
Ontologie-basierte Informationsintegration in der Form eines Social Network of Business Objects (SoNBO)



von

Berit Gebel-Sauer

Genehmigte Dissertation zur Verleihung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.),

Fachbereich 4: Informatik

Universität Koblenz-Landau

Vorsitzender des Promotionsausschusses:	Prof. Dr. Ralf Lämmel
Vorsitzender der Promotionskommission:	PD Dr. Matthias Thimm
Berichterstatterin und Betreuerin:	Prof. Dr. Petra Schubert
Weiterer Berichterstatter:	Prof. Dr. Patrick Delfmann
Datum der wissenschaftlichen Aussprache:	12.02.2021

Danksagung

An erster Stelle möchte ich mich bei meiner Doktormutter Prof. Dr. Petra Schubert bedanken, die mich während meiner Zeit als Doktorandin im CEIR¹-Team bei der Dissertation unterstützte. Dazu gehörten intensive Gespräche über die Inhalte und Struktur meiner Dissertation, eine wertvolle Begleitung bei den Praxispartnern vor Ort und nicht zuletzt ein einmaliger Zugang zur Wissenschaft durch das CEIR-Projekt IndustryConnect. IndustryConnect verbindet die Praxis und die Wissenschaft auf eine außergewöhnliche Weise und nicht zuletzt durch diesen Verbund war die enge wissenschaftliche Zusammenarbeit mit Unternehmen im Rahmen dieser Dissertation möglich. Daher möchte ich als nächstes Bernd Gewehr und seinem IT-Team des Ingenieurbüros *Vössing Ingenieurgesellschaft mbH* danken, deren entwickelte Individualsoftware der Auslöser dieses Forschungsprojektes war und die für einen Austausch immer offen waren. Weiterhin möchte ich Dirk Gerlach von der *Franckh-Kosmos Verlags-GmbH & Co. KG* danken, der durch seine Bereitschaft die Evaluation in der Praxis ermöglichte. Ebenso geht mein Dank an Prof. Dr. Patrick Delfmann, der mich als Zweitgutachter mit wertvollem Input aus einer weiteren wissenschaftlichen Perspektive unterstützte.

Ein weiterer Dank geht an Simon Meier, der mich bei der Evaluation in der Praxis technisch im Rahmen seiner HiWi-Tätigkeit und mit seinem Engagement darüber hinaus unterstützte. Dem IPZ danke ich für die Förderlinie „Doc mit Kind“, durch welches die finanziellen Mittel für die studentische Hilfskraft Katharina Gorjatshev bereitgestellt wurde, die mich ebenfalls unterstützte.

Ein großer Dank geht auch an meine Familie, die mich bei der Erstellung der Dissertation auf verschiedene Arten unterstützte. In die Zeit dieses Dissertationsprojektes fiel die Geburt unserer Tochter Marlene und mein Mann Max, meine Eltern und meine Schwiegereltern räumten mir immer wieder Zeit ein, um an der Erstellung der Dissertation zu arbeiten. Ein weiterer Dank geht an meine Schwestern und meine Freunde, die mir in dieser Zeit zur Seite standen.

Abschließend möchte ich mich beim ganzen CEIR-Team und insbesondere bei den Professorinnen Prof. Dr. Petra Schubert und Prof. Dr. Susan Williams und meinen Kollegen danken, mit denen ein reger inhaltlicher Austausch stattfand und mit denen ich eine unvergessliche Zeit an der Universität und auch auf wissenschaftlichen und praxisorientierten Konferenzen hatte.

¹ = Center for Enterprise Information Research, initiiert durch Prof. Dr. Petra Schubert und Prof. Dr. Susan Williams

Abstract

Die Umsetzung einer flexiblen Integration von Informationen aus verteilten und komplexen Informationssystemen stellt Unternehmen aktuell vor große Herausforderungen. Das im Rahmen dieser Dissertation entwickelte Ontologie-basierte Informationsintegrationskonzept SoNBO (Social Network of Business Objects) adressiert diese Herausforderungen. Bei einem Ontologie-basierten Konzept werden die Daten in den zu integrierenden Quellsystemen (z. B. betriebliche Anwendungssysteme) mithilfe eines Schemas (= Ontologie) beschrieben. Die Ontologie in Verbindung mit den Daten aus den Quellsystemen ergibt dann einen (virtualisierten oder materialisierten) Knowledge Graph, welcher für den Informationszugriff verwendet wird. Durch den Einsatz eines Schemas ist dieses flexibel auf die sich ändernden Bedürfnisse des Unternehmens bezüglich einer Informationsintegration anpassbar. SoNBO unterscheidet sich von existierenden Konzepten aus dem Semantic Web (OBDA = Ontology-based Data Access, EKG = Enterprise Knowledge Graph) sowohl im Aufbau der unternehmensspezifischen Ontologie (= Social Network of Concepts) als auch im Aufbau des nutzerspezifischen Knowledge Graphen (= Social Network of Business Objects) unter der Verwendung von sozialen Prinzipien (bekannt aus Enterprise Social Software). Aufbauend auf diesem SoNBO-Konzept wird das im Rahmen dieser Dissertation entwickelte SoNBO-Framework (nach Design Science Research) zur Einführung von SoNBO in einem beliebigen Unternehmen und die aus der Evaluation (im Unternehmen KOSMOS Verlag) gewonnenen Erkenntnisse vorgestellt. Die Ergebnisse (SoNBO-Konzept und SoNBO-Framework) basieren auf der Synthese der Erkenntnisse zu Ontologie-basierter Informationsintegration aus dem Status quo in Praxis und Wissenschaft: Für den Status quo in der Praxis wird mithilfe einer Tiefenfallstudie (Ingenieurbüro Vössing) die grundlegende Idee zu SoNBO in Form einer vom Fallstudienunternehmen entwickelten und dort seit Jahren eingesetzten Individualsoftware analysiert. Für den Status quo in der Wissenschaft wird das Ergebnis einer im Rahmen der Dissertation durchgeführten strukturierten Literaturanalyse zu Ontologie-basierten Informationsintegrationsansätzen präsentiert. Diese Dissertation liefert damit einen Beitrag sowohl für die Wissenschaft (Erkenntnisgewinn im Bereich der Ontologie-basierten Informationsintegrationsansätze für die Wirtschaftsinformatik u. a. durch die Entwicklung eines evaluierten Artefaktes) als auch für die Praxis (Schaffung eines evaluierten Artefaktes).

Abstract (Englisch)

The flexible integration of information from distributed and complex information systems poses a major challenge for organisations. The ontology-based information integration concept SoNBO (Social Network of Business Objects) developed and presented in this dissertation addresses these challenges. In an ontology-based concept, the data structure in the source systems (e.g. operational application systems) is described with the help of a schema (= ontology). The ontology and the data from the source systems can be used to create a (virtualised or materialised) knowledge graph, which is used for information access. The schema can be flexibly adapted to the changing needs of a company regarding their information integration. SoNBO differs from existing concepts known from the Semantic Web (OBDA = Ontology-based Data Access, EKG = Enterprise Knowledge Graph) both in the structure of the company-specific ontology (= Social Network of Concepts) as well as in the structure of the user-specific knowledge graph (= Social Network of Business Objects) and makes use of social principles (known from Enterprise Social Software). Following a Design Science Research approach, the SoNBO framework was developed and the findings documented in this dissertation. The framework provides guidance for the introduction of SoNBO in a company and the knowledge gained from the evaluation (in the company KOSMOS Verlag) is used to demonstrate its viability. The results (SoNBO concept and SoNBO framework) are based on the synthesis of the findings from a structured literature review and the investigation of the status quo of ontology-based information integration in practice: For the status quo in practice, the basic idea of SoNBO is demonstrated in an in-depth case study about the engineering office Vössing, which has been using a self-developed SoNBO application for a few years. The status quo in the academic literature is presented in the form of a structured literature analysis on ontology-based information integration approaches. This dissertation adds to theory in the field of ontology-based information integration approaches (e.g. by an evaluated artefact) and provides an evaluated artefact (the SoNBO Framework) for practice.

Inhaltsverzeichnis

Danksagung	i
Abstract	iii
Abstract (Englisch).....	v
Inhaltsverzeichnis.....	vii
Abkürzungsverzeichnis.....	xi
Abbildungsverzeichnis.....	xiii
Tabellenverzeichnis.....	xvii
1 Einleitung	1
1.1 Motivation	2
1.1.1 Betriebliche Anwendungssysteme für das Informationsmanagement	3
1.1.2 Integrationsansätze für betriebliche Anwendungssystemen	4
1.2 Forschungsziele.....	9
1.3 Aufbau der Dissertation.....	11
2 Forschungsdesign.....	12
2.1 Methodik: Design Science Research	13
2.2 Forschungsmethoden: Einsatz verschiedener komplementärer Methoden.....	14
2.2.1 Qualitatives Interview und Fallstudie	14
2.2.2 IndustryConnect.....	17
2.2.3 Literaturanalyse	18
2.3 Forschungsschritte.....	21
2.4 Forschungsgrenzen	22
3 Status quo zu Ontologie-basierten Integrationsansätzen in der Praxis und Wissenschaft	23
3.1 Status quo in der Praxis: Tiefenfallstudie zu einem Ontologie-basierten Integrationsansatz	23
3.1.1 Hintergründe der Tiefenfallstudie Vössing	23
3.1.2 Beschreibung der SoNBO-Anwendung	25
3.1.3 Zusammenfassung	33
3.2 Status quo in der Wissenschaft: Literaturanalyse zu Ontologie-basierten Integrationsansätzen	33

3.2.1	Grundlagen.....	33
3.2.2	Ontologie-basierte Integrationsansätze	53
3.3	Zusammenfassung des Status quo in der Praxis und Wissenschaft	101
4	SoNBO-Konzept.....	102
4.1	Definitionen und Eigenschaften.....	102
4.1.1	Social Business Object.....	102
4.1.2	Social Concept und Social Subconcept.....	106
4.1.3	Social Network of Concepts und Social Network of Subconcepts	107
4.1.4	Social Network of Business Objects	109
4.2	Prinzipien von Social Software im SoNBO-Konzept	113
4.3	Benutzerfrontend.....	114
4.4	Administrationsfrontend	117
4.5	Anwendungssicht.....	117
4.6	Vergleich SoNBO zu Ontologie-basierten Ansätzen aus dem Semantic Web.....	119
4.7	Zusammenfassung	121
5	SoNBO-Framework.....	122
5.1	Definition Modell, Methode und Tool	123
5.2	SoNBO-Explorer	124
5.2.1	Hintergrund	124
5.2.2	Aufbau des SoNBO-Explorers	124
5.2.3	Konfiguration des SoNBO-Explorers (= Konfigurations-Projekt).....	127
5.3	Framework: Phasen und Akteure	127
5.4	Phase 1: Status quo.....	132
5.4.1	Beteiligte Akteure	132
5.4.2	Methode: Teilstandardisiertes Leitfaden-Interview	133
5.5	Phase 2: Ontologieentwicklung	134
5.5.1	Beteiligte Akteure	134
5.5.2	Modell: Unternehmensontologie	135
5.5.3	Methode: Analyse Business-Sicht und technische Sicht.....	140
5.5.4	Tool: SoNBO-Visualiser.....	148
5.6	Phase 3: Mapping.....	149
5.6.1	Beteiligte Akteure	150
5.6.2	Methode: Mapping und Tool: SoNBO-Mapper.....	150
5.7	Grenzen des Frameworks	155
5.8	Zusammenfassung	155

6	Evaluation des SoNBO-Frameworks	156
6.1	Ziel und Hintergründe der Evaluation	156
6.2	Erkenntnisse für das SoNBO-Framework.....	158
6.3	Phase 1: Status quo	160
6.3.1	Ergebnisse	160
6.3.2	Erkenntnisse.....	162
6.4	Phase 2: Ontologieentwicklung	163
6.4.1	Ergebnisse	163
6.4.2	Erkenntnisse.....	167
6.5	Phase 3: Mapping	169
6.5.1	Ergebnisse	169
6.5.2	Erkenntnisse.....	178
6.6	Zusammenfassung	183
7	Ausblick	184
7.1	Beitrag für die Wissenschaft: Erweiterung des Wissens im Bereich Ontologie- basierte Informationsintegration	184
7.2	Beitrag für die Praxis: Anwendung des SoNBO-Frameworks in der Praxis.....	186
7.3	Grenzen der Dissertation.....	187
7.4	Weiterer Forschungsbedarf.....	188
8	Zusammenfassung	191
	Literaturverzeichnis.....	195
	Anhang	209
A.1	Organisatorische Daten zur Erhebung des Key Case Vössing	209
A.2	Dokumentation des Forschungsprozesses.....	210
A.3	Grundlage für die Entwicklung der Taxonomie im SoNBO-Framework	213
A.4	Grundlage für die Entwicklung der Ontologie (als Referenzmodell) im SoNBO-Framework	216

Abkürzungsverzeichnis

API	<i>Application Programming Interface</i>
ARIS	<i>Architektur integrierter Informationssysteme</i>
BGE	<i>Bernd Gewehr</i>
BO	<i>Business Ontology</i>
BPM	<i>Business Process Management</i>
CEIR	<i>Center for Enterprise Information Research</i>
CMS	<i>Content Management Systems</i>
CORE	<i>Computing Research and Education</i>
CRM	<i>Customer Relationship Management</i>
CRUD	<i>create, read, update, delete</i>
DBMS	<i>Datenbankmanagementsystem</i>
DGE	<i>Dirk Gerlach</i>
DOLCE	<i>Descriptive Ontology for Linguistic and Cognitive Engineering</i>
DSR	<i>Design Science Research</i>
EAI	<i>Enterprise Application Integration</i>
ebd.	<i>ebenda</i>
ECMS	<i>Enterprise Content Management Systems</i>
ECS	<i>Enterprise Collaboration Systems</i>
EKG	<i>Enterprise Knowledge Graph</i>
ERP	<i>Enterprise Resource Planning</i>
ESN	<i>Enterprise Social Network</i>
ESS	<i>Enterprise Social Software</i>
ETL	<i>Extract Transform Load</i>
FG BAS	<i>Forschungsgruppe Betriebliche Anwendungssysteme</i>
GRDDL	<i>Gleaning Resource Descriptions from Dialects of Languages</i>
HCL	<i>Hindustan Computers Limited</i>
IBM	<i>International Business Machines</i>
IBV	<i>Ingenieurbüro Vössing</i>
IoT	<i>Internet of Things</i>
IRESS	<i>Identification of Requirements for Enterprise Social Software</i>
IS	<i>Information Systems</i>
JSON	<i>JavaScript Object Notation</i>
KG	<i>Knowledge Graph</i>
KOSMOS	<i>Franckh-Kosmos Verlags-GmbH & Co. KG</i>
LD	<i>Linked Data</i>
LED	<i>Linked Enterprise Data</i>

LPG *Labeled Property Graph*
NoSQL *Not only SQL*
OBDA *Ontology-Based Data Access*
ORG *Organization Ontology*
OWL *Web Ontology Language*
R2RML *RDB to RDF Mapping Language*
RDB *Relational Database*
RDF *Resource Description Framework*
SAP *Systeme, Anwendungen, Produkte*
SBO *Social Business Object*
SC *Social Concept*
SDL *Semantic Data Lake*
SOA *Service-Oriented Architecture*
SoNBO *Social Network of Business Objects*
SPARQL *SPARQL Protocol and RDF Query Language*
SQL *Structured Query Language*
SSC *Social Subconcept*
SSM *Soft Systems Methodology*
SSO *Single Sign-on*
SUMO *Suggested Upper Merged Ontology*
UPON *Unified Process for ONtology building*
Vössing *Vössing Ingenieurgesellschaft mbH*
W3C *World Wide Web Consortium*
WI *Wirtschaftsinformatik*
XML *Extensible Markup Language*

Abbildungsverzeichnis

Abb. 1.1: Einordnung und Aufbau der Dissertation (eigene Darstellung)	1
Abb. 1.2: Technische Sicht der Integration (Schubert, 2003, p. 9)	5
Abb. 1.3: Beispielhafte Anwendungssicht für eine Ontologie-basierte Integration	6
Abb. 2.1: Argumentation im Design Science Research Zyklus (Kuechler & Vaishnavi, 2008, p. 493)	13
Abb. 2.2: Forschungsvorgehen (Orientierung an Design Science Research)	20
Abb. 3.1: Integration der SoNBO-Anwendung (“Self-Service”) in IBM Connections (Stand: 2016)	25
Abb. 3.2: Benutzerfrontend der SoNBO-Anwendung am Beispiel des Knotens <i>Bernd Gewehr</i> (Stand: 2016)	26
Abb. 3.3: Benutzerfrontend der SoNBO-Anwendung am Beispiel des Knotens <i>IBV</i> (Stand: 2016)	27
Abb. 3.4: Beispiel für Filter: Über- und Unterstunden der Abteilungsleiter (Stand: 2016)	28
Abb. 3.5: Beispiel für Filter: Verteilung der Arbeitsstunden der Abteilungsleiter (Stand: 2016)	29
Abb. 3.6: Mobilfähige SoNBO-Anwendung (Stand: 2016)	29
Abb. 3.7: Anwendungssicht der SoNBO-Anwendung	31
Abb. 3.8: Ausschnitt aus der Entwicklungsumgebung der SoNBO-Anwendung (Stand: 2016)	32
Abb. 3.9: Beispiel einer Taxonomie (Blumauer, 2014a)	37
Abb. 3.10: IDEON Ontologie I (Madni et al., 2001, pp. 41, 42)	42
Abb. 3.11: IDEON Ontologie II (Madni et al., 2001, pp. 43, 44)	43
Abb. 3.12: Organization Ontologie (W3C, 2014b)	44
Abb. 3.13: Ebenen der Business Ontologie (von Rosing & Laurier, 2015, p. 37)	45
Abb. 3.14: RDF Triple Stores vs. Labeled Property Graphs (Barrasa, 2017)	48
Abb. 3.15: ARIS-Haus (Scheer, 2002, p. 41)	50
Abb. 3.16: IRESS-Framework (Glitsch & Schubert, 2017, p. 869)	51
Abb. 3.17: Einordnung von Enterprise Social Software (Schwade & Schubert, 2017, p. 402)	52
Abb. 3.18: Vorgehen bei der Literaturanalyse - in Anlehnung an vom Brocke et al. (2009)	54
Abb. 3.19: Concept Map	55
Abb. 3.20: Überarbeitete Concept Map	56
Abb. 3.21: Zeitliche Entwicklung der Beiträge zur Ontologie-basierten Informationsintegration	58
Abb. 3.22: Verteilung der Beiträge nach Fachgebiet	59

