

# MODUL 158

SOFTWARE-MIGRATION PLANEN &

DURCHFÜHREN LERNSEQUENZ-01 THEORIE

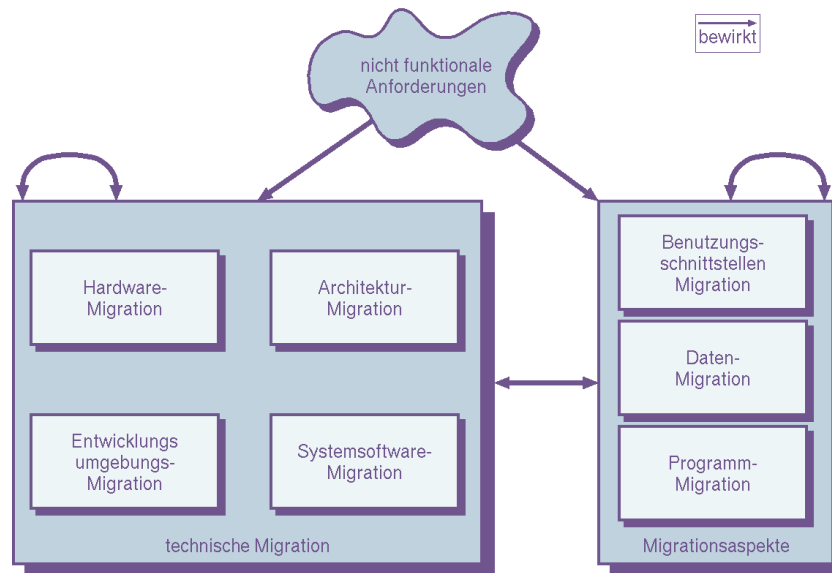
Oliver Schramm

# Inhaltsverzeichnis

<b>1</b>	<b>Einführung ins Thema Migration .....</b>	<b>2</b>
1.1	Workflows der Migration .....	2
1.2	Migrationsstrategie festlegen.....	3
1.3	Zielumgebung festlegen .....	3
1.4	Unterschiede analysieren.....	3
1.4.1	Risikoidentifikation/Risikobewertung.....	3
1.5	Migrationsumfang definieren/festlegen .....	5
1.5.1	Identifizieren wesentlicher Bestandteile der Migration .....	6
1.5.2	Definieren von Ergebnissen/Arbeitspaketen .....	6
1.6	Transformation spezifizieren (Detailkonzept) .....	9
1.7	Transformation umsetzen (Realisierung).....	9
1.8	Migriertes System übergeben .....	9
1.9	Mitarbeiter migrieren .....	9
1.10	Qualität sichern .....	9
1.11	Praktische Relevanz .....	10

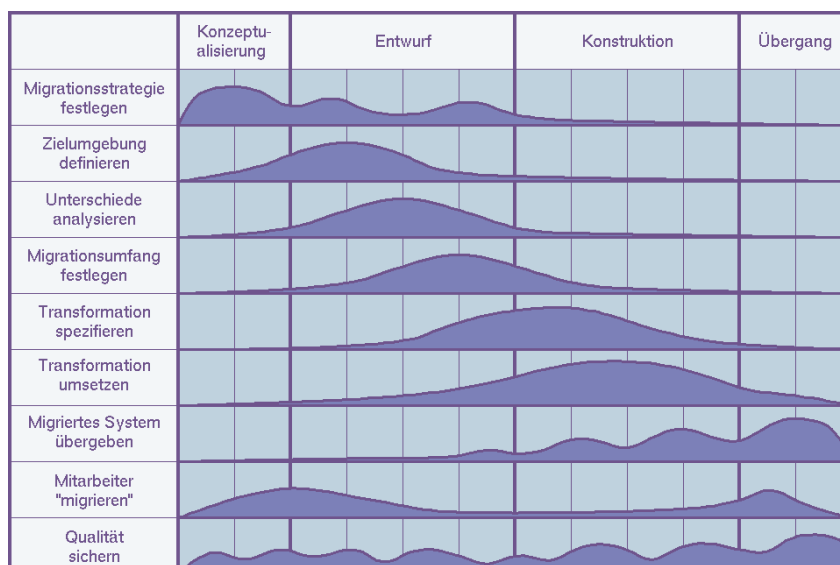
# 1 Einführung ins Thema Migration

Migration bezeichnet die Überführung eines Software-/ Hardwaresystems in eine andere Zielumgebung. Migrationen sind rein technische Transformationen mit einer klaren Anforderungsdefinition. Das zu migrierende Altsystem beschreibt eindeutig die Systemfunktionalität, deren Erhalt nach erfolgreicher Migration durch Regressionstests überprüft wird.



