

RATIONAL USER FORUM 2003
Deutschland – Österreich – Schweiz



FROM ideas TO results



Advanced Use Case Modeling mit Rose, RequisitePro und TestManager

Erfahrungsbericht

Dr. Rudolf Hauber

- OO-Technologie-Berater
- RHauber@ka-muc.de

Kölsch & Altmann

Software & Management Consulting GmbH

- Software & Management Consulting, Projektabwicklung, Schulung
- ca. 40 MA
- Standorte: München, Düsseldorf
- Kunden: Siemens, EADS, Vodafone, AGFA, Bosch, DaimlerChrysler, Datev, ...
- www.ka-muc.de

Inhalt

- Konzept der Use Cases
- Business Object Modell
- Nichtfunktionale Anforderungen
- ReqPro Rose Link
- Testen von Use Cases
- Online Help

Use Cases

Idee des Use Case Konzepts (Ivar Jacobson):

- Anwender **beteiligen**
- Anforderungen **strukturieren**
- Systemgrenzen **festlegen**

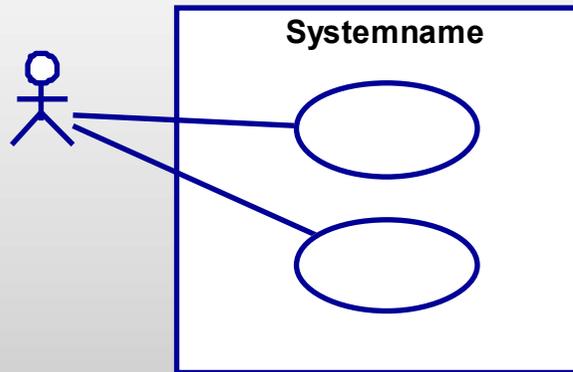
Anwendungsfälle (Use Cases) beschreiben

- aus Sicht der Anwender
- die **funktionalen Anforderungen des Systems**

Use Case-Modell umfaßt Gesamtheit der Use Cases



Use Case Modell



Use Case Documentation

Document Information
<Hierarchy of changes, director views, etc.>

Use Case Name
Bargeld abheben

Actors (Tool supported)
Bankkunde (Hilfsrolle)
Zentralrechner

Priority
1 (hoch)

Goal: K. muss in der Lage sein, Bargeld abzurufen.

Pre-Conditions
• Use Case "Kontostand anfordern" bis zum Ende.

Flow of Events

Basic Path
Der Use Case fungiert als ATM Wahl.
1. Bankkunde wählt Karte aus dem Automaten.
2. Bankkunde gibt Betrag ein.
3. Bankkunde wählt die gewünschte Karte aus.
4. Der Zentralkomputer prüft die Karte.
5. Bankkunde erhält Bargeld.
Karte muss gültig sein und der Betrag muss im Konto sein.

Benefits
Bankkunde hat den gewünschten Betrag abgeholt.

Alternatives/Behaviors
• System kann an der Bruchstelle.

Use Case Documentation

Document Information
<Hierarchy of changes, director views, etc.>

Use Case Name
Bargeld abheben

Actors (Tool supported)
Bankkunde (Hilfsrolle)
Zentralrechner

Priority
1 (hoch)

Goal: K. muss in der Lage sein, Bargeld abzurufen. Anweisung des Kunden wird an den Zentralkomputer weitergegeben.

Pre-Conditions
• Use Case "Kontostand anfordern" bis zum Ende.

Flow of Events

Basic Path
Der Use Case fungiert als ATM Wahl. des Use Case "Kontostand anfordern" aus.
1. Bankkunde wählt Karte aus dem Automaten. "Bitte wählen Sie eine Karte aus."
2. Bankkunde gibt Betrag ein. "Bitte geben Sie den Betrag ein."
3. Der Zentralkomputer prüft die Karte. "Bitte geben Sie den Betrag ein."
4. Der Zentralkomputer prüft die Karte. "Bitte geben Sie den Betrag ein."
5. Bankkunde erhält Bargeld. "Bitte geben Sie den Betrag ein."
Karte muss gültig sein und der Betrag muss im Konto sein.

Benefits
Bankkunde hat den gewünschten Betrag abgeholt.

Alternatives/Behaviors
• System kann an der Bruchstelle.

Bestandteile des Modells:

- Use Case-Diagramme
- Use Case-Beschreibungen
- zusätzliche Informationen



GUI-Skizze

Use Case Beschreibung

Die Beschreibung erfolgt

- in verständlicher Sprache
- präzise und eindeutig
- mit definierten Begriffen (Glossar!)
- nachweisbar
- semi-formal (strukturiert)
- einheitlich

auf Basis eines Templates

Use Case Documentation

Use Case Name

Bargeld abheben.

Actors:

- Bankkunde (Initiator)
- Zentralrechner

Preconditions

- ATM betriebsbereit.

Flow of Events - Basic Path

1. Der Use Case beginnt, wenn der Bankkunde eine Karte am Kartenleser eingibt.
2. Die Karte wird auf Lesbarkeit und Gültigkeit geprüft. ATM fordert zur Eingabe der PIN über Display auf.
3. Bankkunde gibt PIN verborgen am Display ein und bestätigt die Eingabe.
4. Der ATM

Postconditions

- Bankkunde hat Bargeldbetrag erhalten.

Alternative Paths

Stets: Bankkunde bricht ab:

Falls das Geld noch nicht ausgegeben wurde, wird die Transaktion zurückgefahren....

Betrag zu hoch:....

.....

Use Case Template

Seiteneffekte beachten:

- Eigene Formatvorlage für jeden Bereich (ReqPro Import)
- Überschriftenebene und Formate planen (SoDA)
- Feldfunktion für Aufzählung (SoDA)
- Einheitliche Schlüsselwörter (TestManager Export)
- Absatz pro Schritt
- Alternativenbezeichnung
- Sonderzeichen vermeiden

Use Case Name

Bargeld abheben.

Actors:

- Bankkunde (Initiator)
- Zentralrechner

Preconditions

- ATM betriebsbereit.

Flow of Events - Basic Path

1. Der Use Case beginnt, wenn der Bankkunde eine Karte am Kartenleser eingibt.
2. Die Karte wird auf Lesbarkeit und Gültigkeit geprüft. ATM fordert zur Eingabe der PIN über Display auf.
3. Bankkunde gibt PIN verborgen am Display ein und bestätigt die Eingabe.
4. Der ATM

Postconditions

- Bankkunde hat Bargeldbetrag erhalten.

Alternative Paths

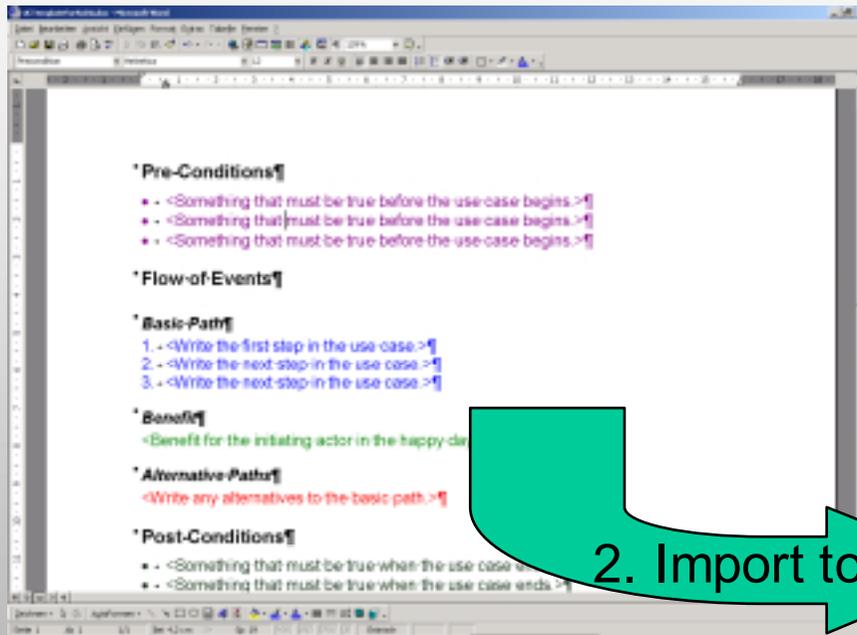
Stets: Bankkunde bricht ab:

Falls das Geld noch nicht ausgegeben wurde, wird die Transaktion zurückgefahren....

Betrag zu hoch:....

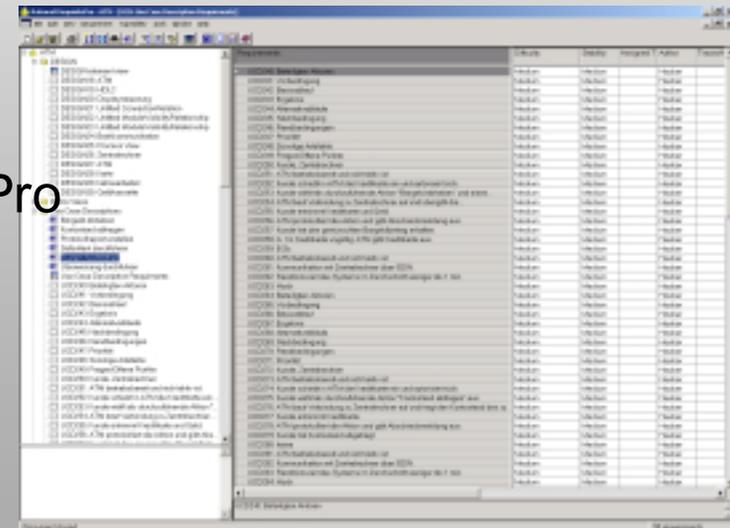
.....

