

# Qualitätsaspekte der Wiederverwendung von Schemata

Martin Bertram

debis Systemhaus Dienstleistungen GmbH  
Lyoner Straße 9; D-60528 Frankfurt

**Zusammenfassung:** Ein erheblicher Anteil des Aufwandes zur Erstellung eines Fachkonzepts wird für die Erarbeitung des Datenmodells verwendet. Dieser Aufwand läßt sich durch eine geplante Wiederverwendung von Schemata reduzieren, denn damit kann das gesammelte Modellierungswissen eines Unternehmens gebündelt und wiederverwendet werden. Wesentliche Voraussetzungen für diese Vorgehensweise ist eine hohe Qualität sowohl des Prozesses, als auch der wiederzuverwendenden Schemata selbst. Dieser Beitrag stellt die wesentlichen Qualitätskriterien für diese Schemata vor und verdeutlicht sie anhand eines Beispiels. Abschließend wird die Vorgehensweise insgesamt bewertet.

## 1 EINLEITUNG

In der heutigen wirtschaftlichen Situation der meisten Unternehmen wächst der Druck zur Reduktion der Kosten der Softwareentwicklung. Insbesondere gilt dies für Beratungsunternehmen, die Software für Kundenunternehmen im Wettbewerb mit anderen Unternehmen erstellen. Ein zentraler Punkt ist hierzu die systematische Wiederverwendung des Wissens, das sich im Laufe vieler Projekte angesammelt hat. Die Wiederverwendung von Wissen geschah bis vor nicht allzu langer Zeit dadurch, daß einzelne Mitarbeiter ihr angehäuftes Wissen in neue Projekte einbrachten, quasi als Wiederverwendung durch Mitarbeitertransfer. Dieses Wissen war aber ausschließlich personenbezogen, mit all den Risiken und Beschränkungen, die sich daraus ergeben. Seit längerem wird nun daran gearbeitet, das Wissen zu systematisieren und personenunabhängig sowie für das gesamte Unternehmen zugänglich zu machen.

Zunächst könnte man daran denken, einfach die einmal erstellten Entwicklungsprodukte direkt wiederzuverwenden. Dies ist aber in der Regel aufgrund

rechtlicher Beschränkungen kaum möglich und es kann, aufgrund der Verschiedenheit der Kundensituationen, nur ein sehr niedriger Wiederverwendungsfaktor erwartet werden. Ein zweiter möglicher Weg ist die Erarbeitung von Referenzmodellen, die das Wissen branchenbezogen bündeln. Dieser Weg erfordert aber vom erstellenden Unternehmen einen erheblichen Aufwand, vor allem da für jedes Anwendungsgebiet ein eigenes, vollständiges und umfassendes Referenzmodell erstellt werden muß<sup>1</sup>. Im folgenden wird ein weiter Weg vorgestellt, nämlich die Wiederverwendung von abstrakter Schemata im Rahmen von Datenmodellen. Solche Schemata werden in einer zentralen Bibliothek gesammelt und den Projekten zugänglich gemacht.

Die semantische Datenmodellierung ist seit langem die verbreitetste Methode, die Anforderungen an Systeme zu beschreiben, konzeptionelle Modelle von Anwendungssystemen zu erstellen sowie auch zur Darstellung unternehmensweiter Datenmodelle. Der Aufwand für diese Modelle stellt einen großen Anteil des Gesamtaufwands eines Fachkonzepts dar und steigt stark mit der Größe der Modelle an. Einer der wesentlichen Gründe dafür liegt in der Tatsache begründet, daß das lehrbuchgemäße Vorgehen eine **Modellerteilung** auf Basis der **Modellierung** einzelner **Tatbestände** - sprich **Entities und Relationships** - fordert: "Ein Entity ist ein Konzept, eine Person, ein Ding, . . ."

Die Wiederverwendung abstrakter Schemata bedeutet eine Möglichkeit, Datenmodelle schneller durch Nutzung größerer komplexer und abstrakter Elemente als Bausteine zu erstellen. Grundlage für diesen Ansatz ist das Vorhandensein einer Bibliothek solcher Elemente. Die Hauptvorteile dieses Ansatzes sind, neben der Verkürzung und Erleichterung der Modellierungsphase, die Möglichkeit zur Aufwandseinsparung sowohl in der Design-, als auch in der Implementierungsphase. Letzteres wird durch die Wiederverwendung vorgefertigter Designs und Implementierungen der einzelnen Schemaelemente erreicht. Als zusätzlicher Effekt wird auch eine spürbare Verbesserung der Qualität des Datenmodells<sup>2</sup> insgesamt erreicht.