## 1.1 Workflows der Migration

Referenzmodelle zur Software-Migration ermöglichen die geplante, dokumentierte und kontrollierte Durchführung von Migrationsvorhaben. Die Bereitstellung von Referenzprozessen, basierend auf „Best Practices“, ermöglicht die zielgerichtete Entwicklung, Anpassung und Optimierung individueller Migrationsprozesse. Vorgehensmodelle zur Software-Migration aus der Praxis werden u.a. bei [Sneed 1999], [Collogia 2004], [RePro 2004] und [Sneed 2004] beschrieben. Diesen Modellen gemeinsam sind die im Folgenden kurz skizzierten Migrations-Workflows in unterschiedlichen Intensitäten der Phasen **Konzeptualisierung**, **Entwurf**, **Konstruktion** und **Übergang** ausgeführt werden



## 1.2 Migrationsstrategie festlegen

Festlegen der Migrationsstrategie, dafür bieten sich grundsätzlich drei Strategien an, Altsysteme in neue Umgebungen zu überführen:

**Neuentwicklung**, d.h. direkte Implementierung in der neuen Umgebung,

**Kapselung** (Wrapping), d.h. Belassen des Altsystems in seiner Umgebung und Bereitstellung von Zugriffsschnittstellen im Neusystem.

**Konversion**, d.h. direkte Übertragung oder (teil-)automatisierte Transformation in die neue Zielumgebung. In Abhängigkeit von der Qualität des vorliegenden Systems, der Daten, der angestrebten Qualität des Neusystems und dem zur Verfügung stehendem Budget ist für jede zu migrierende Komponente das Umsetzungsverfahren festzulegen [Sneed 1999].

## 1.3 Zielumgebung festlegen

Bevor wir eine Migration durchführen, legen wir fest, wie die neue Zielumgebung auszusehen hat. Festzulegen sind hier insbesondere Hardwareumgebung, Systemsoftware, Entwicklungsumgebung und Software-Architektur sowie die davon betroffenen Aspekte der Benutzungsschnittstellen, Daten und Programme.

## 1.4 Unterschiede analysieren

Die Bewertung des Altsystems hinsichtlich der generellen Migrationsfähigkeit ist eine wesentliche Voraussetzung zur Durchführung eines Migrationsprojekts. Dafür ist gegebenenfalls eine Neudokumentation vom Altsystem mit Methoden und Techniken des Reverse Engineering erforderlich. Die Unterschiede zwischen Alt- und Zielsystem beziehen sich beispielsweise auf die Ansteuerung von Benutzungsschnittstellen, die Zugriffstechniken auf Daten und die Kopplung ausführbarer Programme durch Jobsteuerungen oder Programmaufrufe.

### 1.4.1 Risikoidentifikation/Risikobewertung

Bei der Risikoidentifikation wird der Fokus auf die Hauptrisiken des Unternehmens gelegt. Die Checkliste wird oft auch als Gefahrenlisten bezeichnet. Da die personelle Möglichkeit meistens begrenzt sind, konzentrieren wir uns vorab auf die Toprisiken, um diese mit maximalem Erfolg abzudecken.

## Externe und interne Risiken

EXTERNE RISIKEN				
Finanzmarkt	Wirtschaft		Regulierungen	
INTERNE RISIKEN				
Strategisch	Operativ			Finanziell
	Prozesse	Technologie	Ordnungsmässigkeit (Compliance)	
	<ul style="list-style-type: none"><li>• Betriebsunterbruch</li><li>• Produktentwicklung</li><li>• ...</li></ul>	<ul style="list-style-type: none"><li>• Datensicherheit</li><li>• Technologisches Know-how</li><li>• ...</li></ul>	<ul style="list-style-type: none"><li>• Management Fraud</li><li>• Umweltschäden</li><li>• ...</li></ul>	
	Management Informationen	Humankapital		
	<ul style="list-style-type: none"><li>• Budgetierung &amp; Prognosen</li><li>• Investor Relations</li><li>• ...</li></ul>	<ul style="list-style-type: none"><li>• Veränderungsbereitschaft</li><li>• Leadership</li><li>• ...</li></ul>	<ul style="list-style-type: none"><li>• Waren-/Rohstoffpreise</li><li>• Fremdwährungskurse</li><li>• Liquidität</li><li>• Finanzierung</li><li>• ...</li></ul>	