Use Case Beschreibung - Vorgehen



2. Import to ReqPro

3. Change Management unter ReqPro
(History, Discussions)



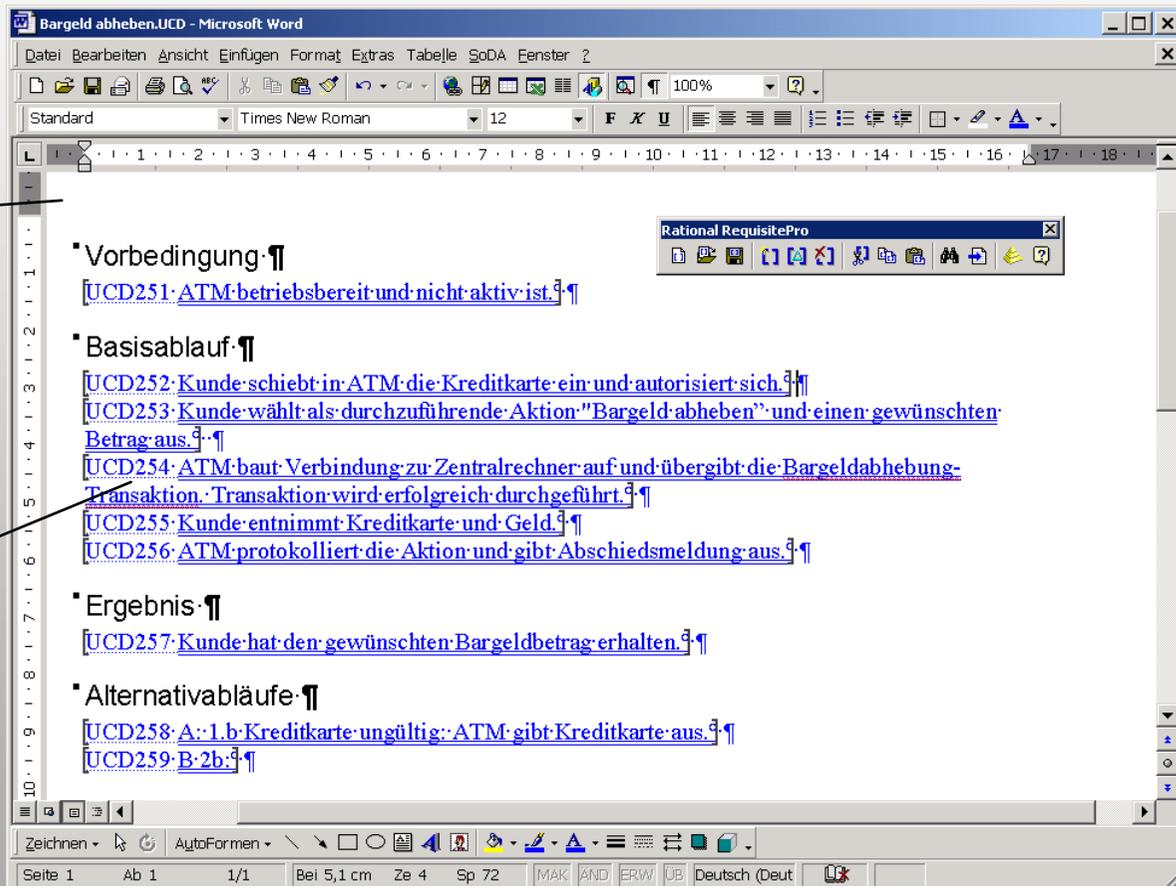
1. Beschreibung mittels Word-Template

Use Case Beschreibung - ReqPro

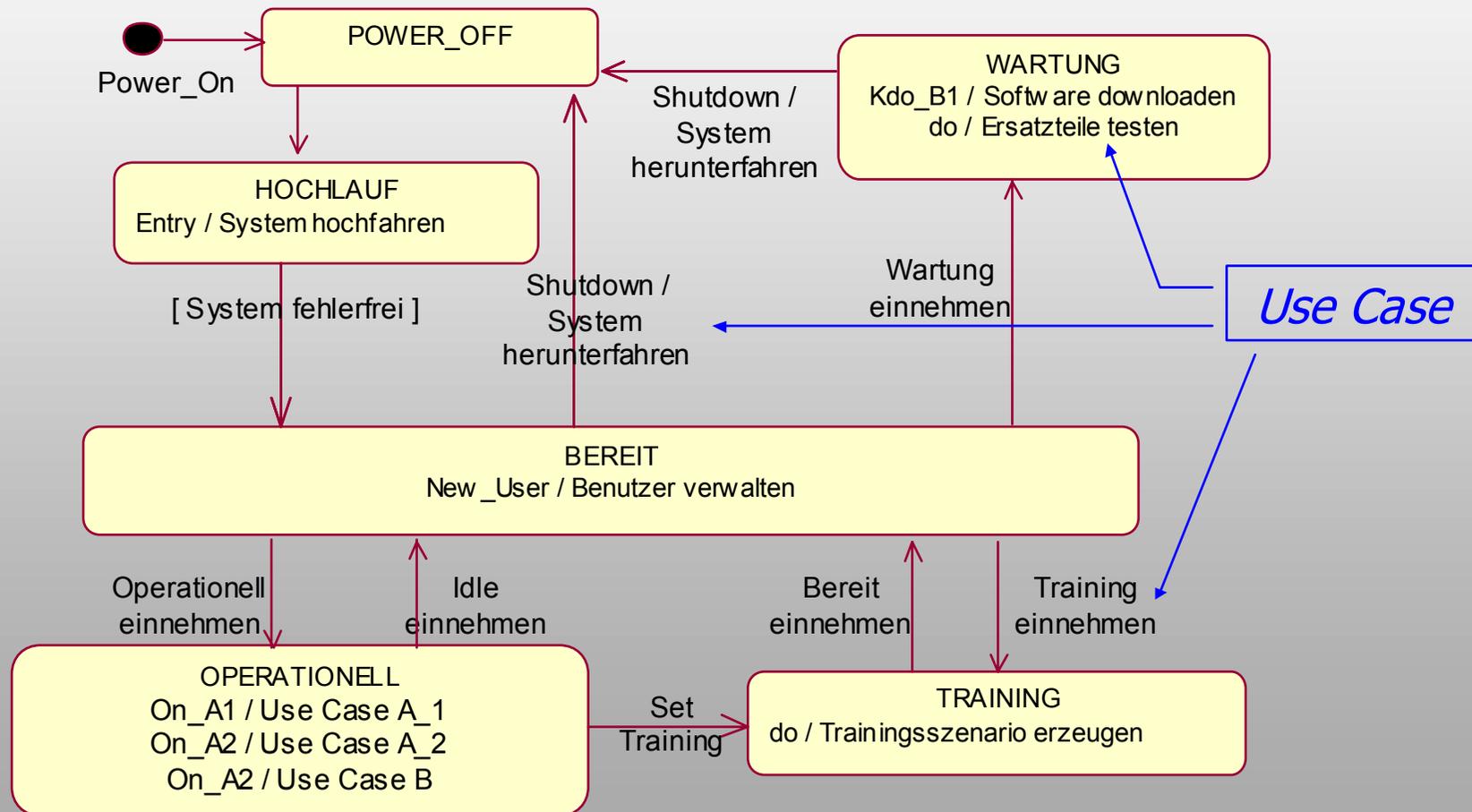
Use Cases
importiert als
"Document and
Requirements"

Beschreibung schrittweise
verwalten:

- Präzise Tracebarkeit
- History und Discussion schrittweise



Zusammenhang von Use Cases



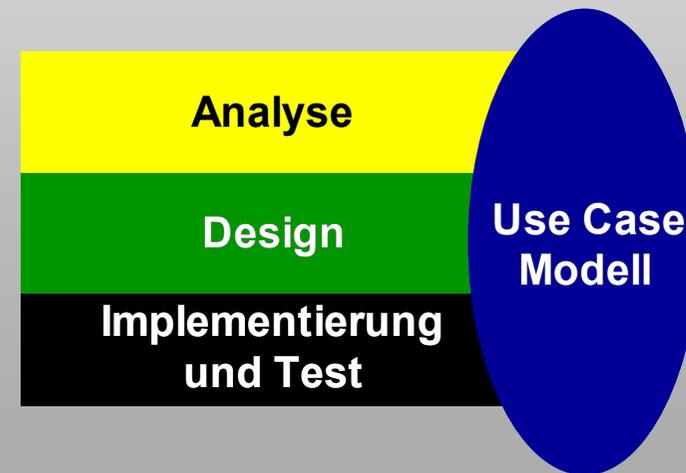
Nutzen des Konzepts

- ✓ Anforderungen sind aus **Anwendersicht** beschrieben
- ✓ **Anwender** ist beteiligt
- ✓ **Vollständigkeit** kann besser beurteilt werden
- ✓ Leichtere **Überprüfbarkeit** durch Anwender
- ✓ **Fortschritt** während der späteren Entwicklung ist meßbar (Use Case basierte Inkrementdefinition)

Einsatz des Use Case-Modells

Use Case-Modell dient zusätzlich als Grundlage für die

- Identifikation von Objekten
- Beschreibung des dynamischen Verhaltens
- Manuale, Dokumentation
- Qualitätsprüfung (Testfälle)
- Projektmanagement

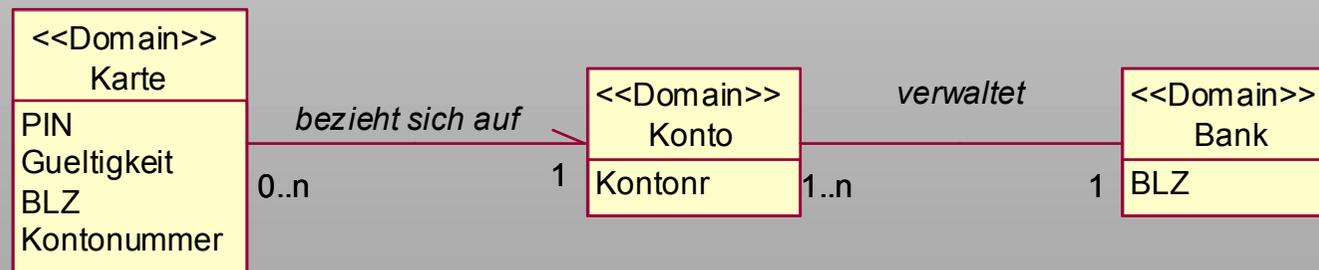


Inhalt

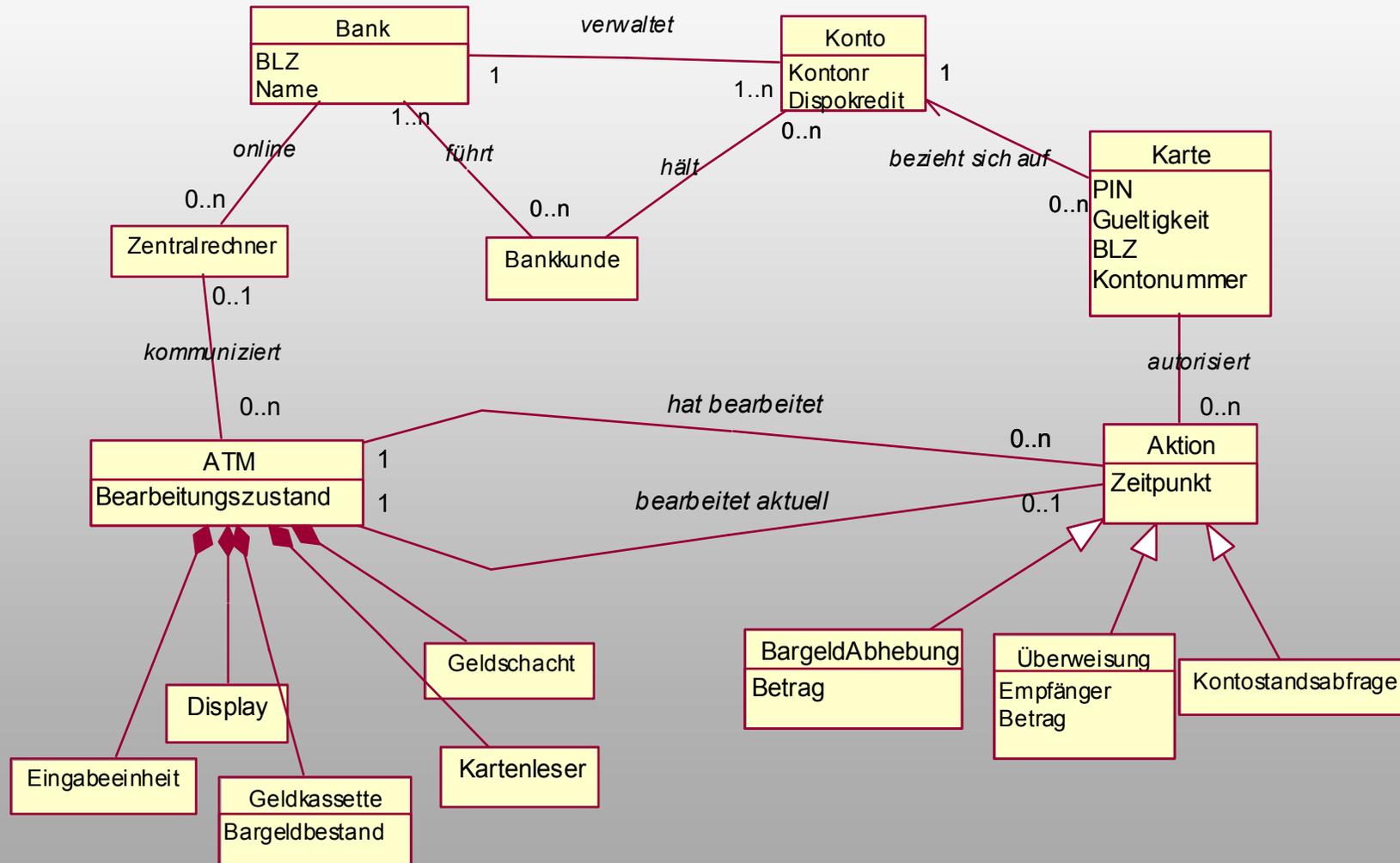
- Konzept der Use Cases
- **Business Object Modell**
- Nichtfunktionale Anforderungen
- ReqPro Rose Link
- Testen von Use Cases
- Online Help

