

# **Der Weg vom Geschäftsprozess zum Workflow**

**Am Beispiel von ARIS und Intalio|BPMS**

**Prof. Dr. Thomas Allweyer**

**Fachhochschule Kaiserslautern**

**Februar 2008**

Kontakt:

Prof. Dr. Thomas Allweyer  
Fachhochschule Kaiserslautern  
Fachbereich Informatik und Mikrosystemtechnik  
Standort Zweibrücken  
Amerikastr. 1  
66482 Zweibrücken  
Tel.: 06332/914-324  
E-Mail: thomas.allweyer@fh-kl.de

Dieser Artikel wurde zur Verfügung gestellt von

**[www.kurze-prozesse.de](http://www.kurze-prozesse.de)**

Das Blog zum Geschäftsprozessmanagement

ARIS und IDS sind eingetragene Marken der IDS Scheer AG.

Intalio, Intalio|BPMS, Intalio|Designer, Intalio|Server und Intalio|Workflow sind Marken oder eingetragene Marken von Intalio, Inc.

Java ist eine eingetragene Marke von Sun Microsystems, Inc.

XML ist eine eingetragene Marke des World Wide Web Consortium.

Alle anderen im Text genannten Namen von Produkten und Dienstleistungen sind Marken oder eingetragene Marken der jeweiligen Eigentümer.

## Inhalt

1	Einleitung .....	4
2	Fachliche Modellierung .....	5
3	Entwicklung des ausführbaren Workflow-Modells .....	7
3.1	Anlegen von Rollen und Benutzern .....	7
3.2	Festlegen der Datenstrukturen.....	7
3.3	Kontrollfluss und Benutzerdialoge.....	8
3.3.1	Dialog Antrag erfassen.....	10
3.3.2	Integration des Dialogs in den Workflow als auslösende Aktivität .....	10
3.3.3	Zuordnung der Antragsdaten zur Prozessvariablen .....	11
3.3.4	Weitere Modellierung.....	12
3.3.5	Das komplette Workflow-Modell .....	13
4	Zusammenfassung und Schlussfolgerungen.....	15

## 1 Einleitung

Moderne *Workflow*- oder *Business Process Management*-Systeme (BPMS) ermöglichen die automatisierte Ausführung von Abläufen auf Grundlage von Modellen. Grafische Modelle erleichtern das Verständnis der implementierten Abläufe auch für Personen, die selbst nicht programmieren können. Die grafische Darstellung eines solchen ausführbaren *Workflow*-Modells sieht auf den ersten Blick sehr ähnlich aus wie das fachliche Modell eines zu unterstützenden Geschäftsprozesses. Teilweise schlagen die BPMS-Hersteller gar vor, dass die ausführbaren Prozessmodelle direkt von Fachanwendern erstellt werden sollen. Zumindest aber sollte dies die korrekte Umsetzung der fachlichen Anforderungen in die informationstechnische Lösung erleichtern.

An einem Beispiel wird die Umsetzung eines einfachen fachlichen Prozessmodells auf Basis von ereignisgesteuerten Prozessketten (EPK) in ein ausführbares Modell für das System Intalio|BPMS gezeigt. Hierbei stehen nicht nur die Modellierung und Umsetzung der reinen Kontrollflusslogik im Fokus, sondern die komplette Definition aller für die spätere Ausführung des Prozesses erforderlichen Aspekte, wie z. B. die Rollenstruktur, Dialoge, Datenstrukturen und Datentransformationen.

Da es für eine derartige Umsetzung von fachlichen in ausführbare Modelle keine einheitliche Vorgehensweise gibt und diese zudem von den verwendeten Notationen und Systemen abhängt, handelt es sich bei dem dargestellten Vorgehensweise um einen anhand des Beispiels entwickelten ersten Vorschlag, keineswegs um eine fertig entwickelte Methodik.

Das verwendete Intalio|BPMS basiert zu einem großen Teil auf *Open Source*-Software-Komponenten, auch wenn es selbst nicht *Open Source* ist. Verwendet wurde die *Community Edition*, die lizenzkostenfrei erhältlich ist, interessierte Leser können die Vorgehensweise somit selbst ausprobieren. Die fachlichen Modelle dienen lediglich als Ausgangspunkt für die weitere Entwicklung, so dass außer der frei erhältlichen Software von Intalio keine weitere spezielle Software erforderlich ist.

Der vorliegende Artikel ist die Kurzfassung eines umfangreichen Arbeitspapiers, in dem das Vorgehen so ausführlich beschrieben ist, dass alle erforderlichen Schritte im Detail nachvollzogen werden können.

Das ausführliche Arbeitspapier „Vom fachlichen Modell zum ausführbaren Workflow“ sowie die Dateien des komplett ausgearbeiteten Beispiels für den Intalio|BPMS Designer können im Internet unter [http://kurze-prozesse.de/?page\\_id=41](http://kurze-prozesse.de/?page_id=41) kostenlos heruntergeladen werden.

## 2 Fachliche Modellierung

Der unterstützende Kreditvergabeprozess ist in Abbildung 1 als ereignisgesteuerte Prozesskette (EPK) abgebildet.