Quelle: KPMG Schweiz, Methodik zur Einführung eines Risikomanagementsystems, 2008

Die Risikoidentifikation beginnt in der Regel mit einer Stärken-Schwächen Analyse (SWOT). Das kann im Rahmen eines Workshops oder Interviews mit den Key Playern durchgeführt werden. Die SWOT-Analyse ist ein einfaches Instrument, welches einerseits das Verständnis des Risiko Managements im Unternehmen schafft, andererseits zeigt sie die Verknüpfungen der relevanten Problemfelder zu den unternehmenseigenen Zielen auf. Der Sinn einer SWOT Analyse ist, die wesentlichen internen und externen Faktoren zu identifizieren, welche einen Einfluss auf die Entwicklung und den Unternehmenswert haben.

Die aus den Workshops, Interviews oder Benchmark Analysen gewonnen Erkenntnisse werden anhand eines Chancen- /Bedrohungsprofils kategorisiert und der richtigen Dimension des Risikomodells zugeordnet

## Chancen-/Bedrohungsprofil

Stufe	Bedrohung	Chance
unbedeutend	Angesichts der Grösse der Organisation zu vernachlässigen, Budget wird kaum beeinträchtigt.	Angesichts der Grösse der Organisation zu vernachlässigen, Budget wird kaum beeinträchtigt.
gering	Schadensfolgen sind begrenzt, sie können aus dem Cash Flow finanziert werden. Das Budget wird leicht beeinträchtigt.	Das Budget fällt in gewissen Positionen leicht günstiger aus wie geplant.
spürbar	Das Jahresergebnis wird beeinträchtigt, der EBIT (Betriebsgewinn vor Zinsen und Steuern) fällt geringer aus als geplant.	Das Jahresergebnis bzw. der EBIT fallen besser aus als geplant.
kritisch bzw. optimistisch	Das Jahresergebnis verschlechtert sich nachhaltig, der EBIT wird durch das Risiko verzehrt.	Das Jahresergebnis bzw. der EBIT fallen deutlich besser aus als geplant.
katastrophal bzw. unerwartet positiv	Die Existenz der Organisation ist bedroht, das Eigenkapital wird ganz oder teilweise verzehrt.	Das Jahresergebnis bzw. der EBIT fallen überraschend positiv und wesentlich besser aus als geplant.

Ist der Risikokatalog definiert, erfolgt zunächst eine grobe Risikobewertung für alle identifizierten Risiken. Die Risikobewertung befasst sich mit der Frage, wann ein Risiko tragbar ist und deshalb von den Stakeholdern akzeptiert werden kann.

