

Metamodelle von Referenzmodellen am Beispiel ITIL

Vorgehen, Nutzen, Anwendung

Matthias Goeken, Stefanie Alter, Danijel Milicevic, Janusch Patas

IT-Governance-Practice-Network, Frankfurt School of Finance & Management
Sonnemannstr. 9-11, 60314 Frankfurt am Main
{ m.goeken, s.alter, d.milicevic, j.patas } @ frankfurt-school.de

1 Einleitung

Die Anforderungen an die Unternehmens-IT haben sich über das letzte Jahrzehnt hinweg drastisch verändert. Erhöhte Flexibilität und Adaptivität der IT-Abteilungen in ihren Technologien, Prozessen und Organisationsstrukturen, werden aufgrund neuer Geschäftsmodelle und der Erschließung neuer Märkte gefordert. Als eine Reaktion darauf hat sich die Ausrichtung vieler IT-Abteilungen in Richtung Serviceorientierung verlagert [Be00]. Diese Entwicklungen wurden von der wissenschaftlichen Seite her nicht hinreichend unterstützt [RB96], so dass die Entstehung sogenannter Best-Practice-Referenzmodelle aus der Praxis heraus nicht verwundert. Dennoch ist es überraschend, dass es weiterhin an wissenschaftlichen Methoden zur Bewältigung dieser Herausforderungen im IT-Management fehlt.

Der vorliegende Artikel möchte zur Überwindung dieses Mangels beitragen, indem er an der theoretischen Fundierung von Best-Practice-Referenzmodellen ansetzt. Zunächst werden grundlegende Eigenschaften von Best-Practice-Referenzmodellen und Fragestellungen, die sich in der Praxis aus ihrem Einsatz ergeben, erläutert. Dabei wird die Metamodellierung als eine Möglichkeit, auf Herausforderungen im Umgang mit besagten Referenzmodellen zu reagieren, erörtert. Metamodelle erscheinen als ein wichtiger Ansatzpunkt, den Umgang mit Referenzmodellen wissenschaftlich zu fundieren. Am Beispiel der IT Infrastructure Library (ITIL) wird das Vorgehen zur Modellierung eines Metamodells dargestellt.

2 Best Practice-Referenzmodelle

2.1 Grundlagen zu Referenzmodellen

In den letzten zehn Jahren entstanden verschiedene Best-Practice-Referenzmodelle und sie finden vermehrt Anwendung in IT-Abteilungen [IG08]. Darunter befinden sich offene Referenzmodelle, bspw. ITIL, COBIT, CMMI, VAL-IT oder RISK-IT, sowie proprietäre Referenzmodelle wie das Microsoft Operations Framework (MOF) oder das HP IT-Service Management Reference Model (ITSM) von Hewlett-Packard. Diese und auch weitere Modelle und Standards wie die ISO 2700x, PMBOK und Prince 2 lassen sich als Referenzmodelle der IT-Governance auffassen [JG07]. Ein wesentliches Merkmal solcher Referenzmodelle ist, dass ihre Entwicklung auf der Konsolidierung von Praxiserfahrung beruht. Das so entstehende Modell besteht aus einer Sammlung von Prozessen, Regeln und Organisationsstrukturen, die im Konsens gebildet wurden und insofern breit akzeptiert sind.

Die unterschiedlichen Referenzmodelle legen ihren Schwerpunkt jeweils auf unterschiedliche Aspekte. So widmet sich ITIL dem IT-Servicemanagement, COBIT ist als Modell für IT-Governance positioniert, ohne dabei aber seine Wurzeln im Auditing zu verbergen. CMMI fokussiert v.a. die Systementwicklung und die genannten ISO 2700x Standards unterstützen das Sicherheitsmanagement.

Referenzmodelle sind aufgrund ihrer Entstehung sehr pragmatisch und vernachlässigen wissenschaftliche Aspekte wie den Gebrauch von Formalsprachen, eine stringente Methode bei der Entwicklung oder eine abschließende Validierung. Während diese Aspekte zunächst keinen direkten Einfluss auf die Anwendbarkeit in der Praxis haben, sind sie jedoch ein Hindernis beim weiteren Einsatz dieser Modelle, bspw. bei deren Erweiterung.

2.2 IT Infrastructure Library

Als Beispiel für Best-Practice Referenzmodelle steht ITIL im Mittelpunkt unserer Untersuchung. Als das bekannteste und am weitesten verbreitete Referenzmodell im Bereich IT-Governance [IG08] dient ITIL als eine Art Vorbild für ähnliche Referenzmodelle. ITIL ist ein generisches Modell für das IT-Servicemanagement und macht Vorgaben, indem es durch die Beschreibung von Prozessen, Aktivitäten und Metriken, Anregungen für dessen Implementierung gibt.

In der zweiten Version, die maßgeblich für den heutigen Erfolg von ITIL verantwortlich ist, lag der Fokus hauptsächlich auf den zwei Komponenten “Service Delivery” und “Service Support”, neben “Application Management” und “ICT Infrastructure Management”. Die in 2007 veröffentlichte ITIL v3 reflektiert die fortwährende Weiterentwicklung im Bereich IT-Servicemanagement, die das Hauptaugenmerk weg von einem primär prozessorientierten Modell in Richtung eines vollwertigen Service-orientierten Modell gelegt hat. Das Betätigungsfeld wurde um Gebiete wie IT-Strategie erweitert und die Bestandteile orientieren sich nun am IT-Service-Lebenszyklus, der dem Deming Cycle ähnelt.