Der Beitrag beschränkt sich auf einen wesentlichen Aspekt der Wiederverwendung abstrakter Schemata, nämlich auf die Beschreibung der für ihre effektive Anwendung notwendigen Qualitätskriterien. Diese sind aus der Erfahrung mit

1 Im debis Systemhaus existieren einige Referenzmodelle, u.a. ICIS für den Versicherungsbe- reich. [dSH94]

2 Bezüglich der Qualität von Datenmodellen, speziell für unternehmensweite Datenmodelle, vergleiche [BER93]



## ANWENDUNGSREGELN

Das sind alle Regeln, die beschreiben, unter welchen Umständen und wie die Schablone innerhalb eines Datenmodells genutzt werden kann und wie nicht. Dazu gehören auch Beispiele für einen sinnvollen Einsatz und solche, in denen die Schablone nicht angewandt werden kann.

Beispiel:

- Nur die ausgezeichneten Aggregate und die Elementarbausteine können anderen Elementen des Modells zugeordnet werden.
- Jeder Elementarbaustein kann nur genau einem Element zugeordnet werden.

## VARIANTEN DER BASISCHABLONE

Varianten dienen dazu, zusätzliche Anwendungsbereiche abdecken zu können, die von der Basisschablone selbst nicht abgedeckt werden. Diese Anwendungsbereiche unterscheiden sich aber nicht grundsätzlich von denen der Basisschablone, sondern stellen eine Erweiterung bzw. eine Spezialisierung dar. Die Basisschablone sollte in diesem Sinne minimal sein.

Beispiel:

- In der Basisvariante sind nur Beziehungen der Art "enthält" zwischen Komponenten erlaubt.
- Eine Variante bietet zusätzlich beliebige variable Beziehungen zwischen den Komponenten.

Die Beschreibung sollte sich im Sinne einer Vererbung der Eigenschaften auf die Darstellung der Unterschiede zur Basisschablone beschränken.

## KLASSIFIKATION

Die Klassifikation der Schablone selbst und ihrer Varianten gemäß einem facetierten Schema dient zur Auffindung der in einer konkreten Situation anwendbaren Schablonen. Dies kann um Synonyme der zugrundeliegenden Fachbegriffe ergänzt werden.

## VERWEISE AUF ANDERE SCHABLONEN

Wenn sich ein vorgegebenes Problem weder von der gefundenen Basisschablone, noch von den zugehörigen Varianten lösen lässt, erleichtert ein Verweis auf ähnliche Schablonen der Bibliothek die weitere Suche.

## 3.2 DATENMODELL

Das Datenmodell einer Schablone beinhaltet die folgenden Bestandteile:

## DATENMODELL

Ein semantisches Datenmodell mit graphischer Darstellung, abstrakten Entities und Relationships einschließlich der Definitionen.

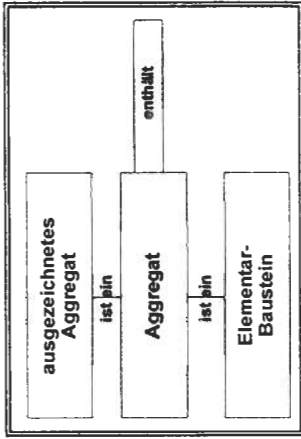


Abbildung 5: Datenmodell der Stückliste

Dieses beschreibt alle Konstrukte des Datenmodells, die die Systemgrenze der Schablone überschreiten. Dies ist die Modellierung der Schnittstellendefinition. Dazu zählen die folgenden Komponenten:

- Eine Beschreibung aller "offenen" Relationships, die die Beziehungen zwischen den Entities der Schablone und denen des Modells herstellen. Hierunter können neben den assoziativen Relationships auch Sub- und Supertyp-Beziehungen fallen.