Business Object Modell

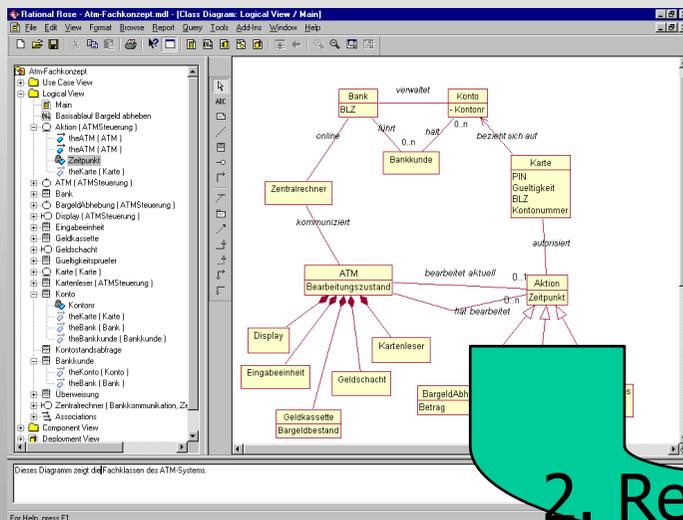
- Business oder Domain Objects (BO) beschreiben die Konzepte des Anwendungsbereichs
- BO Model definiert BOs und deren Beziehungen
- Use-Cases operieren mit BOs.
- Use-Cases Modell und BO Modell ergänzen einander!
- Tracebarkeit von Use-Cases auf BOs.



Beispiel ATM



- BO Modell ist High-Level Glossar!
- Glossar über ReqPro-Rose Link erstellbar
- UML erlaubt Tool-mäßig auswertbare Semantik



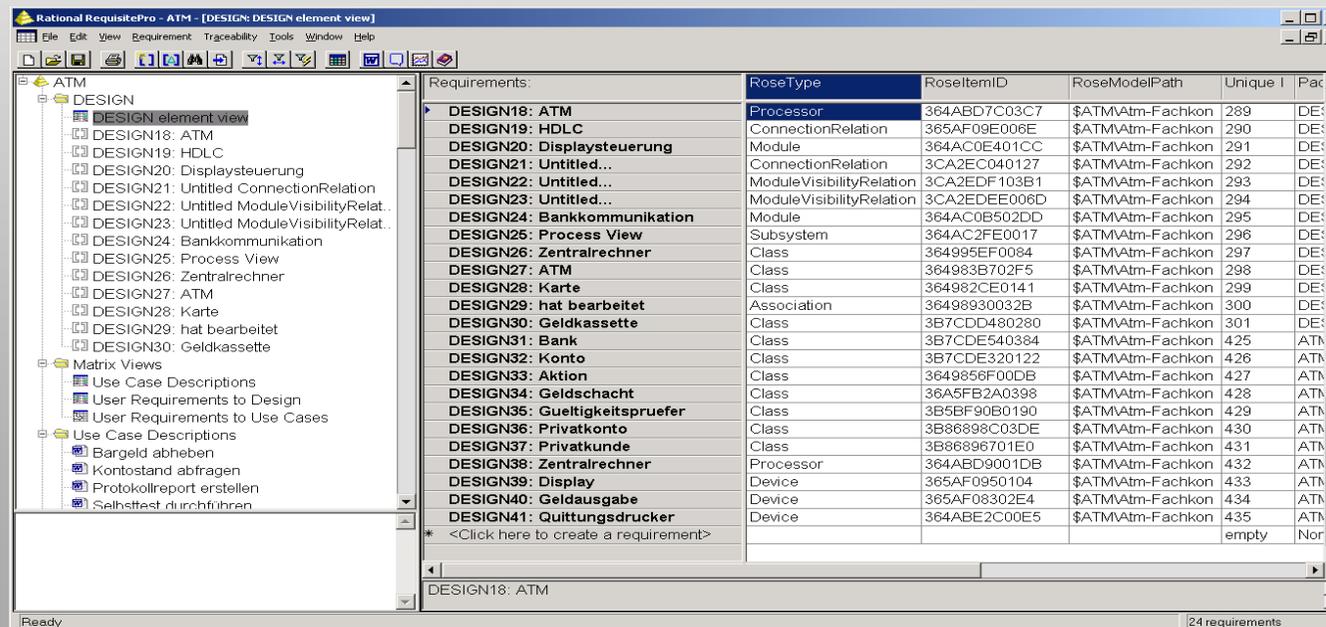
1. Domain Modeling

2. ReqPro Rose Link

3. ReqPro Tracing

Requirement ID	Requirement Name	Requirement Type	Version	Status
DEBS001	ATM	Equipment	1.0	Open
DEBS002	ATM	Equipment	1.0	Open
DEBS003	ATM	Equipment	1.0	Open
DEBS004	ATM	Equipment	1.0	Open
DEBS005	ATM	Equipment	1.0	Open
DEBS006	ATM	Equipment	1.0	Open
DEBS007	ATM	Equipment	1.0	Open
DEBS008	ATM	Equipment	1.0	Open
DEBS009	ATM	Equipment	1.0	Open
DEBS010	ATM	Equipment	1.0	Open
DEBS011	ATM	Equipment	1.0	Open
DEBS012	ATM	Equipment	1.0	Open
DEBS013	ATM	Equipment	1.0	Open
DEBS014	ATM	Equipment	1.0	Open
DEBS015	ATM	Equipment	1.0	Open
DEBS016	ATM	Equipment	1.0	Open
DEBS017	ATM	Equipment	1.0	Open
DEBS018	ATM	Equipment	1.0	Open
DEBS019	ATM	Equipment	1.0	Open
DEBS020	ATM	Equipment	1.0	Open
DEBS021	ATM	Equipment	1.0	Open
DEBS022	ATM	Equipment	1.0	Open
DEBS023	ATM	Equipment	1.0	Open
DEBS024	ATM	Equipment	1.0	Open
DEBS025	ATM	Equipment	1.0	Open
DEBS026	ATM	Equipment	1.0	Open
DEBS027	ATM	Equipment	1.0	Open
DEBS028	ATM	Equipment	1.0	Open
DEBS029	ATM	Equipment	1.0	Open
DEBS030	ATM	Equipment	1.0	Open
DEBS031	ATM	Equipment	1.0	Open
DEBS032	ATM	Equipment	1.0	Open
DEBS033	ATM	Equipment	1.0	Open
DEBS034	ATM	Equipment	1.0	Open
DEBS035	ATM	Equipment	1.0	Open
DEBS036	ATM	Equipment	1.0	Open
DEBS037	ATM	Equipment	1.0	Open
DEBS038	ATM	Equipment	1.0	Open
DEBS039	ATM	Equipment	1.0	Open
DEBS040	ATM	Equipment	1.0	Open
DEBS041	ATM	Equipment	1.0	Open
DEBS042	ATM	Equipment	1.0	Open
DEBS043	ATM	Equipment	1.0	Open
DEBS044	ATM	Equipment	1.0	Open
DEBS045	ATM	Equipment	1.0	Open
DEBS046	ATM	Equipment	1.0	Open
DEBS047	ATM	Equipment	1.0	Open
DEBS048	ATM	Equipment	1.0	Open
DEBS049	ATM	Equipment	1.0	Open
DEBS050	ATM	Equipment	1.0	Open
DEBS051	ATM	Equipment	1.0	Open
DEBS052	ATM	Equipment	1.0	Open
DEBS053	ATM	Equipment	1.0	Open
DEBS054	ATM	Equipment	1.0	Open
DEBS055	ATM	Equipment	1.0	Open
DEBS056	ATM	Equipment	1.0	Open
DEBS057	ATM	Equipment	1.0	Open
DEBS058	ATM	Equipment	1.0	Open
DEBS059	ATM	Equipment	1.0	Open
DEBS060	ATM	Equipment	1.0	Open
DEBS061	ATM	Equipment	1.0	Open
DEBS062	ATM	Equipment	1.0	Open
DEBS063	ATM	Equipment	1.0	Open
DEBS064	ATM	Equipment	1.0	Open
DEBS065	ATM	Equipment	1.0	Open
DEBS066	ATM	Equipment	1.0	Open
DEBS067	ATM	Equipment	1.0	Open
DEBS068	ATM	Equipment	1.0	Open
DEBS069	ATM	Equipment	1.0	Open
DEBS070	ATM	Equipment	1.0	Open
DEBS071	ATM	Equipment	1.0	Open
DEBS072	ATM	Equipment	1.0	Open
DEBS073	ATM	Equipment	1.0	Open
DEBS074	ATM	Equipment	1.0	Open
DEBS075	ATM	Equipment	1.0	Open
DEBS076	ATM	Equipment	1.0	Open
DEBS077	ATM	Equipment	1.0	Open
DEBS078	ATM	Equipment	1.0	Open
DEBS079	ATM	Equipment	1.0	Open
DEBS080	ATM	Equipment	1.0	Open
DEBS081	ATM	Equipment	1.0	Open
DEBS082	ATM	Equipment	1.0	Open
DEBS083	ATM	Equipment	1.0	Open
DEBS084	ATM	Equipment	1.0	Open
DEBS085	ATM	Equipment	1.0	Open
DEBS086	ATM	Equipment	1.0	Open
DEBS087	ATM	Equipment	1.0	Open
DEBS088	ATM	Equipment	1.0	Open
DEBS089	ATM	Equipment	1.0	Open
DEBS090	ATM	Equipment	1.0	Open
DEBS091	ATM	Equipment	1.0	Open
DEBS092	ATM	Equipment	1.0	Open
DEBS093	ATM	Equipment	1.0	Open
DEBS094	ATM	Equipment	1.0	Open
DEBS095	ATM	Equipment	1.0	Open
DEBS096	ATM	Equipment	1.0	Open
DEBS097	ATM	Equipment	1.0	Open
DEBS098	ATM	Equipment	1.0	Open
DEBS099	ATM	Equipment	1.0	Open
DEBS100	ATM	Equipment	1.0	Open