Abb. 3.23: Verteilung der Beiträge nach Themen	59
Abb. 3.24: Verteilung der Beiträge nach CORE Ranking.....	60
Abb. 3.25: Detaillierte zeitliche Entwicklung der Beiträge zu OBDA.....	61
Abb. 3.26: Detaillierte zeitliche Entwicklung der Beiträge zu L(E)D	61
Abb. 3.27: Detaillierte zeitliche Entwicklung der Beiträge zu (E)KG	62
Abb. 3.28: Ontology-Based Data Access (Calvanese et al., 2011, p. 43, 2015, p. 2)	69
Abb. 3.29: OBDA Paradigma (Sequeda & Miranker, 2017, p. 93)	70
Abb. 3.30: Query processing in OBDA (Kharlamov, Hovland, et al., 2015, p. 97)	70
Abb. 3.31: Pay-as-you-go Methodology für OBDA (Sequeda & Miranker, 2017, p. 94)	72
Abb. 3.32: Ontop (Calvanese et al., 2015, p. 2, 2017, p. 3)	74
Abb. 3.33: Optique (Kharlamov, Jimenez-Ruiz, et al., 2015, p. 2).....	75
Abb. 3.34: Vernetzung von Daten und Informationen im Intranet (Auer et al., 2014, p. 91).....	78
Abb. 3.35: Die Phasen des Linked Data-Lebenszyklus (Auer et al., 2014, p. 93; Frischmuth et al., 2012, p. 4).....	80
Abb. 3.36: Datenbankintegration (Auer et al., 2014, p. 96; Frischmuth et al., 2012, p. 14)	81
Abb. 3.37: Customer Information Portal Architecture (Hu & Svensson, 2010, p. 135)	83
Abb. 3.38: Knowledge Graph im Vergleich (Villazon-Terrazas, Garcia-Santa, Ren, Faraotti, et al., 2017, p. 55).....	87
Abb. 3.39 ARA für Knowledge Graph Management in Organisationen (Denaux et al., 2017, p. 58).....	88
Abb. 3.40: Position eines EKG in der Enterprise Information System Architektur (Galkin et al., 2017, p. 90).....	89
Abb. 3.41: Knowledge construction and maintenance lifecycle (Villazon-Terrazas, Garcia-Santa, Ren, Srinivas, et al., 2017, p. 80)	91
Abb. 3.42: High-Level Architecture eines Linked Data Warehouses (Blumauer, 2014b, p. 11)	93
Abb. 3.43: Metaphactory Systemarchitektur (Haase, Herzig, Kozlov, Nikolov, & Trame, 2019, p. 3).....	94
Abb. 3.44: Strukturierte semantische Suche in den Artefakten des British Museums (Haase et al., 2019, p. 12).....	95
Abb. 3.45: RDF-Datensätze vs. Linked Data vs. Knowledge Graph (Villazon-Terrazas, Garcia- Santa, Ren, Faraotti, et al., 2017, p. 52)	96

Abb. 3.46: Schlüsselkonzepte der Ontologie-basierten Integrationsansätze aus dem Semantic Web	97
Abb. 3.47: Kategorien für den Vergleich von OBDA und EKG	99
Abb. 4.1: Social Network of Business Objects	109
Abb. 4.2: Datenzugriff bei SoNBO durch Materialisierung	110
Abb. 4.3: Datenzugriff bei SoNBO durch Virtualisierung	111
Abb. 4.4: Zusammenhang zwischen den drei Graphen bei SoNBO	112
Abb. 4.5: Benutzerfrontend eines SoNBO-Explorers als Mockup	115
Abb. 4.6: Anwendungssicht (einfache Darstellung)	117
Abb. 4.7: Anwendungssicht (detaillierte Darstellung)	118
Abb. 5.1: Klassifikation von Referenzmodellen (Schütte (1998, p. 183) in Rosemann & Schütte (1998, p. 24))	123
Abb. 5.2: Logik des SoNBO-Explorers (Virtualisierung) – verändert nach Götz & Gebel-Sauer (2018, p. 2002)	126
Abb. 5.3: SoNBO-Prozessmodell - verändert nach Gebel-Sauer & Schubert (2019b, p. 4)	128
Abb. 5.4: SoNBO-Framework	130
Abb. 5.5: Referenzmodell: Social Network of Concepts	139
Abb. 5.6: Referenzmodell: Social Network of Subconcepts	140
Abb. 5.7: World-Café – Fragen zur Ermittlung der Social Concepts.....	144
Abb. 5.8: Benutzerfrontend SoNBO-Visualiser (für Ebene: Social Network of Concepts)	149
Abb. 5.9: Konfigurationstabelle für das semantische Mapping (I).....	153
Abb. 5.10: Konfigurationstabelle für das semantische Mapping (II).....	154
Abb. 6.1: Ergebnis Business Analyse	160
Abb. 6.2: Ergebnis Analyse IT-Landschaft	162
Abb. 6.3: KOSMOS-Start- und Zieltaxonomie (als Social Concepts und Social Subconcepts).....	164
Abb. 6.4: KOSMOS-Startontologie (Social Network of Concepts).....	165
Abb. 6.5: KOSMOS-Startontologie (Social Network of Subconcepts)	165
Abb. 6.6: KOSMOS-Zielontologie (Social Network of Concepts).....	166
Abb. 6.7: KOSMOS-Zielontologie (Social Network of Subconcepts)	166
Abb. 6.8: Benutzerfrontend SoNBO-Explorer für den KOSMOS Verlag als Mockup.....	167

Abb. 6.9: Ausschnitt aus dem <i>SoNBO-Mapper für KOSMOS</i> : Konfiguration zu SSC Mitarbeiter (SC: Person)	171
Abb. 6.10: Ausschnitt aus dem <i>SoNBO-Mapper für KOSMOS</i> : Konfiguration der Kanten zwischen den SSC	172
Abb. 6.11: Ausschnitt aus dem <i>SoNBO-Mapper für KOSMOS</i> : Übertragung der Konfiguration auf die Struktur im SoNBO-Explorer.....	173
Abb. 6.12: Screenshot des Administrationsfrontends vom SoNBO-Explorer für KOSMOS.....	174
Abb. 6.13: Benutzerfrontend SoNBO-Explorer für den KOSMOS Verlag als Screenshot (Knoten: <i>Mitarbeiter Dirk Gerlach</i>)	175
Abb. 6.14: Benutzerfrontend SoNBO-Explorer für den KOSMOS Verlag als Screenshot (Knoten: <i>Projekt Glibber-Organe</i>)	176
Abb. 6.15: Erfolgreich konfigurierte und getestete Ontologie.....	182

Tabellenverzeichnis

Tab. 1.1: Forschungsziele	10
Tab. 3.1 Übersicht über existierende Unternehmensontologien (Gorjatshev, 2019, p. 28).....	40
Tab. 3.2: Autoren zu Ontology-Based Data Access	63
Tab. 3.3: Autoren zu Linked Data	63
Tab. 3.4: Autoren zu Linked Enterprise Data	64
Tab. 3.5: Autoren zu Knowledge Graph	64
Tab. 3.6: Autoren für Enterprise Knowledge Graph.....	64
Tab. 3.7: Zentrale Beiträge	66
Tab. 3.8: Vergleich der Ontologie-basierten Integrationsansätze	100
Tab. 4.1: Verwendung von Business Objects für betriebliche Anwendungssysteme (Gebel-Sauer & Schubert, 2019a, p. 205)	103
Tab. 4.2: Voraussetzung (Quellsystem, Business Object, Klasse)	104
Tab. 4.3: Beispiel zu Quellsystem, Klasse, Business Object, Attribut und Attributwert	105
Tab. 4.4: Definition (Social Business Object).....	106
Tab. 4.5: Spezifizierung der auszuwählenden Attribute	106
Tab. 4.6: Definition (Social Concept und Social Subconcept).....	107
Tab. 4.7: Vergleich der Ebenen und verwendeten Begriffe	107
Tab. 4.8: Definition (Social Network of Concepts und Social Network of Subconcepts)	108
Tab. 4.9: Definition (Social Network of Business Objects)	110
Tab. 4.10: Übersicht über die verwendete Terminologie	112
Tab. 4.11: Vergleich von <i>Social Software</i> und einer generischen <i>SoNBO-Anwendung</i>	113
Tab. 5.1: Charakterisierung der einzelnen Akteure	131
Tab. 5.2: Vergleich der hier verwendeten Begriffe (eigene Darstellung)	135
Tab. 5.3: Taxonomie bestehend aus Social Concepts und Subconcepts (eigene Darstellung).....	137
Tab. 5.4: Fragen zur Erhebung der Social Concepts.....	145
Tab. 5.5: Tabelle zur Erhebung der Social Concepts bei den Mitarbeitern.....	146
Tab. 5.6: Matrix für die Identifizierung der Beziehungen	147
Tab. 6.1: Projektablauf	157

Tab. 6.2: Rückfragen zu SoNBO im Rahmen der Evaluation.....	159
Tab. 6.3: Ablauf Phase 3 (3. und 4. Meeting)	177
Tab. 6.4: Herausforderungen bei der Startkonfiguration und daraus abgeleitete Erkenntnisse.....	179
Tab. 6.5: Zugriff auf verschiedene Arten von Attributen	180
Tab. 6.6: Verwendung der Attribute für den jeweiligen Zweck zur Spezifizierung der Ontologie.....	181

1 Einleitung

Die fehlende Umsetzung einer flexiblen Integration von Informationen aus verteilten und komplexen Informationssystemen, um eine zusammenhängende und konsistente Sicht auf diese Informationen zu erhalten, stellt Unternehmen aktuell vor große Herausforderungen. Unter *flexibel* wird in dieser Dissertation die Eigenschaft eines Integrationsansatzes verstanden, auf sich ändernde Bedürfnisse eines Unternehmens bezüglich der Integration zu reagieren. Die Herausforderungen sind sowohl in den Wissenschaftsdisziplinen *Informatik* und *Wirtschaftsinformatik* als auch in der *Praxis* vorhanden (Frischmuth et al., 2012; Holten, 2003; Linthicum, 2000; Wache et al., 2001), was in Abb. 1.1² dargestellt ist.

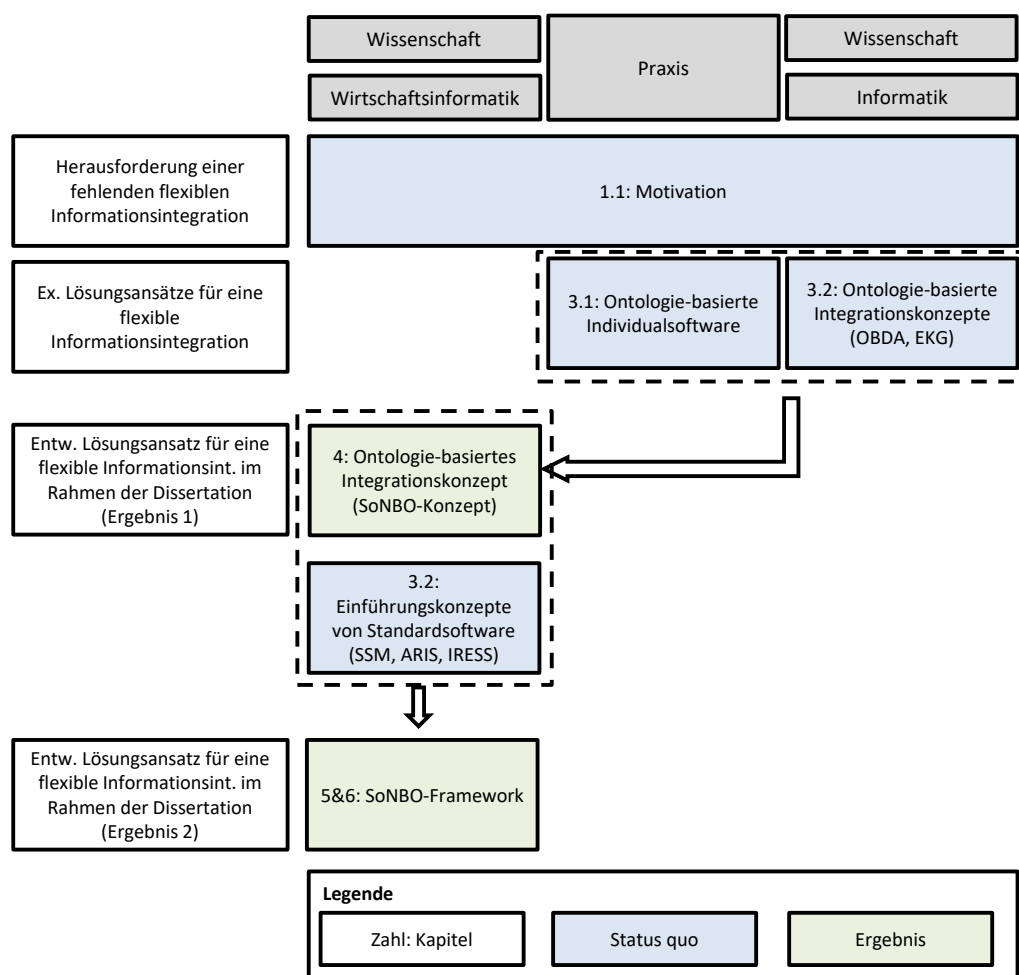


Abb. 1.1: Einordnung und Aufbau der Dissertation (eigene Darstellung)

² In dieser Abbildung sind der Aufbau und die Einordnung der Dissertation in die verschiedenen Fachgebiete dargestellt. Diese Visualisierung entstand als Erkenntnis im Rahmen der Dissertation.

Die Lösungsansätze für die Herausforderungen entwickelten sich in den genannten Disziplinen unterschiedlich (vgl. Abb. 1.1): Aus der Perspektive der Wirtschaftsinformatik wird diese Thematik gegenwärtig kaum in Journals oder auf Konferenzen behandelt, während sich in der Informatik Ontologie-basierte Integrationsansätze etablierten und in der Praxis unternehmensspezifische Individuallösungen existieren, welche die Informationen Ontologie-basiert integrieren. In dieser Dissertation wird darauf aufbauend ein weiterer Ansatz (Ergebnis 1: SoNBO-Konzept; SoNBO = Social Network of Business Objects) vorgestellt: SoNBO ist ein Ontologie-basiertes Informationsintegrationskonzept, um ausgewählte Informationen aus den existierenden betrieblichen Anwendungssystemen zu integrieren und einem Mitarbeiter (dem User) mit Hilfe einer speziellen Software (SoNBO-Explorer) zur Verfügung zu stellen. Auf diesem Konzept aufbauend wird ein Rahmenwerk für ein Einführungsprojekt (Ergebnis 2: SoNBO-Framework) entwickelt. Für das Erreichen der beiden Ergebnisse wird folgendermaßen vorgegangen: In einer von einem Unternehmen (Vössing) entwickelten Individualsoftware wurden Merkmale einer Ontologie-basierten Informationsintegration im Rahmen dieser Dissertation identifiziert, die unter der Zuhilfenahme der existierenden Konzepte aus der Informatik in dieser Dissertation zum *SoNBO-Konzept* (Ergebnis 1) verknüpft und dadurch wissenschaftlich beschrieben werden. Die identifizierten Merkmale binden bei der Umsetzung einer Ontologie-basierten Informationsintegration in einem Unternehmen neben den technischen Gegebenheiten verstärkt die Unternehmensanforderungen mit ein. Das SoNBO-Konzept in Verbindung mit existierenden Einführungskonzepten aus der Wirtschaftsinformatik wird zum *SoNBO-Framework* (Ergebnis 2) weiterentwickelt, welches ein beliebiges Unternehmen bei der Einführung einer SoNBO-Anwendung (als Standardsoftware) unterstützt. Das SoNBO-Framework und die SoNBO-Anwendung wurden in einem Fallstudienunternehmen (KOSMOS) bezüglich Anwendbarkeit und Nützlichkeit evaluiert. Das SoNBO-Konzept und das zugehörige Framework leisten einen Beitrag für die Wirtschaftsinformatik, weil bei SoNBO die Einbindung der Unternehmenssicht und damit verbunden die Perspektive der Mitarbeiter von hoher Relevanz ist, um die zu integrierenden Informationen zu identifizieren. Die Dissertation überträgt damit aktuelle Erkenntnisse aus dem Fachgebiet der Informatik in die Wirtschaftsinformatik und bindet gleichzeitig Erfahrungen aus der betrieblichen Praxis mit ein.

In der Einleitung wird zunächst detailliert die Motivation für die hier beschriebene Entwicklung eines Ontologie-basierten Integrationskonzeptes angeführt. Dazu werden betriebliche Anwendungssysteme für das Informationsmanagement und existierende Integrationsansätze vorgestellt. Im Anschluss daran werden die Forschungsziele formuliert und der Aufbau der Dissertation beschrieben.

1.1 Motivation

Informationsmanagement hat im Unternehmen das Ziel, den bestmöglichen Einsatz der Ressource Information unter Berücksichtigung der Unternehmensziele zu gewährleisten. Dazu soll die richtige Information zum richtigen Zeitpunkt in bearbeitbarer Konzentration dargestellt werden (Bächle & Kolb, 2015). Das Erreichen dieses Ziels wird durch das Management (1) der *Informationswirtschaft*, (2) der *Informationssysteme* und (3) der *Informations- und Kommunikationstechniken* unterstützt unter Be-

rücksichtigung von *übergreifenden Führungsaufgaben* (Krcmar, 2015). Bei der Betrachtung der verschiedenen Ebenen wird die Verzahnung der betriebswirtschaftlichen und technischen Perspektive deutlich. Die Ebene (1) *Management der Informationswirtschaft* hat zur Aufgabe, den Informationsbedarf im Unternehmen zu identifizieren und die Mechanismen in Gang zu setzen, um ein Informationsangebot zu schaffen. Das Informationsangebot wird mit Hilfe von (2) *Informationssystemen* und der dazugehörigen (3) *Informations- und Kommunikationstechnik* bereitgestellt (Krcmar, 2015).