Diese Entwicklung wird auch an den strukturellen Änderungen in ITIL deutlich. Insbesondere wirken sich solche Änderungen als Herausforderung für IT-Abteilungen aus, die ihre ITIL-v2-Implementierung nach ITIL v3 erweitern wollen. Während [Ta06] ITIL v3 als ein Add-on zu ITIL v2 beschreibt, zeigen Mappings, dass nicht lediglich eine Erweiterung gegeben ist, sondern neue Bestandteile und Zuordnungen stattfinden, die nun dem IT-Service-Lebenszyklus folgen (Tabelle 1). Dadurch ist ein simples Eins-zu-Eins Mapping nicht mehr möglich [IT08]. Die Prozentzahlen in der linken Spalte beschreiben jeweils den Überlappungsgrad der sieben ITIL v2 Bücher zu den fünf ITIL v3 Büchern, während die Prozentangaben in der Zeile den Anteil neuer Inhalte in ITIL v3 darstellen.

ITIL v2	ITIL v3	SS	SD	ST	SO	CSI
		Service Strategy 70% neu	Service Design 40% neu	Service Transition 40% neu	Service Operation 30% neu	Continuous Service Improvement 70% neu
70%	Service Support		●	●	●	●
70%	Service Delivery		●	●	●	●
40%	App Mgmt		●	●	●	
30%	Software Asset Mgmt		●	●	●	
20%	Sec Mgmt		●	●	●	
40%	Business Perspective	●	●	●		
40%	ICTIM		●	●	●	

Tabelle 1: ITIL v2-nach-v3 Mapping [IT08]

2.3 Herausforderungen im Umgang mit Referenzmodellen

In der Praxis sind Bedarfe und Anforderungen erkennbar, die Fragen zur Handhabung solcher Modelle aufwerfen, und Lösungsansätze erfordern. In diesem Teil werden einige dieser Anforderungen beschrieben und es wird skizziert, wie Metamodelle hier als Unterstützung dienen können.

Bereits die Einführung eines Best-Practice-Referenzmodells im Unternehmen stellt eine Herausforderung dar, da es in der Regel nur selten vollkommen unverändert zum Einsatz kommt. Vielmehr sind häufig Anpassungen an unternehmensspezifische Belange nötig, bspw. dadurch, dass das Modell erweitert oder aber dass Teile weggelassen werden. Die Erweiterung kann u. a. bedeuten, dass ganze Prozesse hinzu kommen [KW08] oder dass neue Metriken, Rollen und Ziele für beschriebene Prozesse definiert werden.

Da in der Praxis oftmals die Referenzmodelle als Ausgangspunkt für die Entwicklung eigener Modelle genutzt werden [IG08], scheint es ratsam, diese Anpassungen – im Sinne von unternehmensspezifischen Varianten – methodisch zu fundieren.

Hierbei können Metamodelle ein erstes unterstützendes Werkzeug sein. Ein Metamodell bei der Anpassung zugrunde zu legen, kann dabei helfen, die vorgenommenen Erweiterungen konsistent zu den vorhandenen Strukturen zu halten. Dies erlaubt bspw., neue Prozesse mit vorhandenen Prozessen angemessen systematisch und strukturell konsistent zu verbinden.

Darüber hinaus werden verschiedene Referenzmodelle häufig parallel eingesetzt. Dies ist zum einen deshalb der Fall, weil unterschiedliche Modelle jeweils der Perspektive unterschiedlicher Stakeholder im Unternehmen entsprechen (CMMI entspricht der Entwickler-/Projektleiterperspektive; COBIT v. a. der Perspektive von internen und externen Auditoren etc.). Zum anderen kann es zweckmäßig sein, die verschiedenen Referenzmodelle im Sinne eines „Best-of-Breed“-Ansatzes parallel zu verwenden. Da sie jeweils unterschiedliche inhaltliche Schwerpunkte haben, können Sie so für einen unternehmensspezifischen Einsatz gemäß ihren jeweiligen Stärken kombiniert werden.

Dass in der Praxis ein Bedarf besteht, vorhandene Modelle miteinander zu kombinieren, zeigt sich an einer Reihe von sog. „Mapping-Initiativen“, bei denen unterschiedliche Referenzmodelle jeweils miteinander in Beziehung gesetzt werden.¹ Dieses In-Beziehung-Setzen geschieht auf der Ebene der konkreten Inhalte, sodass zum einen nicht alle Modellkomponenten einbezogen werden können; zum anderen sind die „Mappings“ sehr detailliert und umfangreich und ihre Praxistauglichkeit nicht unbedingt gegeben (zu methodischen Problemen hierbei siehe [JG07]).

Auch an dieser Stelle können Metamodelle eine Unterstützung bieten. Sie sind eine höhere Abstraktionsebene und so finden Vergleich und Kombination bzw. Integration unterschiedlicher Referenzmodelle nicht mehr nur auf Detailebene statt. Mittels Metamodellen können vielmehr auch die Modellstrukturen systematisch analysiert und verglichen werden und eine Kombination erfolgt konsistent zu diesen zugrundeliegenden Strukturen.