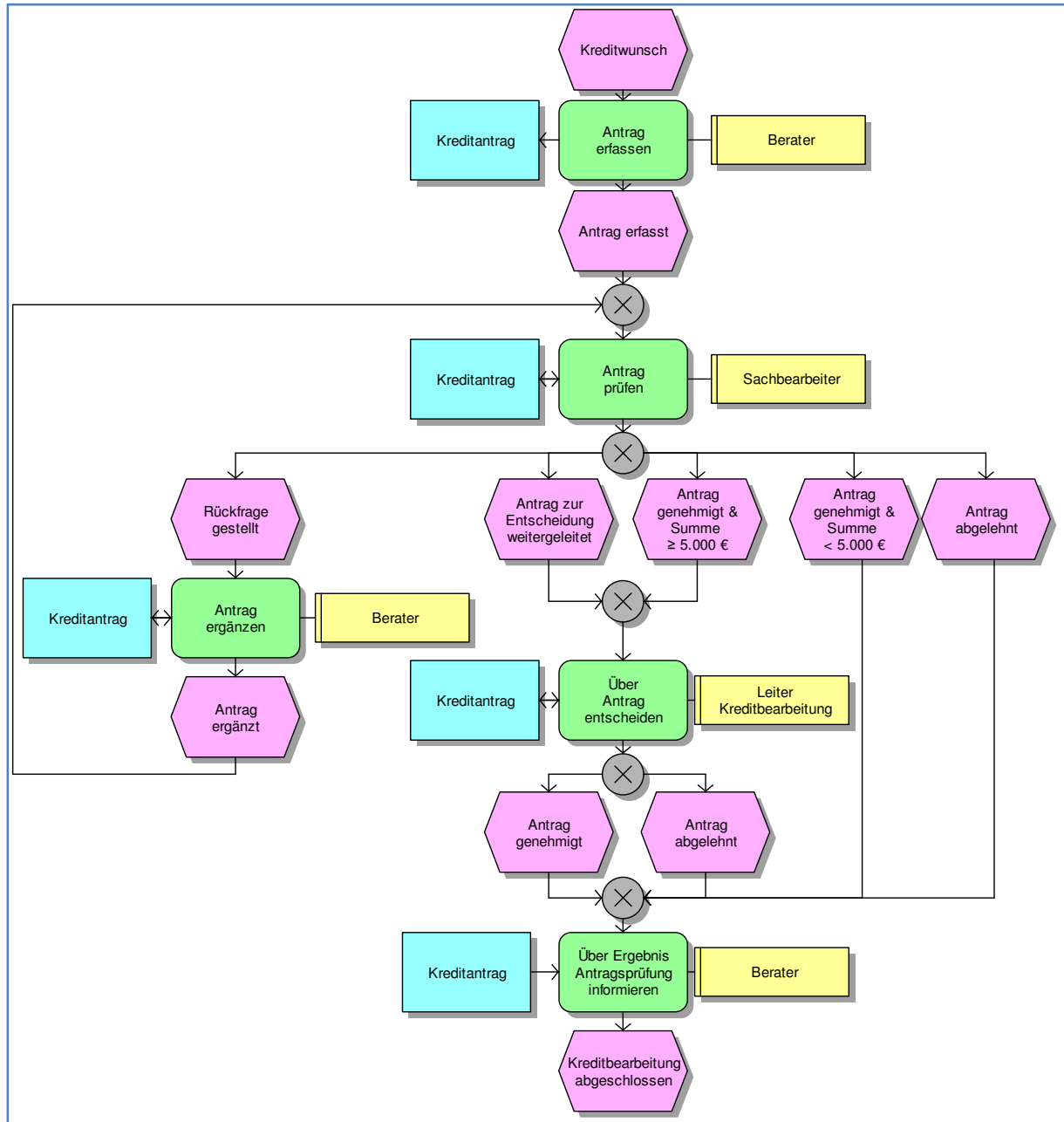


Abbildung 1: Geschäftsprozess Kreditvergabe

Das Modell ist unabhängig von einer konkreten Technologie. So ist beispielsweise nicht festgelegt, ob es sich bei dem Informationsobjekt „Kreditantrag“ um ein Papierdokument oder um einen Datensatz in einer Datenbank handelt. Ebenso wenig ist dargestellt, ob die Weiterleitung des Antrags manuell in Papierform, per E-Mail oder durch ein BPMS erfolgt.

Struktur und Inhalt des Informationsobjektes „Kreditantrag“ sind in Abbildung 2 in Form eines UML Klassendiagramms dargestellt. Der Kreditantrag enthält eine Reihe von Attributen wie Antragsdatum,

Antragssumme usw. Außerdem enthält ein Kreditantrag genau ein Informationsobjekt „Kunde“. Dieses ist hier über eine Komposition als Bestandteil des Kreditantrages dargestellt. Für die einzelnen Attribute wurde außerdem beschrieben, welche Datentypen zu verwenden sind, ob es sich um Mussfelder handelt und welche Restriktionen einzuhalten sind (z. B. dass die Postleitzahl immer fünfstellig sein muss).

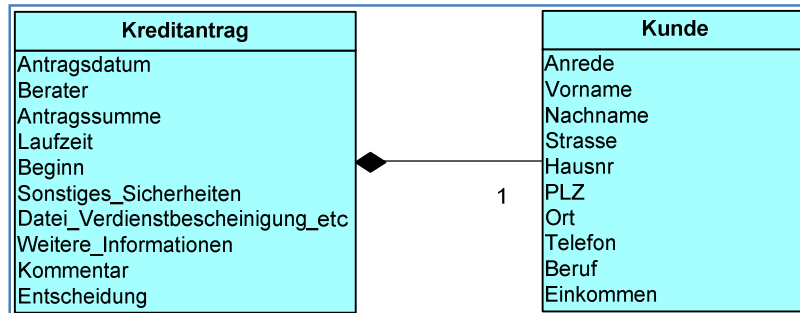


Abbildung 2: Informationsobjekte Kreditantrag und Kunde

Das Rollenmodell und die zugeordneten Mitarbeiter sind in Abbildung 3 dargestellt. Die durch einen horizontalen Pfeil modellierte Beziehung drückt aus, dass die Rolle „Sachbearbeiter“ eine Verallgemeinerung der Rolle „Leiter Kreditbearbeitung“ darstellt. Dies bedeutet, dass ein Leiter der Kreditbearbeitung auch alle Eigenschaften eines Sachbearbeiters besitzt, insbesondere kann er auch die der Rolle Sachbearbeiter zugeordneten Funktionen ausführen.