1.1.1 Betriebliche Anwendungssysteme für das Informationsmanagement

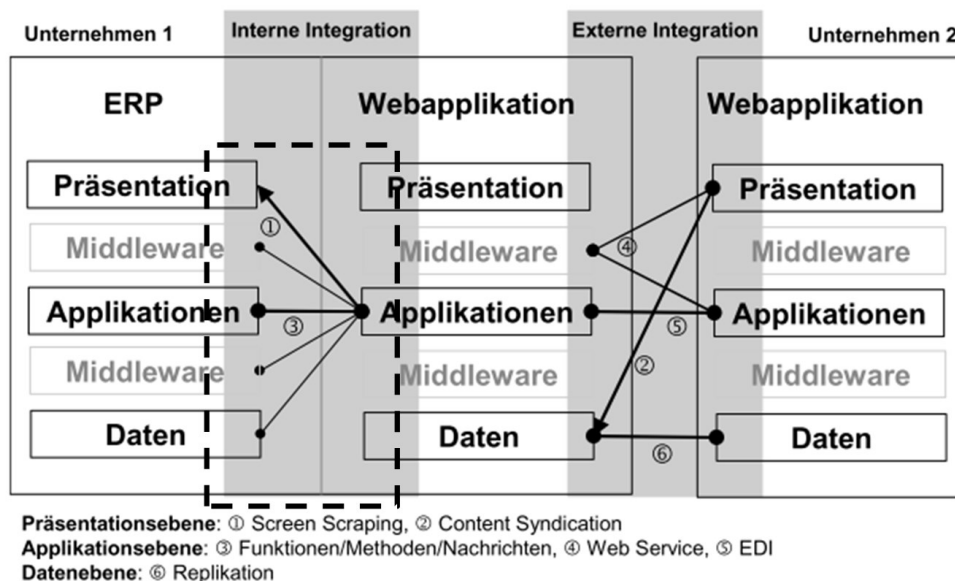
Eine Kernaufgabe des Informationsmanagements besteht darin, ein betriebliches Anwendungssystem (als Individual- oder Standardsoftware) einzuführen, sodass dieses zur Beschaffung, Verarbeitung, Übertragung, Speicherung und/oder Bereitstellung von Informationen genutzt werden kann (Schwarze, 2000). Der Begriff *betriebliches Anwendungssystem* umfasst unterschiedliche Softwaretypen (Gewehr, Gebel-Sauer, & Schubert, 2017; Schubert & Winkelmann, 2020). Systeme für *Enterprise Resource Planning* (ERP, wie z. B. SAP ERP) bestehen aus unterschiedlichen Modulen, die funktionale Bereiche eines Unternehmens unterstützen wie z. B. Finanzmanagement, Personalmanagement, Logistik, Lagerverwaltung oder Verkauf/Marketing (Davenport, 1998). In diesen Systemen liegen die für ein Unternehmen entscheidenden Informationen über alle Geschäftsobjekte (engl. Business Objects, wie z. B. Personen oder Produkte), wodurch sie eine tragende Rolle einnehmen. Beispielsweise kann ohne ein ERP-System keine Rechnung an den Kunden erstellt und somit kein Umsatz erwirtschaftet werden. Sie unterstützen also die Kernprozesse für die Wertschöpfung in einem Unternehmen. Enterprise Collaboration Systems (ECS, wie z. B. HCL Connections) ergänzen ERP-Systeme, indem sie die Mitarbeiter in ihrer gemeinsamen Arbeit unterstützen. Dazu gibt es Module für die Kommunikation, Kooperation, Koordination und das Erstellen und Verwalten von gemeinsamen Inhalten (Williams, 2011). Die Vorläufer der ECS war die Groupware (wie z. B. E-Mail oder Gruppenkalender), die durch die Ergänzung mit Social-Software-Komponenten (wie z. B. soziale Profile, Wiki, Blog) (Schubert & Glitsch, 2015; Schwade & Schubert, 2017) zu den gegenwärtigen ECS weiterentwickelt wurden. Enterprise Content Management Systeme (ECMS) (wie z. B. Alfresco) bilden das Bindeglied zwischen ERP und ECS, indem sie Software bereitstellen, um Content wie z. B. Grafiken oder Dokumente zu verwalten oder zu archivieren. Zu dieser Gattung gehören auch Content Management Systeme (CMS). Business Process Management (BPM) Systeme (wie z. B. Camunda) sind auf die Unterstützung und Koordination von Geschäftsprozessen ausgerichtet und werden durch explizite Prozessdarstellung/-definition gesteuert (Weske, 2012). Darüber hinaus gibt es in den meisten Unternehmen noch spezifische Produktivsysteme (z. B. für die Fertigungssteuerung). Eine Einteilung von Softwaretypen in Unternehmen ist nicht trennscharf, da einige Softwareprodukte Funktionalitäten aus verschiedenen Softwaregattungen anbieten (Schubert & Winkelmann, 2020).

1.1.2 Integrationsansätze für betriebliche Anwendungssystemen

Seit dem ersten Einsatz von betrieblichen Anwendungssystemen für das Informationsmanagement ist die IT-Landschaft in Unternehmen durch eine heterogene Vielfalt an komplexen Systemen (wie z. B. SAP ERP) gewachsen, in denen sich die unternehmensrelevanten Informationen befinden. Allerdings wird eine umfassende und konsistente Sicht für die Mitarbeiter auf die Informationen durch die komplexen und verteilten Systeme zunehmend erschwert (Chaves & de Carvalho, 2016). Daher besteht des Weiteren eine Kernaufgabe des Informationsmanagements darin, die benötigten geschäftskritischen Informationen in den vorhandenen betriebswirtschaftlichen Anwendungssystemen zu identifizieren und durch Integrationslösungen den Mitarbeitern zur Verfügung zu stellen. Um diese Herausforderung zu adressieren, existieren verschiedene Integrationsansätze, die im Folgenden vorgestellt werden.

1.1.2.1 Klassische Integrationsansätze

Seitdem Unternehmen mit mehr als zwei betrieblichen Anwendungssystemen innerhalb eines Netzwerkes arbeiten, gibt es technische Ansätze zur Integration dieser Systeme (Linthicum, 2003) wie z. B. Enterprise Application Integration (EAI) oder Service-Oriented Architecture (SOA). EAI thematisiert das Teilen von Informationen, Prozessen und Anwendungsservices zwischen Applikationen (He & Xu, 2014; Holten, 2003; Jung, 2006; Schubert, 2003) und stellt Werkzeuge, Methoden und Konzepte für die Informationsintegration von betriebswirtschaftlichen Anwendungssystemen bereit (Linthicum, 2000). Es gibt verschiedene Arten von EAI in Abhängigkeit davon, welcher Prozess und/oder Daten integriert werden sollen. Typische Ebenen der Integration sind hier die Datenebene, Anwendungsschnittstelle, Methoden oder Benutzerschnittstelle (Linthicum, 2000). Im Zuge dessen entstand Middleware als eine mögliche Lösung für EAI. Diese technische Lösung versteckt die Komplexität des eigentlichen Systems und stellt APIs (Application Programming Interface) bereit, um auf technischer Ebene client-to-server, client-to-client oder server-to-server durch ein Netzwerk zu verknüpfen (Linthicum, 2000). In Abb. 1.2 ist in dem gestrichelten Kasten zu sehen, welche Möglichkeiten für diese technische Integration existieren. In der Mitte ist die Applikation dargestellt, die die Daten und/oder Informationen aus den verschiedenen Ebenen (Präsentationsebene, Applikationsebene, Datenebene) sowie der Middleware integriert. Unternehmensportale sind in diesem Zusammenhang das moderne Frontend, die die durch EAI integrierten Informationen in einem gemeinsamen Interface darstellen (Schelp & Winter, 2002). Ein weiterer, ganzheitlicher Ansatz für eine verteilte Systemarchitektur und Kommunikation ist die Serviceorientierte Architektur (SOA), die in der Regel mithilfe von Webservices umgesetzt wird (Liebhart, 2007; Linthicum, 2009; Newcomer & Lomow, 2004).



(gestrichelter Kasten: Verschiedene Integrationsmöglichkeiten bei klassischen Integrationsansätzen)

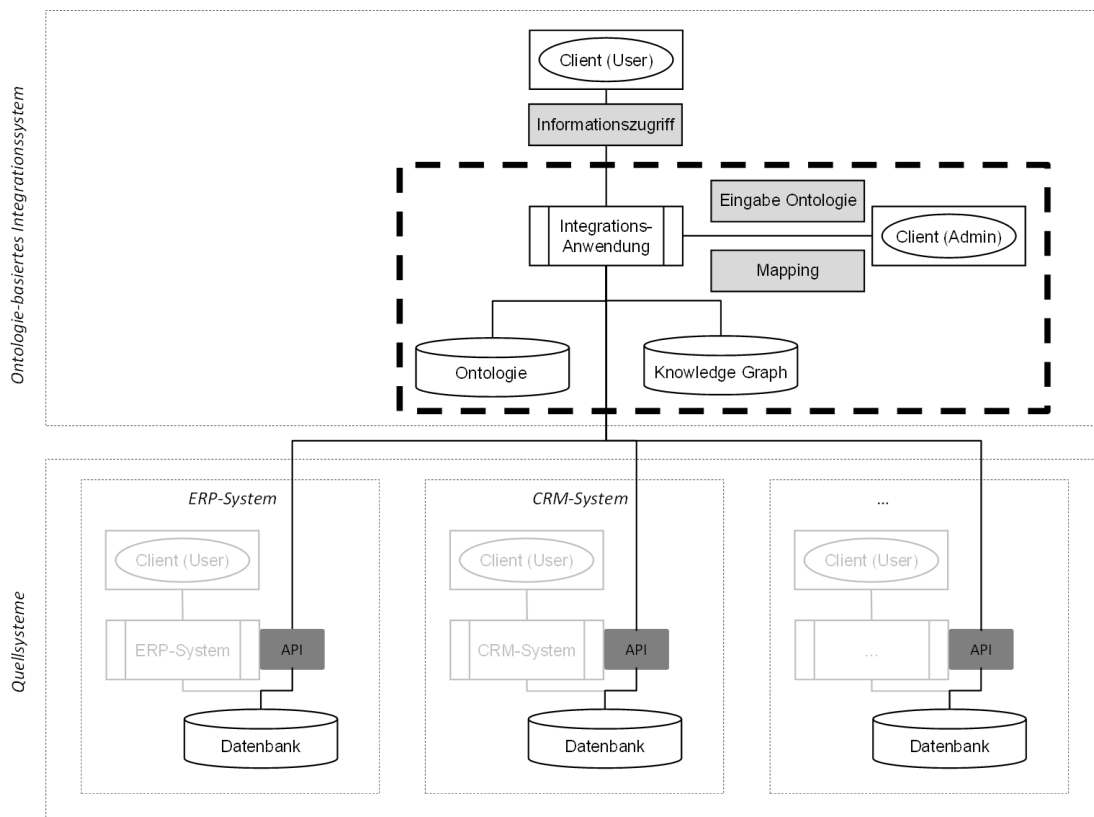
Abb. 1.2: Technische Sicht der Integration (Schubert, 2003, p. 9)

Diese bisherigen Ansätze für eine Verbesserung des Managements von Informationen bestehen darin, einen Informationsbedarf im Unternehmen zu ermitteln, um dann eine Integrationslösung zu entwickeln und einzuführen (Holten, 2003; Krömar, 2015). Daraufhin wird das Informationsangebot bereitgestellt. Eine solche Lösung setzt also ein antizipiertes oder bekanntes Informationsbedürfnis seitens des Managements voraus. Im Laufe der Zeit entsteht im Unternehmen allerdings ein weiterer Informationsbedarf, woraufhin sich der Ablauf (Entwicklung einer Integrationslösung) wiederholt. Zwischen dem Zeitpunkt der Entstehung des Informationsbedarfs und der Fertigstellung der Lösung liegt allerdings eine gewisse Zeit, in der bereits ein neuer Bedarf entstanden sein kann (Gewehr et al., 2017). Der mit den klassischen Lösungen (wie z. B. SOA) verbundene Aufwand ist daher oftmals zu hoch, um eine flexible und effektive Datenintegration in der heutigen dynamischen Unternehmenswelt zu realisieren (Auer, Petzsch, & Unbehauen, 2014).

1.1.2.2 Ontologie-basierte Integrationsansätze

Ein weiterer Integrationsansatz ist die Ontologie-basierte Informationsintegration, welche die im Abschnitt zuvor genannten Herausforderungen adressiert. Dazu wird eine Ontologie verwendet, in welcher die zur Integration benötigten Informationen aus den Quellsystemen beschrieben werden. In Abb. 1.3 ist das Prinzip von Ontologie-basierter Integration als Anwendungssicht dargestellt: Im unteren Bereich befinden sich die *Quellsysteme*, indem die Gattung von zwei betrieblichen Anwendungssystemtypen (ERP- und CRM-System) beispielhaft genannt werden und weitere mit drei Punkten angedeutet werden. Im oberen Bereich ist im gestrichelten Kasten eine *Integrations-Anwendung* dargestellt. Diese besitzt zwei Datenbanken, wobei die Inhalte auch in einer liegen können: In der einen

(links) wird das Schema, also die *Ontologie*, gespeichert, mit dem auf die Daten zugegriffen wird. In der anderen (rechts) kann der *Graph* gespeichert werden, der durch die Ontologie in Verbindung mit den Daten aus den Quellsystemen entsteht. In diesem Fall handelt es sich dann um eine Materialisierung. Falls diese Speicherung nicht erfolgt, wird von einer Virtualisierung gesprochen. Dies wird detailliert in Abschnitt 3.2 erklärt. Für die Eingabe der Ontologie gibt es ein Administrationsfrontend (*Admin-Client*), über das die *Ontologie* und das *Mapping* der Informationen in der Ontologie auf die Daten in den Datenbanken eingegeben werden. Über diese Oberfläche kann auch die Ontologie flexibel verändert oder erweitert werden. Außerdem gibt es einen *Client* für den Endbenutzer (auch: *User*), der mithilfe der Ontologie auf die Daten zugreift.



(gestrichelte Linie: Integrationsschicht durch Beschreibung der Daten mit Hilfe einer Ontologie)

Abb. 1.3: Beispielhafte Anwendungssicht für eine Ontologie-basierte Integration

Solche Integrationsansätze sind in der Praxis noch nicht weit verbreitet (Galkin, Auer, & Scerri, 2016). In einem Unternehmen (Ingenieurbüro Vössing) wurde im Rahmen der IndustryConnect-Initiative³

³ IndustryConnect ist ein Verbund aus CEIR (Center for Enterprise Information Research) und über 20 namhaften Unternehmen (z. B. Bosch, Lufthansa), die sich regelmäßig über die Gestaltung des Digital Workplace austauschen.

(vgl. Abschnitt 2.2.2) von CEIR (Schubert & Williams, 2020; Williams & Schubert, 2017) allerdings eine Software (Eigenentwicklung) zur Integration der Informationen aus den betrieblichen Anwendungssystemen identifiziert, welche die eben genannten Merkmale besitzt und die im Abschnitt zuvor angesprochenen Herausforderungen adressiert: Diese auf das Unternehmen zugeschnittene Individualsoftware ermöglicht die zuvor genannte flexible und effektive Datenintegration in der heutigen dynamischen Unternehmenswelt⁴. Dies wird erreicht, indem eine sinnvoll gewählte Teilmenge der Daten aus den existierenden betrieblichen Anwendungssystemen (im Kontext der Informationsintegration auch: Quellsystem) wie z. B. ERP-System oder CRM-System durch eine Ontologie (vergleichbar mit einem Schema) beschrieben werden⁵. Diese Ontologie kann flexibel auf sich ändernde Anforderungen reagieren. Die bei Vössing entwickelte Ontologie-Anwendung lässt sich aus folgender Motivation heraus charakterisieren: Das Ziel bei der Entwicklung der Software bestand bei Vössing darin, einen Zugriff auf die Informationen der transaktionalen Systeme (z. B. ERP-System) durch das Kollaborationssystem IBM Connections zu ermöglichen. Dadurch entstand die Idee, die Informationen in den transaktionalen Systemen als Teil eines sozialen Netzwerks und nicht mehr nur als Daten in relationalen Datenbanken zu verstehen. Die Übersetzung der Informationen aus den transaktionalen Systemen (Business Objects) in ein soziales Netzwerk (Social Business Objects) erfolgt mithilfe einer Ontologie, die das Schema für das soziale Netzwerk beschreibt. Auf diese Weise erhalten die Mitarbeiter intuitiven Zugriff auf die transaktionalen Informationen, indem sie von Knoten zu Knoten (bzw. z. B. vom Social Business Object *Peter Müller* zu Social Business Object *Rechnung 1725*) navigieren und auf die Informationen (in Anlehnung an ein soziales Netzwerk) aus der Perspektive eines Knotens (bzw. Social Business Object) als *Social Profile* über das Benutzerfrontend zugreifen. Diese Ontologie ist zum einen von den Daten in den Quellsystemen abhängig, aber auch von den Anforderungen der Nutzer in dem Unternehmen. Das entstehende Netzwerk aus der Ontologie und den Daten aus den Quellsystemen unterscheidet sich von Nutzer zu Nutzer, da die Kanten individuell (nutzerspezifisch) sind. Außerdem handelt es sich bei Vössing um eine Virtualisierung, da das Netzwerk nicht gespeichert wird, sondern lediglich die Ontologie. Das Potenzial dieser Individualsoftware wurde von den Mitarbeitern des Unternehmens und von weiteren Unternehmensvertretern auf Konferenzen als zielführend bewertet (Gewehr et al., 2017), um eine innovative Informationsintegration in Unternehmen umzusetzen. Dieser *Key Case Vössing*, in dem es um die Erhebung der dort entwickelten Ontologie-basierten Integrati-

⁴ Bei der Entwicklung der Software von dem Ingenieurbüro Vössing waren praktische Fragestellungen relevant und wissenschaftliche Konzepte (wie z. B. Informationsintegration oder Ontologien) wurden nicht bewusst eingesetzt. Nichtsdestotrotz wird in diesem Abschnitt die Anwendung schon aus Sicht der Wissenschaft beschrieben, um den Zusammenhang zu den existierenden Ontologie-basierten Integrationsansätzen herzustellen.

⁵ Bei Vössing wurde das Schema nicht „Ontologie“ genannt und es wurden auch keine Semantic-Web-nahe Technologien eingesetzt. Um die Ansätze hier zu vergleichen, wird an dieser Stelle aber der Begriff „Ontologie“ verwendet.

onssoftware geht, wird in Abschnitt 3.1 detaillierter erläutert und der darin beschriebene Ansatz in Kapitel 4 als SoNBO (Social Network of Business Objects) konzeptualisiert.

