

Konzept und vergleichende Analyse eines Wissensgraph-basierten Modulkatalogs

Vera G. Meister¹, Jan Malte Beckert²

Zusammenfassung: Modulkataloge gehören zu den wesentlichen konstituierenden und qualitätssichernden Dokumenten für Studiengänge an Hochschulen. Darüber hinaus erfüllen sie valide Informationsbedürfnisse verschiedenster Stakeholder. Grundlegende Qualitäts- und Strukturanforderungen an Modulkataloge werden durch Richtlinien der EU und der nationalen Gremien festgelegt, die jedoch den Akteuren vor Ort einen großen Gestaltungsspielraum lassen. In der Praxis sind Modulkataloge zumeist als Dokumente implementiert, wobei Dokumenten-Management-Systeme die Einhaltung von Qualitätsanforderungen unterstützen. Weitergehende Anforderungen der Stakeholder werden jedoch nicht oder nur unzureichend erfüllt. Auch die Bereitstellung von Modulkatalogen als Feature von Campus-Management-Systemen ist kritisch zu sehen. Der Beitrag analysiert einen Wissensgraph-basierten Ansatz zur Implementierung von Modulkatalogen.

Abstract: Module catalogs are among the most important constituent and quality-assuring documents for study programs at higher educational institutions. In addition, they meet the valid information needs of a wide range of stakeholders. Basic quality and structural requirements for module catalogs are laid down by EU and national guidelines, which leave the local actors a great deal of leeway. In practice, module catalogs are usually implemented as documents, with document management systems supporting compliance with quality requirements. However, further requirements of the stakeholders are not or only insufficiently fulfilled. The provision of module catalogs as a feature of a campus management system should also be viewed critically. The paper analyses a knowledge graph-based approach to the implementation of module catalogs.

Keywords: Wissensgraph, Informationsmanagement, Wissensmanagement, Wissensmodellierung, Modulkatalog, Modulhandbuch, Informationssystem, Campus Management System.

1 Einleitung

Modulkataloge – auch Modulhandbücher genannt – gehören zu den konstituierenden Dokumenten eines jeden Studiengangs an Universitäten und Hochschulen. Während die Studien- und Prüfungsordnung den allgemeinen, formalen Rahmen eines Studiengangs definiert – häufig in Verbindung mit einer Rahmenprüfungsordnung der Hochschule, beschreibt ein Modulkatalog das Curriculum im Detail. Er liefert Informationen zum

¹ Technische Hochschule Brandenburg, Fachbereich Wirtschaft, Magdeburger Str. 50, 14770 Brandenburg a.d.H., vera.meister@th-brandenburg.de

² ebenda, beckert@th-brandenburg.de

Ablauf eines Studiums nach Zielen, Inhalten, Methoden, Ressourcen, Anforderungen, Abhängigkeiten und Wertigkeiten der einzelnen Module eines Studiengangs. Als Modul wird hier eine einzelne, fachlich differenzierbare, durch Prüfung abzuschließende Lerneinheit eines Studiengangs angesehen. (vgl. [Eu09] S. 18, [Be10], S. 15)

Eine Vielzahl gestalterischer, organisatorischer und administrativer Prozesse in der und um die Hochschule stützt sich auf Modulkataloge als zentrale Informationsdokumente. Neben der Akkreditierung eines Studiengangs seien hier vor allem die Orientierung im Studium, die Gestaltung eines Auslandssemesters oder die Anerkennung von außerhalb der Hochschule erworbenen Leistungen genannt.

Erschwerend kommt hinzu, dass Modulkataloge lebende Dokumente sind, ja sein müssen, denn Studiengänge sind keine statischen Gebilde, sondern werden vom verantwortlichen Kollegium stetig weiterentwickelt. Diese Veränderungen können auf Modulebene stattfinden, indem Lehrende Lernziele, -inhalte oder -methoden in Ausübung ihres verfassungsmäßigen Rechts der Freiheit von Lehre und Forschung oder in Orientierung an aktuellen Entwicklungen einer Fachdisziplin anpassen. Sie können auch struktureller Natur sein, wenn Module gestrichen, neu entwickelt oder in ihrer Gewichtung modifiziert werden. So betrachtet gibt es nicht den Modulkatalog für einen Studiengang, sondern es muss differenziert werden im Hinblick auf

- ein konkretes Studienjahr oder -semester,
- eine/n konkreten Studierende/n und seinen/ihren Studienverlauf,
- eine bestimmte Akkreditierung oder Studien- und Prüfungsordnung etc.

Das stellt eine große Herausforderung an das Dokumenten- und Informationsmanagement auf allen eingebundenen administrativen Ebenen der Hochschulen dar, nicht zuletzt im aktuellen Kontext der digitalen Transformation von akademischer Lehre und Verwaltung. Die Erfahrungen aus der langjährigen Mitwirkung der Autorin in Verfahren der Akkreditierung von Studiengängen der Wirtschaftsinformatik sowie aus einer semantischen Textanalyse von Modulkatalogen dieser Disziplin im deutschsprachigen Raum [MJK17] haben gezeigt, dass selbst in einem engen fachlichen Rahmen die Ausprägungsformen von Modulkatalogen sehr stark differieren können.

Dies betrachtend ergeben sich für die vorliegende Arbeit folgende Forschungsfragen:

1. Welche Stakeholder mit welchen Anforderungen an Modulkataloge sind zu unterscheiden?
2. Wie gut setzen aktuell verfügbare IT-Systeme die Anforderungen an Modulkataloge um?
3. Wie wäre ein IT-System für Modulkataloge zu konzipieren, dass die Anforderungen der Stakeholder bestmöglich erfüllt?