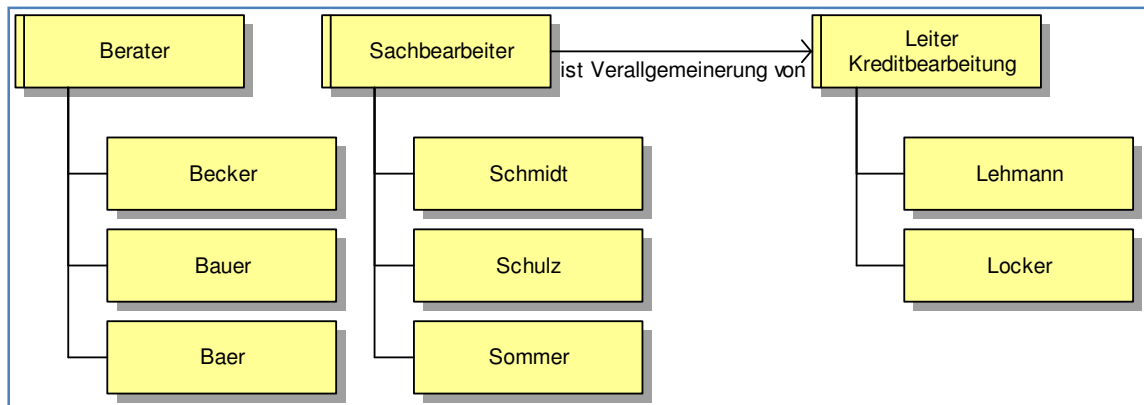


Abbildung 3: Rollenmodell und Rollenbesetzung

## 3 Entwicklung des ausführbaren Workflow-Modells

### 3.1 Anlegen von Rollen und Benutzern

Das Anlegen von Rollen und Benutzern ist in der frei erhältlichen *Community Edition* von Intalio|BPMS etwas spartanisch realisiert. Sämtliche Benutzer werden samt Passwörtern und Rollenzuordnungen einfach in eine XML-Datei auf dem *Server* eingetragen. Für einen produktiven Einsatz wäre dies wenig geeignet. Im laufenden Betrieb ist es erforderlich, Benutzer über eine entsprechende Oberfläche verwalten zu können. Ebenso sollten Benutzer ihrer eigenen Daten und Passwörter pflegen können. Das Ablegen unverschlüsselter Passwörter in einer zentralen Datei ist zudem unter Sicherheitsaspekten abzulehnen.

Für den produktiven Einsatz bietet Intalio u. a. einen kostenpflichtigen *LDAP-Connector* an, über den das BPMS mittels *LDAP (Lightweight Directory Access Protocol)* an eine zentral genutzte Benutzerverwaltung angeschlossen werden kann. Für Test- und Entwicklungszwecke wie in dem vorliegenden Beispiel kann die Datei-basierte Benutzerverwaltung verwendet werden.

Listing 1 zeigt Ausschnitte aus der XML-Datei mit der Definition zweier Rollen und dem Eintrag eines Benutzers.

```
<realm identifier="kreditbank">
  <role identifier="leiter-kredit">
    <description>Leiter Kreditbearbeitung</description>
    <descendantRole>sachbearbeiter</descendantRole>
  </role>

  <role identifier="sachbearbeiter">
    <description>Sachbearbeiter</description>
  </role>
</realm>

...
<user identifier="lehmann">
  <name>Lorenz Lehmann</name>
  <email>lorenz.lehmann@examples.com</email>
  <password>password</password>
  <assignRole>leiter-kredit</assignRole>
</user>
```

Listing 1: Definition von Rollen und Benutzern

### 3.2 Festlegen der Datenstrukturen

Zur Umsetzung der Datenstrukturen sind geeignete Datentypen für den Kreditantrag und den Kunden gemäß Abbildung 2 zu spezifizieren. Anschließend wird eine Prozessvariable vom Typ „Kreditantrag“ angelegt, die während des Prozesses die Daten des Antrags aufnimmt.

Zur Definition von Datenstrukturen wird eine XML Schema-Datei angelegt. Die Inhalte werden mit Hilfe des im Designer enthaltenen grafischen *XML Schema Editors* gepflegt. Die Darstellung der

Struktur und Zusammenhänge der Datentypen im XML Schema Editor wird in Abbildung 4 gezeigt. Hier finden sich die Attribute aus Abbildung 2 wieder.

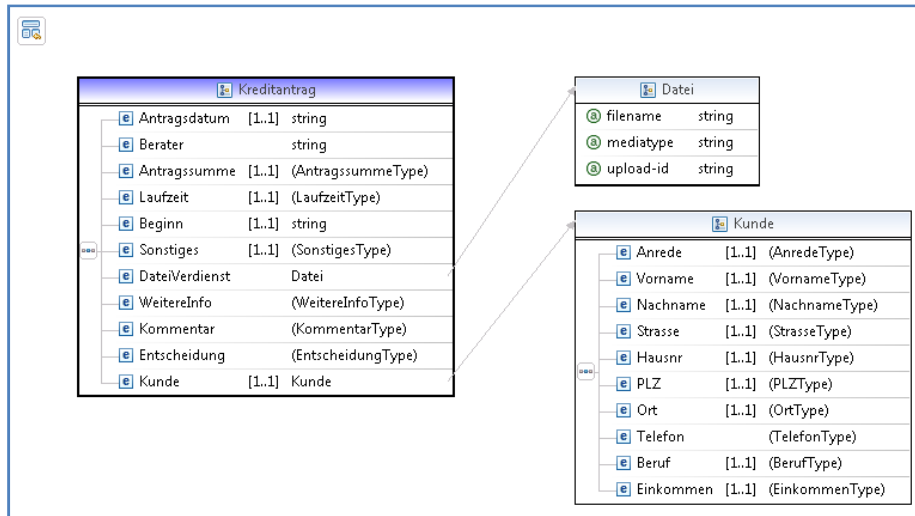


Abbildung 4: Struktur und Zusammenhang der Datentypen

Für jedes Attribut werden Datentypen, erlaubte Feldlängen und einzuhaltende Restriktionen eingegeben. Im Gegensatz zum fachlichen Modell ist nun eine formale Beschreibung erforderlich. Z. T. können diese über entsprechende Masken erfasst werden, wie in Abbildung 5 gezeigt, z. T. ist gar eine direkte Eingabe im XML Code erforderlich. Komplexe Restriktionen, wie z. B. der Aufbau einer gültigen Telefonnummer mit Angabe der Vorwahl, werden mit Hilfe von regulären Ausdrücken definiert. Hierzu sind jeweils fundierte XML-Kenntnisse erforderlich.