- BOs und Actors als ReqPro Objekte
- evtl. Dokumentation in ReqPro (z.B. bei multilanguage-support)
- Synchronisierung über ReqPro-Rose Link
- History und Discussion

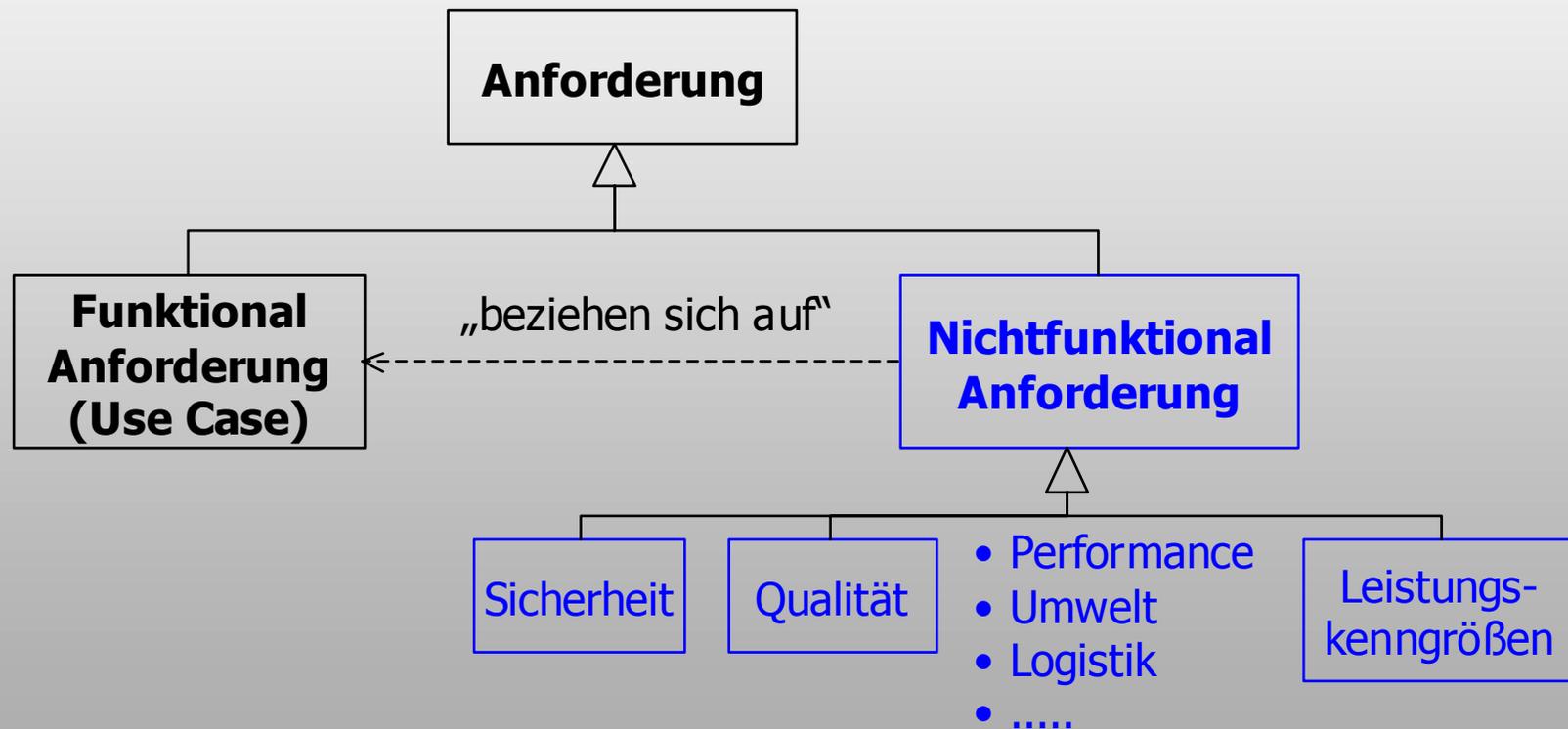


Requirements:	RoseType	RoseItemID	RoseModelPath	Unique I	Pac
DESIGN18: ATM	Processor	364ABD7C03C7	\$ATMAtm-Fachkon	289	DES
DESIGN19: HDLC	ConnectionRelation	365AF09E006E	\$ATMAtm-Fachkon	290	DES
DESIGN20: Displaysteuerung	Module	364AC0E401CC	\$ATMAtm-Fachkon	291	DES
DESIGN21: Untitled...	ConnectionRelation	3CA2EC040127	\$ATMAtm-Fachkon	292	DES
DESIGN22: Untitled...	ModuleVisibilityRelation	3CA2EDF103B1	\$ATMAtm-Fachkon	293	DES
DESIGN23: Untitled...	ModuleVisibilityRelation	3CA2EDEE006D	\$ATMAtm-Fachkon	294	DES
DESIGN24: Bankkommunikation	Module	364AC0B502DD	\$ATMAtm-Fachkon	295	DES
DESIGN25: Process View	Subsystem	364AC2FE0017	\$ATMAtm-Fachkon	296	DES
DESIGN26: Zentralrechner	Class	364995EF0084	\$ATMAtm-Fachkon	297	DES
DESIGN27: ATM	Class	364983B702F5	\$ATMAtm-Fachkon	298	DES
DESIGN28: Karte	Class	364982CE0141	\$ATMAtm-Fachkon	299	DES
DESIGN29: hat bearbeitet	Association	36498930032B	\$ATMAtm-Fachkon	300	DES
DESIGN30: Geldkassette	Class	3B7CDD480280	\$ATMAtm-Fachkon	301	DES
DESIGN31: Bank	Class	3B7CDE540384	\$ATMAtm-Fachkon	425	ATM
DESIGN32: Konto	Class	3B7CDE320122	\$ATMAtm-Fachkon	426	ATM
DESIGN33: Aktion	Class	3649856F00DB	\$ATMAtm-Fachkon	427	ATM
DESIGN34: Geldschacht	Class	36A5FB2A0398	\$ATMAtm-Fachkon	428	ATM
DESIGN35: Gueltigkeitspruefer	Class	3B5BF90B0190	\$ATMAtm-Fachkon	429	ATM
DESIGN36: Privatkonto	Class	3B86898C03DE	\$ATMAtm-Fachkon	430	ATM
DESIGN37: Privatkunde	Class	3B86896701E0	\$ATMAtm-Fachkon	431	ATM
DESIGN38: Zentralrechner	Processor	364ABD9001DB	\$ATMAtm-Fachkon	432	ATM
DESIGN39: Display	Device	365AF0950104	\$ATMAtm-Fachkon	433	ATM
DESIGN40: Geldausgabe	Device	365AF08302E4	\$ATMAtm-Fachkon	434	ATM
DESIGN41: Quittungsdrucker	Device	364ABE2C00E5	\$ATMAtm-Fachkon	435	ATM
* <Click here to create a requirement>				empty	Nor

Inhalt

- Konzept der Use Cases
- Business Object Modell
- **Nichtfunktionale Anforderungen**
- ReqPro Rose Link
- Testen von Use Cases
- Online Help

Anforderungsarten



Nichtfunktionale Anforderungen

Use Cases beschreiben

- aus Sicht der Anwender
- die **funktionalen Anforderungen** des Systems

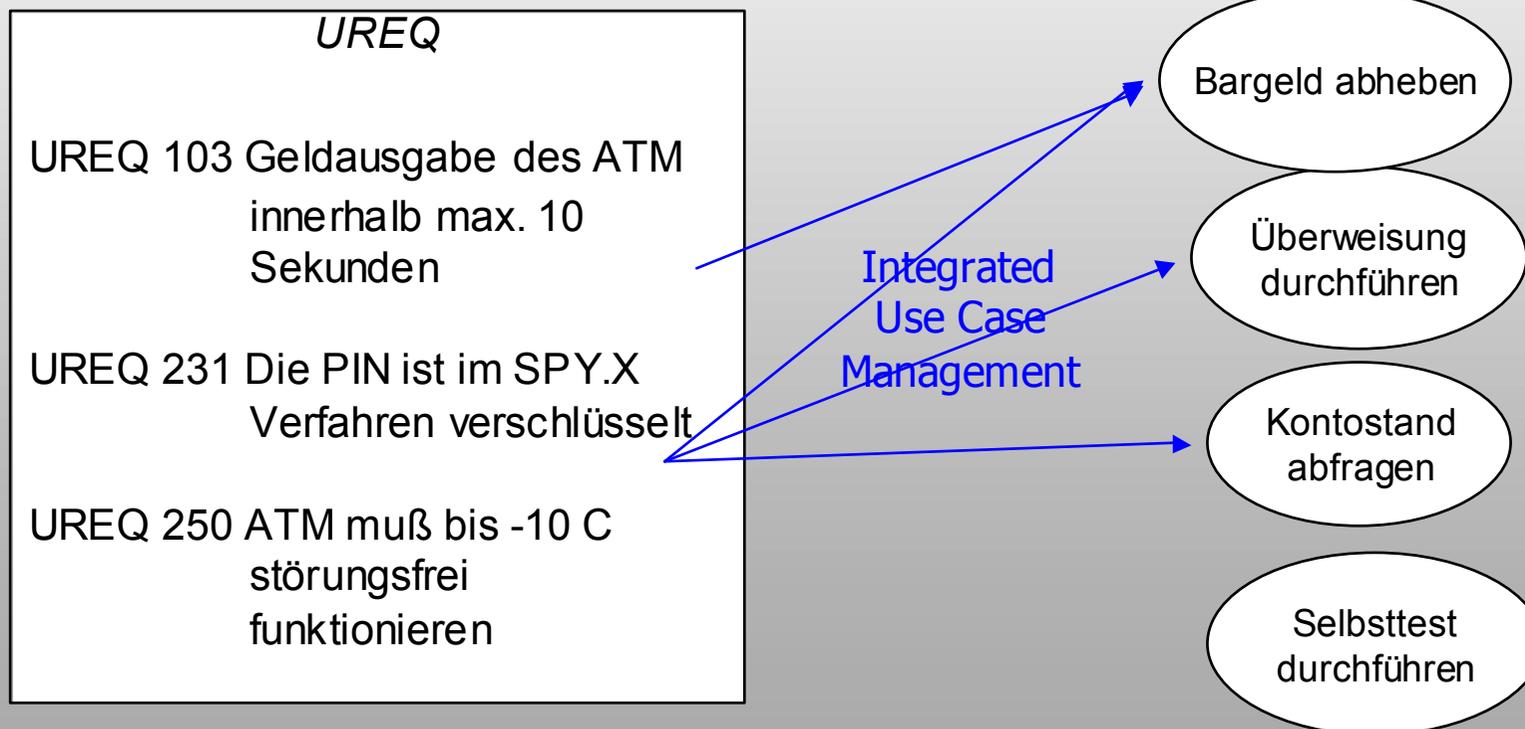
Nichtfunktionale Anforderungen treten hinzu

- Können sich beziehen auf
 - ◆ einzelne Use Cases
 - „Geldausgabe des ATM innerhalb max. 10 Sekunden“
 - ◆ mehrere Use Cases
 - „Die PIN ist im SPY.X Verfahren verschlüsselt“
 - ◆ das System als Ganzes
 - „ATM muß bis -10 C störungsfrei funktionieren“