Der in dem Unternehmen identifizierte Informationsintegrationsansatz lässt sich wie bereits erwähnt als Ontologie-basierter Integrationsansatz einordnen und es existieren in der Literatur ähnliche Konzepte, zum einen der (Enterprise) Knowledge Graph (EKG) unter der Verwendung von Linked (Enterprise) Data (LED) und zum anderen der Ontology-Based Data Access (OBDA). Bei beiden Konzepten werden ebenfalls Ontologien verwendet, um die Daten aus den Quellsystemen zu beschreiben und dem Nutzer in einer anderen Struktur zur Verfügung zu stellen, und nutzen dafür Standards aus dem Web. Beide Ansätze adressieren also auch die im Abschnitt zuvor beschriebene Herausforderung, eine flexible und effektive Datenintegration umzusetzen, da eine Ontologie flexibel erweitert werden kann. Im Web kristallisierten sich seit dem Beginn der 1990er Jahre ähnliche Herausforderungen bezüglich der Informationsintegration heraus: Die Informationen liegen im Web über zahlreiche Webseiten verteilt und durch das Web 2.0 kam das Phänomen Big Data hinzu. Die Herausforderung des Zugriffs auf die Informationen in den verschiedenen Webseiten und insbesondere die Suche nach Informationen adressieren vor allem Suchmaschinen-Unternehmen wie z.B. Google. Diese nutzen dafür Web-Standards, die das W3C (W3-Konsortium) und insbesondere Tim Berners-Lee im Rahmen des Semantic Web ins Leben riefen (Berners-Lee, 2006, 2008). Dafür sollen idealerweise die Informationen einer Webseite als Linked Data in einer vom W3C entwickelten Sprache (RDF) veröffentlicht (quasi „übersetzt“) werden. Wenn die Informationen aller Webseiten in einer einheitlichen Sprache und Struktur vorliegen, sollen diese anschließend mit den Informationen der anderen Webseiten zu einem webumfassenden Knowledge Graphen verknüpft werden. Für die Verknüpfung der Informationen soll eine (die Welt erklärende) Ontologie als Schema dienen (Weller, 2014). Somit besteht also ein Knowledge Graph aus der Faktenbasis (= Informationen aus den Webseiten) und einer Ontologie, mit deren Hilfe die Faktenbasis zu einem Knowledge Graphen wird. Auf diese Konzepte (insbesondere Ontologie, Linked Data, Knowledge Graph) wird genauer in Kapitel 3 eingegangen. In der Realität allerdings existiert keine Ontologie, die die Welt in einer einheitlichen Struktur definiert. Stattdessen bildeten sich in der Informatik verschiedene Ontologien für verschiedene Domänen (z.B. Unternehmen, Wikipedia, ...) heraus. Dadurch existiert auch (noch) kein umfassendes Semantic Web oder ein webumfassender Knowledge Graph, sondern es bildeten sich vereinzelte Knowledge Graphen heraus. Einer der bekanntesten Knowledge Graphen ist Googles Knowledge Graph, der für die Suche nach Informationen verwendet wird. Bei einer Suche über Google wird beispielsweise in diesem Knowledge Graph gesucht und nicht in den Webseiten selbst. Google wiederum greift auf existierende Linked Data und Knowledge Graphen wie z.B. DBpedia zu. DBpedia gilt als semantische Schwester von Wikipedia. Wenn das Semantic Web, Knowledge Graphen und der Einsatz von Ontologien als Konzepte für die Integration von Informationen betrachtet werden, indem verschiedene Datenstrukturen in ein einheitliches Datenschema übersetzt und schließlich miteinander verknüpft werden, um besser darin zu suchen, können diese Konzepte auch auf Unternehmen übertragen werden. Dies wurde von Forschern aus der Informatik auch bereits gemacht und es entwickelten sich Ontologie-basierte Integrationsansätze für Unternehmen in der Forschung (Galkin, Auer, Kim, & Scerri, 2016; Galkin, Auer, & Scerri, 2016;

Villazon-Terrazas, Garcia-Santa, Ren, Srinivas, et al., 2017). Diese existieren allerdings noch insbesondere in der Theorie und es fehlen systematische Umsetzungen in der Praxis (Galkin, Auer, & Scerri, 2016). Ein Integrationsansatz, der ebenfalls eine Ontologie als Schema verwendet, ist OBDA (Ontology-Based Data Access) und entstand in der Wissenschaft, indem unter der Verwendung von semantischen Technologien der Datenzugriff auf verschiedene Quellsysteme vereinfacht wird. Die Idee ist dabei, dass die Ontologie die Datenbankabfragen der Nutzer übersetzt. In der Ontologie sind die Abfragen auf die Quellsysteme durch ein Mapping übersetzt, sodass der Nutzer für den Informationszugriff lediglich Abfragen auf die Ontologie formulieren muss. Somit können durch die Ontologie auch verschiedene Quellsysteme angeschlossen werden, aber der Nutzer muss sich nicht mit den Abfragesprachen dieser verschiedenen Systeme auseinandersetzen. Auch bei OBDA werden Semantic-Web-Technologien genutzt. Die Anwendungssicht in Abb. 1.3 kann ebenfalls für beide Konzepte (EKG und OBDA) herangezogen werden.

Das Konzept in der Software von Vössing und die Konzepte aus dem Semantic Web haben die Gemeinsamkeit, dass eine Ontologie zur Beschreibung der Daten in den zu integrierenden Quellsystemen genutzt wird. Dies wird *Ontologie-basierte Integration* genannt (Leser & Naumann, 2007). In Verbindung mit den Daten entsteht ein virtualisierter oder materialisierter Knowledge Graph, den die Nutzer für den Informationszugriff auf die Quellsysteme verwenden. Dieses Prinzip ist in allen Ansätzen gleich und in Abb. 1.3 dargestellt. Sie unterscheiden sich allerdings insbesondere in dem Aufbau der Ontologie und damit verbunden in dem Aufbau des Knowledge Graphen (unabhängig davon, ob dieser abgespeichert wird oder nicht). Bei Vössing ist der Knowledge Graph ein soziales Netzwerk und der Nutzer navigiert für den Informationszugriff von Knoten zu Knoten. Die dahinterstehende Ontologie ist auf die spezifischen Anforderungen des jeweiligen Unternehmens zugeschnitten und der entstehende Knowledge Graph ist durch die unterschiedlichen Zugriffsrechte für jeden Nutzer individuell und einzigartig. Auf diese Weise werden dem Nutzer sinnvoll ausgewählte Informationen intuitiv präsentiert. Diese Prinzipien (identifiziert in einer Individualsoftware) stellen allerdings noch kein wissenschaftliches Konzept dar, wodurch dieses Potenzial der Informationsintegration noch nicht auf weitere Unternehmen übertragen werden kann. An dieser Stelle setzt die Dissertation an, indem eine wissenschaftliche Beschreibung von SoNBO entwickelt und umgesetzt wird (auf der Basis der identifizierten Individualsoftware und den Semantic-Web-Konzepten) und somit die systematische Einführung von SoNBO in einem beliebigen Unternehmen ermöglicht wird. Zentral dabei ist u. a. die Erstellung einer Ontologie, was auch nach Sequeda & Miranker (2017) ein offener Forschungsbereich ist.

1.2 Forschungsziele

Das Forschungsziel dieser Dissertation adressiert die zuvor beschriebenen Herausforderungen und es besteht somit in der Weiterentwicklung des Wissens im Bereich der Ontologie-basierten Informationsintegrationsansätze, indem die existierenden Ansätze aus dem Semantic Web durch das SoNBO-Konzept um soziale Prinzipien (vgl. Abschnitte 3.2.1.7 und Abschnitt 4.2) und die Einbindung der Unternehmens- und Nutzer-Sicht erweitert werden. Dies wird erreicht, indem SoNBO zunächst konzept-

tualisiert (Kapitel 4) wird. Mithilfe dieser Konzeptualisierung wird das SoNBO-Framework (bestehend aus Modellen, Methoden und Tools) daraufhin entwickelt und evaluiert (Kapitel 5 und 6). Dieses soll bei der Einführung von SoNBO in einem beliebigen Unternehmen unterstützen.

Um dieses Ziel zu erreichen sind folgende Teilziele notwendig (vgl. Tab. 1.1): Zunächst (1a) wird der Status quo in der Praxis mithilfe der Fallstudienmethodik erhoben, indem der Key Case (vgl. 2.2.1) mit der identifizierten SoNBO-Anwendung beschrieben wird. Außerdem wird (1b) der Status quo in der Wissenschaft identifiziert, indem die existierenden Semantic-Web-Ansätze mithilfe einer Literaturlanalyse erhoben werden. Im nächsten Schritt (2a) erfolgt die Konzeptualisierung von SoNBO, indem die aus dem Key Case relevanten Aspekte mithilfe der Ontologie-basierten Integrationsansätze beschrieben und somit die neuen Eigenschaften (soziale Prinzipien sowie Berücksichtigung der Unternehmens- und Nutzer-Sicht) miteingebunden werden. Das nächste Teilziel (2b) ist die Entwicklung des Frameworks, welches aus (Referenz-) Modellen, Methoden und Tools besteht, um SoNBO in einem Unternehmen einzuführen. Im nächsten Schritt (3) wird dieses Framework evaluiert, indem es in einem Unternehmen angewandt wird. Das letzte Teilziel (4) ist eine Diffusion, indem die Erkenntnisse und die Grenzen der Konzeptualisierung und des Frameworks identifiziert werden.

Die Konzeptualisierung wird technologieunabhängig formuliert. Das Framework wird sowohl technologieunabhängig also auch technologiebezogen erläutert, da eine Evaluation ohne Technologiebezug nicht möglich ist.

Tab. 1.1: Forschungsziele

#	Forschungsziel und -teilziele
	Weiterentwicklung des Wissens im Bereich Ontologie-basierte Informationsintegration durch die Erweiterung dieses Integrationsansatzes um das Konzept Social Network of Business Objects (SoNBO) und durch die Entwicklung von wissenschaftlich fundierten und praxisrelevanten Modellen, Methoden und Tools (= Framework), die eine Einführung von SoNBO in ein beliebiges Unternehmen ermöglichen
1a	Status quo Praxis: Identifikation eines erweiterten Ontologie-basierten Integrationsansatzes in der Praxis
1b	Status quo Wissenschaft: Identifikation von existierenden Ontologie-basierten Integrationsansätzen zur Informationsintegration und deren Modellen, Methoden und Tools zur Einführung in Unternehmen
2a	Entwicklung: Konzeptualisierung eines erweiterten Ontologie-basierten Integrationskonzeptes (= SoNBO)
2b	Entwicklung: Entwicklung eines Frameworks bestehend aus Modellen, Methoden und Tools zur Einführung einer SoNBO-Anwendung in Unternehmen
3	Evaluation: Evaluation des Frameworks mit den für SoNBO geeigneten Modellen, Methoden und Tools
4	Diffusion: Anwendung der Erkenntnisse und Identifikation der Grenzen des Frameworks

Der Beitrag für die Wissenschaft besteht in der Erweiterung des Wissens im Bereich der Ontologie-basierten Informationsintegration durch die Entwicklung und Evaluation eines neuen Ansatzes (Konzept und Framework) für eine Ontologie-basierte Informationsintegration. Dadurch wird ein aus der Informatik stammender Integrationsansatz mit Aspekten aus einer Individualsoftware für die Wirtschaftsinformatik angereichert. Bei SoNBO wird nicht nur bottom-up aus den Datenbanken heraus eine Ontologie und somit der Knowledge Graph erzeugt, sondern durch Modelle, Methoden und Tools

werden auch das Wissen und die Anforderungen der Mitarbeiter (top-down) miteinbezogen. SoNBO stellt dann die Informationen unter Zuhilfenahme von sozialen Funktionalitäten zur Verfügung. Außerdem besteht der Beitrag in der Bereitstellung eines nützlichen Artefaktes für die Praxis (belegt durch die Evaluation) in Form des Frameworks, welches zur Einführung von SoNBO in einem Unternehmen dient. Somit stellt die Dissertation auch ein Beitrag für die Praxis dar.

1.3 Aufbau der Dissertation

Die Dissertation ist wie folgt aufgebaut: Zunächst wird das Forschungsdesign (vgl. Kapitel 2) dargelegt, indem die verwendete Methodik (Design Science Research) und die darin verwendeten Methoden als Mixed Methods Ansatz beschrieben werden. Anschließend (Kapitel 3) wird der Status quo in der Praxis und in der Wissenschaft erläutert. Der Status quo in der Praxis bezieht sich auf den Key Case zum SoNBO-Ansatz und für den Status quo in der Wissenschaft werden die Ergebnisse aus einer strukturierten Literaturanalyse vorgestellt. In Kapitel 4 wird die Konzeptualisierung des SoNBO-Konzepts beschrieben und im darauffolgenden Kapitel 5 erfolgt die Präsentation des entwickelten Frameworks, indem die Phasen und die sich darin befindenden Modelle, Methoden und Tools beschrieben werden. Anschließend werden die Ergebnisse der Evaluation des Frameworks in dem Unternehmen KOSMOS Verlag (Kapitel 6) präsentiert. In Kapitel 7 erfolgt die Diffusion, indem der Beitrag für die Praxis und die Wissenschaft sowie die Grenzen und der weitere Forschungsbedarf erläutert werden. Die Dissertation schließt mit einem Fazit (Kapitel 8).

2 Forschungsdesign

Es gibt unterschiedliche Ansätze, den Forschungsprozess zu strukturieren (Blaikie, 2010; Creswell, 2009; Crotty, 1998). Crotty (1998) schlägt vor, sich im Forschungsprozess an den vier Elementen *Epistemologie* (Erkenntnistheorie), *theoretische Perspektive*, *Methodik* und *Methoden* zu orientieren. Es geht dabei darum, sich mit den verschiedenen Ausprägungen dieser Elemente zu beschäftigen und die für das Forschungsthema relevanten begründet auszuwählen.

Für die Auswahl der Erkenntnistheorie ist es relevant, in welcher Art von Beziehung der Mensch zu dem zu erforschenden Objekt steht. Da das übergeordnete Ziel von Forschung darin besteht, einen Erkenntnisgewinn zu erzielen, ist es wichtig, sich darüber im Klaren zu sein, welche Arten von Wissen möglich sind und wie sichergestellt werden kann, dass diese sowohl angemessen als auch legitim sind (Maynard, 1994). Daher beschäftigt sich die Erkenntnistheorie mit der Bereitstellung einer philosophischen Grundlage dafür. Es gibt dazu verschiedene Strömungen: Im Objektivismus gibt es eine objektive Realität, die unabhängig von der Wahrnehmung des Subjekts entdeckt werden kann. Der Konstruktivismus geht davon aus, dass keine objektive Wahrheit entdeckt werden kann, sondern diese konstruiert wird. Dies geschieht durch die Interaktion des Menschen mit der Welt und mit anderen Menschen. Ein weiterer epistemologischer Standpunkt ist der Subjektivismus, bei dem die Erkenntnis nicht durch die Interaktion eines Subjekts und eines Objekts entsteht, sondern indem das Subjekt dem Objekt Bedeutung zuschreibt. Durch eine theoretische Perspektive werden im Forschungsprozess in Abhängigkeit der gewählten Erkenntnistheorie verschiedene Annahmen getroffen, die für die gewählte Methodik relevant sind. Es geht dabei um die Sichtweise auf die menschliche Welt und das soziale Leben (Crotty, 1998). Die verschiedenen Strömungen prägen auch die Entwicklung der deutschsprachigen Wirtschaftsinformatik (WI) und der englischsprachigen Information Systems (IS). Während die IS stark behavioristisch und somit objektivistisch geprägt ist, versteht sich die WI als konstruktionsorientierte Forschungsdisziplin, die sich auf die Gestaltung von Informationssystemen fokussiert. Das primäre Gestaltungsziel ist dabei die Nützlichkeit, indem die konstruierten Artefakte konkrete und relevante Probleme lösen sollen (Baskerville, Lyytinen, Sambamurthy, & Straub, 2011; Frank, 2006; Junglas et al., 2011; Nett & Stevens, 2009; Orlikowski & Baroudi, 1991; Österle et al., 2011). Da das Ziel der Dissertation die Entwicklung nützlicher Artefakte (Konzept und Framework) zur Lösung des in Abschnitt 1.1 dargestellten Problems ist und somit an der Gestaltung von Informationssystemen beteiligt ist, ist der konstruktionsorientierte Ansatz sinnvoll. Dieser ist im Rahmen der Dissertation dem Subjektivismus unterzuordnen, da der Forschungsgegenstand objektiv ist und die Autorin der Dissertation als Subjekt diesem Bedeutung zuschreibt. Für die Entwicklung der nützlichen Artefakte stellt Design Science Research eine geeignete Methodik dar. Außerdem werden bei der Entwicklung verschiedene Methoden verwendet. Im Folgenden wird die für die Arbeit ausgewählte Methodik Design Science Research sowie die verwendeten Methoden im Sinne des Mixed Methods Ansatzes vorgestellt.

2.1 Methodik: Design Science Research

Die Methodik hat für den Forschungsprozess die Bedeutung, eine Strategie für das Erreichen der Forschungsziele bereitzustellen. Sie ist eng mit der gewählten Epistemologie und der theoretischen Perspektive verknüpft. Die Methodik Design Science Research (DSR) stammt ursprünglich aus den Ingenieurwissenschaften und es wird damit ein nützliches Artefakt für ein realweltliches Problem entwickelt. Es gibt verschiedene Strömungen, doch ein gemeinsamer Konsens ist, dass DSR das Lernen durch den Akt des Bauens beinhaltet. Demgegenüber wurde in den letzten Jahren diskutiert, was zu bauen ist (= Definition des DSR-Artefakts) und wie es aufgebaut werden soll (= DSR-Methodik) (Hevner, March, Park, & Ram, 2004; Kuechler & Vaishnavi, 2008; Nunamaker, Briggs, Derrick, & Schwabe, 2015). Bezüglich der Definition des DSR-Artefaktes existieren verschiedene Verständnisse, die von Peffers, Tuunanen & Niehaves (2018) in fünf Genres klassifiziert werden. Diese sind: *IS design theory*, *design science research methodology*, *design-oriented IS research*, *explanatory design theory* und *action design research*. In den verschiedenen Genres wird unter einem Artefakt beispielsweise eine Anwendung, System, Prototyp oder auch eine Theorie oder Methode verstanden. Ein unterscheidendes Merkmal zwischen den Artefakten ist die Abstraktion. Während die soeben zuerst aufgeführten Artefakte nicht abstrakt sind und unter das klassische Verständnis eines Artefaktes fallen, gewinnen zunehmend auch abstrakte Artefakte (wie z. B. Theorien) an Bedeutung (Alter, 2017; Gericke & Winter, 2009; Peffers et al., 2018; Vaishnavi & Kuechler, 2015). In der Dissertation entstehen zwei Artefakte. Zum einen wird ein abstraktes Konzept und zum anderen ein Framework mit konkreten Modellen, Methoden und Tools entwickelt.