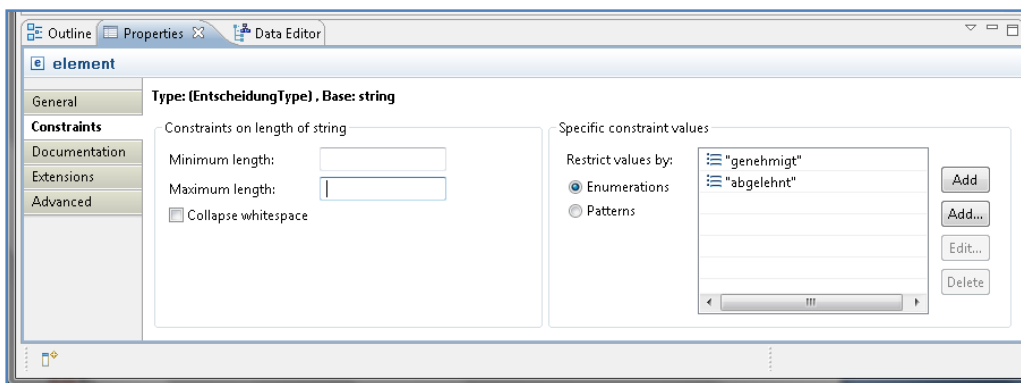


Abbildung 5: Eigenschaften des Attributs „Entscheidung“

Schließlich wird eine Variable „Antragsdaten“ mit der gerade definierten Datenstruktur „Kreditantrag“ definiert, die die entsprechenden Werte für jeden Prozess aufnehmen wird.

### 3.3 Kontrollfluss und Benutzerdialoge

Das Prozessmodell wird in der Modellierungskomponente von Intalio|BPMS erstellt, dem BPMS Designer. Die Modellierung erfolgt grafisch in der Business Process Modeling Notation (BPMN). Abbildung 6 zeigt den BPMS Designer mit einem Modell, in dem zunächst vier Pools angelegt

wurden. Den oberen drei *Pools* werden die oben definierten Rollen zugeordnet, der vierte *Pool* steht für das System selbst.

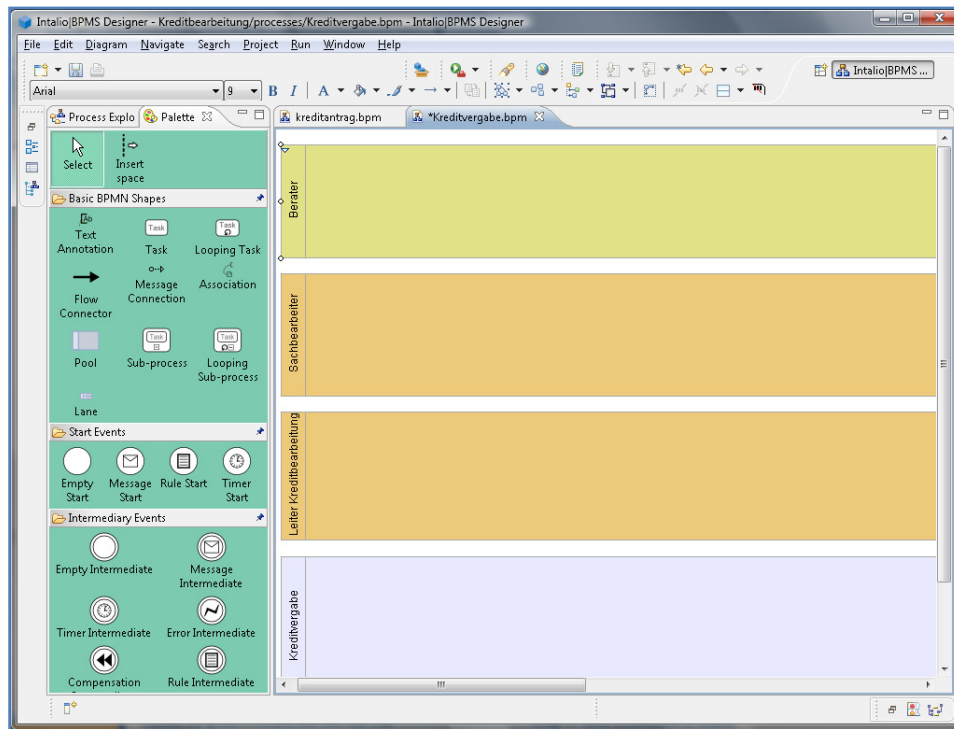


Abbildung 6: Pools im BPMS Designer

Der eigentliche Kontrollfluss wird nun grafisch modelliert. In den Kontrollfluss ist eine Reihe von Benutzer-Aktionen eingebunden. Für diese müssen jeweils Benutzerdialoge bereitgestellt werden. Der Kontrollfluss wird schrittweise entwickelt, wobei jeweils die benötigten Benutzerdialoge entworfen und eingebunden werden. Grundlage für die Entwicklung des Kontrollflusses ist die EPK aus Abbildung 1.