Nichtfunktionale Anforderungen

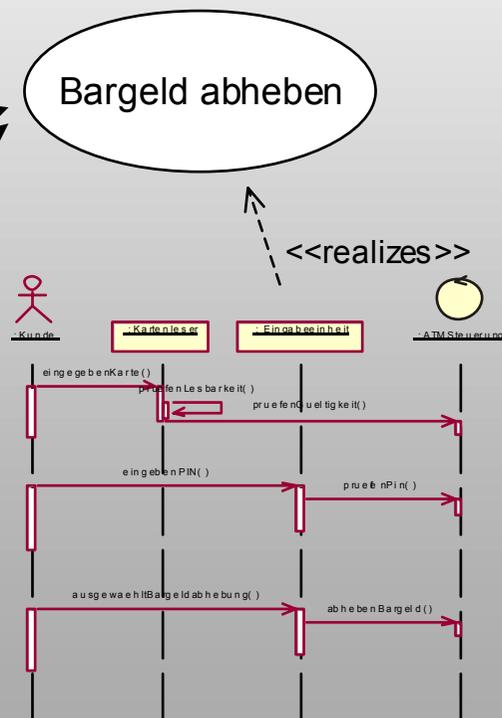
Verlinkung der nichtfunktionale Anforderungen mit Use Cases



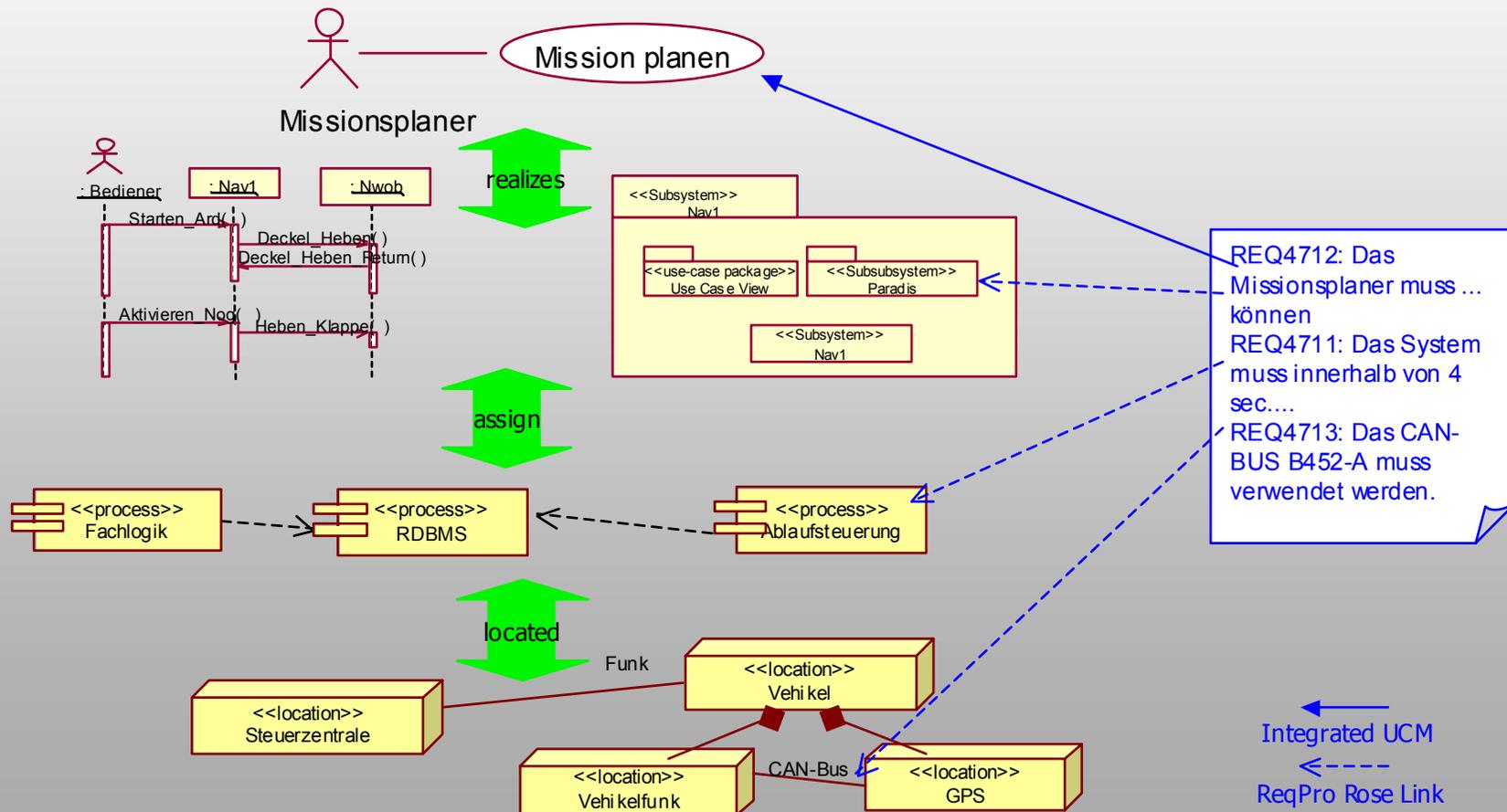
Nichtfunktionale Anforderungen

Tracebarkeit der nichtfunktionale Anforderungen bis auf Code-Ebene dank UML

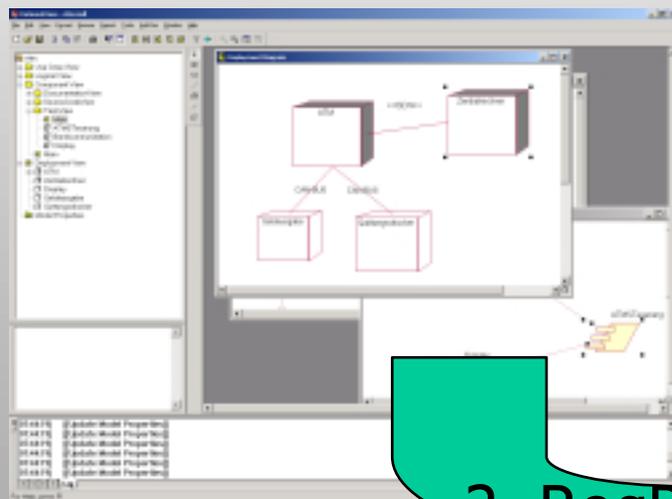
Id	UREQ
UREQ 103	Geldausgabe des ATM innerhalb max. 10 Sekunden
UREQ 231	Die PIN ist im SPY.X Verfahren verschlüsselt
UREQ 250	ATM muß bis -10 C störungsfrei funktionieren



Systemmodellierung und Requirements



- Modellierung weiterer Sichten
 - ◆ Subsysteme
 - ◆ Prozess
 - ◆ Hardware
- Tracing in ReqPro



1. System Modeling

2. ReqPro Rose Link

ReqID	ReqText	ReqType	ReqStatus	ReqModPath
DEFGA00	hat keine Funktion	connection	Not Implemented	SA\Mem.Funktion.de
DEFGA01	Interaktion	Conn	Not Implemented	SA\Mem.Funktion.de
DEFGA02	ATM	Conn	Not Implemented	SA\Mem.Funktion.de
DEFGA03	Konto	Conn	Not Implemented	SA\Mem.Funktion.de
DEFGA04	Geldkonto	Conn	Not Implemented	SA\Mem.Funktion.de
DEFGA05	Bank	Conn	Not Implemented	SA\Mem.Funktion.de
DEFGA06	Konto	Conn	Not Implemented	SA\Mem.Funktion.de
DEFGA07	Aktion	Conn	Not Implemented	SA\Mem.Funktion.de
DEFGA08	Geldkonto	Conn	Not Implemented	SA\Mem.Funktion.de
DEFGA09	Geldkonto	Conn	Not Implemented	SA\Mem.Funktion.de
DEFGA10	Privatkonto	Conn	Not Implemented	SA\Mem.Funktion.de
DEFGA11	Privatkonto	Conn	Not Implemented	SA\Mem.Funktion.de
DEFGA12	Bankangewandter	Conn	Not Implemented	SA\Mem.Funktion.de
DEFGA13	Titel	Conn	Not Implemented	SA\Mem.Funktion.de
DEFGA14	Kunde	Conn	Not Implemented	SA\Mem.Funktion.de
DEFGA15	Interaktion	Conn	Not Implemented	SA\Mem.Funktion.de
DEFGA16	Speicher	Conn	Not Implemented	SA\Mem.Funktion.de
DEFGA17	HOLC	ConnectionPattern	Not Implemented	SA\Mem.Funktion.de
DEFGA18	Unlimited Kommunikation	ConnectionPattern	Not Implemented	SA\Mem.Funktion.de
DEFGA19	Display	Device	Not Implemented	SA\Mem.Funktion.de
DEFGA20	Geldkonto	Device	Not Implemented	SA\Mem.Funktion.de
DEFGA21	Geldkonto	Device	Not Implemented	SA\Mem.Funktion.de
DEFGA22	Speicher	Device	Not Implemented	SA\Mem.Funktion.de
DEFGA23	Bankkommunikation	Node	Not Implemented	SA\Mem.Funktion.de
DEFGA24	Unlimited Modultastigkeit	Node	Not Implemented	SA\Mem.Funktion.de
DEFGA25	Unlimited Modultastigkeit	Node	Not Implemented	SA\Mem.Funktion.de
DEFGA26	ATM	Processor	Not Implemented	SA\Mem.Funktion.de
DEFGA27	Interaktion	Processor	Not Implemented	SA\Mem.Funktion.de
DEFGA28	Prozess	Subsystem	Not Implemented	SA\Mem.Funktion.de