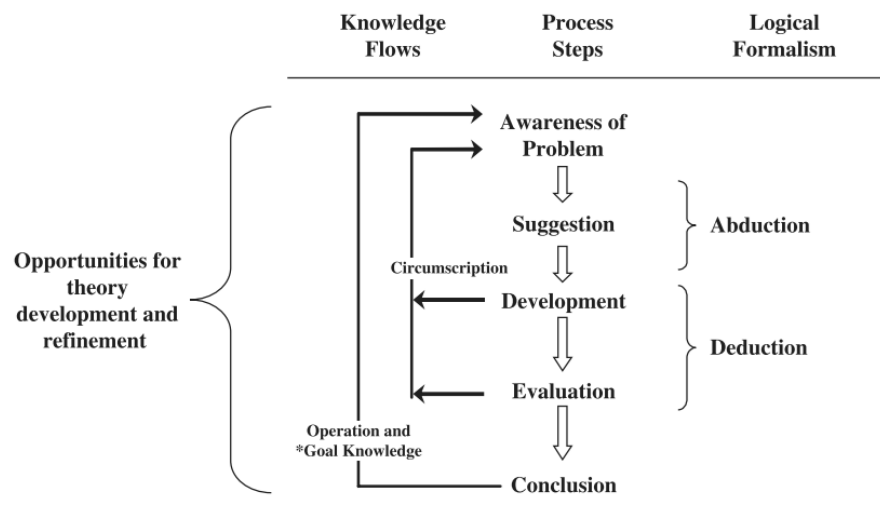


Abb. 2.1: Argumentation im Design Science Research Zyklus (Kuechler & Vaishnavi, 2008, p. 493)

Mit der DSR-Methodik wird die Schaffung von neuem Wissen durch das Entwerfen von neuartigen oder innovativen Fakten erreicht (Vaishnavi & Kuechler, 2004). Für die DSR-Methodik schlagen Kuechler & Vaishnavi (2008) folgende Argumentation vor (vgl. Abb. 2.1): In der ersten Phase wird das Be-

wusstsein für ein existierendes realweltliches und relevantes Problem in der Praxis geschaffen. In der zweiten Phase (Suggestion) werden verschiedene existierende Ansätze für dieses Problem identifiziert, die von der Forschung bereits zu ähnlichen Themen gewonnen wurde. Auf diese Weise kann in einem Gedankenexperiment eine erste Lösung aus diesen Ansätzen entwickelt werden. In der Entwicklungsphase werden die Vorschläge aus der Phase zuvor durch die Entwicklung von Artefakten konkretisiert, indem diese konstruiert und iterativ verfeinert werden. Eine Evaluation des entwickelten Artefaktes erfolgt in der nächsten Phase. Die Evaluation zur Demonstration der Nützlichkeit ist ein entscheidender Schritt von Design Science (Peppers, Rothenberger, Tuunanen, & Vaezi, 2012). Der Übergang der Phasen erfolgt nicht wie in einem Wasserfallmodell, sondern der Forschungsprozess iteriert häufig (vor allem zwischen Entwicklungs- und Evaluationsphase). Es soll also ein Wechsel zwischen den Phasen stattfinden bis die Lösung des Problems (also das entwickelte Artefakt) den Anforderungen genügt und somit das identifizierte Problem löst. Auf diese Weise entsteht Wissen, das durch die letzte Phase (Conclusion) verbreitet werden soll (Kuechler & Vaishnavi, 2008; Vaishnavi & Kuechler, 2004). Diese beschriebenen Schritte lassen sich den bereits dargestellten Forschungszielen aus Abschnitt 1.2 zuordnen, woraus sich die Forschungsschritte ergeben (vgl. Abb. 2.2). Die Forschungsschritte werden in Abschnitt 2.3 detailliert beschrieben.

2.2 Forschungsmethoden: Einsatz verschiedener komplementärer Methoden

Die verschiedenen Forschungsziele verlangen verschiedene Methoden für eine geeignete Bearbeitung, weshalb ein Mischung aus verschiedenen komplementären Methoden (Creswell, 2009; Creswell & Clark, 2011) gewählt wurde. Auch wenn bestimmte Techniken ursprünglich für die Anforderungen konkreter Forschungsparadigmen entwickelt wurden (wie z. B. Methode quantitativer Fragebogen für Epistemologie Objektivismus), können diese auch für andere genutzt werden (Seale, 1999). Im Folgenden werden die in der Dissertation verwendeten Methoden vorgestellt und es wird dargelegt, welche Bedeutung sie im Forschungsprozess innehaben. Die hier vorgestellten Methoden werden im Forschungsprozess selbst für den Erkenntnisgewinn verwendet (= *Forschungsmethoden* nach Wilde & Hess (2007)). Sie werden vorrangig allgemein beschrieben und der Verweis auf den Abschnitt, in dem sie verwendet werden, gesetzt. In dem betreffenden Abschnitt werden diese dann konkret bezogen auf die Dissertation ausgeführt.

Im Folgenden werden die Methoden erläutert, die für den Forschungsprozess zur Datenerhebung relevant sind.

2.2.1 Qualitatives Interview und Fallstudie

Nach Wilde & Hess (2007) ist die Fallstudie eine spezielle Form der qualitativ-empirischen Methodik und untersucht einzelne Merkmalsträger intensiv. Ein *Fall* (im Folgenden: Case bzw. Key Case nach Thomas (2011)) kann auf der Basis eines qualitativ durchgeführten Interviews entstehen. Dann stellt das Interview keine eigenständige Methode dar, sondern wird als Datenerhebungserfahren betrachtet, welches qualitativ ausgewertet wird (Wilde & Hess, 2006, 2007).

Nach Flyvbjerg (2006) besitzt die Methode *Fallstudie* für die Forschung ein großes Potenzial. Da das in der Praxis gewonnene kontextbezogene Wissen genauso wertvoll ist wie kontextunabhängiges und theoretisches Wissen, kann auf der Basis eines konkreten *Case* eine Generalisierung und somit ein Beitrag für die Wissenschaft stattfinden. (Key) Cases untersuchen oftmals „komplexe, schwer abgrenzbare Phänomene in ihrem natürlichen Kontext“ (Wilde & Hess, 2007, p. 282). Bei einem verhaltenswissenschaftlichen Zugang steht die möglichst objektive Untersuchung von Thesen im Mittelpunkt. Liegt der Fokus auf einer Konstruktionsorientierung steht „die Interpretation von Verhaltensmustern als Phänotypen konstruierter Realitäten“ im Vordergrund (Wilde & Hess, 2007, p. 282). Auch eine Theoriebildung ist auf der Basis von einer Fallstudienforschung möglich (Eisenhardt, 1989; Flyvbjerg, 2006; Kvale & Brinkmann, 2009; Miles, Huberman, & Saldaña, 2014; Walsham, 1995). Eine Fallstudie ist also eine eingehende Untersuchung eines begrenzten Systems (z. B. Ereignis, Prozess oder Einzelperson) und die Grundlage dazu ist eine umfassende Datenerfassung. Mit „begrenzt“ ist gemeint, dass der Fall für die Forschung in Bezug auf Zeit, Ort oder weitere physikalische Grenzen getrennt betrachtet wird (Creswell, 2012). Es ist nützlich, verschiedene Arten von Fällen zu betrachten, die mehrfach untersucht werden. Zunächst kann ein Fall (auch: Case⁶) einzelne Personen, mehrere Personen, eine Gruppe, ein Programm, Ereignisse, Aktivitäten oder ein Prozess sein. Die Fälle an sich lassen sich in drei Arten klassifizieren: Entweder wird ein Fall ausgewählt, weil dieser ungewöhnlich ist und der Fall selbst von Wert und Interesse ist. Diese Art wird (1) *intrinsisch* genannt. Wenn der Schwerpunkt einer qualitativen Studie auf einem bestimmten Thema liegt und ein oder mehr Fälle zur Veranschaulichung des Problems verwendet werden, wird der Fall als (2) *Instrumentalfall* bezeichnet. Dieser dient dann dazu, ein bestimmtes Problem zu beleuchten. Außerdem können Fallstudien auch mehrere (Key) Cases umfassen, die dann als (3) *kollektive Fallstudie* bezeichnet werden. Darin werden mehrere (Key) Cases beschrieben und verglichen, um einen Einblick in ein Thema zu erhalten. Darüber hinaus gibt es noch die Möglichkeit, eine Tiefenfallstudie durchzuführen, indem der Forscher ein tiefgreifendes Verständnis zu einem Case entwickelt. Dazu sammelt er mehrere Arten von Daten. Für das tiefgreifende und gründliche Verständnis müssen lediglich wenige Fälle untersucht werden. Abschließend ist es notwendig, dass der Forscher den Fall in den übergeordneten Kontext einordnet (Creswell, 2012). In dieser Dissertation werden mit der Methode Tiefenfallstudie zwei Key Cases (Vössing und KOSMOS) erhoben, die in den entsprechenden Kapiteln beschrieben werden (vgl. Abschnitte 3.1 und 6.1).

Das Interview als qualitative Querschnittsanalyse ist eine Erhebungstechnik. Dabei handelt es sich um eine einmalige Erhebung, welche anschließend ausgewertet wird. Das Ergebnis ist ein Querschnittsbild über die Probanden hinweg (Wilde & Hess, 2007). Qualitative Interviews werden in der konstruktionsorientierten Forschung angewendet, um an das Wissen zu gelangen, welches der Forscher nicht antizipierte. Die Herausforderung ist dabei, eine Gruppe mit Probanden zu identifizieren, die dieses besitzt. Dieses Wissen wird dann durch das qualitative Interview erhoben, indem der Forscher durch Im-

⁶ „Fall“ und „Case“ wird hier synonym verwendet.

pulse und offene Fragen den Interviewpartner frei reden lässt, sodass die Perspektiven des Forschers oder Ergebnisse aus früheren Forschungsergebnissen so wenig Einfluss wie möglich haben. Diese Interviews werden aufgezeichnet und anschließend transkribiert, sodass der Inhalt auf diese Weise möglichst genau nachvollzogen werden kann (Creswell, 2009). Für die Auswertung werden in der Regel die Informationen zur Analyse in Wörter umgeschrieben (Coding). Ein Vorteil der qualitativen Forschung ist, dass die Probanden nützliche Informationen liefern, wenn sie offen auf eine Frage antworten können. Allerdings besteht auch hier wie bei einer quantitativen Forschung der Nachteil, dass die Ansichten des Interviewers durch den Forscher gefiltert werden. Interviewdaten können also täuschen und die Perspektive bieten, die der Befragte vom Forscher hören lassen möchte (Creswell, 2009). Außerdem kann die Anwesenheit des Forschers die Reaktion des Befragten beeinflussen. Weiterhin können die Antworten des Befragten auch möglicherweise nicht klar sein, sodass der Interpretationsspielraum zu groß ist (Creswell, 2009). Bei der Durchführung gibt es verschiedene Arten von Interviews: One-to-One Interview, Focus Group Interview, Telefoninterview, E-Mail-Interview und Fragebögen mit offenen Fragen (Creswell, 2009). Bei allen Arten von Interviews sind allgemeine Schritte erforderlich: Zunächst (1) müssen die Interviewpartner identifiziert werden. Anschließend (2) wird der Interviewtyp ausgewählt. Daraufhin erfolgt (3) die Durchführung des Interviews und damit verbunden die Aufnahme der Fragen und Antworten. Während des Interviews sollten (4) kurze Notizen des Forschers erstellt werden. Für das Interview sollte (5) ein angemessener Ort ausgewählt werden, der ruhig ist, sodass sich alle Parteien auf das Interview konzentrieren können. Weiterhin muss (6) im Rahmen des Interviews die Zustimmung des Befragten zur Teilnahme an der Studie eingeholt werden. Außerdem gilt, dass der Interviewer (7) zwar einen Plan für das Interview haben, aber auch flexibel sein sollte. Eine Vorbereitung mit Unterfragen (8) hilft dabei. Abschließend gilt (9), dass der Forscher professionell auftreten sollte, indem er das Interview mit einem Dank schließt, die Vertraulichkeit der Daten versichert und eine Zusammenfassung der Ergebnisse anbietet.

Bei der qualitativen Forschung und insbesondere bei der Durchführung eines Interviews und Erstellung eines (Key) Cases muss dem Forscher also bewusst sein, dass die Subjektivität und somit eventuell vorgefasste Ansichten miteinfließen, sobald der Forscher in den Forschungsprozess eingreift. Dies geschieht beispielsweise bei der Formulierung der Impulse oder Fragen während des offenen Interviews, der Transkription, der Erstellung des (Key) Cases und schließlich der Interpretation der Ergebnisse und lässt sich schwer vermeiden. Durch ein Aufrechterhalten dieses Bewusstseins während des Forschungsprozesses kann dieser Einfluss der Subjektivität reduziert werden.

Die hier aufgeführten Methoden werden ursprünglich für die Beantwortung einer verhaltenswissenschaftlichen Forschungsfrage verwendet. Beispielsweise wird im Anschluss an eine Forschungsfrage die qualitativen (oder quantitativen) Methoden verwendet, um eine Querschnittsanalyse zu erstellen und die Forschungsfrage zu beantworten. Nach Wilde & Hess (2007) sind diese qualitativen Methoden dadurch dem verhaltenswissenschaftlichen Ansatz unterzuordnen, da sie unterstützen, ein existierendes Phänomen zu erklären. In der Dissertation werden diese offenen Methoden hingegen für das Erreichen eines *Teilziels* verwendet, indem sie den Status quo in der Praxis erheben. Mit diesen Erkennt-

nissen aus dem (Key) Case wird dann ein Konzept (und anschließend ein Framework) entwickelt. Somit werden diese Methoden nicht im Sinne eines verhaltenswissenschaftlichen Forschungsansatzes verwendet, sie dienen vielmehr als Ausgangsbasis für das Ergebnis der Dissertation. Auf einem Zeitstrahl betrachtet, war die Erhebung des (Key) Case Vössing der eigentliche Auslöser der SoNBO-Forschung. Der Case KOSMOS diente zur Evaluation der Ergebnisse in der Praxis. Das konkrete Vorgehen zur Erstellung der im Rahmen der Dissertation durchgeführten Tiefenfallstudie wird in Abschnitt 3.1.1 (für den Key Case Vössing) und in Abschnitt 6.1 (für den Key Case KOSMOS) erläutert.

2.2.2 IndustryConnect

Die in dieser Dissertation mit der Fallstudien-Methode erhobenen Key Cases entstanden durch die Arbeit mit IndustryConnect, welche eine praxisorientierte Forschungsgemeinschaft bestehend aus mehreren Organisationen darstellt (Williams & Schubert, 2017). Da diese Key Cases und somit auch die Arbeit mit IndustryConnect als Methode eine tragende Rolle in dieser Dissertation spielen, wird diese Forschungsinitiative im Folgenden erklärt: IndustryConnect ist ein Verbund aus Vertretern der Wirtschaft und Wissenschaft, in dem seit mehr als fünf Jahren eine praxisorientierte Langzeitforschung im Themengebiet *Digitaler Arbeitsplatz* und deren Entwicklung durchgeführt wird. Ursprünglich lag der Fokus auf der Erforschung von Problemen im Zusammenhang mit der Implementierung und Einführung von ECS sowie den Erkenntnissen über den komplexen sozio-technischen Wandel rund um ECS (Williams & Schubert, 2017). Während dieses Thema weiterhin ein zentraler Bestandteil ist, wurden weitere Themen (z. B. Informationsintegration) aufgenommen, wodurch das übergeordnete Thema die Erforschung des digitalen Arbeitsplatzes wurde. Es finden jedes Jahr mindestens zwei physische Treffen statt, an denen sich die Praxisvertreter und die Wissenschaftler zu einem von den Wissenschaftlern organisierten Workshop treffen und sich strukturiert austauschen. Darüber hinaus finden bilaterale Treffen zwischen Praxisvertreter und Wissenschaftler zu Fokusthemen statt. Im Jahr 2020 umfasste die Community 68 Praxisvertreter aus 38 Organisationen sowie 2 Universitätsprofessorinnen, 9 wissenschaftliche Mitarbeiter und 9 studentische Hilfskräfte (Schubert & Williams, 2020). IndustryConnect zeichnet sich dadurch aus, dass eine Forschung mit (Menschen in) Unternehmen und nicht über Unternehmen durchgeführt wird. Dies ist durch die vertrauensvolle Gemeinschaft von Forschenden und Praktikern möglich, die durch das Langzeitprojekt aufgebaut wurde, und auf diese Weise können wissenschaftlich und praktisch relevante Forschungsfragen kombiniert mit rigorosen Forschungsmethoden behandelt werden (Schubert & Williams, 2020). Die Forschungsfragen werden inspiriert durch die Praxis und durch die Praktiker und Wissenschaftler gemeinsam erarbeitet. Die Forschungsergebnisse werden mit rigorosen Methoden erhoben und basieren auf ausgewerteten empirischen Daten, welche in die Praxis zurücktransferiert und evaluiert werden (Schubert & Williams, 2020). Dieser Praxisbezug im letzten Forschungsschritt erfüllt die Forderung von Nunamaker et al. (2015), die letzte Meile zu gehen (Schubert & Williams, 2020). Für die Beantwortung der Forschungsfragen werden folgende Methoden verwendet: Aktionsforschung, Fokusgruppen, Workshops, (eXperience) Fallstudien und Milestones (Schubert & Williams, 2020). Sowohl für die Praxisvertreter als auch

für die Wissenschaftler gibt es durch IndustryConnect einen Mehrwert: An den Workshops haben Praxisvertreter die Gelegenheit, sich mit anderen Praxisvertretern über Strategien, Maßnahmen oder Werkzeuge auszutauschen. Die Wissenschaftler haben an den Workshops und in den bilateralen Treffen die Gelegenheit, Einblicke in den Alltag der Praxis zu gewinnen und empirische Daten zu sammeln (Schubert & Williams, 2020). Die Forschungsergebnisse nehmen verschiedene Formen an: Erklärungs- und Klassifikationsmodelle/Theoriebildung, Methoden oder Prototypen (Schubert & Williams, 2020).