¹ In den Mappings der ISACA (vgl. www.isaca.org/mappings), wird angegeben, welche Komponenten verschiedener Referenzmodelle miteinander korrespondieren bzw. sich ergänzen. Hierbei werden jeweils nur Teile und ausgewählte Komponenten der Modelle in die Betrachtung einbezogen. Gleichzeitig ergibt sich dabei eine enorme Detailtiefe, da die Betrachtung auf Ebene der Modelle selbst erfolgt. Ein Vergleich der den Modellen zugrundeliegenden Strukturen findet in der Regel nicht statt.

Darüber hinaus bieten Metamodelle eine Vergleichsgrundlage und einen Ausgangspunkt für eine Integration, da sie die „einheitliche Darstellung verschiedener Formalismen erlauben“ [St96]. Dies ist hilfreich, da unterschiedliche Referenzmodelle in jeweils unterschiedlichem Maße strukturiert sind. COBIT bspw. folgt streng einer Formularstruktur, während die ITIL-Inhalte im Wesentlichen „prosaisch“ und in Fließtext beschrieben sind.

Um die genannten Anforderungen der Praxis – unternehmensspezifische Anpassung, Erweiterung, Kombination und Integration von Referenzmodellen – erfüllen zu können, scheint ein methodisch gestützter, wissenschaftlicher Umgang mit ihnen ratsam.

3 Metamodellierung

3.1 Modell und Metamodell

In der Informatik werden Modelle genutzt, um von Sachverhalten zu abstrahieren. Die Darstellung eines Sachverhalts als ein Modell ist gewöhnlich einer der ersten Schritte zur Entwicklung einer Applikation. Im Folgenden betrachten wir die Modellierung von Modellen. Ziel dabei ist es, die theoretische Fundierung von IT-Governance-Referenzmodellen zu stärken, indem eine Abstraktion vorgenommen wird.

Wenn der Sachverhalt bzw. der Untersuchungsgegenstand ein Modell ist – also selbst bereits eine Abstraktion – dann werden Modelle von Modellen gebildet. Für gewöhnlich wird ein solches Modell eines Modells als Metamodell bezeichnet. Hieraus ergeben sich dann mehrere Modellebenen (Abbildung 1). [St96] betrachtet wie diese Hierarchien konstruiert sind und prägt dabei den Begriff Metaisierungsprinzip für den Vorgang der Konstruktion, der wiederholt angewendet werden kann: das Metaisierungsprinzip beschreibt „denjenigen Aspekt eines Modells, der in der übergeordneten Modellierungsstufe abgebildet wird.“ Sie betont, dass verschiedene Möglichkeiten bestehen, wie man von den Instanzen der Wirklichkeit zu Modellen und schließlich zu Metamodellen kommt [siehe auch Kü06].

Das Metaisierungsprinzip definiert demnach in einer Modellhierarchie den Abstraktionsmechanismus zur Strukturierung der Objekte der jeweils darunterliegenden Ebene. Der Abstraktionsmechanismus, der in der Informatik am häufigsten genutzt wird, ist die linguistische Abstraktion. Daraus entstehen linguistische Metamodelle, die die Sprache der Modelle auf der tiefer liegenden Ebene beschreiben. Wenn beispielsweise ein ungerichteter Graph benutzt (M2) wird, um einen Teil der Wirklichkeit zu repräsentieren (M0), können auf Modellebene (M1) nur noch Kanten und Knoten genutzt werden. Dadurch kann kein Kontrollfluss abgebildet werden. Insofern zwingt die Ebene Mx, die Objekte auf Ebene Mx-2 entsprechend den vordefinierten Bausteinen zu strukturieren.

Jedoch ist die linguistische Abstraktion nicht die einzige Möglichkeit, die sich im Rahmen der Modellhierarchie ergibt. [KK02] wie auch [KH06] beschreiben die „ontologische Metamodellierung“. Andere Autoren hingegen betreiben Forschung zur sog. „physikalischen Metamodellierung“ [AK03a, GH05]. Für die hier betrachtete Fragestellung ist insbesondere die ontologische Metamodellierung von Bedeutung: Im Gegensatz zur linguistischen Metamodellierung behandelt sie die Abstraktion von Modellelementen gemäß ihres Inhalts [AK03a, AK03b]. Mit der ontologischen Metamodellierung werden auf der Ebene Mx Metatypen definiert, welche die auf Mx definierten Metatypen sowie deren Eigenschaften beschreiben.

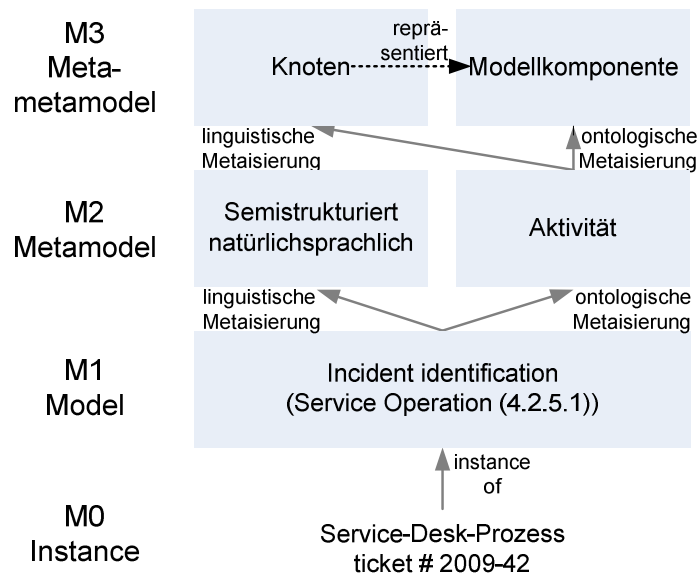


Abbildung 1: Ontologische und linguistische Metamodellierung in Anlehnung an [AK03a]