Zur Beantwortung der Forschungsfrage 1 werden Methoden der Anforderungs- und Prozessanalyse eingesetzt. Forschungsfrage 2 wird literatur- und fallbasiert sowie argumentativ-deduktiv beantwortet. Für Forschungsfrage 3 wird ein Wissensgraph-basierter

Modulkatalog konzipiert und mit den zuvor analysierten IT-Systemen für Modulkataloge verglichen. Die Autoren lassen sich dabei von der in [RM17] aufgestellten Hypothese leiten, dass Wissensgraph-basierte Informationssysteme klare Mehrwerte für Teilnehmer wissensintensiver Prozesse erwarten lassen. Daraus abgeleitet ist die vorliegende Arbeit wie folgt aufgebaut: In Abschnitt 2 werden die Anforderungen an einen idealen Modulkatalog systematisch zusammengetragen. Abschnitt 3 thematisiert und evaluiert IT-Systeme für Modulkataloge. In Abschnitt 4 werden exemplarisch die aktuellen Prozesse der Pflege und Nutzung von Modulkatalogen im Fachbereich Wirtschaft der TH Brandenburg diskutiert. Dabei werden die Defizite beim Einsatz von Datei- bzw. Dokumenten-Management-Systemen deutlich herausgearbeitet. Die Abschnitte 5 und 6 sind der Entwicklung und Evaluation des Konzeptes eines Wissensgraph-basierten Modulkatalogs gewidmet. Der Beitrag schließt mit einem Fazit und Ausblick.

2 Anforderungen an einen idealen Modulkatalog

Die Analyse von Anforderungen an ein (IT-)System folgt klassischerweise drei Schritten: (i) Identifikation der Stakeholder und ihrer Bedürfnisse, (ii) Erhebung und Priorisierung von Anwendungsfällen und Analyse der zugehörigen Prozesse, (iii) Spezifikation und Formalisierung der Anforderungen ([RS14], [CM10]). Eine vollständige Analyse der Anforderungen würde den Rahmen dieser Arbeit sprengen. Deshalb wird beginnend mit dem zweiten Schritt eine Reduktion auf exemplarische Aspekte vorgenommen.

2.1 Die Stakeholder von Modulkatalogen

Modulkataloge sind keine rein internen Steuerungsdokumente für einen Studiengang. Sie gehören zu den Dokumenten die verpflichtend zu veröffentlichen sind. In [St15, S. 15] heißt es: „Institutions should publish information about their activities, including programmes, which is clear, accurate, objective, up-to date and readily accessible. Information ... is useful for prospective and current students as well as for graduates, other stakeholders and the public.“ Zu den anderen Stakeholdern sind interne Akteure innerhalb der Studiengänge und ihrer Verwaltung sowie externe Träger der Qualitätssicherung zu rechnen. Tab. 1 gibt einen Überblick über die Stakeholder und ihre Bedürfnisse.

2.2 Analyse exemplarischer Anwendungsfälle

In Anlehnung an den Lebenszyklus im Dokumentenmanagement (z. B. [KS14]) lassen sich fünf Phasen im Leben eines Modulkatalogs identifizieren. Die in 2.1 ermittelten Bedürfnisse der Stakeholder adressieren die mittleren Phasen (s. Abb. 1).

	Nr.	Stakeholder	Bedürfnis(se)
Interne	1	Studieninteressierte	- Informationen über alternative Studienangebote
	2	Studierende	- Informationen über zukünftige Module - Dokumentation bislang belegter Module
	3	Interne Lehrende	- Pflege der eigenen Modulbeschreibungen - Informationen über vor-/nachgelagerte Module
	4	Studiengangleitung	- Steuerung der Studiengangentwicklung - Bereitstellung von Dokumentationen
	5	Mitarbeiter des Studiengangs	- Erstellung und Pflege verbundener Medien, z. B. Lehrportale, Webseiten, Broschüren
	6	Prüfungs-/Studierendenamt	- Administration von Prüfungen - Anerkennung/Dokumentation von Leistungen
	7	Leitungsgremien der Hochschule	- Steuerung und Dokumentation der Entwicklung des Fachbereichs/ der Fakultät/ der Hochschule
Externe	8	Alumni	- Dokumentation aller belegten Module - Informationen über Studiengangentwicklung
	9	Öffentlichkeit	- allgemeines Informationsinteresse, z. B. als Steuerzahler, als Bürger der Hochschulstadt etc.
	10	Qualitätssicherung	- Prüfung von Standards/Richtlinien - Erstellung von Qualitätsberichten
	11	Externe Lehrende	- Orientierung/Begutachtung in einer Disziplin
	12	Forscher	- Zugang zu Forschungsmaterial bzw. -daten

Tab. 1: Stakeholder von Modulkatalogen und ihre Bedürfnisse

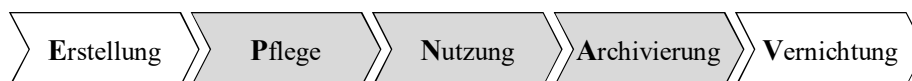


Abb. 1: Management-Lebenszyklus für Dokumente

Auch wenn Modulkataloge nicht zwingend primär als Dokumente gepflegt werden (z. B. bei Einsatz eines Campus-Management-Systems – Genauerer folgt in Abschnitt 3), werden sie doch von vielen Nutzern als solche wahrgenommen. Im Folgenden sollen drei exemplarische Anwendungsfälle näher analysiert werden.