2.2.3 Literaturanalyse

Eine Literaturanalyse stellt im Forschungsprozess eine wichtige wissenschaftliche Forschungsmethode dar. Sie ist während des gesamten Forschungsprozesses relevant und wird in unterschiedlichen Ausprägungen durchgeführt. Huff (2009) unterscheidet vier Arten bei der Literaturanalyse: (1) Survey, (2) Critical review, (3) Systematic review sowie (4) Supporting search. Zu Beginn einer Forschung wird eine initiale Literaturrecherche durchgeführt, um einen Überblick zu erhalten und erste zentrale Beiträge zu identifizieren (Survey, Critical review). Außerdem wird während des Forschungsprozesses eine unterstützende Suche durchgeführt. Die systematische Literaturrecherche soll für jede weitere Person reproduzierbar sein, weshalb das Vorgehen dokumentiert werden muss (Fink, 2014; Okoli & Schabram, 2010). Der Zweck eines systematischen Reviews ist zu identifizieren, was bereits in dem Forschungsgebiet bekannt ist (Huff, 2009). Dazu werden von Huff (2009) verschiedene Regeln aufgestellt. Das Forschungsziel muss vor der Recherche formuliert werden. Außerdem sollten bedeutende Forscher in dem zu untersuchenden Gebiet identifiziert werden. Weiterhin können Experten gefragt werden, ob diese das Vorhaben ebenfalls als sinnvoll erachten. Es gibt verschiedene Vorschläge für die Vorgehensweise bei einer systematischen Literaturanalyse (Kitchenham & Charters, 2007; Torraco, 2005; vom Brocke et al., 2009, 2015; Webster & Watson, 2002). Nach vom Brocke et al. (2009) wird zunächst der Umfang der Literaturanalyse festgelegt. Anschließend wird das Thema z. B. mithilfe einer Concept Map festgelegt. Daraufhin wird die Literaturrecherche durchgeführt. Dazu werden mit sinnvollen Schlüsselwörtern die einschlägigen Literaturdatenbanken, Journals und Konferenzen durchsucht. Außerdem können in den Journals und Konferenzen selbst auch ohne Keyword-Suche anhand der Inhalte relevante Beiträge identifiziert werden. Anschließend werden zentrale Beiträge für eine Schneeballsuche (Vorwärts- und Rückwärtssuche) verwendet. Wichtig ist, die zentralen Autoren bzw. die Gruppe darum herauszufiltern, um davon ausgehend weitere relevante Beiträge zu erhalten. Bei der Eingabe der Suchbegriffe werden die Beiträge anhand des Titels ausgewählt. Die Auswahl wird anhand des Abstracts und schließlich des Beitrags selbst weiter in Abhängigkeit des Themengebietes gefiltert (vom Brocke et al., 2009). Nach der Literaturanalyse erfolgt dann die Literatursynthese, indem durch die Darstellung der Literatur bezogen auf die Fragestellung neue Erkenntnisse abgeleitet werden. Abschließend wird wieder Bezug zur Forschungsagenda genommen. Ein weiteres Vorgehen nach Randolph (2009) beinhaltet folgende Schritte: (1) Problem formulation, (2) Data collection, (3) Data evaluation, (4) Analysis and interpretation sowie (5) Public presentation. Die Inhalte dieser Schritte sind weitestgehend deckungsgleich mit den zuvor beschriebenen Schritten. Eine Literaturanalyse soll-

te wie bereits erwähnt nachvollziehbar durchgeführt werden, sodass ein weiterer Forscher mit demselben Vorgehen auch zu denselben Ergebnissen und somit Erkenntnissen gelangen kann. Wichtig ist allerdings anzumerken, dass nicht unbedingt alles durch eine strukturiert durchgeführte Literaturanalyse erfasst werden kann, da beispielsweise die Keywords nicht das komplette Forschungsfeld abdecken. Dies muss sich immer bewusst gemacht werden und der Forscher sollte flexibel und offen gegenüber den verwendeten Suchbegriffen sein.

In der Dissertation wird den beschriebenen Empfehlungen gefolgt und für die Erhebung des Status quo eine strukturierte Literaturanalyse durchgeführt. Das genaue Vorgehen bei der Literaturanalyse wird in Abschnitt 3.2.2.1 auf der Basis der hier vorgestellten Hintergründe vorgestellt. Nach der systematischen Literaturrecherche wird bis zum Ende der Dissertation eine unterstützende Literaturrecherche durchgeführt, sodass die Suche nach der wissenschaftlichen Literatur in diesem Themengebiet den gesamten Forschungsprozess (wie auch für eine qualitative Forschung empfohlen) begleitet (Blaikie, 2010; Fink, 2014; Hart, 1998, 2001; Huff, 2009; Jalali & Wohlin, 2012; Randolph, 2009; Torraco, 2005; vom Brocke et al., 2009, 2015; Webster & Watson, 2002; Wohlin, 2014).

Phase	Vorbereitung	Status quo	Entwicklung	Evaluation	Beitrag & Grenzen		
Ergebnis	Problemstellung, Forschungsziel, Forschungsfrage	Beschreibung des in der Praxis identifizierten Ansatzes	Beschreibung des Ontologie-basierten Integrationskonzeptes SoNBO (mit den neuen Merkmalen)	Framework mit für SoNBO geeigneten Modellen, Methoden und Tools zur Einführung eines SoNBO-Explorers in einem beliebigen Unternehmen	Beschreibung des Erkenntnisgewinns		
		Vergleich der identifizierten wissenschaftlichen Ansätzen			Beschreibung der Grenzen des Framework		
Forschungsschritt	Verständnis und Definition des Problems	Identifikation Forschungsziele und -fragen	Identifikation eines Ontologie-basierten Integrationsansatz (mit neuen Merkmalen) in der Praxis	Entwicklung des SoNBO-Konzepts	Entwicklung des SoNBO-Frameworks	Evaluation des SoNBO-Frameworks	Identifikation des Beitrags für die Wissenschaft und Praxis
			Identifikation von ex. Ontologie-basierten Integrationsansätze und deren Modellen, Methoden & Tools in wissenschaftlicher Literatur				Identifikation der Grenzen des Framework
Methode	Qualitatives Interview, Tiefenfallstudie (Case: Vössing), initiale Literaturanalyse	Qualitatives Interview, Tiefenfallstudie, strukturierte Literaturanalyse	Synthese (Zusammenführung des Status quo aus der Praxis & Status quo aus der Wissenschaft)	Evaluation des Prototyps in der Praxis durch Anwendung; Tiefenfallstudie (Case: KOSMOS)	Analyse der vorangegangenen Ergebnisse		
DSR	Awareness of Problem	Suggestion	Development	Evaluation	Conclusion		

Abb. 2.2: Forschungsvorgehen (Orientierung an Design Science Research)

2.3 Forschungsschritte

Die Forschungsschritte orientieren sich an der zuvor beschriebenen Methodik Design Science Research (DSR). Diese sind mit der Zuordnung zu den Schritten im DSR in Abb. 2.2 visualisiert. In den einzelnen Schritten werden verschiedene Methoden verwendet, mit denen ein Ergebnis in den jeweiligen Schritten erreicht wird. Im ersten Schritt findet eine *Vorbereitung* der Forschung statt, indem ein Verständnis und Definition des Problems und darauf aufbauend die Forschungsziele erstellt werden. Das Ergebnis ist eine klare Formulierung der Problemstellung sowie des Forschungsziels und -teilziele. Als Methode dazu dient die Tiefenfallstudie mit einem Unternehmen (basierend auf einem qualitativen Interview), durch welche neue Merkmale eines Informationsintegrationskonzepts ermittelt wurden, und eine initiale Literaturanalyse, durch welche ein Überblick über weitere, bereits existierende Integrationsansätze geschaffen wird. Dies ist in der Dissertation in Kapitel 1 dargelegt. Mit dem zweiten Forschungsschritt wird der *Status quo* in der Praxis und in der Wissenschaft erhoben. Dazu wird mithilfe einer Tiefenfallstudie ein neuartiges Integrationskonzept in Form einer Software beschrieben, die im Unternehmen Vössing entwickelt wurde und Merkmale eines Ontologie-basierten Informationsintegrationskonzeptes aufweist. Dieser Fall ist *intrinsisch* (vgl. Abschnitt 2.2.1), weil die untersuchte Individualsoftware selbst von Wert und Interesse ist. Für die Erstellung des Key Cases dient ein qualitatives Fokusgruppen-Interview (vgl. Abschnitt 2.2.1) mit dem Entwicklungsteam der Individualsoftware. Außerdem wird der Status quo in der wissenschaftlichen Literatur ermittelt, indem eine strukturierte Literaturanalyse durchgeführt wird. Nach Design Science Research (vgl. Abschnitt 2.1) liegt der Fokus in dieser Phase auf der Erhebung von ersten Vorschlägen zur Lösung des Problems auf der Basis der wissenschaftlichen Literatur. Dies wird in der Dissertation darüber hinaus durch den Status quo in der Praxis ergänzt. Die Ergebnisse dieses Kapitels sind zum einen die Beschreibung der in dem Unternehmen entwickelten Software durch den Key Case und zum anderen die Kategorisierung der Ergebnisse aus der Literaturanalyse, indem die Merkmale der existierenden Ontologie-basierten Integrationsansätze aus dem Semantic Web herausgearbeitet werden. Im dritten Schritt *Entwicklung* erfolgt zum einen die Konzeptualisierung von SoNBO auf der Basis des Status quo in der Praxis und der Literatur. Dazu werden die Elemente der beschriebenen Individualsoftware aus der vorherigen Phase abstrahiert, verallgemeinert und mit den Aspekten der Ontologie-basierten Integrationsansätze beschrieben. Außerdem erfolgt die Entwicklung des Frameworks mit den für die Einführung von SoNBO relevanten Modellen, Methoden und Tools. Dies geschieht durch die Synthese der Ergebnisse des vorangegangenen Forschungsschrittes. Die Ergebnisse sind die Beschreibung des SoNBO-Konzeptes und des SoNBO-Frameworks. Im vierten Schritt *Evaluation* wird insbesondere das SoNBO-Framework aber auch indirekt das SoNBO-Konzept evaluiert. Dies erfolgt, indem das SoNBO-Framework in der Praxis im Unternehmen KOSMOS Verlag für die Einführung eines SoNBO-Explorers angewandt wird. Bei der Durchführung der Gespräche mit den ausgewählten Mitarbeitern wird das SoNBO-Konzept zudem indirekt evaluiert. Das Ergebnis dieser Phase sind die Erkenntnisse und ggf. Anpassungen des entwickelten SoNBO-Frameworks. Der letzte Schritt arbeitet den *Beitrag* des SoNBO-Frameworks für die Wissenschaft und

Praxis sowie die *Grenzen* heraus. Dies erfolgt durch die Analyse der vorangegangenen Ergebnisse. Das Ergebnis ist die Beschreibung des Beitrags und der Grenzen.

2.4 Forschungsgrenzen

In der Dissertation wird ein Konzept und darauf aufbauend ein Framework zur Einführung einer SoN-BO-Anwendung in einem beliebigen Unternehmen entwickelt. Als Informationsquellen dienen eine Literaturanalyse zu Ontologie-basierten Integrationsansätzen und eine Tiefenfallstudie zu einem neuartigen Ontologie-basierten Integrationsansatz in Form einer Individualsoftware. Die Evaluation des Artefaktes erfolgt in der Praxis (in einem Unternehmen). Es steht dabei die Nützlichkeit des Artefaktes für die Herausforderung der flexiblen Integration von Informationen in Unternehmen im Vordergrund. Das dahinterstehende Paradigma ist der Subjektivismus, indem der Forschungsgegenstand als objektiv betrachtet wird und die Art der Beschreibung (Semantik) durch die Autorin der Dissertation beeinflusst wird. Der Forschungsgegenstand ist zum einen im Status quo vorhanden und stellt die Individualsoftware dar, die durch die Tiefenfallstudie dokumentiert wird. Zum anderen handelt es sich dabei um die existierenden wissenschaftlichen Ansätze in der Literatur, die im Rahmen der Literaturanalyse kategorisiert wurden. Außerdem werden die Ergebnisse (Konzept und Framework) als Forschungsgegenstand betrachtet, die auf eine nachvollziehbare Art entwickelt werden. Auch wenn diese als objektiv betrachtet werden, werden diese durch die Semantik (Art der Beschreibung) beeinflusst. Dieser subjektive Einfluss durch die Semantik lässt sich im Forschungsprozess nicht vermeiden. Allerdings kann durch die Evaluation neben der Nützlichkeit der Ergebnisse selbst (und damit auch der Ergebnisse des Status quo) auch die Nützlichkeit der Semantik demonstriert werden, wodurch der Evaluationsphase eine große Bedeutung zukommt. Somit ist das Ziel des Forschungsvorgehens in der Dissertation, dass die Ergebnisse mit dem hier beschriebenen Vorgehen von jedem anderen Forscher ebenfalls erreicht worden wären (Reproduzierbarkeit). Die Art der Beschreibung der Ergebnisse allerdings würde vermutlich variieren, worin auch die Grenzen der hier durchgeführten Forschung und somit der Ergebnisse liegen.

3 Status quo zu Ontologie-basierten Integrationsansätzen in der Praxis und Wissenschaft

In diesem Kapitel wird der Key Case zu der Individualsoftware („Self-Service“) von Vössing beschrieben (Status quo Praxis), welche Aspekte eines Ontologie-basierten Informationsintegrationsansatzes beinhaltet. Außerdem wird die für das Forschungsziel relevante Literatur zu Ontologie-basierten Informationsintegrationsansätzen (Status quo Wissenschaft) vorgestellt. Dieses Kapitel dient als Ausgangsbasis für die Konzeptualisierung von SoNBO und für das SoNBO-Framework, das ein beliebiges Unternehmen bei der strukturierten Einführung einer SoNBO-Anwendung unterstützt.

3.1 Status quo in der Praxis: Tiefenfallstudie zu einem Ontologie-basierten Integrationsansatz

Im Folgenden wird die Tiefenfallstudie zu der Anwendung vorgestellt, die in dem Unternehmen *Vössing Ingenieurgesellschaft mbH* (im Folgenden referenziert als Vössing) entwickelt wurde und die Informationen aus verschiedenen Quellsystemen in eine Oberfläche integriert. Diese Anwendung wird im Folgenden *SoNBO-Anwendung* genannt. Intuitiv wurden bei der Entwicklung dieser Individualsoftware Aspekte der Ontologie-basierten Informationsintegrationsansätze verwendet, obwohl diese Idee aus dem Semantic Web in dem Unternehmen nicht explizit bekannt war. Außerdem lassen sich bei dieser Software weitere prägende Merkmale (soziale Prinzipien, Berücksichtigung Unternehmens- und Nutzer-Sicht) finden. Die Individualsoftware mit diesen Merkmalen wird in den nächsten Abschnitten beschrieben.

3.1.1 Hintergründe der Tiefenfallstudie Vössing

Die Idee des „Social Network of Business Objects“ und die selbstentwickelte Anwendung, die diese Idee softwaretechnisch umsetzt, wurden vom IT-Leiter Bernd Gewehr (Kürzel: BGE) im April 2016 im Rahmen eines IndustryConnect-Workshops (Schubert & Williams, 2020; Williams & Schubert, 2017) vorgestellt (vgl. Abschnitt 2.2.2). Im Oktober 2016 wurde dazu ein qualitatives Interview (ca. 3 Stunden) mit dem Entwicklungsteam geführt, das anschließend transkribiert wurde. Die organisatorischen Daten zur Erhebung des Interviews sind im Anhang aufgeführt (A.1). Die daraufhin entstandene eXperience-Fallstudie wurde als Arbeitsbericht dokumentiert (Gebel-Sauer & Schubert, 2016). Die SoNBO-Anwendung wird in dem Unternehmen „Self-Service“ genannt und erlaubt den Zugriff auf die Informationen in den transaktionalen Systemen (selbstentwickelte ERP-/ CRM-Systeme) aus einem Kollaborationssystem (HCL Connections) heraus. Die Integrationslösung wurde u.a. mit dem Ziel geschaffen, die Akzeptanz und Nutzung (engl. Adoption) der Kollaborationsplattform zu fördern. Beispielsweise erfolgt die Arbeitszeiterfassung heute über diesen Self-Service. Somit ist es für die Mitarbeiter mittlerweile notwendig, HCL Connections zu nutzen, um ihre tägliche Arbeit zu erledigen. Ein zweiter Aspekt bei der Entwicklung der SoNBO-Anwendung war der Gedanke, die Informationen in den relationalen Datenbanken als „social“ zu betrachten. Damit sollte ein Grundkonzept von HCL Connections, einer

Enterprise Social Software (vgl. Abschnitt 3.2.1.7), auf transaktionale Systeme übertragen werden. Es entstand die Idee, ein Geschäftsobjekt (auch: Business Object) wie einen Knoten in einem Netzwerk zu betrachten und dieselben Attribute und Aktivitäten wie eine Person in einem Netzwerk auf diese Knoten zu übertragen. Ein Zitat von Bernd Gewehr aus dem Interview verdeutlicht dies:

„(...) diese Objekte sind alle (...) da und sie sind eng miteinander vermascht, stark vernetzt (...) und haben normalerweise diese Ausprägungen: Es gibt (...) Stammdaten zu den Objekten – (...) ein Titel, eine Identitätsnummer (...). Es gibt weitere Attribute (...): Wie es beschaffen ist, wo es beschafft wurde usw. Und es gibt (...) Kennzahlen, wenn wir bei dem Asset bleiben: Es hat 3.000,00 Euro gekostet und ist noch 36 Monate im Leasing (...). Und (...) für ein Objekt in einem *social network* ist das wenig. (...) wenn wir (...) es schaffen, zu abstrahieren: Was haben Teilnehmer in social networks darüber hinaus? Dann haben sie (...) ein Profil. Ich kann mir das anschauen: Wer bist du? Ich kann mir genauere Einblicke verschaffen. Ich weiß über die Beziehungen - das Netzwerk dieses Objektes – und (...) [es] ist an dem Objekt erkennbar, was damit alles passiert ist. Es gibt einen *activity stream*. (...) diese eng vermaschten Objekte werte ich auf, indem ich ihre Ereignisse im activity stream sammle - „Ich bin erzeugt worden.“, „Ich bin bearbeitet worden.“, „Ich bin vernichtet worden.“, „Ich bin bezahlt worden.“, „Ich bin gekauft worden.“ – und [wenn ich] innerhalb ihrer Beziehungen das Objekt (...) darstelle, dann bekomme ich ein (...) *social network of business objects*.“

Dieses Prinzip sollte dazu dienen, den Mitarbeitenden einen intuitiveren Zugang auf die Informationen zu ermöglichen. Dies wird im folgenden Zitat von Bernd Gewehr deutlich:

„(...) wenn man (...) unsere Beschäftigten betrachtet, dann sind die meisten davon nur zeitweilig an Geschäftsprozessen beteiligt, die in Systemen stattfinden. Die andere Zeit (...) telefonieren sie, produzieren sie, kommunizieren sie (...), sind in Meetings (...). Und dann müssen sie ein Angebot erstellen, eine Freigabe machen, eine Rechnung zur Zahlung freigeben, ein Beschaffungsantrag usw. (...) Die [meisten] machen das nur einmal im Monat oder einmal im Quartal - diese Transaktion. Also sollen diese Transaktionen möglichst leicht (...) in diesen neuen Gedanken von vernetzten Objekten angeboten werden, so dass ich gar nicht mehr in die Tiefe der Kernapplikation abtauchen muss und trotzdem eine Geschäftsprozessbeteiligung vollständig ausüben kann. Zum Glück sind wir ja in der „era of APIs“. Also man kann eine ganze Menge heute von außen triggern. Es gibt *REST-APIs* und tolle Sachen darüber hinaus, um Sachen zu feuern, auch von Oberflächen, die (...) unten im Core-System passieren.“

Es geht also darum, „einen *digital workplace* [zu] gestalten, der es einfacher und besser möglich macht, den Tagesjob zu erfüllen. Das ist die eigentliche Zielsetzung.“ (Bernd Gewehr, aus dem transkribierten Interview). Dies implizierte in der Umsetzung, dass Informationen integriert werden müssen.