Die Unterschiede zwischen ontologischer und linguistischer Metamodellierung sind in Abbildung 1 dargestellt, die auch unseren Ansatz demonstriert, der im Folgenden weiter konkretisiert wird. In dem Beispiel befindet sich auf Ebene M0 eine konkrete Aktivität in einem IT-Prozess (ein spezifisches Ticket im Service Desk # 2009-42). Die ausgeführten Aktivitäten sind Instanzen des ITIL-Prozessschrittes „Incident Identification“ innerhalb des „Incident Managements“ im Buch Service Operation. In ITIL sind die Definitionen der Prozesse und ihrer Aktivitäten in Prosa beschrieben und somit ist ITIL semi-strukturiert. In Bezugnahme auf die ontologische Metaisierung kann „Incident Identification“ als Aktivität auf Metamodellebene klassifiziert werden.

Die ontologische Metaisierung von M1 (wo sich ITIL befindet) zur Metaebene M2 (wo sich das hier vorgestellte Metamodell befindet) ist der Kern der Modellierungsaktivitäten und wird im folgenden Abschnitt beschrieben.

Zur Modellierung von Ausschnitten der Realität benötigt man eine Sprache sowie Methoden im Sinne von Vorgehensweisen, die die Identifikation wie auch die

Repräsentation der relevanten Objekte, im Modell unterstützen. Die Sprache kann man als „*way of modelling*“ und die Vorgehensweisen als „*way of working*“ bezeichnen [VHW91]. In der Informatikforschung liegt der Fokus zumeist auf dem „*way of modelling*“, das heißt auf der Sprache bzw. Notation, und nicht so sehr auf der Fragestellung wie aus der Diskurswelt die relevanten Sachverhalte extrahiert werden können, die die wesentlichen Modellinhalte bilden sollen. Nachfolgend legen wir zunächst den Fokus zunächst auf den „*way of thinking*“, indem wir Richtlinien und Prinzipien im Rahmen der Metamodellierung diskutieren. Dies wird als eine Vorbedingung für die Definition des „*ways of working*“ gesehen. Letzterer wird im darauffolgenden Abschnitt 4 beschrieben und vor dem Hintergrund der hier betrachteten Problemstellung beleuchtet.

Beim „*way of modelling*“ sind in dieser Arbeit die Freiheitsgrade gering. Da das Werkzeug Protégé zum Einsatz kommt, wird das Modell als ungerichteter Graph repräsentiert. Insofern konzentrieren wir uns auf die statischen Aspekte des Referenzmodells und sind nicht in der Lage, Steuerungs- und Informationsflüsse direkt einzufangen. An anderer Stelle wurde von den Verfassern ein Metamodell vorgestellt, dass die E/R-Notation verwendet [GA09].

3.2 Grundsätze der Metamodellierung

Wie bereits erwähnt, liegt der Schwerpunkt in der Informatik und Wirtschaftsinformatik häufig auf der Entwicklung von Modellierungssprachen. Die Fragestellung, wie das Wissen der Domäne in das Modell gelangt, wird häufig nicht weiter betrachtet. Auf der anderen Seite kümmert sich die Disziplin „Requirements Engineering“, die sich mit der Erfassung des Wissens der Anwender und Benutzer befasst, häufig nur im geringeren Maße um die semiformale Darstellung mittels Modellierungssprachen. Zwar weisen die beiden Gebiete „Modellierung“ und „Requirements Engineering“ Überlappungen auf [RP00], eine durchgängige methodische Unterstützung ist nicht gegeben.

Um diese Lücke zu füllen, schlagen einige Forscher eine schrittweise Methode zur Metamodellierung vor [KW07], während andere eine epistemologische Beschreibung an den Anfang stellen. [SR98]. Auf letzterer aufbauend, leiten wir im Folgenden einige Grundsätze für die Metamodellierung ab.

Da ein Metamodell als „Modell eines Modells“ definiert ist, können die sogenannten „Grundsätze ordnungsmäßiger Modellierung“ Anwendung finden. [Sc98] und [SR98] schlagen sechs Grundsätze zur Qualitätssteigerung von Informationsmodellen vor. Diese Grundsätze sind:

- (1) Grundsatz der Richtigkeit
- (2) Grundsatz der Relevanz
- (3) Grundsatz der Wirtschaftlichkeit
- (4) Grundsatz der Klarheit

(5) Grundsatz des systematischen Aufbaus

(6) Grundsatz der Vergleichbarkeit

Die Autoren vertreten die Ansicht, dass für die Metamodellierung eine Erweiterung dieser sechs Grundsätze notwendig ist. Um den Schritt von einem Modell zu einem Metamodell zu vollziehen, nutzt der Modellierer den Prozess der Abstraktion. Dieses Prinzip der Metaisierung, z.B. in seinen Formen der ontologischen oder linguistischen Metaisierung, hat immense Konsequenzen für das konstruierte Metamodell (siehe folgende Absätze). Vor allem die ontologische Metamodellierung, welche wir zur Entwicklung unseres Metamodells nutzen, erfordert eine Vielzahl von Entscheidungen durch den Modellierer. Der Nutzer des Metamodells benötigt entsprechend die Information, welches Prinzip der Metaisierung der Modellierer genutzt hat, um das Metamodell zu entwickeln. Der erste Grundsatz der Metamodellierung ist dementsprechend: *Ein Metamodell muss das ihm zugrundegelegte Prinzip der Metaisierung darlegen.*