Die Risikobewertung wird normalerweise in zwei Dimensionen durchgeführt

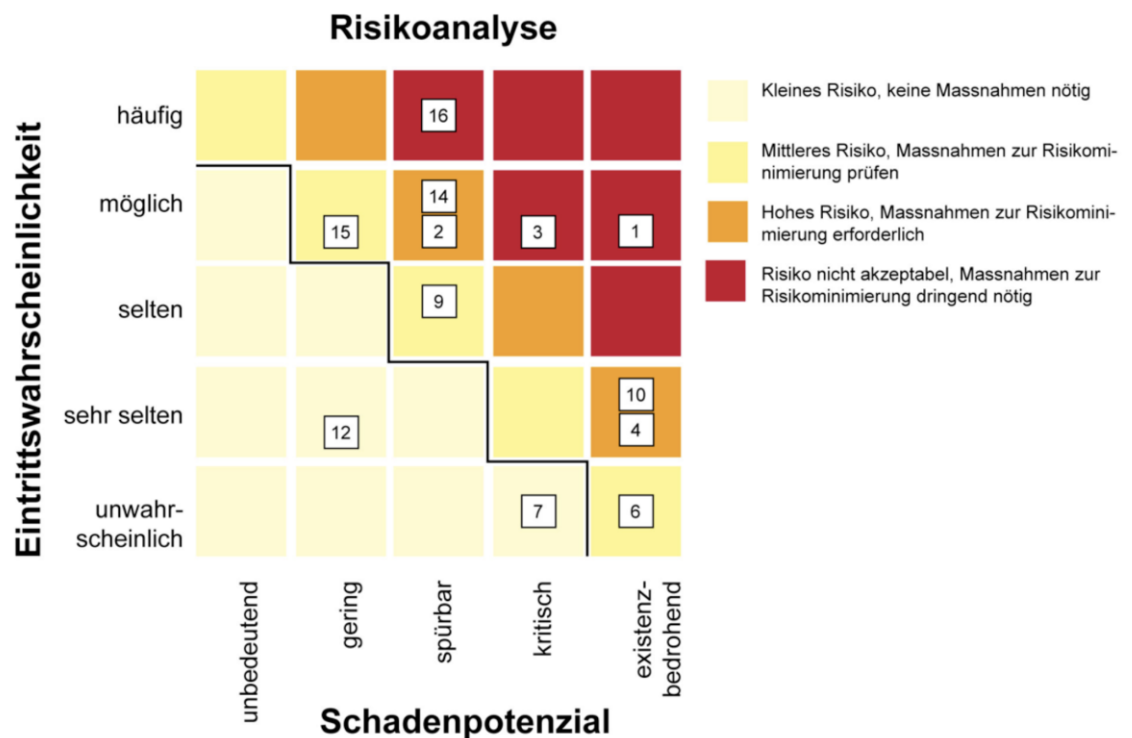
Eintretenswahrscheinlichkeit:

Die Eintrittswahrscheinlichkeit ist eine quantitative oder qualitative Angabe über die Wahrscheinlichkeit, mit der ein Risikoereignis innerhalb eines bestimmten Zeitraums eintritt

Auswirkung (Tragweite):

Tragweite eines Risikos ist die qualitative Prognose des Umfangs, den die Auswirkungen des Risikoereignisses haben, wenn es eintritt. Üblicherweise wird die Tragweite eines Risikos in Kategorien angegeben, z.B. "niedrig", "mittel" und "hoch"

Das Ergebnis der Risikobewertung wird grafisch in Form einer Risikolandkarte dargestellt.



## 1.5 Migrationsumfang definieren/festlegen

Die qualitative und quantitative Analyse der zu migrierenden Softwareobjekte wie z.B. Benutzerschnittstellen, Frontend (UI's) oder Backend (Server) Komponenten / Programme, Jobs, Dateien und Datenbanken und der sich daraus ergebenden Abhängigkeiten und Vergleiche mit früheren Migrationsprojekten erlauben es uns Rückschlüsse über den Aufwand des Migrationsvorhabens [Sneed 2003] zu ziehen. Die Analyse der Systemunterschiede auf der Ebene einzelner Artefakte ist mit entsprechend hohem Detaillierungsgrad auszuführen. Zentrale Kriterien sind hier insbesondere Funktionsumfang, Qualität und Wiederverwendbarkeit aller relevanten Komponenten des Altsystems. Durch Verwendung Metrik-basierter Verfahren kann entschieden werden, ob einzelne Komponenten durch Neuentwicklung, Wrapping oder Konvertierung migriert werden. Im Rahmen von Migrationsvorhaben ist immer ein möglichst hoher Anteil an Wiederverwendung anzustreben.

## 1.5.1 Identifizieren wesentlicher Bestandteile der Migration

IT-Migration was ist zu tun

Eine Migration beinhaltet die Phasen Initialisierung, Konzept, Realisierung und Einführung

IT-Migration nach Hermes 5.1

Initialisierung	Konzept	Realisierung	Einführung
	Migrationskonzept erarbeiten	Migrationsverfahren realisieren	Migration durchführen Entscheid zur Abnahme der Migration treffen Altsystem ausser Betrieb setzen

Tabelle 25: Aufgaben Modul IT-Migration

## 1.5.2 Definieren von Ergebnissen/Arbeitspaketen

Die Ergebnisse Arbeitspakete einer IT-Migration ergeben sich aus den jeweiligen Phasen:

*Initialisierung, Konzept, Realisierung und Einführung*

Initialisierung	Konzept	Realisierung	Einführung
	Migrationskonzept	Detailspezifikation Migrationsverfahren	Migration durchgeführt Abnahmeprotokoll Checkliste Projektentscheid Führung & Ausführung Altsystem abgebaut

Tabelle 26: Ergebnisse Modul IT-Migration

### Aus der Phase Initialisierung

Die Initialisierung schafft eine definierte Ausgangslage für das Projekt und stellt sicher, dass die Projektziele mit den Zielen und Strategien der Organisation abgestimmt sind. Die Projektgrundlagen und der Projektauftrag werden erarbeitet und der Entscheid zur Projektfreigabe wird getroffen.