Anwendungsfall A – Pflege von Modulbeschreibungen durch interne Lehrende. Es sind drei Szenarien denkbar: (i) die Lehrenden erhalten Schreibberechtigungen auf ein Dokument oder (ii) in einem System oder (iii) sie arbeiten einer schreibberechtigten Person zu und nutzen dafür idealerweise eine strukturierte Vorlage. Szenario (i) erlaubt keine feingranulare Rechtesteuerung, was zu Zusatzaufwand bei der Qualitätssicherung führen kann. Szenario (iii) kann mit erhöhtem Aufwand bei der Übertragung von Daten oder dem Zusammenführen der Zuarbeiten einhergehen. Szenario (ii) wäre von diesen Män-

geln frei, allerdings könnte im Vorfeld ein enormer Aufwand bei der Anpassung des Systems an unterschiedlich strukturierte Modulbeschreibungen entstehen.

Anwendungsfall B – Aktualisierung eines Studienführers unter Verwendung aggregierter Daten bzw. Informationen aus dem Modulkatalog. In diesem Fall muss zunächst sichergestellt werden, dass eine aktualisierte und abgestimmte Version des Modulkatalogs vorliegt, aus der die notwendigen Daten bzw. Informationen extrahiert werden können. Sofern die Modulbeschreibungen nur Dokumenten-basiert vorliegen, erfordert das eine erhebliche redaktionelle Arbeit bei entsprechend großer Fehlerwahrscheinlichkeit.

Anwendungsfall C – Anrechnung einer Studienleistung für einen anderen Studiengang auf Basis einer Modulbeschreibung. Das setzt voraus, dass zum Zeitpunkt der Anrechnung die entsprechende Modulbeschreibung verfügbar ist. Konsequenterweise müssten die jeweils gültigen Modulbeschreibungen für jede Kursinstanz persistent gespeichert werden. Dafür müssten zumindest in den Szenarien (i) und (iii) dedizierte Archivierungsprozesse implementiert werden.

2.3 Ableitung kritischer Anforderungen an Modulkataloge

Wie in 2.1 eingangs aus [St15] zitiert, sollen Informationen über Studiengänge klar, präzise, objektiv, aktuell und leicht zugänglich bereitgestellt werden. Zugleich stehen Hochschulen immer knappere personelle Ressourcen zur Verfügung. Die Informationen müssen somit sowohl in der erwarteten Qualität als auch unter effizientem Ressourceneinsatz gepflegt und bereitgestellt werden. Aus 2.2 lassen sich in Tab. 2 die folgenden exemplarischen funktionalen Anforderungen an einen Modulkatalog ableiten:

Nr.	Funktionale Anforderungen (Teilmenge)	Stakeholder	Phase
FA1	Lehrende sollen dedizierte Editierrechte für die von ihnen verantworteten Module erhalten.	3	P
FA2	Studiengangverantwortliche sollen die Struktur des Modulkatalogs erweitern oder anpassen können.	4, 7	P
FA3	Für jedes Semester/Studienjahr sollen die Instanzen von Modulkatalogen gespeichert werden.	2, 6, 8	A
FA4	Die aktuell gültige Version des Modulkatalogs soll auf der Studiengangwebseite veröffentlicht werden.	alle	N
FA5	Studierende sollen sich den Katalog der von ihnen individuell belegten Module herunterladen können.	2, 8	A
FA6	Studiengangmitarbeiter Daten aus Modulkatalogen für Info-Materialien effizient nachnutzen können.	1, 5, 9	N

Tab. 2: Funktionale Anforderungen an Modulkataloge (Auswahl)

3 IT-Systeme für Modulkataloge

Eine Untersuchung zum Einsatz von IT-Systemen für das Management des Lebenszyklus von Modulkatalogen an Hochschulen konnte nicht in der Literatur gefunden werden und wurde auch von den Autoren nicht angestrebt. Die Vorgaben aus europäischen und nationalen Richtlinien und Standards beschränken sich auf den Mindestumfang an Informationen [Be10] und auf die Informationsqualität [St15]. Die Ressourcenknappheit an Hochschulen wurde bereits angesprochen. In diesem Kontext ist auch die generelle Vorgabe für das Verwaltungshandeln im öffentlichen Dienst zu erwähnen, die nach § 7 der BHO den Grundsätzen der Wirtschaftlichkeit und Sparsamkeit zu folgen hat [Bu17]. Beide Aspekte lassen den Hochschulen einen großen Gestaltungsspielraum im Hinblick auf Struktur und technische Umsetzung von Modulkatalogen.

Ein Blick auf die Angebotsseite zeigt, dass einige Hersteller von Campus-Management-Systemen (CMS) die Unterstützung von Modulkatalogen im Rahmen ihres Portfolios [Au17, S. 41] anbieten, allerdings kann die Diskrepanz zwischen hochschulinterner Gestaltung von Modulkatalogen und technischer Implementierung im CMS so groß sein, dass diese Funktionalität nicht oder zumindest nicht primär eingeführt wird. Kaum praktikabel erscheint die Eigenentwicklung eines Datenbank-basierten Systems, da Struktur, Pflege- und Nutzungsprozesse für Modulkataloge auch innerhalb einer Hochschule von Studiengang zu Studiengang stark voneinander abweichen können. Es bleibt zu konstatieren, dass die überwiegende Mehrheit von Modulkatalogen mit Dokumenten-basierten Systemen erstellt, gepflegt, bereitgestellt und archiviert wird. Die Palette an technischen Ausprägungen reicht hier von Standard-Office-Systemen in Verbindung mit lokalen oder zentralen Dateisystemen über Content-Management-Systeme bis zu Dokumenten-Management-Systemen (DMS). Tab. 3 gibt einen Überblick über die Unterstützungsqualität der Ausprägungsformen: Dateisystem, DMS und CMS. Die Evaluation erfolgt im Hinblick auf die (ersten vier) Phasen des Dokumenten-Lebenszyklus sowie auf die in 2.3 ermittelten exemplarischen Anforderungen.