3. ReqPro Tracing

UML Stellvertreter Requirements

Erweitertes ReqPro Tool

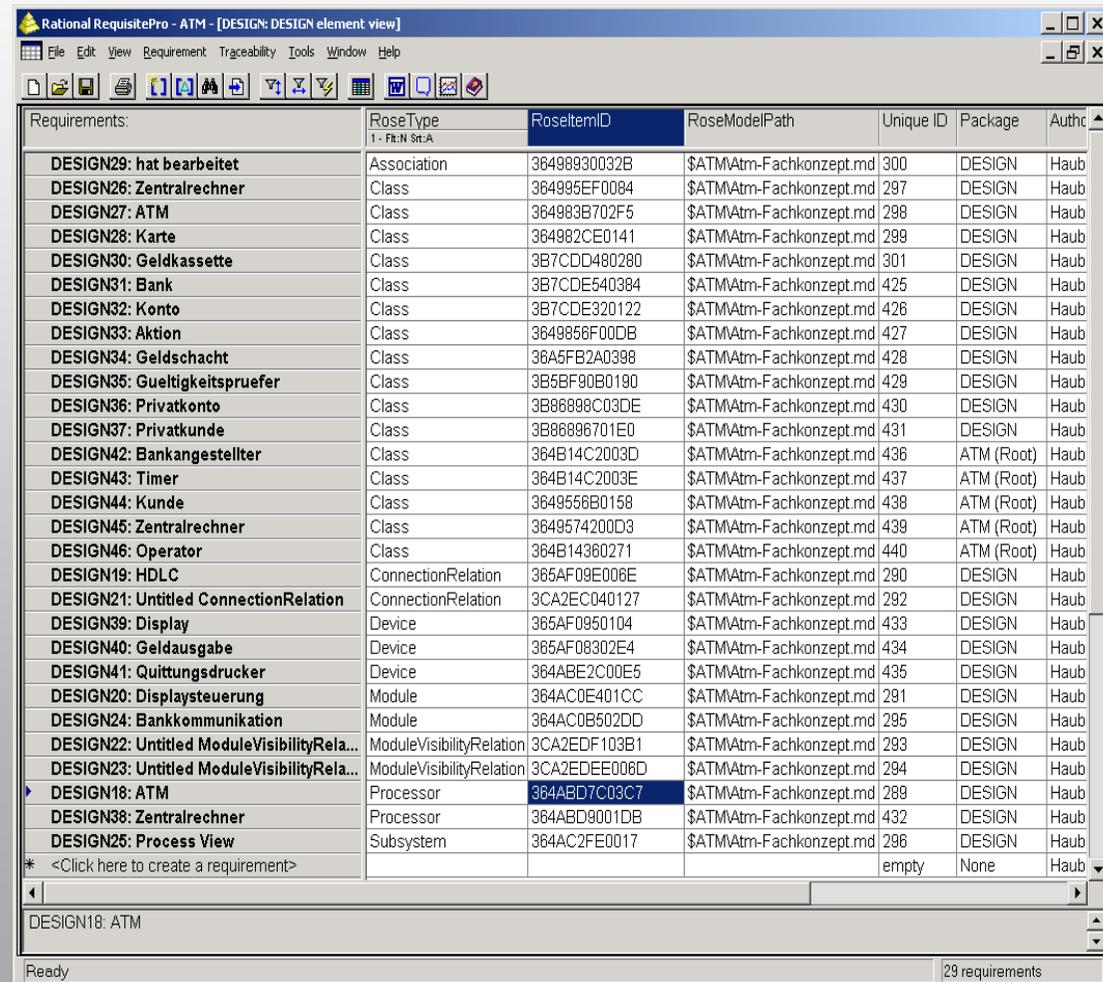
- Sync Traceability
- Access Traceability

Minimalinformation

- Rose-Identifizier
- UML Elementname
- UML Elementtyp

Weitere Informationen

- Stereotype
- Dokumentation
- Parentpackage
- ...



Requirements:	RoseType	RoseItemID	RoseModelPath	Unique ID	Package	Auth
DESIGN29: hat bearbeitet	Association	36498930032B	\$ATMAtm-Fachkonzept.md	300	DESIGN	Haub
DESIGN26: Zentralrechner	Class	364985EF0084	\$ATMAtm-Fachkonzept.md	297	DESIGN	Haub
DESIGN27: ATM	Class	364983B702F5	\$ATMAtm-Fachkonzept.md	298	DESIGN	Haub
DESIGN28: Karte	Class	364982CE0141	\$ATMAtm-Fachkonzept.md	299	DESIGN	Haub
DESIGN30: Geldkassette	Class	3B7CDD480280	\$ATMAtm-Fachkonzept.md	301	DESIGN	Haub
DESIGN31: Bank	Class	3B7CDE540384	\$ATMAtm-Fachkonzept.md	425	DESIGN	Haub
DESIGN32: Konto	Class	3B7CDE320122	\$ATMAtm-Fachkonzept.md	426	DESIGN	Haub
DESIGN33: Aktion	Class	3649856F00DB	\$ATMAtm-Fachkonzept.md	427	DESIGN	Haub
DESIGN34: Geldschacht	Class	36A5FB2A0398	\$ATMAtm-Fachkonzept.md	428	DESIGN	Haub
DESIGN35: Gueltigkeitspruefer	Class	3B5BF90B0190	\$ATMAtm-Fachkonzept.md	429	DESIGN	Haub
DESIGN36: Privatkonto	Class	3B88898C03DE	\$ATMAtm-Fachkonzept.md	430	DESIGN	Haub
DESIGN37: Privatkunde	Class	3B88896701E0	\$ATMAtm-Fachkonzept.md	431	DESIGN	Haub
DESIGN42: Bankangestellter	Class	364B14C2003D	\$ATMAtm-Fachkonzept.md	436	ATM (Root)	Haub
DESIGN43: Timer	Class	364B14C2003E	\$ATMAtm-Fachkonzept.md	437	ATM (Root)	Haub
DESIGN44: Kunde	Class	3649556B0158	\$ATMAtm-Fachkonzept.md	438	ATM (Root)	Haub
DESIGN45: Zentralrechner	Class	3649574200D3	\$ATMAtm-Fachkonzept.md	439	ATM (Root)	Haub
DESIGN46: Operator	Class	364B14360271	\$ATMAtm-Fachkonzept.md	440	ATM (Root)	Haub
DESIGN19: HDLC	ConnectionRelation	365AF09E006E	\$ATMAtm-Fachkonzept.md	290	DESIGN	Haub
DESIGN21: Untitled ConnectionRelation	ConnectionRelation	3CA2EC040127	\$ATMAtm-Fachkonzept.md	292	DESIGN	Haub
DESIGN39: Display	Device	365AF0950104	\$ATMAtm-Fachkonzept.md	433	DESIGN	Haub
DESIGN40: Geldausgabe	Device	365AF08302E4	\$ATMAtm-Fachkonzept.md	434	DESIGN	Haub
DESIGN41: Quittungsdrucker	Device	364ABE2C00E5	\$ATMAtm-Fachkonzept.md	435	DESIGN	Haub
DESIGN20: Displaysteuerung	Module	364AC0E401CC	\$ATMAtm-Fachkonzept.md	291	DESIGN	Haub
DESIGN24: Bankkommunikation	Module	364AC0B502DD	\$ATMAtm-Fachkonzept.md	295	DESIGN	Haub
DESIGN22: Untitled ModuleVisibilityRela...	ModuleVisibilityRelation	3CA2EDF103B1	\$ATMAtm-Fachkonzept.md	293	DESIGN	Haub
DESIGN23: Untitled ModuleVisibilityRela...	ModuleVisibilityRelation	3CA2EDEE008D	\$ATMAtm-Fachkonzept.md	294	DESIGN	Haub
DESIGN18: ATM	Processor	364ABD7C03C7	\$ATMAtm-Fachkonzept.md	289	DESIGN	Haub
DESIGN38: Zentralrechner	Processor	364ABD9001DB	\$ATMAtm-Fachkonzept.md	432	DESIGN	Haub
DESIGN25: Process View	Subsystem	364AC2FE0017	\$ATMAtm-Fachkonzept.md	296	DESIGN	Haub
* <Click here to create a requirement>				empty	None	Haub

Inhalt

- Konzept der Use Cases
- Business Object Modell
- Nichtfunktionale Anforderungen
- ReqPro Rose Link
- **Testen von Use Cases**
- Online Help

Use Case basierte Testcases

Use Case Test Bargeld abheben - Basisablauf

Vorbedingung

- **ATM ist in Idle-Zustand.**

Basisablauf

- **Bankkunde gibt gültige Karte in Kartenleser ein.**
- **Die Karte wird auf Lesbarkeit und Gültigkeit geprüft. ATM fordert zur Eingabe der PIN über Display auf.**
- **Bankkunde gibt PIN 4711 verborgen am Display ein und bestätigt die Eingabe.**
- **Die Gültigkeit der PIN wird geprüft. ATM fordert zur Eingabe der auszuführenden Aktion über Display auf.**
- **Kunde wählt als durchzuführende Aktion "Bargeld abheben"**
- **ATM fordert zur Eingabe des abzuhebenden Betrags über Display auf.**
- **Kunde wählt als gewünschten Betrag 300 DM aus.**
- **ATM baut Verbindung zu Zentralrechner auf und übergibt die Bargeldabhebung-Transaktion.**
- **Transaktion wird erfolgreich durchgeführt.**
-

Ergebnis

- **Kunde hat den gewünschten Bargeldbetrag erhalten.**
- **ATM ist in Idle-Zustand.**

Testen von Use Cases



Use Case Documentation

Documentation for: Informant
 ID: Informant; Name: Informant; Role: Informant; ...

Use Case Name:
 Bargeld abheben

Actor (User/period):
 Informant

Priority:
 1

Preconditions:
 Informant is logged in and has a valid PIN.

Postconditions:
 Informant has received cash.

Flow of Steps:

Start/End:
 Start: Informant requests cash.
 End: Informant receives cash.

Flow:
 1. Informant enters PIN.
 2. System checks PIN.
 3. System dispenses cash.
 4. Informant receives cash.

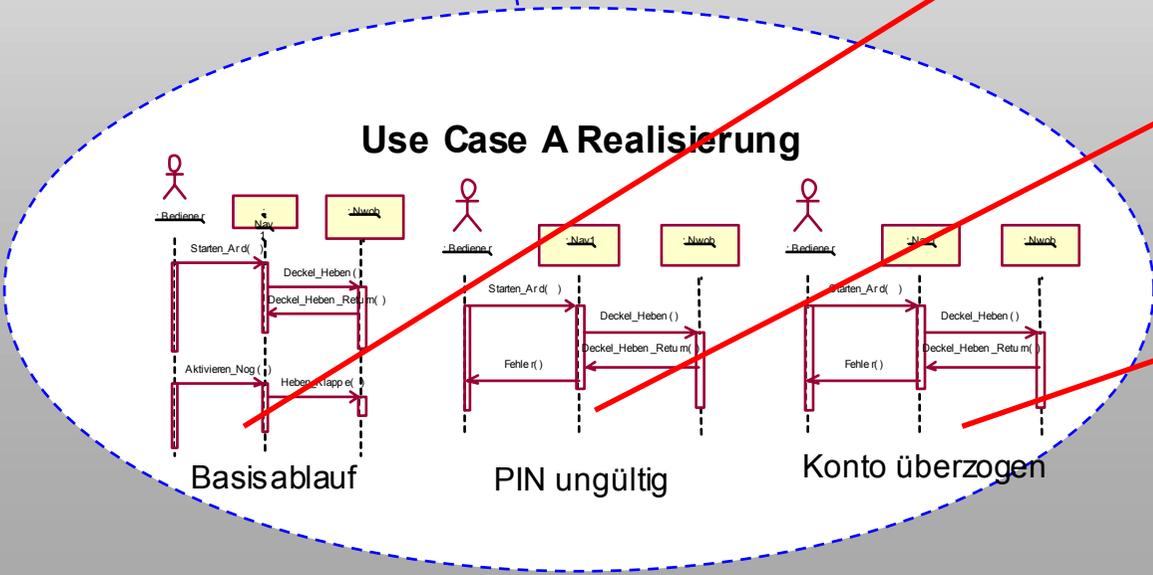
Exceptions:
 - Invalid PIN: System displays error message.

Notes:
 - Informant must be logged in.