1. Auf der Grundlage des Projektinitialisierungsauftrags gibt der Auftraggeber die Ressourcen für die Phase Initialisierung frei. Er beauftragt einen Projektleiter mit der Durchführung der Phase Initialisierung.
2. Die Studie mit Situationsanalyse, Zielen und groben Anforderungen sowie die Varianten werden erarbeitet. Die Beschreibung der Varianten erfolgt so detailliert, dass sie nachvollziehbar und transparent bewertet werden können. Unter anderem werden die Projekt- und Betriebsrisiken ermittelt, die Rechtsgrundlagenanalyse und die Schutzbedarfsanalyse erarbeitet und in die Entscheidung einbezogen. Der Entscheid zur Variantenwahl wird getroffen.
3. Auf der Basis der gewählten Variante werden Projektmanagementplan und Projektauftrag erarbeitet und mit den Strategien, Vorgaben und übergeordneten Zielen der Stammorganisation abgeglichen. Die Stakeholder Interessen werden analysiert und Zielkonflikte bereinigt.
4. Der Entscheid zur Projektfreigabe wird getroffen und der Projektauftrag unterzeichnet. Die Freigabe erfolgt durch die Stammorganisation und den Auftraggeber.

## Aus der Phase Konzept

### *Migrationskonzept*

Die in der Phase Initialisierung gewählte Variante wird konkretisiert. Die Ergebnisse werden so detailliert erarbeitet, dass die Projektbeteiligten das Produkt bzw. das IT-System auf einer verlässlichen Grundlage planen, offerieren und realisieren können

[Migrationskonzept Beziehungen](#)

[Word Vorlage Migrationskonzept](#)

Modul	Aufgabe	Verantwortung Aufgabe	Ergebnis	Beteiligt an der Ergebniserstellung
IT-Migration	Migrationskonzept erarbeiten	IT-Architekt	Migrationskonzept	Business Analyst, Entwickler

## Aus der Phase Realisierung

### *Detailspezifikation*

Die Detailspezifikation beschreibt die detaillierten Systemeigenschaften. Sie basiert auf den Systemanforderungen und der Detailstudie und bildet die Grundlage für die Realisierung und die Erstellung von detaillierten Testfallbeschreibungen.

[Detailspezifikation Beziehungen](#)

Modul	Aufgabe	Verantwortung Aufgabe	Ergebnis	Beteiligt an der Ergebniserstellung
IT-System	System realisieren	Entwickler	Detailspezifikation	Business Analyst, IT-Architekt
	Systemintegration vorbereiten	Entwickler	Detailspezifikation	Business Analyst, IT-Architekt
IT-Migration	Migrationsverfahren realisieren	Entwickler	Detailspezifikation	Business Analyst, IT-Architekt

### *Migrationsverfahren*

Das realisierte Migrationsverfahren ist durch den Entwickler bzw. den Tester des Erstellers getestet worden. Er erbringt den Nachweis seiner Tests. Sie bilden die Voraussetzung für die Vorabnahme

Modul	Aufgabe	Verantwortung Aufgabe	Ergebnis	Beteiligt an der Ergebniserstellung
IT-Migration	Migrationsverfahren realisieren	Entwickler	Migrationsverfahren	Betriebsverantwortlicher

## Aus der Phase Einführung

Der sichere Übergang vom alten zum neuen Zustand wird gewährleistet. Der Betrieb wird aufgenommen und so lange durch das Projekt unterstützt, bis er stabil ist.

### *Migration durchgeführt*

Die Migration vom alten auf das neue System ist durchgeführt und gemäss den Vorgaben der Controlling- und Koordinationsstellen dokumentiert. Die Nachvollziehbarkeit der Migration wird sichergestellt. Die erfolgreiche Migration ist die Voraussetzung für den Entscheid zur Betriebsaufnahme und die Abnahme der Migration.