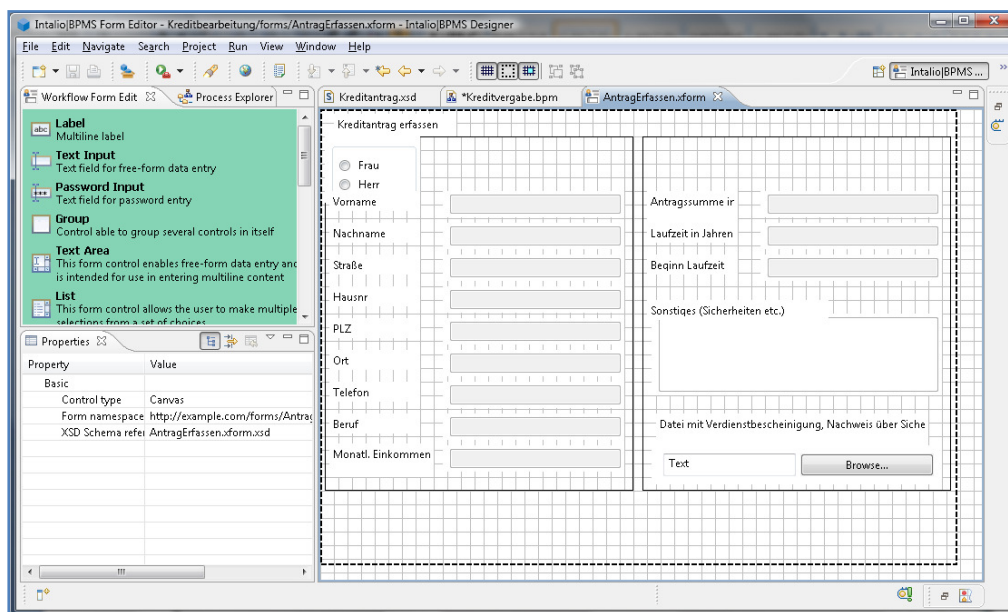


Abbildung 7: Der Form-Editor

### 3.3.1 Dialog Antrag erfassen

Liegt ein Kreditwunsch vor, beginnt der Ablauf mit dem Erfassen des Antrags. Hierzu wird zunächst der Benutzerdialog entwickelt. Dies erfolgt mit Hilfe des im *BPM Designer* enthaltenen *Form Editors* (Abbildung 7). Hier lassen sich die gewünschten Dialog-Elemente mit der Maus in den Dialog-Entwurf ziehen und mit weiteren Angaben versehen.

Die vom Benutzer zu erfassenden Werte sollen bereits bei der Eingabe überprüft werden. Hierzu sind für jedes Eingabefeld wiederum Datentypen, Feldlängen und einzuhaltende Restriktionen anzugeben, wie dies bereits bei der Festlegung der Datenstrukturen beschrieben wurde.

### 3.3.2 Integration des Dialogs in den Workflow als auslösende Aktivität

Der erstellte Dialog kann direkt mit der Maus in den *Pool* der betreffenden Rolle gezogen werden (Abbildung 8). Hierdurch wird automatisch eine mit dem Dialog verbundene Aktivität angelegt, die zur weiteren Modellierung verwendet werden kann.

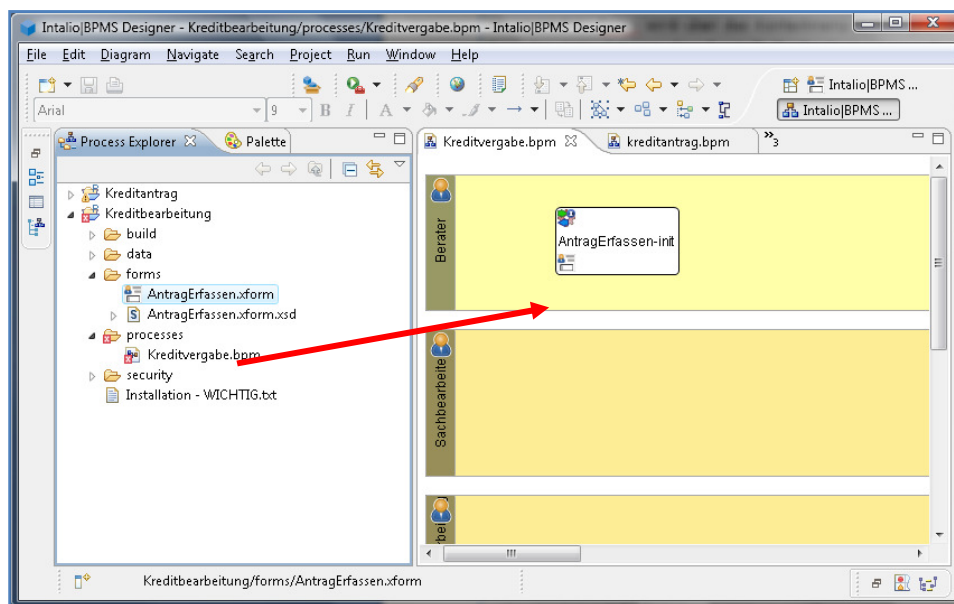


Abbildung 8: Zuordnen des Dialogs zum Pool

Um den Dialog im Workflow verarbeiten zu können, muss das System das Ergebnis der vom Berater durchgeführten Aktivität entgegennehmen. Hierzu wird im System-Pool „Kreditverarbeitung“ eine weitere Aktivität angelegt. Dieser wird mittels Nachrichtenverbindungen mit „AntragErfassen-init“ verbunden (Abbildung 9).