<<Test Case>>
 Bargeld abheben::
 Basisablauf

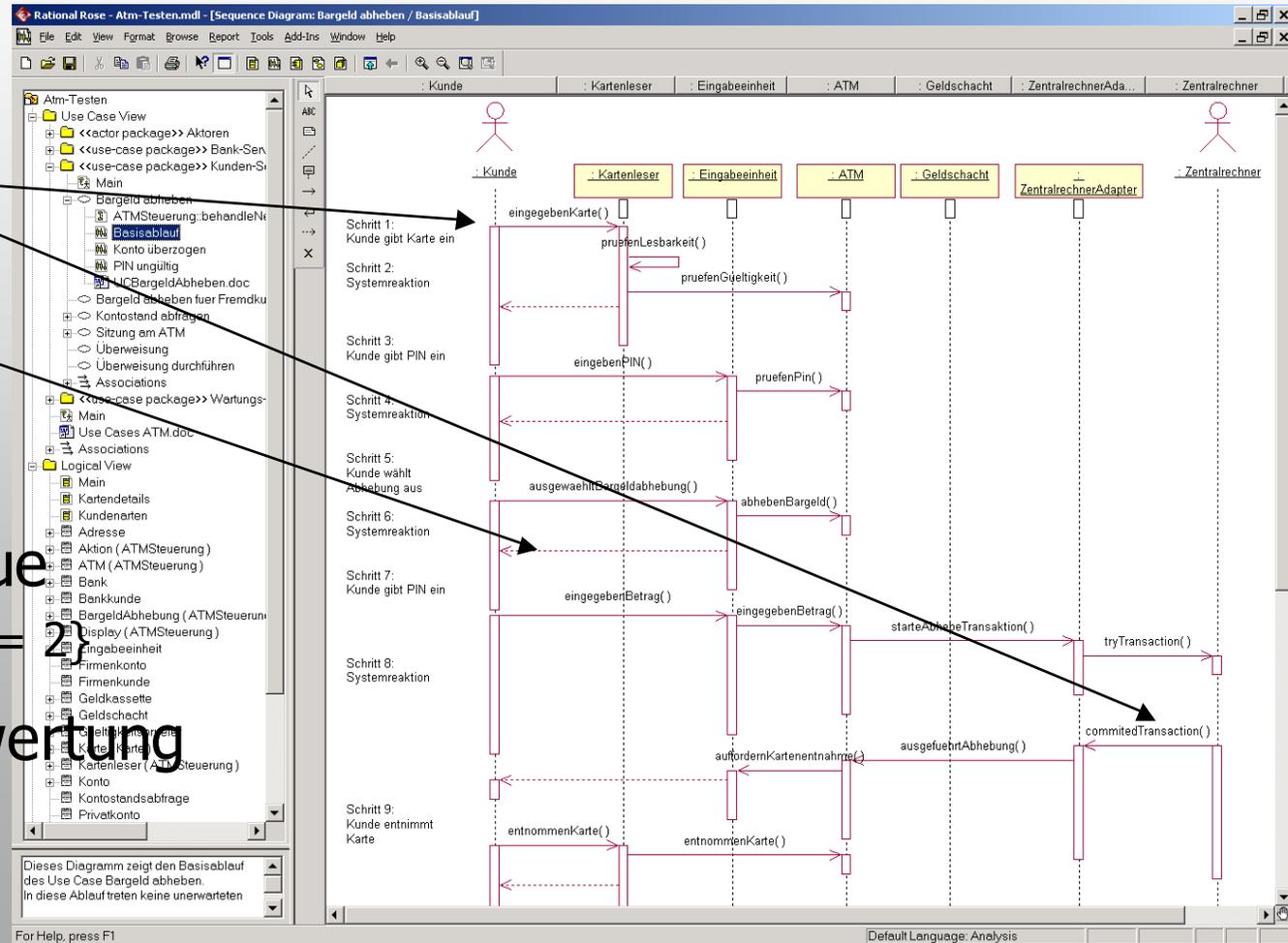
<<Test Case>>
 Bargeld abheben::
 PIN ungültig

<<Test Case>>
 Bargeld abheben::
 Konto überzogen



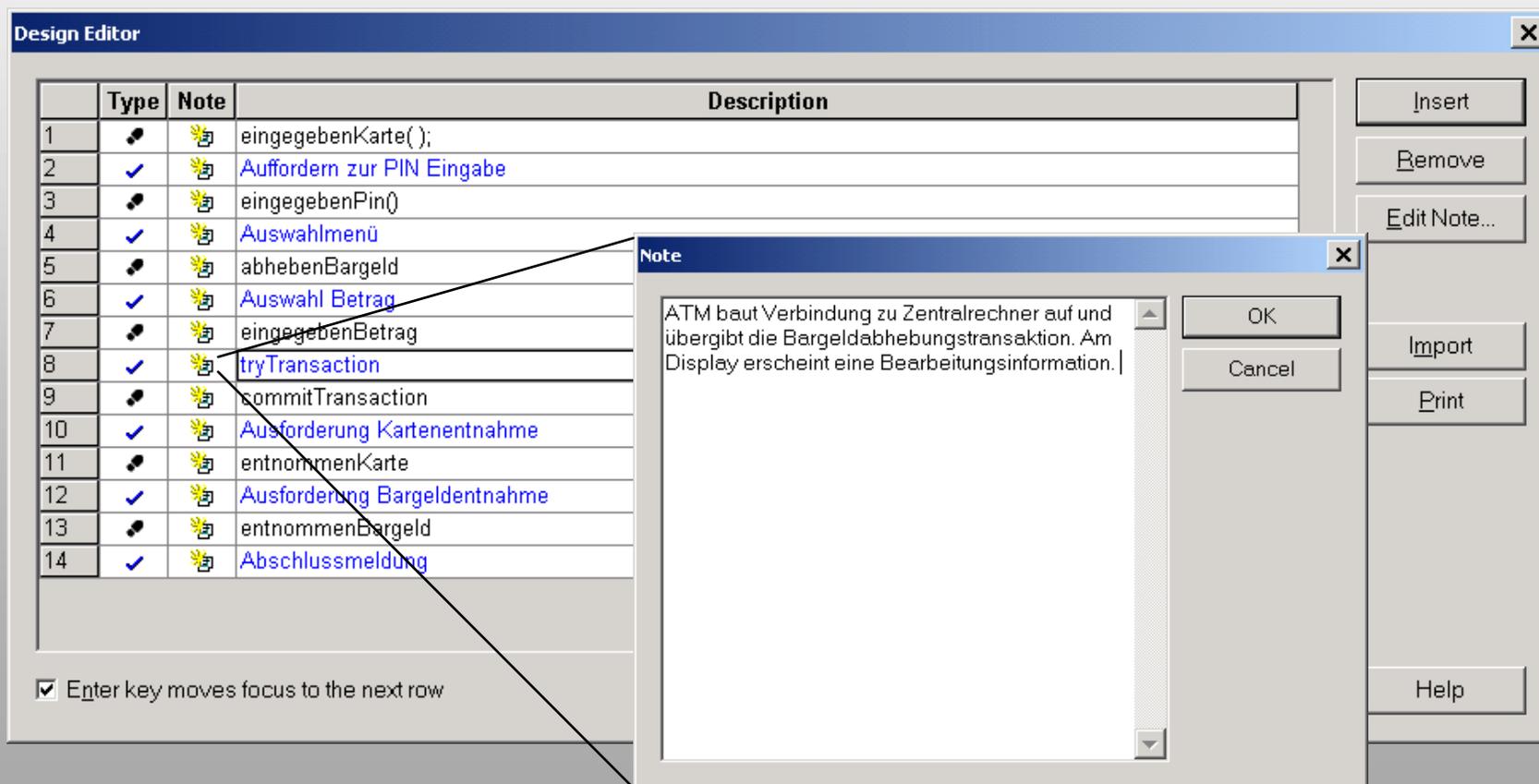
Ableitung der Testcases aus Use Case Realisierungen

- Steps
- Verification Points
- Use Case tagged value
{UseCaseStep = 2}
- Word Auswertung
Precondition
Postcondition



Ableitung der Testcases aus Use Case Realisierungen

- Test Case Design mit Steps und Verificationpoints

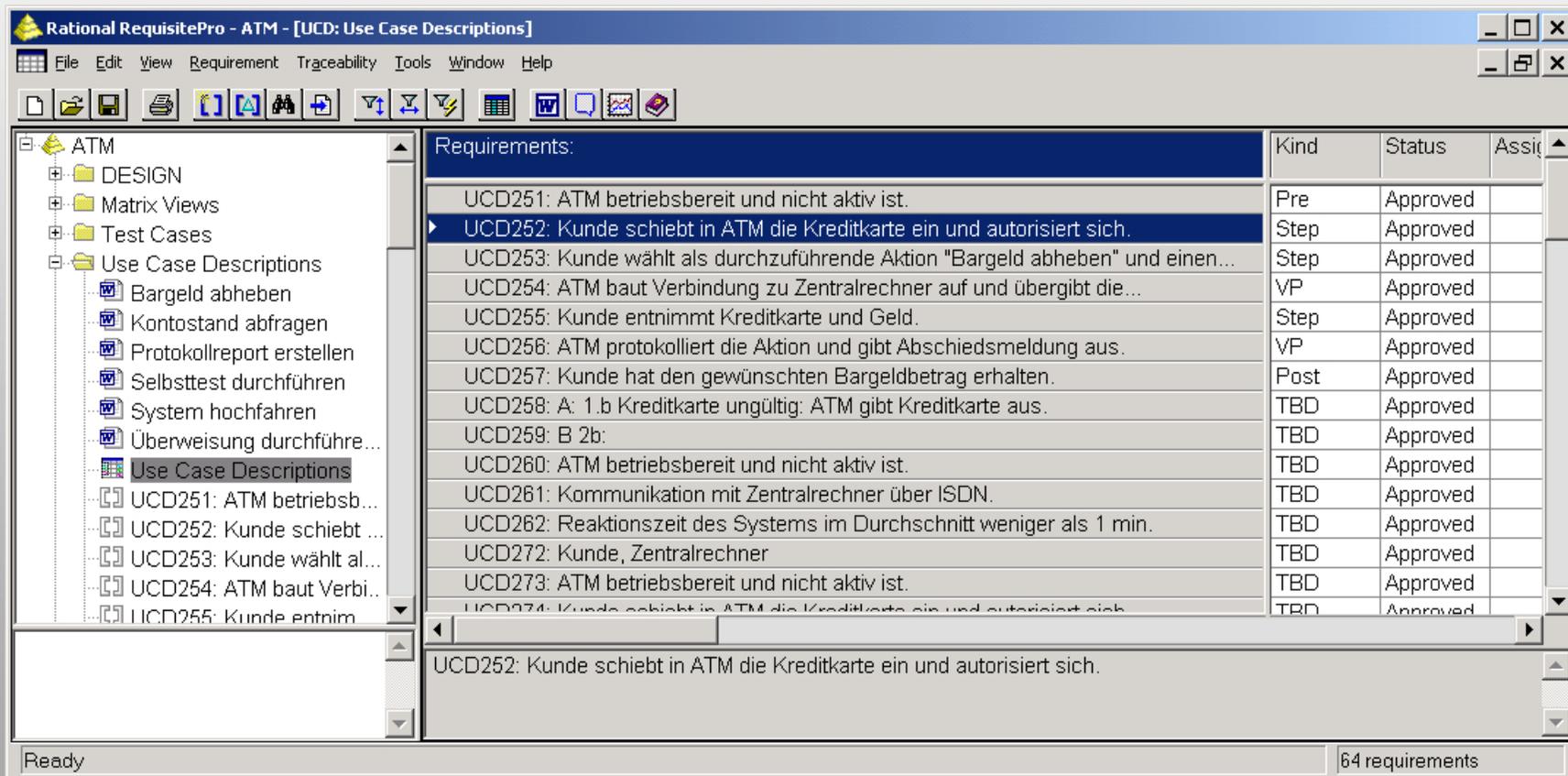


The screenshot shows the 'Design Editor' window with a table of test steps and a 'Note' dialog box. The table has columns for 'Type', 'Note', and 'Description'. The 'Note' dialog box is open over the 'tryTransaction' step, displaying the text: 'ATM baut Verbindung zu Zentralrechner auf und übergibt die Bargeldabhebungstransaktion. Am Display erscheint eine Bearbeitungsinformation.'