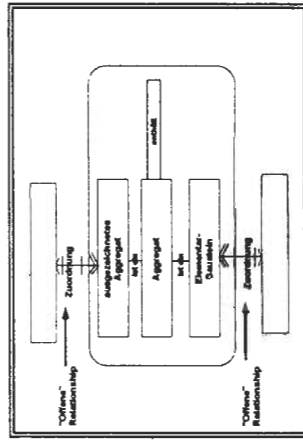


Abbildung 4: Schnittstellen zur Umwelt

- Die Definition aller Integritätsbedingungen, die Elemente sowohl innerhalb als auch außerhalb der Schablone betreffen. Dazu zählen sowohl die impliziten, als auch die expliziten Integritätsbedingungen.

## INTEGRITÄTSBEDINGUNGEN INNERHALB DER SCHABLONE

- implizite Integritätsbedingungen, wie z.B. die Bedingungen an die Kardinalitäten.
- explizite Integritätsbedingungen, d.h. solche, die sich nicht mit dem vorgegebenen Formalismus abbilden lassen, sondern die durch eigene Konstrukte, wie z.B. Prädikatenlogik, formuliert werden müssen.

## ATTRIBUTE

- Definition und Beschreibung der abstrakten Attribute mit den zugehörigen Datentypen.

## 3.3 OPTIONALE KOMponentEN<sup>7</sup>

Um den Nutzen der Schablonen zu maximieren, sind über die oben beschriebenen Komponenten hinaus die folgenden Bestandteile sinnvoll:

### FUNKTIONALITÄT

Zur Erleichterung der Funktionsmodellierung sollten zu einer Schablone auch die anwendbaren Funktionen angegeben werden. Die vorhandenen Varianten spiegeln sich auch in den Funktionen wieder.

### ZUSTANDSMODELL

Zur Beschreibung der Lebensläufe der Elemente dient das Zustandsmodell für das der Schablone zugrundeliegende Konzept (falls anwendbar), sowie gegebenenfalls für die Teilkonzepte einschließlich der definierten Varianten.

### DESIGNKONZEPTE UND IMPLEMENTIERUNG

Zur Erleichterung des Systemdesigns sollten zu einer Schablone auch die möglichen Designkonzepte mit ihren Stärken und Schwächen angegeben werden. Diese können für unterschiedliche Varianten verschieden sein. Aufbauend auf diesen Designkonzepten ist es sinnvoll, vorgedachte Implementierungen anzubieten.

## 3.4 BIBLIOTHEK

Zentrales Werkzeug der Wiederverwendung von Schemata ist die Bibliothek, die die Schablonen enthält und die Infrastruktur zur Wiederverwendung zur Verfügung stellt. Die Qualität der Schema-Schablonen bestimmt nicht nur die

Effizienz und Qualität aller Prozesse zur Wiederverwendung von Schemata, sondern auch die Qualität des Modells der Anwendung als Ganzes.

Qualität in diesem Zusammenhang bedeutet vor allem Qualität der Infrastruktur, d.h. sowohl die Ergonomie der Benutzeroberfläche, als auch der Umfang der angebotenen Funktionalität.

## 4 QUALITÄTSKRITERIEN

Die Qualität einer Schema-Schablone kann durch eine Menge an Anforderungen beschrieben werden, die alle diejenigen Charakteristiken betreffen, die sicherstellen, daß die Wiederverwendung eines Schemas möglich und effizient bzw. effektiv ist.

Die Anforderungen lassen sich teilweise aus den Anforderungen an wiederverwendbare Softwarekomponenten<sup>8</sup> bzw. aus den Kriterien einer guten Softwarearchitektur<sup>9</sup> herleiten. Sie lassen sich in Anforderungen an die Semantik und an die Konstruktion der Schablone gliedern.

Als Beispiel für die folgenden Qualitätskriterien soll ebenfalls die Schablone "Stückliste" dienen.

### 4.1 QUALITÄTSKRITERIEN BEZÜGLICH DER SEMANTIK

#### WOHLDEFINIERTER ANWENDUNGSSEMANTIK

Die Schema-Schablone muß ein Modell eines eindeutig definierten Ausschnittes des Betrachtungsgegenstandes sein, d.h. es muß eine eindeutig definierte Menge von Anwendungsfakten repräsentieren. Dieser Ausschnitt sollte weder zu groß - und damit zu schwer anwendbar -, noch zu klein bzw. trivial, sein um nützliche Bausteine abzugeben.