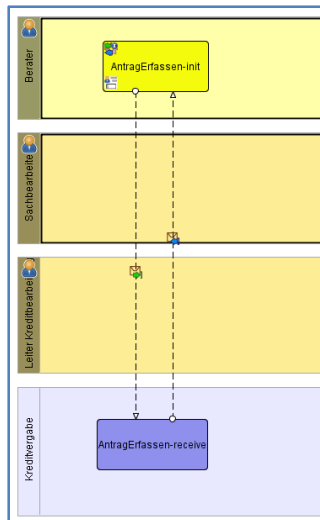


Abbildung 9: Nachrichtenfluss zwischen Dialog-Aktivität und System-Aktivität

An dieser Stelle kann der *Workflow* bereits testweise auf den *Server deployed* werden, um die Erfassung der Antragsdaten zu testen.

### 3.3.3 Zuordnung der Antragsdaten zur Prozessvariablen

Für die weitere Verwendung der eingegebenen Daten im Prozess sollen diese nun in der Prozessvariablen „Antragsdaten“ gespeichert werden. Der *BPM Designer* bietet hierfür ein Tool, den sogenannten Mapper. Wie Abbildung 10 zeigt, werden hierin die vorher angelegten Eingabefelder des Dialogs (links) mit den zugehörigen Datenfeldern der Variablen (rechts) verbunden.

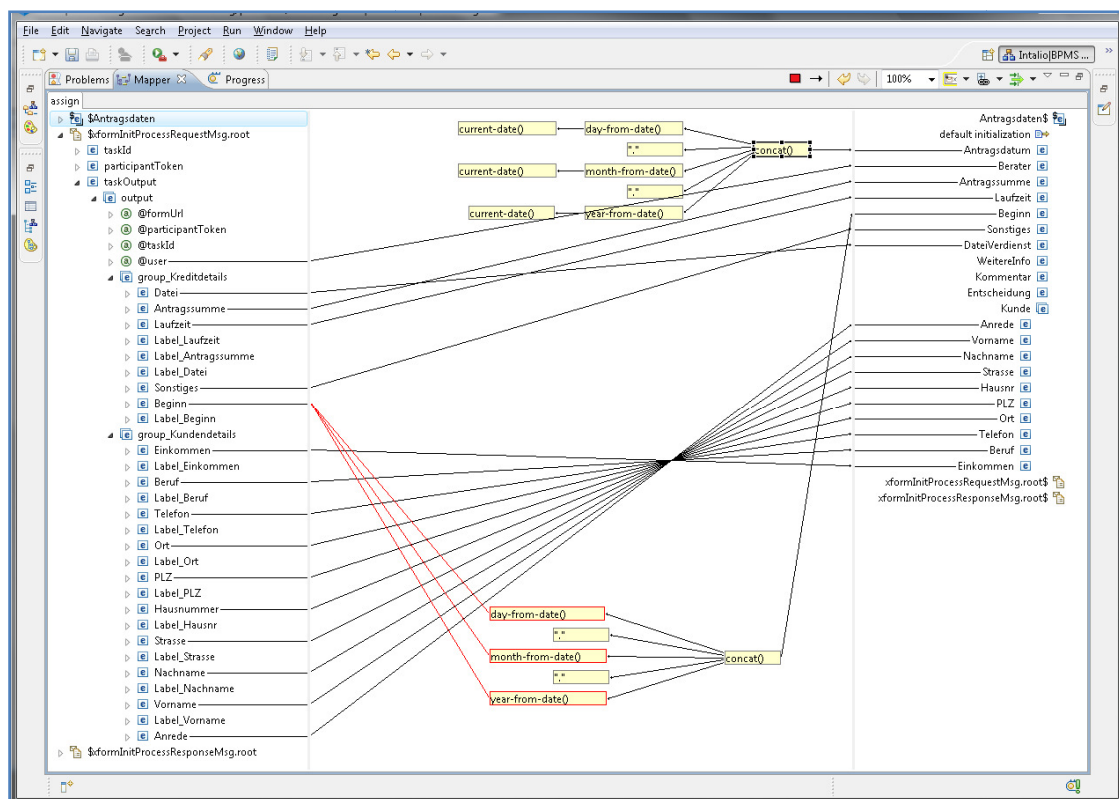


Abbildung 10: Zuordnung von Eingabewerten zu Variablenfeldern

Neben diesen direkten Zuordnungen, bei denen einfach Werte unverändert übernommen werden, lassen sich auch Transformationen und Berechnungen im *Mapper* spezifizieren. Im Beispiel mussten etwa Datumsangaben für eine spätere korrekte Darstellung konvertiert werden.

### 3.3.4 Weitere Modellierung

Die weitere Entwicklung des *Workflow*-Modells erfolgte inkrementell: Es wurde jeweils ein neuer Dialog entworfen und in das Modell integriert, anschließend wurden die notwendigen Daten-Transformationen (*Mappings*) erstellt. Sobald eine in sich funktionsfähige Version des Modells vorlag, wurde sie auf den Server *deployed* und getestet.

Das gesamte Modell wird in Abbildung 11 und Abbildung 12 gezeigt. Hier fallen gegenüber den bisherigen Erläuterungen noch einige Besonderheiten auf. Für die meisten Dialoge wurden zwei Aktivitäten angelegt: eine, die dem Benutzer die Aufgabe in seine Aufgabenliste einfügt, und eine, die seine Eingabedaten verarbeitet.

Die in der EPK durch einen Rücksprung modellierte Schleife (Abbildung 1) kann im BPMS *Designer* nicht in der gleichen Weise modelliert werden. Für die Schleife ist ein eigener Unterprozess erforderlich. Dieser ist in Abbildung 11 und Abbildung 12 als Aktivität zu sehen, die selbst wieder Aktivitäten enthält. Bei den darin enthaltenen Kreisen handelt es sich um Endereignisse, bei deren Erreichen der jeweilige Schleifendurchlauf beendet wird.