	Type	Note	Description
1	●	☞	eingegebenKarte();
2	✓	☞	Auffordern zur PIN Eingabe
3	●	☞	eingegebenPin()
4	✓	☞	Auswahlmenü
5	●	☞	abhebenBargeld
6	✓	☞	Auswahl Betrag
7	●	☞	eingegebenBetrag
8	✓	☞	tryTransaction
9	●	☞	commitTransaction
10	✓	☞	Ausforderung Kartententnahme
11	●	☞	entnommenKarte
12	✓	☞	Ausforderung Bargeldentnahme
13	●	☞	entnommenBargeld
14	✓	☞	Abschlussmeldung

Alternativ: Ableitung aus ReqPro Use Case Beschreibung

■ Attribute UCD::Kind



The screenshot shows the Rational RequisitePro interface for an ATM use case. The main window displays a table of requirements with columns for 'Kind', 'Status', and 'Assignee'. The 'Kind' column lists various requirement types such as Pre, Step, VP, Post, and TBD. The 'Status' column for all entries is 'Approved'. The 'Assignee' column is empty. The table is filtered to show requirements for the 'UCD252' use case.

Requirements:	Kind	Status	Assignee
UCD251: ATM betriebsbereit und nicht aktiv ist.	Pre	Approved	
UCD252: Kunde schiebt in ATM die Kreditkarte ein und autorisiert sich.	Step	Approved	
UCD253: Kunde wählt als durchzuführende Aktion "Bargeld abheben" und einen...	Step	Approved	
UCD254: ATM baut Verbindung zu Zentralrechner auf und übergibt die...	VP	Approved	
UCD255: Kunde entnimmt Kreditkarte und Geld.	Step	Approved	
UCD256: ATM protokolliert die Aktion und gibt Abschiedsmeldung aus.	VP	Approved	
UCD257: Kunde hat den gewünschten Bargeldbetrag erhalten.	Post	Approved	
UCD258: A: 1.b Kreditkarte ungültig: ATM gibt Kreditkarte aus.	TBD	Approved	
UCD259: B 2b:	TBD	Approved	
UCD260: ATM betriebsbereit und nicht aktiv ist.	TBD	Approved	
UCD261: Kommunikation mit Zentralrechner über ISDN.	TBD	Approved	
UCD262: Reaktionszeit des Systems im Durchschnitt weniger als 1 min.	TBD	Approved	
UCD272: Kunde, Zentralrechner	TBD	Approved	
UCD273: ATM betriebsbereit und nicht aktiv ist.	TBD	Approved	
UCD274: Kunde schiebt in ATM die Kreditkarte ein und autorisiert sich.	TBD	Approved	

The status bar at the bottom of the window indicates 'Ready' and '64 requirements'.

Automatisierung

- Skriptgestützte Ermittlung des Test Design und Export der Test Cases in XML
- Import der Test Cases in TestManager
- Analoges Vorgehen nichtfunktionaler Nachweis

Probleme

- Zuordnung Schritte <-> Messages nötig
- Aktualisierung bei instabilen Use Case Realisierungen
- Test Cases sind als "initial" zu betrachten
- Reicht nicht für komplexe Use Cases
 - Ansatz mit operationellen Variablen

Inhalt

- Requirements Management
- Konzept der Use Cases
- Business Object Modell
- Nichtfunktionale Anforderungen
- RM Datenbankstruktur
- Testen von Use Cases
- **Online Help**

Online-Dokumentation

Die Beschreibung besteht aus

- Use Case Beschreibung
- Relevaten Aktionen
- Relevaten GUIs
- Relevaten Fachkonzepten

Generierbar aus Modell

Action Bargeld abheben.

Preconditions

- ATM betriebsbereit.

Flow of Events - Basic Path

1. Der Use Case beginnt, wenn der Bankkunde eine Karte am Kartenleser eingibt.

.....

Relevante Aktionen

[How to choice the amount](#)

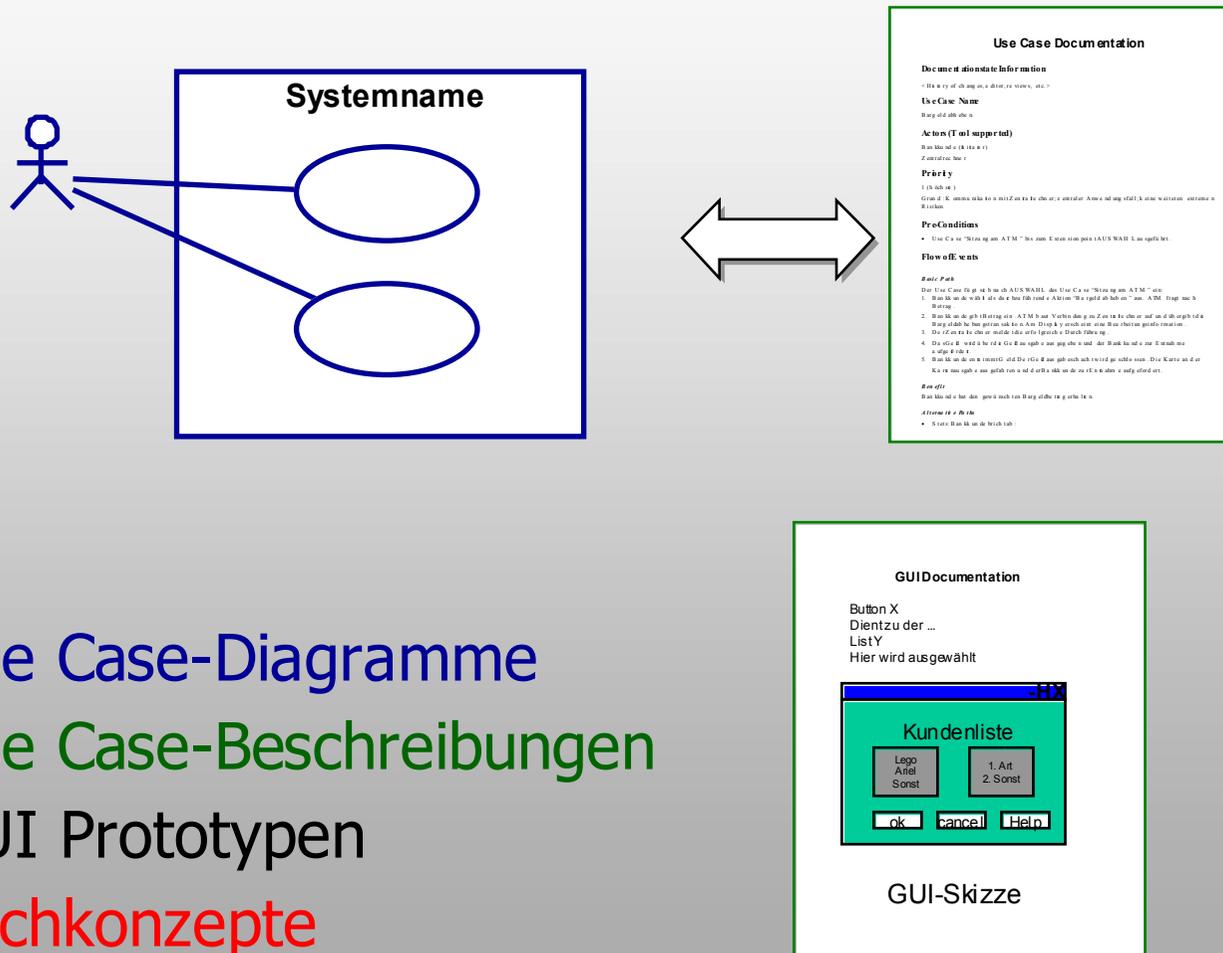
Relevante GUIs

[Aktionsauswahl](#) , [Betragsauswahl](#) ,
[Kartentnahme](#)

Relevante Fachkonzepte

[Konto](#), [Karte](#), [Kunde](#), ...

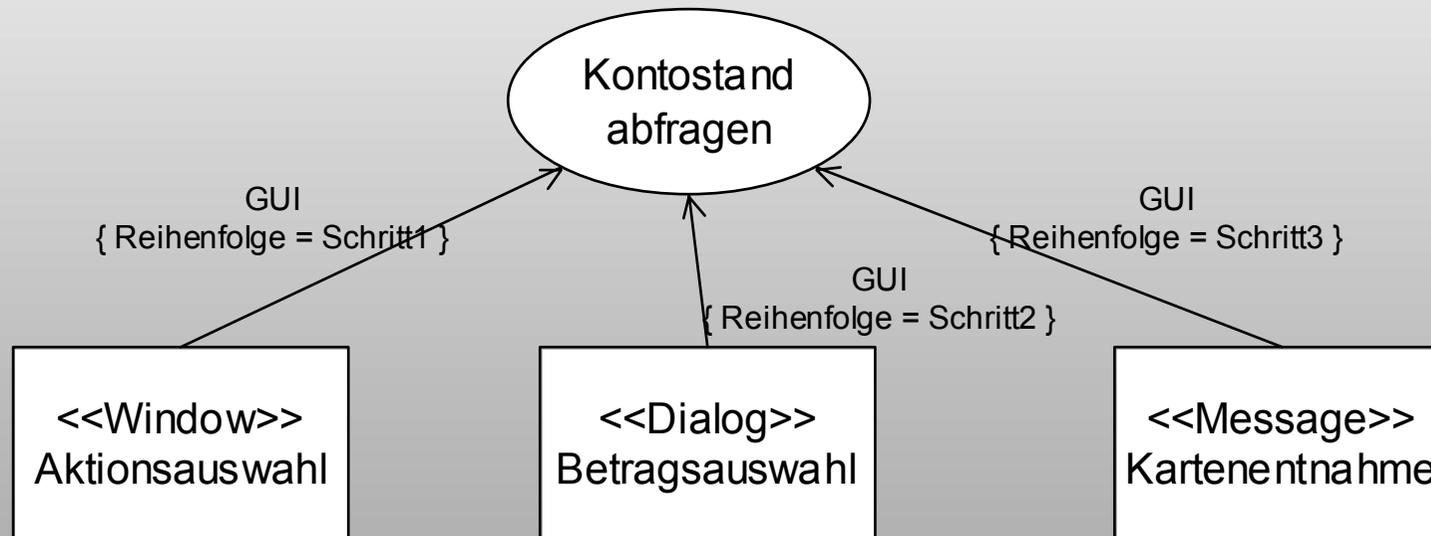
Use Case Modell



- Use Case-Diagramme
- Use Case-Beschreibungen
- GUI Prototypen
- Fachkonzepte

GUI-Zuordnung

Schrittzuordnung durch VModell->Reihenfolge



Zusammenfassung

- Klassisches RE und UC Modellierung sind kein Gegensatz
- Use Case Modellierung **ergänzt** klassisches RE
 - + Bessere Anwendereinbindung
 - + Unterstützung des Entwicklungsprozesse
 - inkrementelle Entwicklung
 - Übergang zur Systemmodellierung
 - Testen
 - Dokumentation
 - + Direkte Integration in ReqPro
 - + Anwendbar mit jedem UML CASE Tool
 - + Automatisierte Unterstützung für Rose