Modul	Aufgabe	Verantwortung Aufgabe	Ergebnis	Beteiligt an der Ergebniserstellung
IT-Migration	Migration durchführen	Entwickler	Migration durchgeführt	Betriebsverantwortlicher, Business Analyst, Anwendungsverantwortlicher

### *Abnahmeprotokoll*

Das Abnahmeprotokoll dokumentiert die Erfüllung der Vereinbarung über die Produkt-/Systemeigenschaften und bestehende Mängel. Es ist ein rechtlich verbindliches Dokument.

[Wordvorlage Abnahmeprotokoll](#)



Modul	Aufgabe	Verantwortung Aufgabe	Ergebnis	Beteiligt an der Ergebniserstellung
Einführungsorganisation	Entscheid zur Vorabnahme treffen	Projektleiter	Abnahmeprotokoll	Betriebsverantwortlicher, Anwendervertreter, Anwendungsverantwortlicher, Entwickler, Qualitäts- und Risikomanager
	Entscheid zur Abnahme treffen	Projektleiter	Abnahmeprotokoll	Betriebsverantwortlicher, Anwendervertreter, Anwendungsverantwortlicher, Entwickler, Qualitäts- und Risikomanager
IT-Migration	Entscheid zur Abnahme der Migration treffen	Projektleiter	Abnahmeprotokoll	Betriebsverantwortlicher, Anwendervertreter, Anwendungsverantwortlicher, Entwickler, Qualitäts- und Risikomanager

### Checkliste

Die nach Entscheidungsaufgaben gegliederte Checkliste unterstützt die Entscheidungsprozesse und dient der systematischen Durchführung von Prüfungen. Vor jeder Prüfung wird die allgemeingültige Checkliste an entsprechender Stelle mit projektspezifischen Kriterien ergänzt.

#### Wordvorlage Checkliste

Einführungsorganisation	Entscheid zur Vorabnahme treffen	Projektleiter	Checkliste	
	Entscheid zur Betriebsaufnahme treffen	Auftraggeber	Checkliste	Projektleiter
	Entscheid zur Abnahme treffen	Projektleiter	Checkliste	
IT-Migration	Entscheid zur Abnahme der Migration treffen	Projektleiter	Checkliste	
Informationssicherheit und Datenschutz	Entscheid zum ISDS-Konzept treffen	Projektleiter	Checkliste	
	ISDS-Konzept überführen	ISDS-Verantwortlicher	Checkliste	Projektleiter

### Projektentscheid

Die Projektentscheide der Projektführung dokumentieren die Ergebnisse der Entscheidungsaufgaben der Hierarchieebene Führung und Ausführung. Die während eines Projekts zu treffenden Entscheide werden als Entscheidungsaufgaben definiert. Das Resultat der Entscheide wird dokumentiert. Das Dokument Projektentscheid Führung & Ausführung wird für die ganze Dauer eines Projekts verwendet.

#### Wordvorlage Projektentscheid

Modul	Aufgabe	Verantwortung Aufgabe	Ergebnis	Beteiligt an der Ergebniserstellung
Projektführung	Entscheid zur Variantenwahl treffen	Projektleiter	Projektentscheid Führung & Ausführung	Qualitäts- und Risikomanager
Entwicklung Agil	Entscheid zur agilen Entwicklung mit SCRUM treffen	Projektleiter	Projektentscheid Führung & Ausführung	Qualitäts- und Risikomanager
IT-System	Entscheid zur Systemarchitektur treffen	Projektleiter	Projektentscheid Führung & Ausführung	Qualitäts- und Risikomanager
Einführungsorganisation	Entscheid zur Vorabnahme treffen	Projektleiter	Projektentscheid Führung & Ausführung	Qualitäts- und Risikomanager
	Entscheid zur Abnahme treffen	Projektleiter	Projektentscheid Führung & Ausführung	Qualitäts- und Risikomanager
IT-Migration	Entscheid zur Abnahme der Migration treffen	Projektleiter	Projektentscheid Führung & Ausführung	Qualitäts- und Risikomanager
Informationssicherheit und Datenschutz	Entscheid zum ISDS-Konzept treffen	Projektleiter	Projektentscheid Führung & Ausführung	Qualitäts- und Risikomanager