Das Ziel der Metamodellierung ist, die oben (2.3) genannten Anforderungen aus der Praxis wie unternehmensspezifische Anpassung, Erweiterung, Kombination und Integration von Referenzmodellen methodisch zu unterstützen. Aufgrund dieses Anwendungsfeldes, der verschiedenen Nutzer und dem hohem Abstraktionsgrad, erfordert das Metamodell eine klare Zuordnung zwischen der Diskurswelt und den Wörtern und Symbolen, die es benutzt [FS06, RS03]. Dies bedeutet, dass sprachliche Defekte wie Synonyme und Homonyme minimiert oder entfernt werden. Das folgende Beispiel zeigt die Relevanz der Metamodellierung: Viele Best-Practice-Referenzmodelle verwenden die Komponente „Ziel“. Allerdings hat jedes dieser Referenzmodelle seine eigene Definition für das Wort Ziel mit entsprechend abweichender Semantik. Folglich hat man Komponenten, die zwar denselben Namen tragen, allerdings verschiedenen Bedeutungen innehaben. Falls ein Vergleich zwischen den *Generic Goals* aus dem CMMI Model mit den *Business Objectives* aus ITIL erstellt wird, erkennt man, dass sie die gleiche Bedeutung haben, jedoch unterschiedliche Konzepte verfolgen. Der zweite Grundsatz der Metamodellierung sollte daher lauten: *Ein Metamodell hat eine eindeutige Zuordnung zwischen seiner Diskurswelt und den Wörtern und Symbolen, die das Metamodell beschreiben.*

Als dritten Grundsatz schlagen die Autoren vor: *Ein Metamodell wählt die jeweils stärkste/reichste semantisch Beziehung.* Aufgrund der Tatsache, dass das Modell (Diskurswelt), welches metamodelliert wird, nicht gänzlich klar und konsistent ist, da es nicht systematisch entworfen wurde, steht der (Meta-)Modellierer des Öfteren vor der Entscheidung, zwischen verschiedenen Beziehungen auswählen zu müssen. Es ist evident, dass Beziehungen wie „ist erzeugt durch“ oder „enthält“ mehr Informationen tragen, als ein Beziehungstyp mit einer Konjunktion wie „hat“ oder Beziehungstypen mit ähnlich allgemeinen Bedeutungen. Um allerdings zwischen zwei semantisch angereicherten Beziehungen zu unterscheiden, werden oft weitere Informationen benötigt und – soweit den Autoren bekannt ist – gibt es keine Methode, mit der sich das Konstrukt „semantische Stärke“ messen und vergleichen ließe.

4 Metamodellierung von ITIL

In diesem Abschnitt wird gezeigt, wie ein Metamodell von ITIL im Sinne der oben dargestellten Modellhierarchie und unter Anwendung der skizzierten Prinzipien und Richtlinien konstruiert werden kann. Das Metamodell wird unter Verwendung der ontologischen Metaisierung abgeleitet. Die fünf ITIL-Publikationen bilden den Startpunkt des Metamodellierungsprozesses. Das ITIL-Referenzmodell umfasst insgesamt mehr als 1000 Seiten, was als Indikator für die Komplexität und Schwierigkeit eines detaillierten Vergleichs zwischen ITIL mit anderen Referenzmodellen dient. Daher kann eine Metamodellierung hier entscheidende Vorteile bei der Nutzung von Referenzmodelle mit sich bringen.

Der Modellierungsprozess orientiert sich an dem für das verwendete Werkzeug vorgeschlagenem Prozessmodell (Vgl. Abbildung 2). Die Entscheidung fiel auf das frei verfügbare Werkzeug Protégé, welches sowohl die Modellierung von Ontologien als auch deren Visualisierung unter Einsatz verschiedener Plug-ins erlaubt.² Für die Visualisierung wurden OntoView, Jambalaya und TGViz verwendet. Beim Vorgehen, um eine Ontologie aus den einzelnen ITIL-Publikationen zu extrahieren, wurde den Empfehlungen des Stanford Center for Medical Informatics Research gefolgt [NM01], an dem Protégé entwickelt wird.³ Bei der konkreten Durchführung des Modellierungsprozesses wurden einige Schritte im Sinne der obigen Modellhierarchie modifiziert.

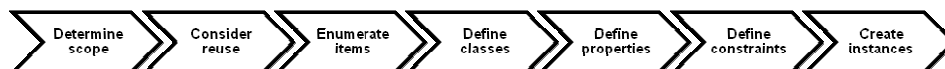


Abbildung 2: Prozess zur Entwicklung von Ontologien [NM01]

Der Entwicklungsprozess ist evolutionär und muss daher wiederholt durchlaufen werden, um die sich schrittweise entwickelnde Ontologie zu analysieren, zu bewerten und zu verfeinern. Letztlich kann so eine reife Ontologie aus dem Untersuchungsgegenstand extrahieren werden. Ein Durchlauf des Prozesses umfasst insgesamt sieben Einzelschritte.