Zusammenfassend lässt sich sagen, dass keines der evaluierten IT-Systeme eine hohe Unterstützungsqualität bei vertretbarem Aufwand bietet. CMS erlauben zwar potenziell ein hohes Maß an Unterstützungsqualität, jedoch um den Preis eines hohen Anpassungsaufwandes, der nicht von Fachverantwortlichen geleistet werden kann und zudem die Nachhaltigkeit des gesamten Systems beeinträchtigen kann. DMS unterstützen nur die Erstellung, Versionierung und Archivierung, nicht jedoch die Pflege durch Mitarbeiter und Lehrende. Nur drei der sechs exemplarischen Anforderungen werden erfüllt. Am schlechtesten schneiden erwartungsgemäß Dateisysteme ab. Ihr wesentlicher Vorteil liegt in der niedrigschwelligen Nutzbarkeit.

IT-System	Unterstützungsqualität				
	Dokumenten-Lebenszyklus		exempl. Anford.		
			+	○	-
Dateisystem	Erstellung	einfach und flexibel	FA2	FA3 FA4	FA1 FA5 FA6
	Pflege	problematisch, fehlerbehaftet			
	Nutzung	Versionierung schwierig, Individualisierung nicht möglich			
	Archivierung	unsicher			
DMS	Erstellung	einfach und flexibel	FA2 FA3 FA4		FA1 FA5 FA6
	Pflege	problematisch, fehlerbehaftet			
	Nutzung	Versionierung wird unterstützt, Individualisierung nicht möglich			
	Archivierung	sicher			
CMS	Erstellung	und Anpassung sehr aufwändig	FA3 FA5	FA1 FA4 FA6	FA2
	Pflege	ev. hohe Lernkurve, Lizenzkosten			
	Nutzung	Versionierung wird unterstützt, Individualisierung durch Abfragen			
	Archivierung	sicher			

Tab. 3: Evaluation der Unterstützungsqualität von IT-Systemen für Modulkataloge

4 Fallstudie: Modulkataloge im Fachbereich Wirtschaft der THB

Die Technische Hochschule Brandenburg (THB) gehört zu den kleinen Hochschulen mit klarem regionalen Entwicklungsauftrag. Eine aktuelle Evaluation der Verwaltungs-IT an Brandenburger Hochschulen, die vom zuständigen Ministerium für Wissenschaft, Forschung und Kunst beauftragt wurde, bescheinigt der Hochschule eine insgesamt positive Wahrnehmung der IT-Infrastruktur. Tatsächlich gehört die THB trotz substanzieller Ressourcenengpässe zu einem der Vorreiter bei der Einführung eines integrierten CMS. Das betrifft nach aktueller Planung jedoch (noch) nicht die Bereiche der Lehr- und Veranstaltungsplanung sowie der Modulbeschreibungen. Tatsächlich wird hier mit einem historisch gewachsenen, bunten Flickenteppich an IT-Systemen und Medien gearbeitet.

Der Prozess der Pflege und Bereitstellung von Modulkatalogen im Fachbereich Wirtschaft soll hier exemplarisch dargestellt werden. Bei der initialen Akkreditierung der Studiengänge Mitte der 00-er Jahre wurde eine Struktur entwickelt und als Word-Template im gesamten Fachbereich ausgerollt. Jedes Modul wird in einer standardisierten Tabelle beschrieben. Im Studiengang Wirtschaftsinformatik erhalten die Lehrenden Schreibrechte im DMS für die Modulkatalog-Dokumente (MS Word). Damit greifen alle Lehrenden auf den gesamten Modulkatalog zu. Die Integrität von Änderungen am Dokument wird technisch durch die Funktionalität des Ein- und Auscheckens sowie durch Versionierung sichergestellt. Die Steuerung des Prozesses und die Veröffentlichung der

aktuellen Version der Modulkataloge als PDF-Dokument über die Studiengangwebseite³ obliegt der Studiengangleitung bzw. ihren Mitarbeitern. Die Lehrenden im Studiengang BWL erhalten vorausgefüllte Modulbeschreibungen zu den von ihnen verantworteten Modulen (Word-Tabelle) per E-Mail zugesandt. Mitarbeiter der Studiengangleitung sammeln diese ggf. geänderten Dokumente ein und führen sie zu Modulkatalogen zusammen, die dann im DMS archiviert und auf der Studiengangwebseite veröffentlicht werden. Es kommen somit die in 2.2, Anwendungsfall A beschriebenen Szenarien (i) – im Studiengang Wirtschaftsinformatik – und (iii) im Studiengang BWL zum Einsatz.

Die Schwachpunkte dieses DMS-basierten Managements von Modulkatalogen wurden in Tab. 2 bereits allgemein dargestellt und werden hier konkretisiert:

- Zu FA1: Die Pflege von Modulkatalogen im DMS ist problematisch, weil entweder viele Lehrende auf dasselbe Dokument zugreifen (Szenario i) und damit Fehler entstehen können oder die Sicherung der Dokumentenqualität außerhalb des DMS vorgenommen wird (Szenario iii) und damit sehr aufwändig ist.
- Zu FA 5: Die Modulkataloge werden im DMS als aggregierte Dokumente geführt. Damit wird eine individuelle Ausgabe nicht unterstützt.
- Zu FA 6: Die Modulkataloge liegen als schwach strukturierte Texte vor. Eine differenzierte Auswertung der Informationen ist nur über Methoden der Textanalyse möglich. Die gezielte Nachnutzung durch Nicht-Informatiker wird somit nicht unterstützt.