### Altsystem abgebaut

Modul	Aufgabe	Verantwortung Aufgabe	Ergebnis	Beteiligt an der Ergebniserstellung
IT-Migration	Altsystem ausser Betrieb setzen	IT-Architekt	Altsystem abgebaut	Betriebsverantwortlicher, Business Analyst

#### IT-Migration nach Hermes 5.1

## 1.6 Transformation spezifizieren (Detailkonzept)

Vor der Durchführung der eigentlichen Umsetzungen des Altsystems in das Zielsystem sind die notwendigen Transformationen festzulegen. Je nach Migrationsvorgehen sind Konversionsregeln zu definieren, Kapselungen auf Basis geeigneter Middleware festzulegen, oder das Zielsystem aufbauend auf der Funktionalität des Altsystems zu spezifizieren. Dabei ist eine Vielzahl unterschiedlicher Programmier-, Job-, Datenbank- und Maskensprachen zu berücksichtigen, für deren Transformationen nur in Spezialfällen Werkzeuge vorhanden sind [Sneed 1999].

Generische Programmtrans Transformationsansätze, wie beispielsweise TXL [Cordy et al. 2002], basieren auf der Definition von Transformationsregeln auf Basis der Grammatiken von Quell- und Zieldokumenten. Generelle Transformationsansätze, deren Erprobung in konkreten Migrationsvorhaben jedoch noch ausstehen, sind in [Czarnecki/Helsen 2003] skizziert.

## 1.7 Transformation umsetzen (Realisierung)

Die eigentliche Umsetzung der Migration durch Konversion oder Kapselung der Programme und Daten erfolgt aufgrund der zuvor getroffenen Entscheidungen. Dabei sollten iterativ jeweils zusammengehörende Module migriert werden. Es empfiehlt sich zunächst die Migration der Daten, anschließend die Migration der Programme und zum Schluss die Migration der Benutzungsoberflächen, dieser Migrationsschritt erfordert insbesondere gute Kenntnis der Zielumgebung.

## 1.8 Migriertes System übergeben

Abhängig von der von Ihnen gewählten Migrationsstrategie erfolgt die komplette Übergabe des Zielsystems auf einen Schlag (Big Bang) oder inkrementell, in mehreren Schritten (sanft). Im letzteren Fall existieren beide Systeme zeitweise nebeneinander, so dass der Parallelbetrieb mit geeigneten Maßnahmen synchronisiert werden muss.

## 1.9 Mitarbeiter migrieren

Zur Migration von Softwaresystem ist es erforderlich, dass sowohl die Mitarbeiter, die mit der Migration betraut sind, als auch die Mitarbeiter, die später mit dem migrierten System arbeiten, entsprechend qualifiziert werden. Für die Mitarbeiter in der Software-Entwicklung bezieht sich die Qualifikation neben den Methoden und Techniken zur Softwaremigration auch auf den Umgang mit Alt- und Zielsystem.

## 1.10 Qualität sichern

Qualitätssichernde Maßnahmen der Migration müssen sicherstellen, dass die funktionale Äquivalenz von Alt- und Zielsystem gewährleistet ist. Diese Aktivität gilt als größter Kostenfaktor eines Migrationsprojekts [Sneed et al. 2004]. Soweit teilautomatisierte Transformationen zur Konvertierung eingesetzt werden, sind diese ausführlich zu testen. Generell empfiehlt sich die Erprobung der Transformationen an geeigneten Programmen, Daten und Masken. Das Vorliegen des Altsystems erlaubt die Validierung der funktionalen Äquivalenz durch Regressionstests.

## 1.11 Praktische Relevanz

Das hier skizzierte, abstrahierte Vorgehensmodell für Software- Migrationen wird anhand von komplexen Projektbeispielen auf seine generelle Anwendbarkeit überprüft. Diese beinhalten unterschiedliche Migrationstypen, die in jeweils unterschiedlicher Intensität verschiedene technische Migrationen betreffen und unterschiedliche Migrationsaspekte behandeln.

### Projektbeispiele:

#### *Programm-zentrierte Migration:*

Umstellung zentraler Schlüsselnummern.

#### *Daten-zentrierte Migration:*

(Instanz und Schema): Migration einer dateibasierten Datenhaltung in eine relationale Datenbank.

