Einsatz von ModelIntegra.



ModelIntegra – das Werkzeug

Wir stellen ModelIntegra bereit, so dass Sie produktiv arbeiten können. Dazu gehören die Anpassung der bestehenden Funktionalität von ModelIntegra an Ihre Bedürfnisse und die Integration des Werkzeugs in Ihre bestehende Werkzeuglandschaft.



Training und Beratung

Wir führen eine strukturierte, modellgetriebene Methodik zum Anforderungsmanagement ein. Dazu schulen und coachen wir Ihre Anforderungsanalysten, entwickeln anschauliche Modellierungsbeispiele und passen Vorgehensmodell und Methodenhandbücher entsprechend an.

Kontakt zu ModelIntegra

E-Mail: info@4soft.de

Ansprechpartner: Michael Kempf

4Soft GmbH

4Soft führt innovative Lösungen und Techniken mit methodischem Software-Engineering zusammen. Die Kernkompetenzen des Unternehmens reichen von der Erarbeitung zukunftsfähiger IT-Strategien über die Projektdurchführung und das modellbasierte Software-Engineering bis hin zu innovativen Technologien und Software-Architekturen. In diesen Bereichen bietet 4Soft Beratungsleistungen sowie die eigenständige Projektdurchführung an.