Als weitere Konstrukte sind als Rauten dargestellte *Gateways* vorhanden. Mit einem „X“ versehen symbolisieren sie exklusive Verzweigungen. Für die verschiedenen Pfade sind Bedingungen zu formulieren, unter denen sie ausgewählt werden. Auch für die Schleife muss eine Bedingung für den Schleifenabbruch definiert werden. Die hierfür erforderlichen logischen Ausdrücke werden wiederum mit Hilfe des *Mappers* modelliert. Beispielsweise überprüft eine Bedingung, ob die in der Variablen gespeicherte Antragssumme größer als 5 000 € ist.

Genauere Erläuterungen hierzu finden sich in dem ausführlichen Arbeitspapier „Vom fachlichen Modell zum ausführbaren Workflow“, das unter [http://kurze-prozesse.de/?page\\_id=41](http://kurze-prozesse.de/?page_id=41) kostenlos heruntergeladen werden kann.

### 3.3.5 Das komplette Workflow-Modell

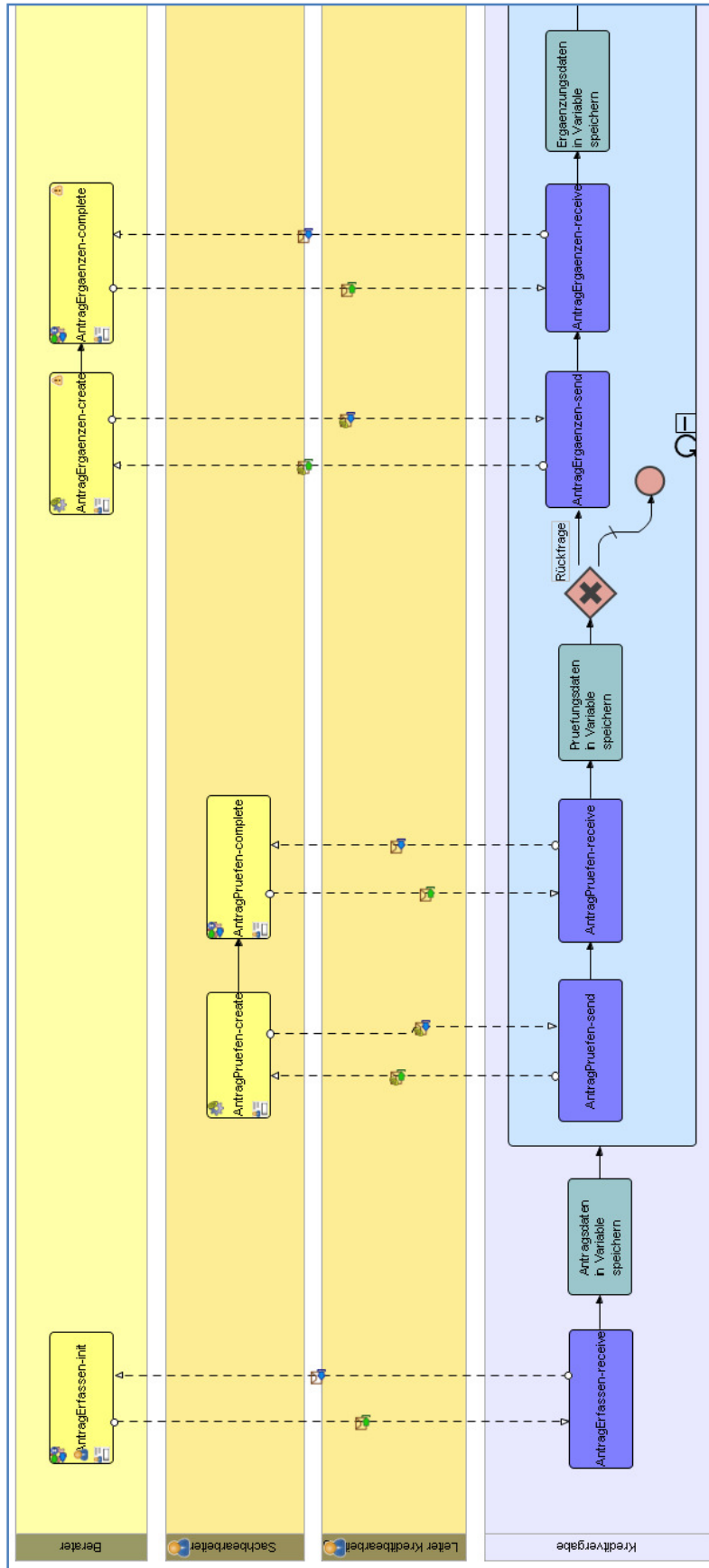


Abbildung 11: Das komplette Workflow-Modell (Teil 1)