5 Konzept eines Wissensgraph-basierten Modulkatalogs

Ein Wissensgraph als technische Basis eines IT-Systems stützt sich auf eine formale, standardspezifizierte Konzeptualisierung der betreffenden Wissensdomäne. Weiterführende Definitionen finden sich in [Pa16] und [JM17]. Vergleichbar zu einem CMS würde ein Wissensgraph-basierter Modulkatalog also auf einem formalen Datenmodell fußen, allerdings wäre dieses Datenmodell nicht nur flexibel erweiterbar, ggf. sogar durch Nicht-Informatiker, sondern auch semantisch reich und damit unter Nutzung offener Schnittstellen umfassend nachnutzbar und auswertbar. Schließlich lassen sich im Vergleich zu einer relationalen Datenbank auch stark vernetzte Strukturen gut abbilden.

Bei der Konzeption eines solchen Systems sind drei wesentliche Schritte begleitet von drei Supportprozessen (Dokumentation, Evaluation und Domänenanalyse) durchzuführen [JM17]. Schritt 1 umfasst die Anforderungsanalyse. Sie wurde bereits zusammen mit einer Domänenanalyse in Abschnitt 2 für Modulkataloge an Hochschulen vorgenommen. Schritt 2 adressiert die Konzeptualisierung und im vorliegenden Ansatz zugleich wesentliche Aspekte der Dokumentation. Er ist Gegenstand dieses Abschnitts und führt die Fallstudie aus Abschnitt 4 fort. Schritt 3 betrifft die Implementierung. Sie wird als Proof of Concept in Abschnitt 6 beschrieben und einer initialen Evaluation unterzogen.

³ siehe z. B.: <https://wirtschaft.th-brandenburg.de/studium/bachelorstudiengaenge/wirtschaftsinformatik-bsc/>

In der Konzeptualisierung geht es zunächst darum, die im Hinblick auf das Entwicklungsziel wesentlichen Konzepte (Entitätstypen, Klassen) einer Domäne zu identifizieren. Neben den Konzepten selbst müssen die Beziehungen (Relationen) zwischen den Konzepten sowie deren Eigenschaften (Attribute) identifiziert werden, auch hier mit Fokus auf das Entwicklungsziel. Es bestätigt sich immer wieder, dass ein iteratives Vorgehen unausweichlich ist. Selbst bei sehr gründlicher Kenntnis und Analyse der Domäne entsteht immer wieder Refactoring-Bedarf, z. B. weil die Relevanz von Konzepten, Relationen oder Attributen falsch eingeschätzt wurde oder weil deren Spezifikation aus Sicht des IT-Systems angepasst werden muss. Glücklicherweise erlauben Wissensgraphbasierte Systeme ein Refactoring zu großen Teilen auch im laufenden Betrieb. Zumindest Ergänzungen sind vollkommen unkritisch. Ausgangspunkt für die vorliegende Konzeptualisierung war das in Abschnitt 4 erwähnte Word-Template für eine Modulbeschreibung. Tab. 4 zeigt einen Ausschnitt einer solchen Modulbeschreibung.

Modul-Kurzkennzeichen:	GPzM
Modulbezeichnung:	Grundlagen der Prozessmodellierung
ggf. Aufteilung in LV:	Vorlesung/ Übung/ Projekt
Dauer des Moduls:	Einsemestrig
Zuordnung zum Curriculum:	WI Ba, 2. Semester, Pflichtmodul
Verwendbarkeit des Moduls:	Dient der Vorbereitung aufbauender Veranstaltungen
Häufigkeit des Angebots:	Jedes Studienjahr
Verantwortlich:	Prof. Dr. Vera G. Meister
Dozent/in:	Prof. Dr. Dietmar Wikarski, Prof. Dr. Vera G. Meister
Lehrsprache:	Deutsch, Englisch
Voraussetzungen:	Systemanalyse
ECTS-Credits:	5

Tab. 4: Ausschnitt aus einer Modulbeschreibung im Studiengang Wirtschaftsinformatik

Als zentrale Konzepte (Knoten im Wissensgraphen) wurden Modul (Veranstaltungskonzept) und Modulinstanz (in Zeit und Ort definierte Veranstaltung), Person, Organisation und Literatur identifiziert. Außerdem wurde deutlich, dass eine Reihe von Entitäten als Bezugsobjekte bzw. als Listenobjekte fungieren müssen, z. B. die Lernziele und -inhalte sowie die Zuordnungen zu Studiengängen und Semestern. Alle anderen Strukturelemente der Vorlage wurden als Relationen bzw. als Attribute klassifiziert. Zugleich ist es von größter Relevanz, dass für die identifizierten Konzepte, Relationen und Attribute geeignete technische Namen (URIs) aus standardisierten, veröffentlichten und weit verbreiteten Vokabularen gefunden werden. Dabei rückt für alle Themen, die über Webseiten publiziert werden, schema.org immer mehr in den Fokus. Im vorliegenden Fall konnten alle spezifischen, Semantik tragenden Graph-Elemente auf das schema.org-Vokabular gemappt werden. Ergänzend werden nur Elemente der Standardvokabulare des W3C RDF, RDFS und OWL genutzt, um grundlegende Strukturen und Dokumentationen zu erzeugen. Tab. 5 zeigt einen Ausschnitt des vorbereitenden Mappings, während Abb. 2 das Schema des Wissensgraphen visualisiert.