Das IT-Team von Vössing besteht aus 14 Mitarbeitern und stellt die technische Infrastruktur für die 500 Mitarbeiter des Ingenieurbüros bereit. Vössing betreut deutschland- und europaweit Großprojekte, wodurch zahlreiche Dokumente (z. B. Berichte oder Pläne) anfallen. In dieser Branche spielt die Angebotserstellung eine zentrale Rolle, wofür seit Ende der 1990er Jahre eine Datenbank mit fertiggestellten Projekten als Referenz angelegt und gepflegt wird. Auf diese ca. 15.000 Projekte kann mit einer Suche (Facetten- und Schlagwortsuche) zugegriffen und ein geeignetes Referenzprojekt für das

Angebot herausgesucht werden. Diese Datenbank wird sukzessive erweitert und angepasst. Es handelt sich dabei um eine Eigenentwicklung, die mit HCL Domino umgesetzt wurde. Neben der Unterstützung der Primärprozesse werden auch die sekundären Geschäftsprozesse wie z. B. Buchhaltung informationstechnologisch unterstützt.

3.1.2 Beschreibung der SoNBO-Anwendung

Im Folgenden wird die SoNBO-Anwendung vorgestellt, indem ausgehend von dem Benutzerfrontend das Administrationsfrontend und schließlich die Anwendungssicht beschrieben wird.

3.1.2.1 Benutzerfrontend

Das Benutzerfrontend der SoNBO-Anwendung ist in dem Kollaborationssystem HCL Connections integriert. Abb. 3.1 zeigt einen Ausschnitt von HCL Connections und wie die SoNBO-Anwendung (bei Vössing „Self-Service“ genannt) zusätzlich zu den HCL Connections-Modulen *Aktivitäten*, *Blogs*, *Lesezeichen*, *Dateien*, *Foren* und *Wikis* integriert ist. In Abb. 3.2 und Abb. 3.3 sind zwei Screenshots dargestellt, auf welchen ersichtlich wird, wie die SoNBO-Anwendung als Webseite integriert ist: Die obere Leiste mit *Portal*, *Profile*, *Communitys*, *Anwendungen* etc. sind native HCL Connections-Funktionalitäten und der Bereich darunter gehört zur SoNBO-Anwendung (XPage) und ist in das Kollaborationssystem integriert.

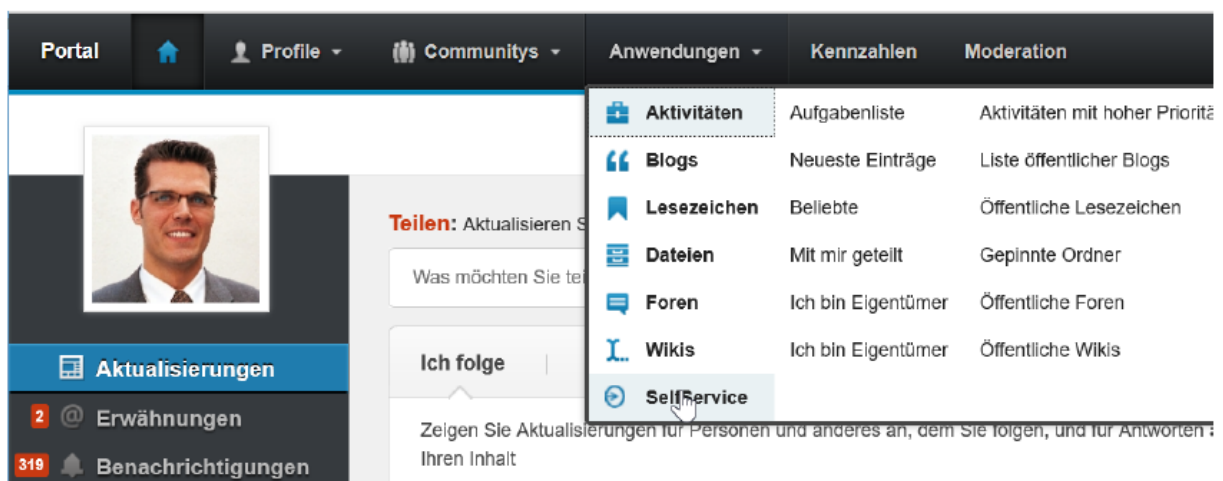


Abb. 3.1: Integration der SoNBO-Anwendung („Self-Service“) in IBM Connections (Stand: 2016)

Für die Vorstellung des Benutzerfrontends wird im Folgenden auf die SoNBO-Anwendung fokussiert. Diese zeigt in Abb. 3.2 den Knoten (auch: Business Object) „Bernd Gewehr“. Der Name des Knotens ist im oberen Bereich angezeigt und darunter befinden sich weitere Informationen (z. B. Leiter IT). In dem Bereich darunter befinden sich die Kategorien, zu denen es für Bernd Gewehr weitere Knoten gibt, die mit diesem verknüpft sind. In diesem Screenshot existieren als Kategorien *aktuell* (auch: *Neuigkeiten*),

Personen, Organisation, Projekte, Referenzen, Partner, Inventar, Bedarf. Da *aktuell* ausgewählt ist, erscheinen in der Leiste darunter Subkategorien, die zu den Neuigkeiten gehören: *Person, Projekt, Referenz, Inventar, Bedarf*. Der Bereich darunter zeigt die Knoten (auch: Business Objects) an, die zu der Kategorie *aktuell* gehören, und diese könnten weiter nach den aufgeführten Subkategorien gefiltert werden. In diesem Screenshot sind 2297 (= 53+1731+245+74+194) Knoten mit Bernd Gewehr verknüpft. Beispielsweise wird eine Neuigkeit zur Person *Dietmar*⁷ (im Screenshot aus Gründen des Datenschutzes entfernt) angezeigt, der demnächst als neuer Mitarbeiter bei Vössing anfangen wird. Dies ist für Bernd Gewehr in seiner Rolle als IT-Leiter relevant, da neue Mitarbeiter Geräte (z. B. Laptop, Handy) erhalten und Zugänge eingerichtet werden müssen. Im linken Bereich können unter *Kennzahlen* und *Filter* noch weitere Filterungen der Knoten im Ergebnisbereich vorgenommen werden.



Abb. 3.2: Benutzerfrontend der SoNBO-Anwendung am Beispiel des Knotens *Bernd Gewehr* (Stand: 2016)

Wie auf dem Screenshot zu sehen ist, ist der Knoten Bernd Gewehr in der SoNBO-Anwendung ausgewählt und in dem Kollaborationssystem ist auch Bernd Gewehr eingeloggt. Dadurch ist der Startknoten in der SoNBO-Anwendung auch Bernd Gewehr selbst. Dies gilt analog für jeden anderen Nutzer.

Das Prinzip bei dieser Anwendung ist, dass ausgehend von einem Startknoten auf einen weiteren Knoten navigiert werden kann, indem beispielsweise auf den Hyperlink IBV unter *Bernd Gewehr – Leiter ZD IT* geklickt wird. In diesem Beispiel ist der nächste Knoten *IBV* (vgl. Abb. 3.3), was für Ingenieurbüro Vössing steht. Daraufhin erscheinen die Informationen zu dem Knoten *IBV*, der dieselben Bereiche wie

⁷ Name geändert

der vorherige Knoten besitzt. Allerdings werden im Ergebnisbereich nur die Knoten angezeigt, zu denen der Knoten IBV Verknüpfungen (also Kanten) besitzt. In diesem Beispiel existieren in der Kategorie *aktuell* 2040 Knoten. Die Informationen werden aus der Sicht des jeweiligen Knotens angezeigt. Der Nutzer sieht allerdings nur die Informationen, zu denen dieser auch berechtigt ist. Navigiert also beispielsweise ein Sachbearbeiter auf den Knoten *Bernd Gewehr*, sieht dieser aufgrund seiner Berechtigungen wesentlich weniger Knoten als Bernd Gewehr selbst (vgl. Abb. 3.2), da die Zugriffsrechte auf die Datenquellen bei der SoNBO-Anwendung vom Quellsystem vorgegeben werden. Bernd Gewehr als Leiter der IT besitzt bedingt durch seine Position Verknüpfungen zu allen Informationsobjekten im Netzwerk. Somit existiert also ein für jeden Nutzer einzigartiges Netzwerk aus Business Objects, durch welches der Nutzer navigiert, und dieses wird Social Network of Business Objects genannt.

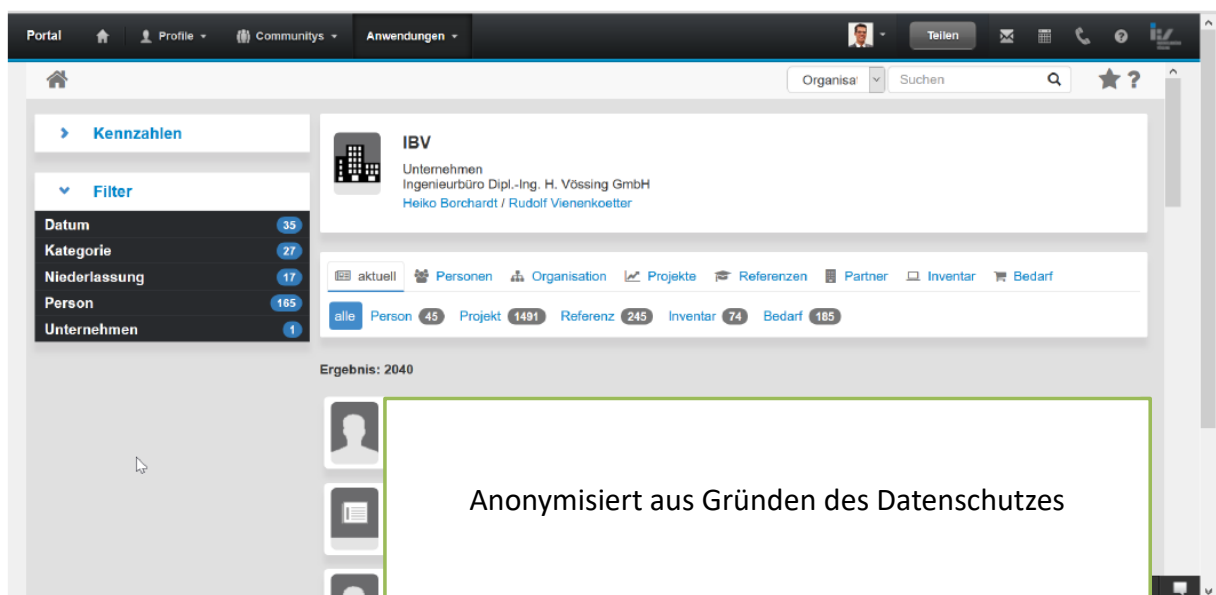


Abb. 3.3: Benutzerfrontend der SoNBO-Anwendung am Beispiel des Knotens *IBV* (Stand: 2016)

Das Enterprise Collaboration System *HCL Connections* besitzt einen Activity Stream, in den Neuigkeiten für den jeweils eingeloggtten Nutzer gepostet werden. Die Informationsobjekte der SoNBO-Anwendung posten ebenfalls Neuigkeiten (z. B. „Rechnung XY wurde bezahlt“) in den Activity Stream der jeweils zugeordneten Mitarbeiter und informieren diese über Neuigkeiten, was zu einer verbesserten Workspace Awareness führt (Gutwin & Greenberg, 1996). Die Neuigkeiten werden erzeugt auf Grund der Tatsache, dass jede Transaktion im Quellsystem ein Ereignis erzeugt. Dieses wird dann in den Activity Stream des Mitarbeiters visualisiert, sofern das Ereignis für den Mitarbeiter von Relevanz ist. Auf diese Weise werden die transaktionalen Ereignisse aus den Quellsystemen in die Wahrnehmung des Mitarbeiters im Frontend gebracht, sodass eine Beteiligung daran über das Frontend der SoNBO-Anwendung möglich wird. Bislang wurde der Mitarbeiter in einer E-Mail darüber informiert, in welchem System eine Beteiligung erforderlich ist, sodass der Mitarbeiter sich daraufhin in dem System anmelden, zu dem Vorgang navigieren und schließlich die Aufgabe ausführen konnte (z. B. Bedarfsan-

forderung genehmigen). Durch die SoNBO-Anwendung wird dort auf die Beteiligung aufmerksam gemacht und in der SoNBO-Anwendung direkt kann die Beteiligung auch ausgeführt werden. Außerdem können über die SoNBO-Anwendung auch diese Neuigkeiten kommentiert werden. Wenn ein Sachbearbeiter beispielsweise eine Bedarfsanforderung erstellt, wird eine solche Neuigkeit als Post erzeugt und der Sachbearbeiter kann einen Verantwortlichen zur Genehmigung auffordern (über die Kommentarfunktion). Dieser Verantwortliche erhält diese Neuigkeit als Post in seinem Kollaborationssystem, woraufhin dieser ebenfalls über die Kommentarfunktion kurze Rückfragen stellen und die Bedarfsanforderung auch genehmigen kann.

Neben den geschäftskritischen Informationen zu Bedarfsanforderungen, Rechnungen, Inventar, Mitarbeiter oder Projektinformationen von laufenden und abgeschlossenen Projekten, sind durch die SoNBO-Anwendung weitere geschäftskritische Vorgänge möglich. Mithilfe der SoNBO-Anwendung müssen die Arbeitszeiterfassung erfasst und Urlaubsanträge gestellt werden. Außerdem kann darüber auf die umfassende Projektdatenbank zugegriffen werden, sodass durch verschiedene Filterfunktionen das für ein Angebot relevante Projekt herausgesucht und verwendet werden kann.

Ein Beispiel für Kennzahlen bieten die Screenshots in Abb. 3.4 und Abb. 3.5 (beide anonymisiert aus Gründen des Datenschutzes). Beispielsweise kann eine Fragestellung in einem Unternehmen nach der Arbeitsauslastung der Abteilungsleiter sein. Dazu werden als Kategorie *Person* und als Subkategorie *Mitarbeiter* ausgewählt. Die Abteilungsleiter werden im Filterbereich unter *Personalgruppe* über eine Facettensuche ausgewählt. Insgesamt gibt es 31 Knoten also 31 Abteilungsleiter. Nun kann die *Produktivität* (vgl. Abb. 3.4) oder der *Verlauf* (vgl. Abb. 3.5) betrachtet werden: Für die Produktivität werden die Über- und Unterstunden in Prozent angezeigt (vgl. Abb. 3.4). Für den Verlauf werden die absoluten Werte für die letzten Monate angezeigt (vgl. Abb. 3.5).

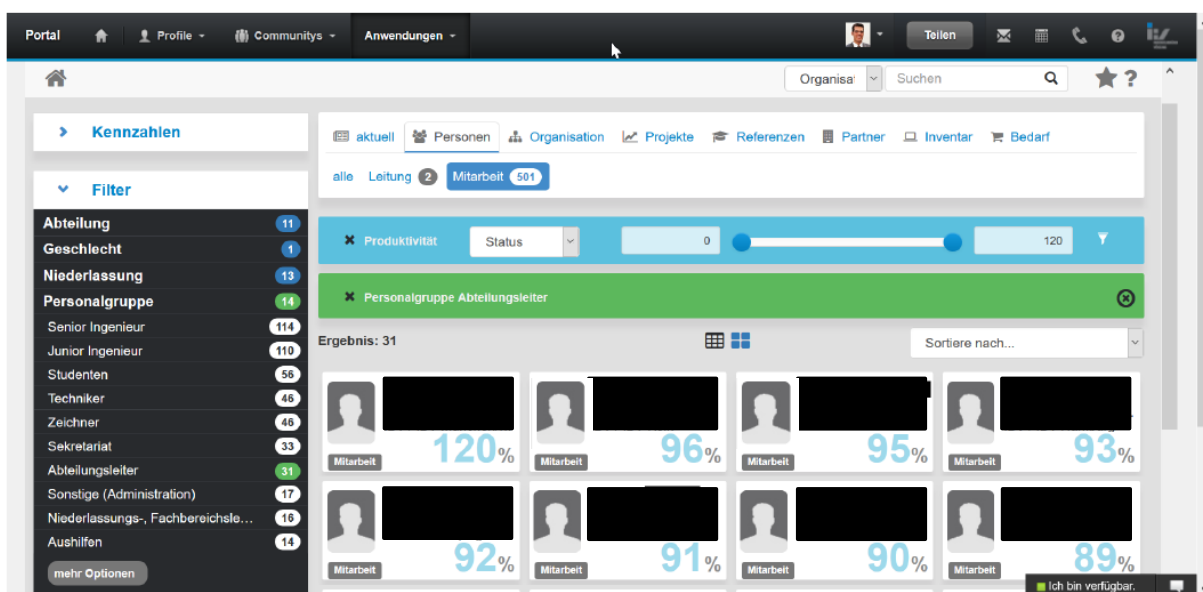


Abb. 3.4: Beispiel für Filter: Über- und Unterstunden der Abteilungsleiter (Stand: 2016)

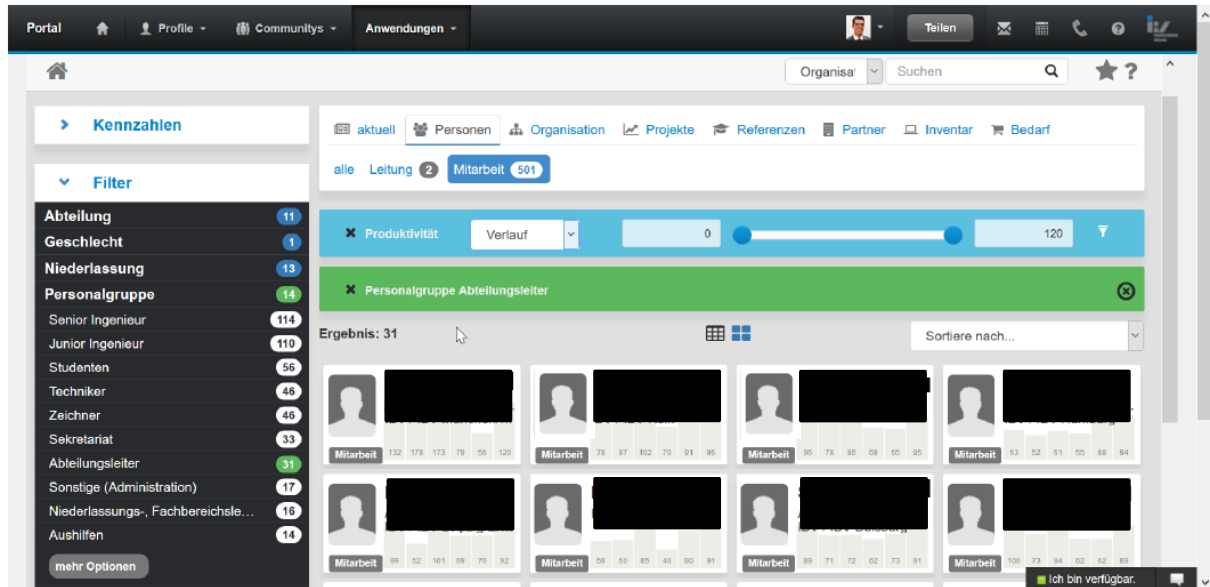


Abb. 3.5: Beispiel für Filter: Verteilung der Arbeitsstunden der Abteilungsleiter (Stand: 2016)

Hervorzuheben ist, dass der Nutzer den Zugriff auf die Informationen durch Navigation (query by navigation, vgl. Abschnitt 3.2.1.4) erhält und nicht durch die Eingabe einer selbst formulierten Suchanfrage (query by construction, vgl. Abschnitt 3.2.1.4).