Das in Abbildung 2 dargestellte Konzept einer Stückliste stellt ein klar abgegrenztes fachliches Konzept dar, das einen angemessenen Umfang hat.

#### PRÄZISE SPEZIFIKATION

Spezifikation bedeutet die Beschreibung dessen, **was** die Schema-Schablone repräsentiert, nicht **wie** sie das tut. Überfragen auf die Schablonen bedeutet dies, daß es nicht nötig sein darf, das interne Datenmodell einer Schablone zu

Typische Funktionen der Stückliste sind:

- Einfügen und Löschen von Komponenten.
- "Explosion" eines Aggregats, d.h. Liste aller elementaren Bestandteile.
- Verwendungsnachweis einer Elementarkomponente.

<sup>7</sup> Auf die Qualitätsaspekte der optionalen Komponenten wird im folgenden nicht näher eingegangen.

<sup>8</sup> Einen Überblick enthält [KRU92].  
<sup>9</sup> Modularisierungskriterien sind beschrieben bei [PAR72a].

kennen, um zu verstehen, wie und wofür sie angewandt werden kann. Eine solche Spezifikation darf nicht nur für die Systemanalytiker, sondern muß auch für die Anwender und Auftraggeber verständlich sein, die die Korrektheit und Konsistenz des Modells zu beurteilen haben.

Die Spezifikation enthält die wesentlichen fachlichen Konzepte und ist auch ohne den Bezug auf das Datenmodell für Fachbereichsmitarbeiter verständlich.

Eine präzise Spezifikation verringert den Dokumentationsaufwand für das konkrete Modell, da sie nur noch in die aktuelle Umgebung übertragen werden muß. Darüber hinaus wird die Qualität der Dokumentation des gesamten Datenmodells als Bestandteil des Fachkonzepts verbessert. Es werden nicht nur einzelne Elemente der Modelle beschrieben, sondern größere Zusammenhänge, die es wesentlich erleichtern, das Modell als Ganzes zu verstehen.

#### MINIMALE KOGNITIVE DISTANZ

Es muß weniger aufwendig sein, eine Schablone zu finden, zu verstehen und anzuwenden, als es ist, den abgebildeten Teil der Anwendung vollständig neu zu modellieren.

Konkret bedeutet das, daß es einfach sein muß, zu den meist unscharf formulierten, fachlichen Anforderungen die passende Schema-Schablone zu finden. Auch dazu dient in erster Linie das Modell des Konzepts und nicht sein Datenmodell, sowie eine präzise Spezifikation mit den Anwendungsregeln.

#### PRÄZISE NUTZUNGSREGELN

Es sind präzise explizite Regeln nötig, die eine Identifizierung derjenigen Ausschnitte der Anwendungswelt erlauben, für deren Modellierung die Schema-Schablone angewandt werden kann. Darüber hinaus sind Regeln für die Anwendung der Schablone selbst sowie für die Grenzen der Anwendbarkeit nötig.

#### HINREICHENDE UNIVERSALITÄT

Eine Schema-Schablone muß hinreichend abstrakt sein, um in möglichst vielen Anwendungsmodellen benutzt werden zu können. Andererseits darf sie nicht zu abstrakt sein, da dadurch die kognitive Distanz zu groß werden kann. Bei sehr abstrakten Schablonen wird darüber

hinaus der Aufwand der Anpassung an ein spezifisches Anwendungsmodell zu groß. Durch eine hohe Universalität in der Anwendung wird die Anzahl der zur Verfügung zu stellenden Schablonen klein gehalten und damit sowohl der Verwaltungsaufwand der Bibliothek selbst, als auch der Suchaufwand innerhalb der Bibliothek verringert.

Das Konzept der Stückliste läßt sich nicht nur auf technische Gegenstände anwenden, sondern auch auf die Zusammenfassung von Firmen zu Konzernen oder auf die Gliederung von Verträgen in Klauseln und Paragraphen.