Strukturelement	Kategorie	schema.org-Mapping
Modul-Kurzzeichen	Attribut von Modul	courseCode
Modulbezeichnung	Attribut von Modul	name
ggf. Aufteilung in LV	Attribut von Modul	learningResourceType
Dauer des Moduls	Attribut von Modulinstanz	duration
Zuordnung zum Curriculum	Zuordnungsrelationen	educationalAlignment
Verwendbarkeit des Moduls	Attribut von Modul	educationalUse
Häufigkeit des Angebots	Attribut von Modulinstanz	courseMode
Verantwortlich	Relation von Modul zu Person	accountablePerson
Dozent/in	Relation von Modulinstanz zu Person	instructor
Lehrsprache	Attribut von Modul	inLanguage
Voraussetzungen	Attribut von Modul/ Relation von Modul zu Modul	coursePrerequisites / isBasedOn
ECTS-Credits	Attribut von Modul	educationalCredentialAwarded

Tab. 5: Ausschnitt aus dem Mapping des Word-Templates auf schema.org-Elemente

Der folgende kleine Auszug aus dem Programmcode zeigt exemplarisch die Deklarationen der Klasse `schema:Course` (Modul), der Relation `schema:accountablePerson` (Verantwortlich) und des Attributs `schema:courseCode` (Modul-Kurzzeichen). Die Serialisierung erfolgt in TURTLE, einem weiteren W3C-Standard.

```

schema:Course a owl:Class ;
  rdfs:label "Course"@en, "Modul"@de ;
  rdfs:subClassOf schema:CreativeWork ;
  rdfs:comment "zentrale Klasse für die Module ..."@de .

schema:accountablePerson a owl:ObjectProperty ;
  rdfs:label "accountable person"@en, "Verantwortlich" @de ;
  rdfs:comment "wird verwendet, um Verantwortliche ..."@de ;
  rdfs:domain schema:Course ;
  rdfs:range schema:Person .

schema:courseCode a owl:DatatypeProperty ;
  rdfs:label "course code"@en, "Modul-Kurzzeichen"@de ;
  rdfs:comment "Kurzzeichen für ein Modul"@de ;
  rdfs:range schema:Text .

```

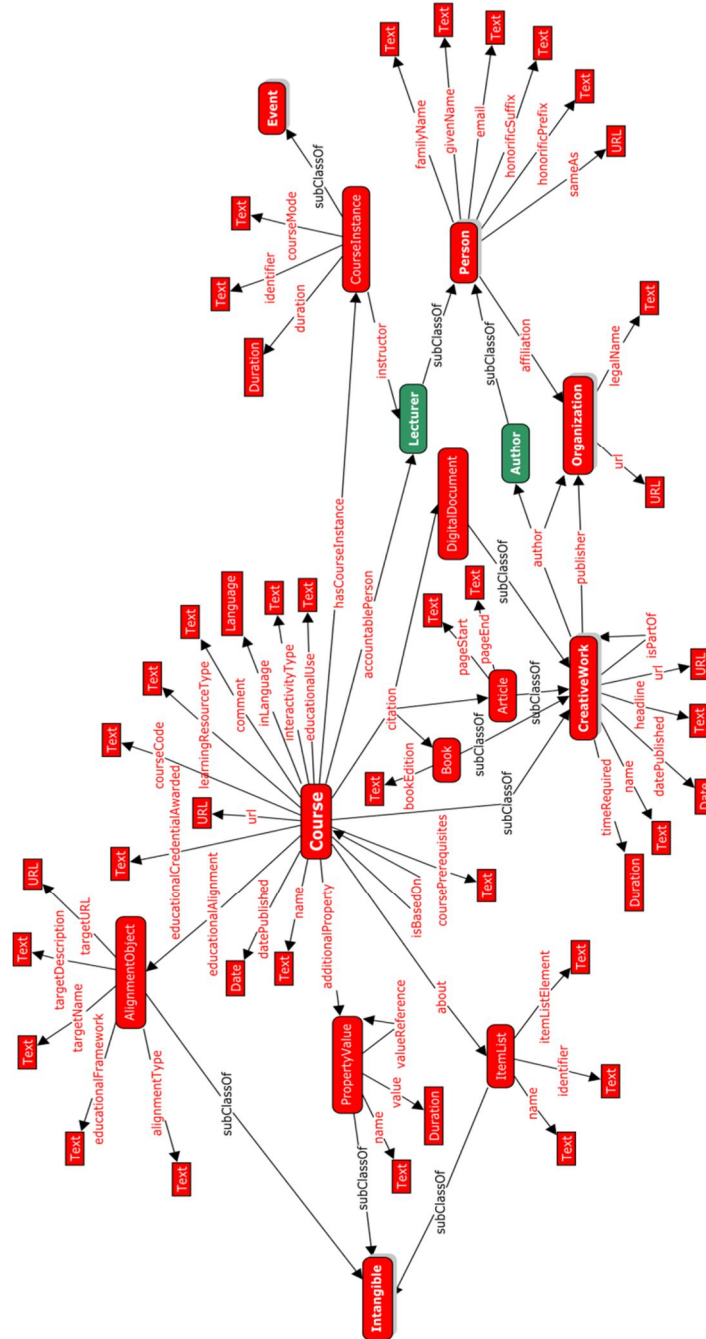


Abb. 2: Schema des Wissensgraphen für Modulkataloge im FB Wirtschaft der THB

Mit Hilfe der Relationen `rdf:type` (Kurzform `a`), `rdfs:subClassOf`, `rdfs:domain` und `rdfs:range` wird die Struktur des Wissensgraphen gebildet, während die Attribut-Statements mit `rdfs:label` und `rdfs:comment` die Dokumentation sicherstellen. Die Konzeptualisierung der Domäne ist damit abgeschlossen. Eine exemplarische Implementierung wird im folgenden Abschnitt beschrieben und gegen die in Abschnitt 2 aufgestellten Anforderungen evaluiert.