Zu Beginn erfolgt die Definition des Zwecks, den die Ontologie erfüllen soll, insbesondere welche Domäne die Ontologie abbilden soll sowie die Identifikation ihrer Stakeholder. Da hier die Modellierung von ITIL erfolgt, soll die durch dieses Referenzmodell abgedeckte Domäne des IT-Servicemanagements mit den entsprechenden Interessensgruppen repräsentiert werden.

² In dem ebenfalls von den Verfassern durchgeführten Projekt „SemGoRiCo“ wird mit dem professionellen Werkzeug K-Infinity der intelligent views gmbh

³ Alternative Ansätze der Ontologieentwicklung werden in [DMN09] diskutiert und gegenüber gestellt.

Der nächste Schritt umfasst u. a. die Recherche nach bereits existierenden Ontologien zur spezifischen Domäne „IT-Servicemanagement“, bspw. in einschlägiger Literatur oder Ontologiedatenbanken. An dieser Stelle lässt sich [BW07] finden, das ein „Metamodel for IT services“ enthält. Dieses bezieht sich jedoch auf ITIL V2 und kann daher hier nur eingeschränkt Eingang in den Modellierungsprozess finden.

Im dritten Schritt werden relevante Objekte erhoben und erfasst. D. h. es wird danach gefragt, welche Objekte bzw. Begriffe besonders charakteristisch für das ITIL-Referenzmodell sind und welche Eigenschaften diese besitzen. Charakteristische Objekte, das sind die M1-Objekte im Sinne der obigen Modellhierarchie. Diese gilt es in den ITIL-Originaldokumenten zu identifizieren. So findet sich dort bspw. die erwähnte „Incident Identification“ neben „Incident Logging“, „Incident Categorization“ und etc.

Darauf aufbauend wird – im nächsten Schritt – die Definition von Klassen vorgenommen. Die drei genannten M1-Objekte werden nun im Sinne der Modellhierarchie zu M2-Objekten bzw. Klassen typisiert und als „Activity“ definiert. Auf dieselbe Art und Weise lassen sich u. a. die Klassen „Process“, „Role“, „Result“ usw. identifizieren. Weiter werden in diesem vierten Schritt semantische und hierarchische Beziehungen zwischen Klassen definiert. Grundsätzlich kann bottom-up, top-down oder combined vorgegangen werden. In ITIL wird, bspw. aus der Vogelperspektive gesehen, die Klasse „Objective“ durch die Klassen „Business Objective“, „Activity Objective“ etc. verfeinert (siehe Taxonomy, Abbildung 3).

Im fünften Schritt müssen den Klassen Eigenschaften zugeordnet werden, um deren Aussagekraft zu erhöhen. Hier werden den Metriken bestimmte Eigenschaften zugeordnet, wie bspw. die Definition einer Berechnung. Zu beachten bleibt, dass sich die Schritte vier und fünf unter Umständen stark überschneiden können und eine Trennung nicht immer möglich ist.

Der sechste Schritt umfasst die Definition von Constraints, d. h. ob eine Eigenschaft bspw. numerisch ist oder aus Zeichen besteht sowie Kardinalitäten zwischen den Klassen. So umfasst in ITIL die Klasse Metrik numerische Werte und eine Aktivität eine Zeichenkette. Constraints durch Kardinalitäten bedeutet hier bspw., dass einer Klasse mehrere Metriken zugeordnet werden können. Andersherum aber eine 1:n-Beziehung vorliegt.

Schließlich beinhaltet der letzte Schritt das Erzeugen von Instanzen für die definierten Klassen. An dieser Stelle kann auf die Ergebnisse der Bottom-Up-Ableitung der Objekte aus den ITIL-Dokumenten in Schritt drei zurückgegriffen werden. Dieser Schritt wird im Rahmen der Metamodellierung hier nur punktuell durchgeführt, da das Ziel zunächst darin besteht, ein Metamodell zu konstruieren, nicht aber alle ITIL-Prozesse vollständig zu modellieren, d. h. die Ebene M1 wird zunächst nicht in Protégé abgebildet.

Im Ergebnis liegen so definierte Klassen vor, die zum einen in einem hierarchischen Verhältnis zueinander stehen. Das Ergebnis ist eine Taxonomie, die in Abbildung 3 ausschnittsweise dargestellt ist.