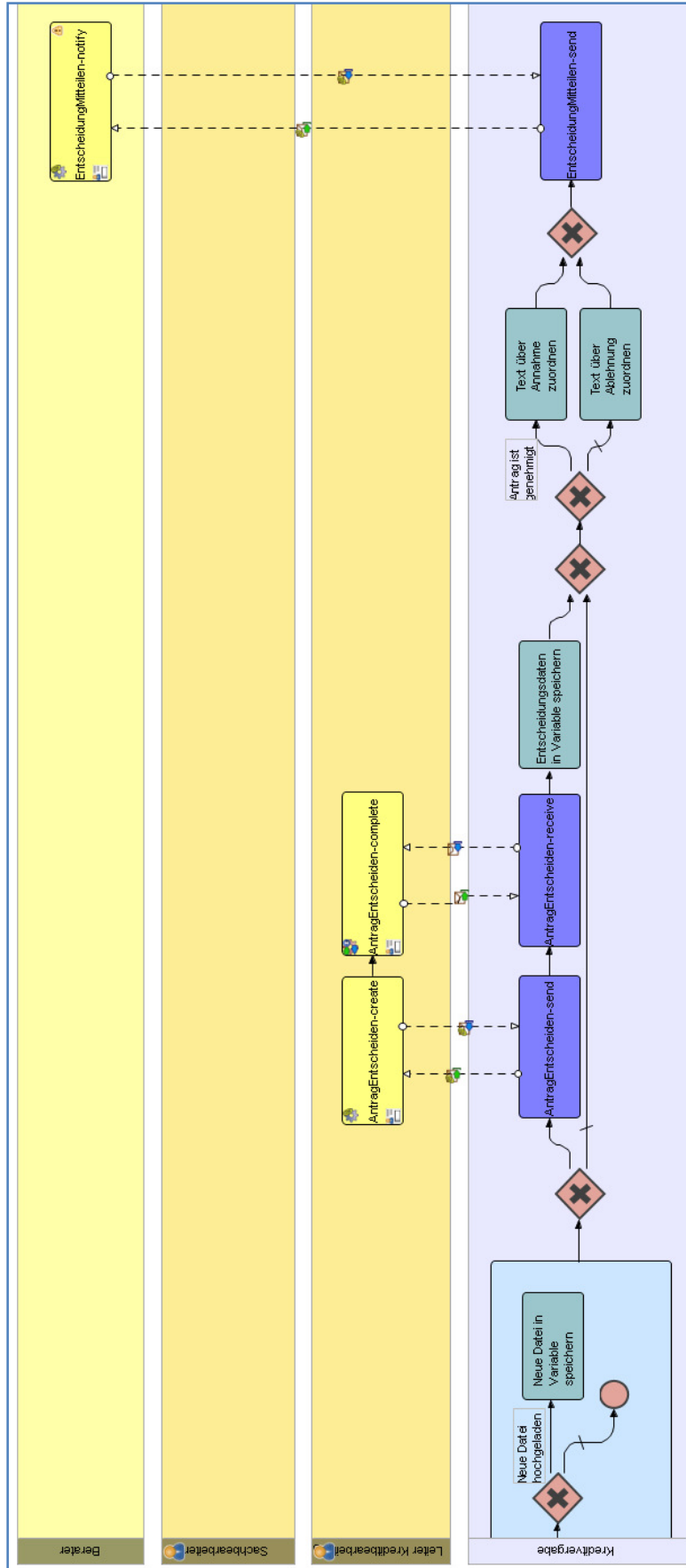


Abbildung 12: Das komplette Workflow-Modell (Teil 2)

## 4 Zusammenfassung und Schlussfolgerungen

Das gewählte Beispiel der Bearbeitung eines Kreditantrages ist einerseits aus Komplexitäts- und Nachvollziehbarkeitsgründen gegenüber einem realen Prozess noch deutlich vereinfacht. Andererseits ist es in der umgesetzten Form wesentlich vollständiger und umfassender als die typischen Demonstrationsbeispiele zum Ausprobieren eines BPMS. So war es erforderlich, eine Reihe von Detailproblemen zu lösen, wie sie in realen Projekten ebenfalls auftreten. Oftmals ist zwar die grundlegende Funktionalität rasch realisiert, aber die Lösung von Problemen und die technische Umsetzung erforderlicher Details erzeugen einen hohen Aufwand. Hierdurch vermittelt das Beispiel zumindest einen Eindruck, wie sich die Entwicklung eines *Workflows* für reale Prozesse gestalten würde.

Die wesentlichen Erfahrungen bei der Entwicklung und Durchführung des beschriebenen Vorgehens unter Verwendung von ARIS-Modellen und Intalio|BPMS lassen sich folgendermaßen zusammenfassen:

- Das Vorgehen erwies sich insgesamt als geeignet. Allerdings überstieg der Zeitaufwand für die Realisierung die ursprünglichen Schätzungen um ein Vielfaches. Hierbei war die Modellierung der Kontrollflusslogik vergleichsweise wenig aufwändig. Die beträchtlichen Aufwände entstanden bei der Erstellung der Dialoge mit den erforderlichen Eingabeüberprüfungen und bei der Definition der *Mappings*, insbesondere bei der Formulierung komplexer Transformationsregeln.
- Die verwendeten Modelle (EPK, Organigramm, Klassendiagramm) bildeten zusammen mit den eingesetzten Attributen eine gute Grundlage für die Entwicklung eines Workflow-Modells. Allerdings konnten nicht alle für die Umsetzung benötigten fachlichen Informationen komplett mit den verwendeten Modelltypen dargestellt werden.
- Die für die fachliche Modellierung angewandten Modellierungskonventionen sollten auf die *Workflow*-Modellierung abgestimmt werden.
- Die inkrementelle Vorgehensweise hat sich bewährt. Hierbei wurde zuerst ein kurzer, aber bereits funktionierender *Workflow* entwickelt, getestet und dann sukzessive weiter entwickelt.
- Die *Workflow*-Modellierung im Intalio|BPMS Designer ist recht intuitiv. Dennoch eignet sich die Modellierungskomponente eher für zumindest IT-affine Modellierer als für reine Fachexperten, da man recht schnell auch mit technischen Details in Berührung kommt.
- Die Umsetzung von Rollenmodell und Datenmodell war recht einfach. Allerdings sind für viele Fragestellungen z. T. recht weitreichende XML-Kenntnisse erforderlich. Letzteres gilt auch für die Datentransformationen (*Mappings*).
- Die Umsetzung der EPK in das BPMN-*Workflow*-Modell war im Gegensatz zu anderen Modellen nicht so direkt möglich. Die Darstellung der Kontrollflusslogik unterscheidet sich grundlegend (z. B. Schleifen, Formulierung von Bedingungen).
- Die Modellierung der Dialoge im *Form Editor* ist ebenfalls sehr intuitiv, allerdings sind die angebotenen Möglichkeiten der Oberflächengestaltung recht eingeschränkt.

Weitere Informationen und Beiträge zum  
Geschäftsprozessmanagement finden Sie unter

**[www.kurze-prozesse.de](http://www.kurze-prozesse.de)**

Das Blog zum Geschäftsprozessmanagement