6 Proof of Concept

Neben dem konstituierenden Strukturelement – dem Wissensgraphen selbst – benötigt ein Wissensgraph-basiertes System mindestens noch eine RDF-Datenbank (Triple Store), Schnittstellen für die Datenpflege und -bereitstellung sowie Komponente(n) die die entsprechende Geschäftslogik implementieren. Inzwischen kommen immer mehr Systeme auf den Markt, die diese Komponenten vollständig integriert mitbringen bzw. bei Bedarf mit weiteren IT-Diensten per Standard-API kommunizieren können. Ein sehr modernes System dieser Art wird von der Leipziger Firma *eccenca GmbH* unter dem Produktnamen *Corporate Memory*⁴ entwickelt. Dieses System erlaubt die parallele Verwaltung verschiedener Wissensgraphen sowie der darauf basierenden IT-Dienste. Für einen Proof of Concept ist es besonders geeignet, da es eine Reihe von Plugins und Features mitbringt, die die Implementierung beschleunigen. Für die Evaluation werden die drei ersten Phasen des in Abschnitt 2 eingeführten Lebenszyklus einzeln diskutiert.

6.1 Erstellung eines Modulkatalogs

Mit Upload des Wissensgraphen als RDF-File wird bereits die interne Datenstruktur für den Modulkatalog angelegt. Nun ist zu klären, wie die Daten aus den vorhandenen Dokumenten migriert werden können. Dafür kann ein Skript auf Basis von regulären Ausdrücken in Verbindung mit SPARQL 1.1 verwendet werden. Hier handelt es sich um eine vom W3C standardspezifizierte Abfrage- und Update-Sprache für RDF-Datenbanken. Im Proof of Concept wurden die Daten für eine kleine Anzahl strukturell verschiedener Module zunächst manuell in TURTLE serialisiert und in die Datenbank geladen. Auch wenn das eine mühevollen Arbeit darstellt, so dient sie doch zugleich sehr effektiv der Validierung des Wissensgraphen. Die Bereitstellung eines graphischen Benutzer-Templates kann sehr elegant über eine SHACL-basierte Systemfunktion definiert werden. SHACL wurde 2017 vom W3C zur Validierung von RDF-Graphen gegen eine Reihe von Bedingungen spezifiziert und basiert selbst auf RDF. Abb. 3 zeigt einen Ausschnitt der so erzeugten Benutzeroberfläche mit einer Reihe von Attributen zu einem exemplarischen Modul sowie einer Relation zu einem vorausgehenden Modul.

⁴ <https://www.eccenca.com/en/products/eccenca-corporate-memory.html>

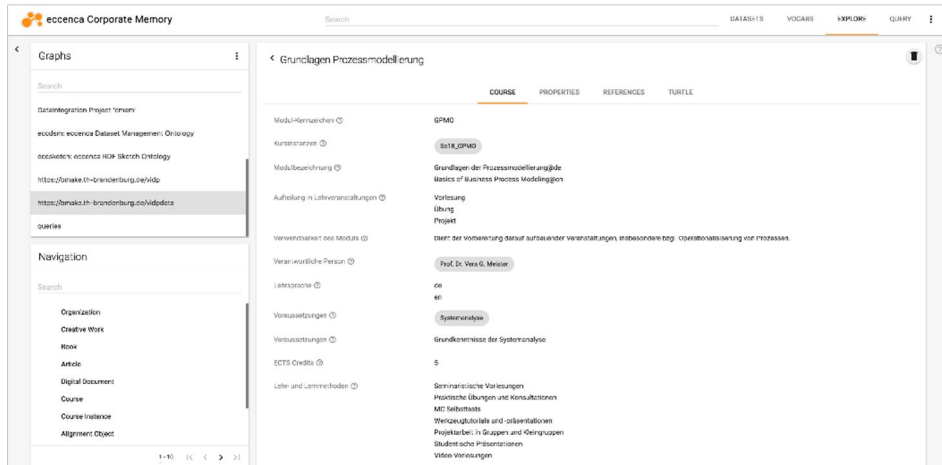


Abb. 3: Ausschnitt aus der Benutzeroberfläche des implementierten Modulkatalogs

Das betrachtete System unterstützt effektiv die Erstellung und Versionierung eines Modulkatalogs. Die Anforderung FA2 wird teilweise umgesetzt. Ein Studiengangverantwortlicher könnte notwendige Änderungen grafisch spezifizieren. Eine Implementierung würde jedoch die Arbeit eines Knowledge Engineers erfordern.

6.2 Pflege des Modulkatalogs

In Abb. 3 sind neben allen Strukturelementen des Modulkatalogs Fragezeichen-Symbole zu sehen, die die im Wissensschema hinterlegten Kommentare anzeigen. Sie unterstützen einen Lehrenden bei der Pflege des Modulkatalogs. Ein mit Schreibrechten ausgestatteter Benutzer des Systems ist in der Lage, die Daten zu verändern, zu ergänzen oder anzupassen. Inwiefern das auf Entitätsebene gesteuert werden kann, sodass Lehrende nur auf die von ihnen verantworteten Module zugreifen können, muss noch erforscht werden. Auch hier erscheint SHACL als probates Mittel. Das System erfasst alle Änderungen und Versionen und erlaubt eine Zurückführung auf frühere Stände. Abb. 4 zeigt einen kleinen Ausschnitt eines Benutzerdialogs mit Schreibrechten und dabei insbesondere die Auswahl bereits angelegter Entitäten bei der Spezifikation einer Relation.