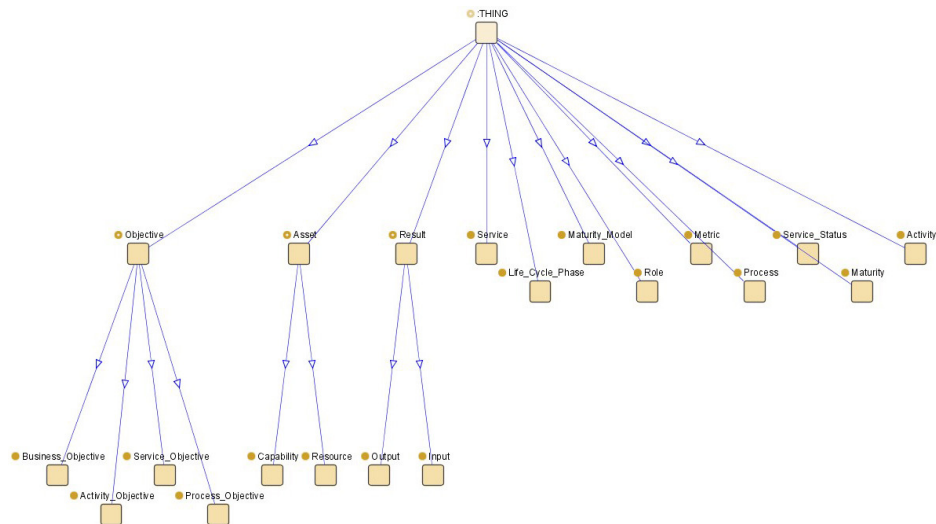


Abbildung 3: Taxonomie der ITIL-Klassen in Protégé

Zum anderen findet sich ein semantisches Netz, welches das Metamodell auf Ebene M2 repräsentiert. Es enthält die semantisch reichen Beziehungen zwischen den Klassen. In Abbildung 4 ist ein großer Ausschnitt des in Protégé abgelegten Metamodells zu sehen. Aus Gründen der Lesbarkeit sind die Beziehungen nicht benannt. Abbildung 5 zeigt einen detaillierteren Ausschnitt und stellt daher auch die Semantik der Beziehungen dar.

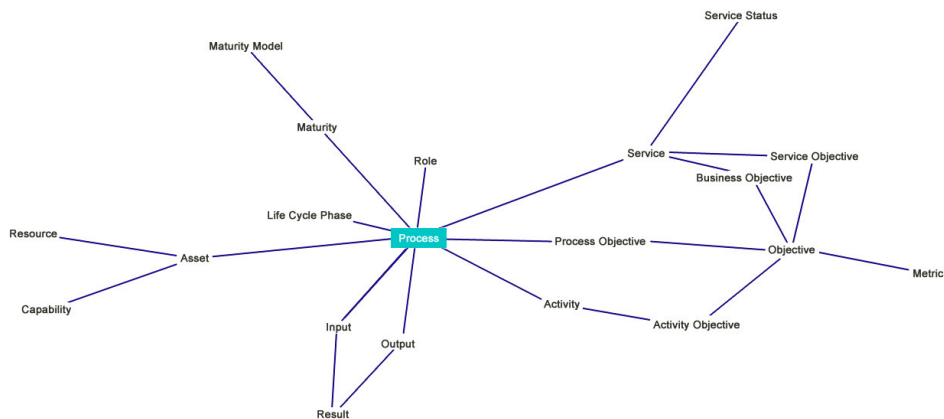


Abbildung 4: ITIL-Metamodell als Graph

Jeder Knoten repräsentiert eine Klasse des ITIL-Metamodells. Die Klasse Prozess kann als die wichtigste Klasse identifiziert werden. Ein starkes Indiz dafür bildet die Anzahl der Kanten, die an diese Klasse andocken. Dies erscheint jedoch nicht sehr überraschend, da Prozesse im ITIL-Referenzmodell bspw. im Gegensatz zu Services besonders umfassend definiert sind. Prozesse kombinieren dabei als zentrales Objekt sowohl Ressourcen mit Rollen als sie auch Services und deren Ergebnisse definieren.

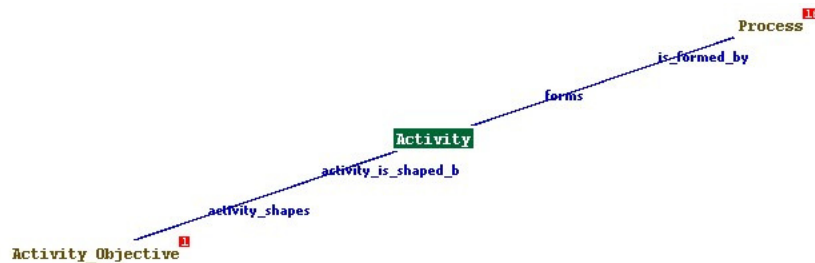


Abbildung 5: Detaillierter Ausschnitt aus dem ITIL-Metamodell

5 Fazit

Die vorgestellte Methode zur Metaisierung von Best-Practice-Referenzmodell, die am Beispiel von ITIL vorgestellt wurde, hat eine Reihe von Anwendungsmöglichkeiten und stiftet Nutzen beim Umgang mit Referenzmodellen. Dabei werden die Modellstrukturen freigelegt, so dass von den konkreten Ausprägungen im Modell abstrahiert wird. Dies unterstützt die Anwendung dadurch, dass sich die Strukturen analysieren, vergleichen und ggf. kombinieren oder integrieren lassen. Insoweit fügt die Betrachtung auf Metaebene den in der Praxis vorzufindenden Mappings eine weitere Sichtweise hinzu.

Weiter sind Metamodelle nötig, um Werkzeuge zur Unterstützung zu bauen. An dieser Stelle wurde die Abbildung in einem Werkzeug zur Repräsentation von Ontologien gezeigt. Dies dient an diesem Punkt der Illustration und dem Experimentieren. Eine weitergehende Werkzeugunterstützung sollte hingegen auf stabileren Plattformen aufsetzen, um den „Betrieb“ von Referenzmodellen im Unternehmensalltag zu gewährleisten.