#### *Architektur-zentrierte Migration:*

schrittweise Umstellung einer Mainframe-Realisierung in eine Serviceorientierte Architektur [Gimnich 2005].

Die konkrete Bearbeitung der Migrations-Workflows ist in allen Beispiel-Projekten sehr unterschiedlich, sie lässt sich aber nach dem skizzierten Vorgehensmodell auf einheitliche Weise beschreiben. Der Workflow Transformationen spezifizieren wird etwa bei den Projekten zur Programm und Daten-zentrierten Migration nur einmal durchlaufen. Die Spezifikation lässt sich z.B. über Legacy Transformation Patterns [Hess 2005] auf Architektur-, Daten- und Programmseite relativ einfach erstellen. Im Projekt zur Architektur-zentrierten Migration ist dagegen dieser Workflow naturgemäß mehrfach im Gesamtprojekt zu durchlaufen und wesentlich umfangreicher, da Geschäftsprozess, Architektur und Programme tiefgreifend betroffen sind.

# Literaturverweise

- [Borchers/Moritz 2005] J. Borchers, Moritz: B. Genauigkeit von Aufwandsschätzungen in Reengineering-Projekten - Erfahrungen aus einer Sprachumstellung von Assembler nach COBOL, Informatik Forschung und Entwicklung, 2005.
- [Brodie/Stonebraker 1995] M.L. Brodie, M. Stonebraker: Migrating Legacy Systems. Morgan Kaufmann, San Francisco, California, 1995.
- [Czarnecki/Helsen 2003] K. Czarnecki, S. Helsen: Classification of Model Transformation Approaches, OOPSLA'03 Workshop on Generative Techniques in the Context of MDA, [http://swen.uwaterloo.ca/~kczarneck/ECE750T7/czarnecki\\_helsen.pdf](http://swen.uwaterloo.ca/~kczarneck/ECE750T7/czarnecki_helsen.pdf).
- [Collogia 2004] collogia AG: Reengineering - Modernisierung von Altsystemen. Köln. Version 5.4. <http://www.collogia.de/95.0.html>.
- [Cordy et al. 2002] J.R. Cordy, T.R. Dean, A.J. Malton and K.A. Schneider: Source Transformation in Software Engineering using the TXL Transformation System, Journal of Information and Software Technology 44(13)2002, pp. 827-837.
- [Ginnich 2005] R. Ginnich: Nutzung von Legacy-Software in Serviceorientierten Architekturen, SQM 2005, <http://www.sqs-conferences.com/abstracts/ginnich.pdf>.
- [Hasselbring et al. 2004] W. Hasselbring, R. Reussner, H. Jaekel, J. Schlegelmilch, T. Teschke, S. Krieghoff: The Dublo Architecture Pattern for Smooth Migration of Business Information Systems: An Experience Report. 26th ICSE, 117-126, Edinburgh, Scotland, 2004.
- [Hess 2005] H. M. Hess: Aligning technology and business: Applying patterns for legacy transformation. IBM Systems Journal, 44(1) 2005.
- [Kruchten 2000] P. Kruchten: The Rational Unified Process, An Introduction. Addison-Wesley, Upper Saddle River, 2nd edition, 2000.
- [RePro 2004] U. Kuhlmann, H. Sneed, A. Winter: Workshop Reengineering Prozesse, Fallstudien, Methoden, Vorgehen, Werkzeuge. Fachberichte Informatik, 11/2004, Universität Koblenz-Landau, 2004.
- [Sneed 1999] H. Sneed: Objektorientierte Softwaremigration, Addison-Wesley, Bonn, 1999.
- [Sneed 2003] H. Sneed: Aufwandsschätzung von Reengineering-Projekten, Wirtschaftsinformatik, 45(6)2003.
- [Sneed et al. 2004] H. Sneed, M. Hasitschka, M. Teichmann: Software-Produktmanagement, Wartung und Weiterentwicklung bestehender Anwendungssysteme. Dpunkt Verlag, Heidelberg, 2004.
- [Winter 2004] A. Winter: Software-Reengineering, Werkzeuge und Prozesse, Workshop der GI-Fachgruppe „Software-Wartung“, 15. Oktober 2004, Stuttgart <http://www.iste.uni-stuttgart.de/se/people/opferkuch/AK-Wartung/Positionspapiere/winter.pdf>.