## 4.2 QUALITÄTSKRITERIEN BEZÜGLICH DER MODELLIERUNG BZW. KONSTRUKTION

### MODELLIERBARKEIT

Die Schema-Schablone muß in dem Formalismus modellierbar sein, der für das Modell der Anwendung verwendet wird. Mindestens aber muß eine Abbildung von dem in der Bibliothek verwendeten Formalismus in den Formalismus der Anwendungsmodelle gegeben sein. Das Datenmodell der Schablone selbst muß den etablierten Qualitätskriterien eines Datenmodells genügen. Die Schablone muß vor allem in sich konsistent sein. Es ist offensichtlich, daß eine Verletzung dieses Kriteriums die Anwendbarkeit in konkreten Situationen stark erschwert, da dann eine Transformation des Formalismus nötig wird.

### ORTHOAGONALITÄT

Die einzelnen Schema-Schablonen müssen so konstruiert sein, daß möglichst viele Kombinationen zwischen ihnen möglich sind. Solche Kombinationen dienen zur Abbildung noch komplexerer Anwendungsprobleme. Auf diese Art kann die Menge der Schema-Schablonen klein gehalten werden, da sich die Anwendungsfälle in der Regel durch solche Kombinationen abdecken lassen. Konsequenz daraus ist, daß jedes abstrakte Konzept nur durch genau eine Schablone abgebildet werden darf, da andernfalls überschneidende Konzepte Widersprüche erzeugen. Jede Schablone muß die Stellen, an denen andere Schablonen eingesetzt werden können, sowie die Minimalanforderungen an die einzusetzende Schablone präzise definieren. Die Beschreibung einer Schablone darf natürlich umgekehrt keinerlei Verweise darauf enthalten, wo sie selbst eingesetzt werden kann. Mann kann sich z.B. neben der Schablone "Stückliste" eine Schablone "Historisierung" vorstellen. Diese beiden Schablonen sind voneinander vollständig unabhängig. Sie können aber zu einer gemeinsamen Schablone "historisierbare Stückliste" kombiniert werden. Andererseits könnte

Beispiel:  
Die Nutzungsregeln im Falle der Stückliste könnten sein:  
• "Die Stückliste sollte bei allen Formen mehrstufiger Aggregation elementarer Bausteine zu einem Aggregat angewandt werden."  
• "Nicht angewandt werden sollte die Stückliste bei einfachen Gruppenbildungen."

die Schablone "Historisierung" mit einer Schablone "mehrdimensionale Tabelle" zu einer Schablone "historisierbare mehrdimensionale Tabelle" kombiniert werden.

#### **KOMPAKTHEIT**

Ein Kriterium für den Umfang einer Schema-Schablone ist die Tatsache, daß in einer Schablone nur solche Konzepte zusammengefaßt sind, die im Sinne einer funktionalen bzw. informalen Kopplung<sup>10</sup> zusammengehören. Dabei handelt es sich um eine weitergehende Bedingung als es die Orthogonalität ist, denn dort wird nur die Überschneidungsfreiheit gefordert. Als Basiskriterium dient die Definition auf einer gemeinsamen Datenmenge. Übertragen auf die Schablone bedeutet dies, daß als notwendige Bedingung das Datenmodell einer Schablone zusammenhängend sein muß, entweder mittels Beziehungen, gemeinsamen Regeln oder durch Integritätsbedingungen. Dadurch wird es unnötig, nur Teilmengen der Schema-Schablonen anzuwenden, die innerhalb dieser Schablonen unter Umständen nur schwer abgrenzbar sind.

#### **VOLLSTÄNDIGKEIT**

Die Schema-Schablone muß genau alle Aussagen über das abgebildete Konzept enthalten, die korrekt und relevant sind. Das beinhaltet die Geschäftsregeln, die die Integrität und die Konsistenz des modellierten Teils des Anwendungsgebiets beschreiben. Diese Regeln definieren sowohl die Einbettung in das Anwendungsmodell, als auch die internen Bedingungen des Konzepts.

Es existieren Regeln, die unabhängig von der konkreten Anwendung sinnvoll sind. Im Einzelfall kann es auch ausreichen, klar festzulegen, welche Regelungen im konkreten Anwendungsfall noch zu spezifizieren sind.

#### **HOHE FLEXIBILITÄT**

Die Schema-Schablone sollte so flexibel sein, daß möglichst viele Veränderungen in der modellierten Anwendungswelt ohne Veränderung oder Austausch der Schablone abgebildet werden können. Solche Flexibilität könnte durch die Definition von Regeln auf der Basis von Datenstrukturen erreicht werden. Auf diese Art können die Regeln teilweise auch von der Mitarbeitern der Fachbereiche gepflegt werden. Solche Datenstrukturen können ebenfalls als Schema-Schablonen aufgefaßt werden.