Des Weiteren sollte die Methode zur Konstruktion von ontologischen Metamodellen erweitert und validiert werden. Eine Erweiterung könnte die Einbeziehung von Verfahren der semantischen Textanalyse (Inhaltsanalyse) aus der qualitativen Sozialforschung oder Text-Mining-Verfahren umfassen. Weiteren Forschungsbedarf sehen wir in der Validierung der GoMM.

Literaturverzeichnis

- [AK03a] Atkinson, C.; Kühne, T.: Calling a Spade a Spade. The MDA Infrastructure, International Workshop “Metamodeling for MDA”, York, November, 2003.
- [AK03b] Atkinson, C.; Kühne, T.: Model-Driven Development: A Metamodeling Foundation. IEEE Software, 20, 5, 36–41, 2003.
- [Be00] Bernhard, M.: Die IT und ihre neue Kernkompetenz – Vom Verwalter zum serviceorientierten Dienstleister. In (Bernhard, M., Lewandowski, W., Mann, H. Hrsg.):

Service-Level-Management in der IT, Düsseldorf, Symposion Publishing, 2000, 253-262.

- [BW07] Braun, C.; Winter, R.: Integration of IT service management into enterprise architecture. In Proceedings of the 2007 ACM symposium on Applied computing, 1215-1219.
- [DMN09] De Nicola, A.; Missikoff, M.; Navigli, R.: A software engineering approach to ontology building. In Information Systems 34, 2, 2009, 258 -279.
- [FS06] Ferstl O.K.; Sinz E.J.: Grundlagen der Wirtschaftsinformatik. 5. Aufl., Oldenbourg, München, 2006.
- [GA09] Goeken, M.; Alter, S.: Towards Conceptual Metamodeling of IT Governance Frameworks. Approach - Use – Benefits. In Proceedings of the 42nd Hawaii International Conference on System Sciences, 2009.
- [GH05] Gitzel, R.; Hildenbrand, T.: A Taxonomy of Metamodel Hierarchies. Working paper, Lehrstuhl für ABWL und Wirtschaftsinformatik, 2005.
- [IG08] IT Governance Institute (Ed.): IT Governance Global Status Report, unter www.isaca.org, Abruf am 03.03.2008.
- [IT08] ITIL.org: ITIL V3-V2 Mapping, unter <http://www.itil.org/de/itilv3-servicelifecycle/itilv3-v2mapping.php>, Abruf am 06.04.2008.
- [JG07] Johannsen, W.; Goeken, M.: Referenzmodelle für IT-Governance. dpunkt-verlag, Heidelberg, 2007.
- [KH06] Karagiannis, D.; Höfferer, P.: Metamodels in action: An overview. ICSOFT, 2006.
- [KK02] Karagiannis, D.; Kühn, H.: Metamodeling Platforms. In (Min Tjoa, A.; Quirchmayer, G. Hrsg.): Lecture Notes in Computer Science, 2455. In Proceedings of the Third International Conference EC-Web, Springer 2002, 451–464.
- [Kü06] Kühne, T.: Matters of (Meta-) Modelling. In Journal on Software and Systems Modeling, 5, 4, 2006, 369-385.
- [KW07] Kurpjuweit, S.; Winter, R.: Viewpoint based Meta Model Engineering. In (Reichert, M.; Strecker, S.; Turowski, K. Hrsg.): Enterprise Modelling and Information Systems Architectures - Concepts and Applications. In Proceedings of the 2nd International Workshop on Enterprise Modelling and Information Systems Architectures, October 8-9, 2007, 143-161.
- [KW08] Kurz, E.; Woltering, A.: ITK-Governance bei der Deutschen Bahn – nachhaltige ITK-Steuerung im Vorstandsressort Personenverkehr. In IT-Governance, 3, 2008, S. 3-8.
- [NM01] Noy, N. F.; McGuinness, D.L.: Ontology Development 101: A Guide to Creating Your First Ontology. Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880, 2001.
- [RB96] Reich, B. H.; Benbasat, I.: Measuring the linkage between business and information technology objectives. In MIS Quarterly, 20, 1, 1996, 75-10.
- [RP00] Rolland, C.; Prakash, N.: From conceptual modelling to requirements engineering. In Annals of Software Engineering, 10, 2000, 151–176.
- [RS03] Rautenstrauch, C.; Schulze, T.: Informatik für Wirtschaftswissenschaftler und Wirtschaftsinformatiker. Heidelberg, 2003.
- [Sc98] Schütte, R.: Grundsätze ordnungsmäßiger Referenzmodellierung. Konstruktion konfigurations- und anpassungsorientierter Modelle. Dissertation, Wiesbaden, 1998.

- [SR98] Schütte, R.; Rotthowe, T.: The Guidelines of Modeling - an approach to enhance the quality in information models. In (Ling, T.W., Ram, S., Lee, M. L. Hrsg.): Conceptual Modeling – ER 98. Singapore, 16.-19.11.1998, 240-254.
- [St96] Strahinger, S.: Metamodellierung als Instrument des Methodenvergleichs. Shaker, Aachen, 1996.
- [Ta06] Taylor, S.: ITIL Version 3. Presentation at itSMF-USA, Salt Lake City, 9-23-2006.
- [VHW91] Verhoef, T. F.; Hofstede, A. H. M. T.; Wijers, G. M.: Structuring Modelling Knowledge for CASE Shells. In (Andersen, R. et al. Hrsg.): Proceedings of the third international Conference CAiSE'91 on Advanced Information Systems Engineering, 502-524.