Abb. 4: Ausschnitt Benutzeroberfläche im Editier-Modus

Das betrachtete System unterstützt effektiv die Pflege eines Modulkatalogs. Die Anforderung FA1 ist dann vollumfänglich erfüllt, wenn die feingranulare Rechtsteuerung

implementiert wird. Anforderung FA3 ist erfüllt, da zu jedem Modul die jeweiligen Instanzen mit referenziert werden. Das erlaubt zugleich eine sichere Archivierung.

6.3 Nutzung des Modulkatalogs

Eine initiale Nutzung des Modulkatalogs ist bereits über Leserechte im Corporate Memory realisierbar. Für die Realisierung der Anforderungen FA4 und FA6 sind weitere Dienste an das System anzubinden. Geeignet ist hier das Jekyll-RDF Plugin, welches das Erstellen von benutzerdefinierten Inhalten für Webseiten und andere Dokumente erlaubt. Der folgende Code-Ausschnitt zeigt das Zusammenspiel verschiedener Web-Technologien zur Erstellung von Dokumenten auf Basis gespeicherter Daten.⁵

```
<tr><th>Literature:</th><td>
{% assign query = ' SELECT DISTINCT ?citation
  (group_concat(?authorName;separator=", ") as ?authors)
WHERE { ?resourceUri schema:citation ?citation .
  ?citation schema:author ?author . ?author rdfs:label ?authorName . }
GROUP BY ?citation ORDER BY ?author ' %}
{% assign resultset = page.rdf | sparql_query: query %}
{% for result in resultset %}
<li> {{ result.authors }}
      {{ result.citation | rdf_property: 'schema:headline' }}
      {{ result.citation | rdf_property: 'schema:bookEdition' }},
      {{ result.citation | rdf_property: 'schema:datePublished'}}. </li>
{% endfor %} </td></tr>
```

Um die FA5 umzusetzen, muss das System an das CMS angebunden werden. Für das an der THB eingesetzte HISinOne sollte das grundsätzlich möglich sein.

7 Fazit und Ausblick

Zusammenfassend sind die Forschungsfragen des Beitrags wie folgt zu beantworten:

1. Mit Methoden der Literatur- und der Systemanalyse wurden zwölf interne und externe Stakeholder für Modulkataloge für Studiengänge an Hochschulen identifiziert. Die Anforderungen adressieren zum einen die Phasen eines Dokumenten-Managements, zum anderen gibt es eine Reihe von Nutzungsanforderungen, wie Information, Dokumentation, Nachnutzung und Publikation.
2. Aktuell verfügbare IT-Systeme auf Dokumenten- oder Datenbankbasis setzen die Anforderungen nur unzureichend oder mit unverhältnismäßig hohem Aufwand um.
3. Die Annahme, ein Wissensgraph-basierter Modulkatalog würde den Stakeholdern substanzielle Mehrwerte bieten, konnte initial bestätigt werden. Konkrete

⁵ Der gesamte Code findet sich unter: <https://github.com/bmake/modulcat-sheet-generator/>.

Aussagen zur Einsparung an Erstellungs- und Pflegeaufwand können noch nicht getroffen werden. Zunächst sind weitere, in Abschnitt 6 skizzierte Entwicklungen sowie eine umfassende Evaluierung unter Stakeholdern nötig.

Literaturverzeichnis

- [Au17] Auth, G.: Campus-Management-Systeme - Prozessorientierte Anwendungssoftware für die Organisation von Studium und Lehre. Die Hochschule, Heft 1/2017, S. 40-58.
- [Be10] Ländergemeinsame Strukturvorgaben für die Akkreditierung von Bachelor und Masterstudiengängen - Beschluss der Kultusministerkonferenz vom 10.10.2003 i. d. F. vom 04.02.2010. 2010.
- [Bu17] Bundeshaushaltsordnung (BHO), www.gesetze-im-internet.de/bho/, Stand 24.06.2018.
- [CM10] CMMI Product Team: CMMI for Development V. 1.3. SEI, CMU, 2010.
- [Eu09] Europäische Kommission: ECTS-Leitfaden. 2009.
- [JM17] Jetschni, J.; Meister, V.: Schema Engineering for Enterprise Knowledge Graphs - A Reflecting Survey and Case Study. In: Proceedings of the IEEE Eighth International Conference on Intelligent Computing and Information Systems, S. 271-277, 2017.
- [KS14] Kaiser, M.; Smolnik S.: Information Lifecycle Management für Dokumente. In: Walter S., Kaiser G. (Hrsg.), Dokumentenlogistik. Springer, S. 89-107, 2014.
- [MJK17] Meister, V.; Jetschni, J.; Kreideweiß, S.: Konzept und Prototyp einer dezentralen Wissensinfrastruktur zu Hochschuldaten für Mensch und Maschine. In: Eibl, M. & Gaedke, M. (Hrsg.), INFORMATIK 2017. GI, Bonn, S. 1717-1732, 2017.
- [Pa16] Paulheim, H.: Knowledge Graph Refinement: A Survey of Approaches and Evaluation Methods. Semantic Web, 2016, S. 1-23.
- [RM17] Rizun, M.; Meister V.: Analysis of Benefits for Knowledge Workers Expected from Knowledge Graph-Based Information Systems. In: Wrycza S., Maślankowski J. (eds) Information Systems: Research, Development, Applications, Education. SIG-SAND/PLAIS 2017. LNBIP, vol. 300. Springer, S. 25-39, 2017.
- [RS14] Rupp, C.; SOPHISTen: Requirements-Engineering und -Management - Aus der Praxis von klassisch bis agil. Hanser, 6. Auflage, 2014.
- [St15] Standards and Guidelines for Quality Assurance in the European Higher Education Area (ESG). Brussels, 2015.