Es sollte zunächst versucht werden, Änderungen in der Anwendungswelt innerhalb der verwendeten Schablone abzubilden. Je weiter diese Flexibilität geht, desto später wird es nötig, von einer Schablone auf eine ihrer Varianten überzugehen, oder gar den Wechsel zu einer anderen Schablone vornehmen zu müssen. Eine hohe Flexibilität verringert den Prüfungsaufwand, da nicht im kleinsten Detail analysiert werden muß, um festzustellen, ob eine Variante einer Schablone anwendbar ist.

#### **WOHLDEFINIERTE SCHMALE SCHNITTSTELLE**

Eine Schema-Schablone muß ein wohldefinierte Schnittstelle - im oben definierten Sinn - zur Umgebung der Schablone innerhalb des Datenmodells besitzen. Die Breite dieser Schnittstelle sollte minimal sein, d.h. so wenige Relationships und Regeln wie sinnvollerweise möglich, enthalten.

Basierend auf einer wohldefinierten Schnittstelle ist es möglich, eine Schablone durch eine andere mit derselben oder einer kompatiblen Schnittstelle auszutauschen, da die Anzahl der herzustellenden bzw. zu ändernden Verbindungen zwischen der Schablone und dem übrigen Modell gering ist. Ein solcher Austausch, der notwendig werden könnte, wenn Änderungen in der Anwendung die eingebaute Flexibilität der benutzten Schablone übersteigen, wird durch eine schmale Schnittstelle somit wesentlich erleichtert.

#### **VORHANDENSEIN VON VARIANTEN**

Es erscheint nicht sinnvoll, jedes Problem sofort mit einem sehr mächtigen Modell zu beschreiben, da dieses Möglichkeiten bietet, die in vielen Fällen nicht genutzt werden können. Durch den Einsatz einer zu mächtigen Schablone erhöht sich der Aufwand sowohl in der Modellierungsphase als auch bei der späteren Implementierung.

Um die Zahl der Modellierungskonstrukte insgesamt zu beschränken, sollte die Zahl der Varianten ihrerseits beschränkt bleiben. Varianten sollten nur bei substantiellen Unterschieden beschaffen werden, die nicht durch die eingebaute Flexibilität behandelt werden können. Die Suche innerhalb der Varianten einer Schema-Schablone ist, selbst in einer gut organisierten Bibliothek, weniger aufwendig, als die Suche in einer Menge unabhängiger Schablonen, da hier immer nur die Unterschiede zur Basisschablone beschrieben und somit zu beurteilen sind.

10 Eine Definition enthält [BAL82].

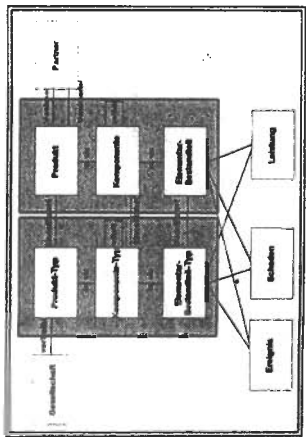


Abbildung 6: Basisvariante

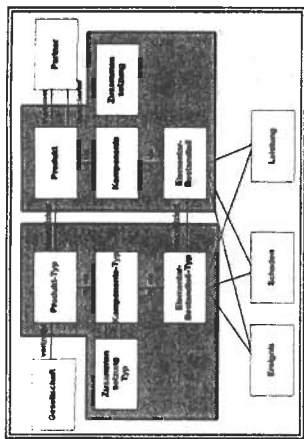


Abbildung 7: Variante mit variablen Beziehungen

Die Basisvariante der Stückliste umfaßt nur die Beziehung "enthält" zwischen den Komponenten, die Variante läßt beliebige flexible und datenmäßig definierbare Beziehungen zwischen den Komponenten zu.

Die Menge der Varianten sollte halbgeordnet sein mit dem Hintergrund einer Aufwärtskompatibilität zwischen den Varianten. Dadurch wird es einfach möglich, eine Variante durch eine leistungsfähigere zu ersetzen, ohne daß das überrige Modell deswegen angepaßt werden muß.

Aufwärtskompatibilität bedeutet hier konkret, daß die leistungsfähigere Variante mindestens das leisten muß, was die Basisvariante kann. Dabei bezieht sich der Begriff "leisten" auf den Umfang der fachlichen Gegebenheiten, die abgebildet werden. Insbesondere müssen mindestens die Relationships und Integritätsregeln definiert sein, die in der Basisvariante vorhanden sind.

## 5 ZUSAMMENFASSUNG UND BEWERTUNG

Im vorliegenden Beitrag wurden die Qualitätskriterien für die Wiederverwendung von abstrakten Teilschemata semantischer Datenmodelle beschrieben und ihr Einfluß auf die Modellierungsprozesse dargestellt. Die zugrundegelegte Vorgehensweise basiert auf einer Bibliothek abstrakter Schema-Schablonen. Die angegebene Qualitätskriterien bestimmen ganz wesentlich die Effizienz und Effektivität des Modellierungsprozesses.

Obwohl noch keine harten quantitativen Daten bezüglich der erzielten oder erzielbaren Einsparungen in den für unser Unternehmen typischen Projekten vorliegen, lassen sich doch auf der vorhandenen Basis folgende qualitative Aussagen zu den Vor- und Nachteilen der vorgestellten Vorgehensweise machen:

- + Der Aufwand der Datenmodellierung des Fachkonzepts ist geringer.

- + Die Qualität der entstandenen Datenmodelle hat sich erhöht. Dabei wird die Schablone oft als "Liste der zu stellenden Fragen" benutzt und damit die vollständige Bearbeitung des Problembereichs sichergestellt.

- + Da die Konzepte der Schablonen als in sich geschlossenes Konzept erläutert werden können, erzielt man eine Verbesserung der Verständlichkeit durch Einführung einer höheren Abstraktionsebene oberhalb der Elementarkonstrukte der Datenmodellierung.

- Der Aufbau und die Pflege der verwendeten Bibliothek sind aufwendig. Dabei wurde vor allem der Aufwand, eine Schemaschablone gemäß den vorgestellten Qualitätskriterien zu beschreiben, zunächst unterschätzt.

- Die Akzeptanz dieser Vorgehensweise litt unter dem "Not Invented Here"-Syndrom, dadurch daß wesentliche Inhalte von außen vorgegeben und nicht im Projekt selbst entwickelt wurden. Die Vorteile sind aber offensichtlich und man kann deshalb von einer wachsenden Akzeptanz ausgehen.

Zusammenfassend läßt sich sagen, daß die Vorteile dieser Vorgehensweise der Modellierung die Nachteile und Probleme bei weitem überwiegen und ein erhebliches Potential für Einsparungen und Qualitätsverbesserung zu erwarten ist. Durch Beachtung der beschriebenen Qualitätskriterien kann der Nutzen der Schemabibliothek vergrößert und der Erfolg dieser Investition sichergestellt werden. Für Beratungsunternehmen wird ein solcher Fundus von Schemaschablonen zukünftig eine hohe Bedeutung für die Wettbewerbsfähigkeit haben.

## 6 LITERATUR

- BAL82 Balzert H., Die Entwicklung von Software-Systemen; Bibliographisches Institut 1982.
- BER93 Bertram M., Aspekte der Qualitätssicherung von Unternehmensdatenmodellen; Proceedings "Wirtschaftsinformatik '93 Münster", pp 230-242
- CDAZ92 Castano S., DeAntonellis V., Zonta B., Classifying and Reusing Conceptual Schema; Proceedings "11th ER-Conference 1992", pp. 121-138.
- COAD92 Coad P., Object Oriented Patterns; Communications of the acm Vol.35(9) 1992, pp.152-159.
- dSH94 ICIS Insurance Company Information System Leistungsübersicht; debis Systemhaus Dienstleistungen GmbH 1994.
- KRU92 Krueger C.W. Software Reuse. acm Computing Surveys; Vol.24 No.2 (1992), pp.131-184.
- PAR72 Parnas D.L. On the Criteria to be used in Decomposing Systems into Modules; Communications of the acm; Vol.15 No.12 (1972), pp. 1053-1058.