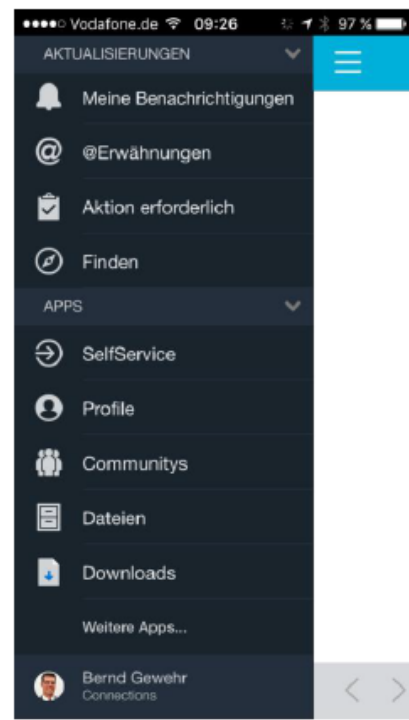


Abb. 3.6: Mobilfähige SoNBO-Anwendung (Stand: 2016)

Vössing entwickelte die SoNBO-Anwendung auch als mobile Applikation. In Abb. 3.6 ist ein Screenshot abgebildet, indem die SoNBO-Anwendung (Self-Service) in die HCL-Connections-Anwendung eingebettet ist.

3.1.2.2 Administrationsfrontend

Damit der Nutzer durch das Social Network of Business Objects navigieren kann, muss ein Schema in der SoNBO-Anwendung hinterlegt werden, welches dieses instanziierte Netzwerk beschreibt. Diese Einrichtung des Schemas bestehend aus Knotentypen und Kanten bzw. die Erweiterung um einen weiteren Knotentyp (Kategorie) und Kanten erfolgt im Administrationsfrontend. Dieses muss nicht programmiert werden, sondern wird konfiguriert. Diese Konfiguration kann auch nachträglich vorgenommen werden, was bei Vössing auch gemacht wurde. Bernd Gewehr erklärt das Prinzip folgendermaßen:

„(...) zunächst Mal geht es darum, eine serverseitige Instanz zu schaffen, in der die Daten in ihrer Vernetzung aufbereitet vorliegen. Und das machen wir über intensive Konfigurationsdokumente, mit der wir dieser allgemeinen Softwarekomponente die Beschaffenheit des Graphen erklären. Also sie ist zunächst (...) abstrakt, (...) wir haben nicht programmiert, dass etwas eine Person ist, sondern wir benennen in einem Konfigurationsdokument ein Objekt, das ist das Objekt, das heißt Person. Und die Daten kommen aus dieser Stelle und es hat zusätzliche Attribute A, B, C, D. (...) wir programmieren eine Anwendung für Objekte und parametrisieren alles, was (...) da erscheinen soll durch Konfigurationsdokumente. Das heißt, die Anwendung kennt erstmal nicht ihre Nutzung. (...) wir haben [also] kein zweites *HR system* programmiert, kein zweites *financial system* programmiert, sondern wir haben etwas programmiert, was Objekte abstrakt darstellen kann und Überschriften usw. kommen aus Konfigurationsdokumenten. (...) wenn ich diese Anwendung öffne, ist es erstmal eine ganz normale Notes-Anwendung. Aber sie besteht eben aus Konfigurationsdokumenten. Und diese Konfigurationsdokumente, die machen die eigentliche Würze von dem ganzen Spiel aus.“

In den Konfigurationsdokumenten liegt bei Vössing das Schema, wie auf die Daten zugegriffen wird, sodass ein Social Network of Business Objects entsteht. In einem Konfigurationsdokument liegt beispielsweise die Beschreibung des Zugriffes auf die Quellsysteme für den Typ Mitarbeiter. Ein wichtiges Element ist z. B. eine ID, sodass ein Objekt eindeutig identifiziert werden kann. Beim Mitarbeiter ist dies idealerweise die E-Mail-Adresse. Wenn ein weiterer Typ dazukommt (z. B. Rechnung) wird dafür ein weiteres Konfigurationsdokument erstellt und darin beschrieben, wie auf die Daten zugegriffen werden soll und welche Beziehungen oder Kanten zu dem Konfigurationsdokument Mitarbeiter bestehen. Dadurch ist die SoNBO-Anwendung gegenüber einer Erweiterung flexibel. Insgesamt gibt es über 30 Konfigurationsdokumente.

3.1.2.3 Anwendungssicht

Das Unternehmen setzt vor allem Software und Hardware von HCL (bis 2019: IBM) ein. Für die Kollaboration nutzen sie das Enterprise Collaboration System HCL Connections und für die transaktionalen Systeme wie das ERP-System wird eine Eigenentwicklung eingesetzt. Außerdem wird eine Datenbank

mit abgeschlossenen Projekten gepflegt, die für die Angebotserstellung relevant ist. Insgesamt existieren etwa zehn Applikationen, die als Quellsysteme dienen. Die Anwendungssicht (vgl. Abb. 3.7) nach der eXperience-Fallstudienmethodik unter der Verwendung der Drei-Schichten-Architektur zeigt, wie die Quellsysteme (transaktionale Systeme) mithilfe der SoNBO-Anwendung in das Kollaborationssystem integriert sind: Auf der linken Seite befinden sich die Quellsysteme, mit denen bei Vössing die Informationen verwaltet werden. Es wird beispielsweise ein ERP- und ein Projektmanagement-Tool genutzt, welche als Notes-Applikation entwickelt wurden. Über einen Client werden die Informationen eingegeben und abgerufen. Dieser Client wird durch die SoNBO-Anwendung nicht ersetzt.

Durch SoNBO werden ausgewählte Informationen aus den Quellsystemen entnommen (vgl. Abb. 3.7, mitte), mithilfe der Idee des sozialen Netzwerkes neu dargestellt, und schließlich in HCL Connections integriert (vgl. Abb. 3.7, rechts). Wie bereits erwähnt ist der SoNBO-Explorer bei Vössing in HCL Connections integriert. Die Integration erfolgt nicht in einer Community, sondern in der Anwendung selbst (vgl. Abb. 3.1).

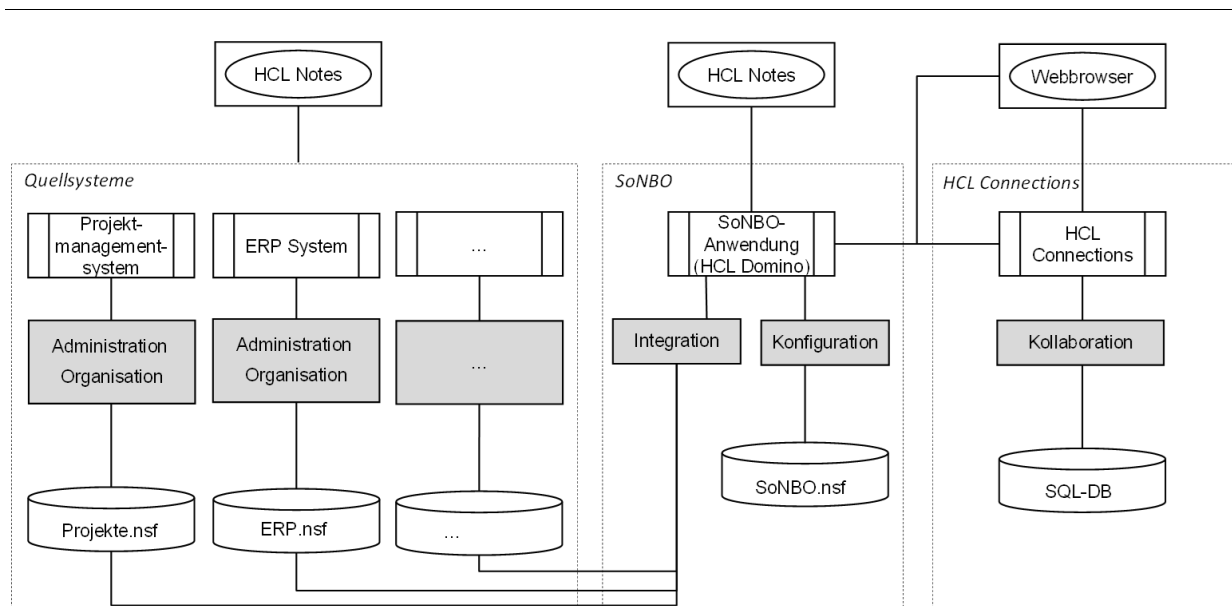


Abb. 3.7: Anwendungssicht der SoNBO-Anwendung

Für die Entwicklung der SoNBO-Anwendung wurde die Entwicklungsumgebung HCL Notes/Domino verwendet. Vössing erteilte Dritten keinen Zugriff auf den Code, stellte dem Team der Universität Koblenz aber einen Auszug auf die Entwicklungsumgebung mit den erstellten Java Klassen, Custom Controls usw. bereit (vgl. Abb. 3.8). Außerdem wurde das Backend vorgeführt. Der SoNBO-Explorer selbst ist eine XPage, wobei es sich um XML-Code handelt, der von einem Domino-Server (oder Notes-X-Client) interpretiert wird. Anschließend wird er in einem Web-Browser oder Notes-Client wiedergegeben. Die Custom Controls gehören zu den XPages, da damit die einzelnen Elemente auf der XPage spezifiziert werden. Für die SoNBO-Anwendung wurden drei XPages entwickelt. Die bei Vössing entwickelte SoNBO-Anwendung wird in einem Web-Browser aufgerufen und ist in HCL Connections integriert.

Bei SoNBO werden die Business Objects (z. B. Mitarbeiter *Bernd Gewehr* und Unternehmen *Vössing*) als Knoten in einem sozialen Netzwerk verstanden, der zur Laufzeit virtualisiert entsteht. Der Aufbau des Netzwerkes wird durch die Spezifizierung der Kategorien (z. B. Mitarbeiter und Unternehmen) in dem Administrationsfrontend (vgl. Abschnitt 3.1.2.2) definiert. Wenn der Nutzer also die XPage und somit z. B. den Knoten *Bernd Gewehr* aufruft, wird mithilfe der Java Klassen die Konfiguration zur Kategorie *Mitarbeiter* aufgerufen. Damit werden die Daten aus den Quellsystemen abgefragt und zum Knoten *Bernd Gewehr* zusammengesetzt. Wenn der Nutzer auf den nächsten Knoten navigiert, wiederholt sich der Prozess. Das soziale Netzwerk wird nicht zwischengespeichert und die Daten werden bei jedem Aufrufen eines neuen Knotens oder dem neuen Laden eines existierenden Knotens aus den Datenbanken geladen. Es handelt sich dabei also um einen Live-Zugriff, was auch Virtualisierung genannt wird (vgl. Abschnitt 3.2.2.3.2 und 3.2.2.6).

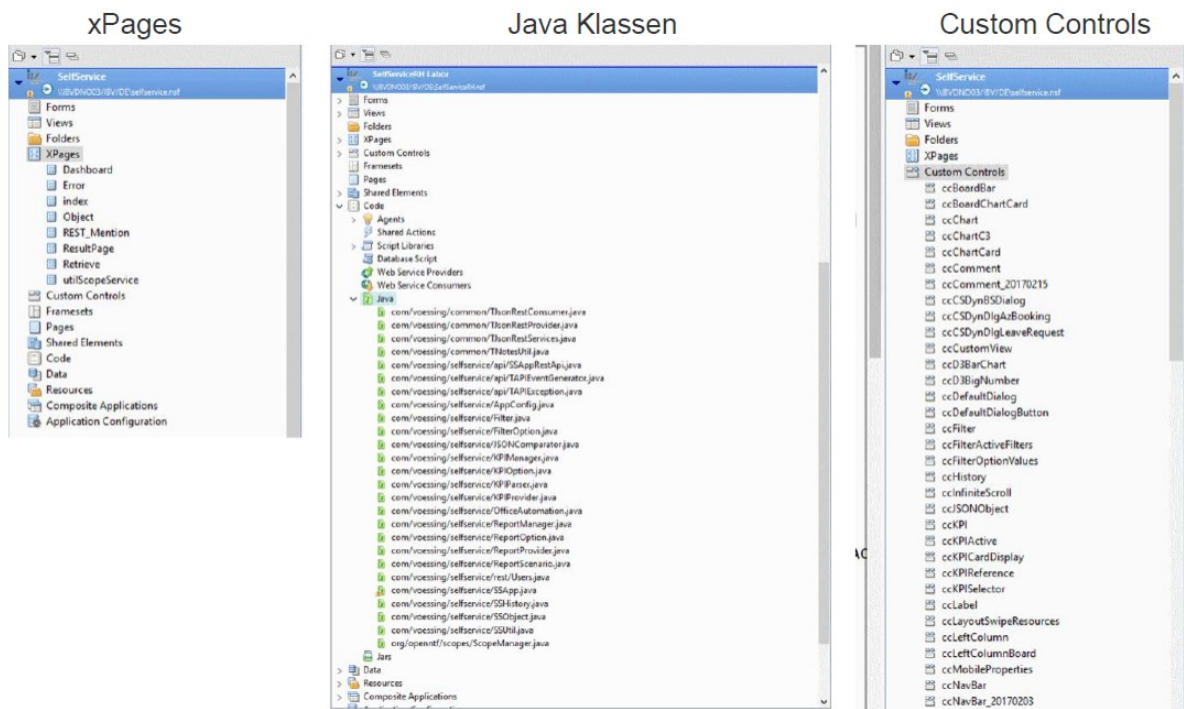


Abb. 3.8: Ausschnitt aus der Entwicklungsumgebung der SoNBO-Anwendung (Stand: 2016)

Da aufgrund der Eigenentwicklungen keine API in den Quellsystemen vorhanden ist, wurden in diesen JSON-Strings (für das Erhalten eines einheitlichen Datenformates zum Datenaustausch zwischen Anwendungen) erstellt, sodass auf ein einheitliches Datenformat zugegriffen werden kann. Außerdem wurde ein SSO (Single Sign-on) mit der Übergabe der Berechtigungen eingerichtet, sodass der Nutzer in der SoNBO-Anwendung auch nur das sehen kann, zu dem dieser berechtigt ist. Durch die Anmeldung des Nutzers in HCL Connections erfolgt also automatisch die Anmeldung an den Quellsystemen. Dies wird bei Vössing umgesetzt, indem die Passwörter verschlüsselt abgespeichert werden.

In Anlehnung an die konzeptionelle Idee des Self-Service von Vössing wurde im Rahmen von Abschlussarbeiten (Götz, 2018; Riedle, 2018) an der Universität Koblenz eine SoNBO-Anwendung (der sogenannte „SoNBO-Explorer“) entwickelt und die Informationen aus zwei Quellsystemen (CRM- und ERP-System) integriert. Entwicklung und Evaluation des SoNBO-Explorers wurden dokumentiert in Götz & Gebel-Sauer (2018) und Gebel-Sauer & Schubert (2019b). Die Erfahrungen (insbesondere die Konfiguration) aus diesen Projekten fließen auch in die Ergebnisse dieser Dissertation (Konzeptualisierung von SoNBO, SoNBO-Framework) mit ein.

3.1.3 Zusammenfassung

Im ersten Teil dieses Kapitels wurde die von Vössing unter der Leitung von Bernd Gewehr entwickelte SoNBO-Anwendung vorgestellt. Diese war der Auslöser des SoNBO-Forschungsprojektes und somit auch dieser Dissertation, da ein Unternehmen mithilfe einer Anwendung die Herausforderungen eines gegenwärtigen Problems (flexible Integration von Informationen) erfolgreich adressierte. Für die Übertragbarkeit dieses Konzeptes auf weitere Unternehmen wird im Rahmen der Dissertation eine Konzeptualisierung (vgl. Kapitel 4) und ein Framework zur Übertragbarkeit/Einführung von SoNBO (vgl. Kapitel 5) erstellt, wofür die Beschreibung dieser existierenden SoNBO-Anwendung und die Erkenntnisse aus der Konfiguration des daraus resultierenden SoNBO-Explorers grundlegend ist.

3.2 Status quo in der Wissenschaft: Literaturanalyse zu Ontologie-basierten Integrationsansätzen

In diesem Abschnitt werden die Ergebnisse aus der Literaturanalyse zu Ontologie-basierten Integrationsansätzen kategorisiert vorgestellt. Die Erkenntnisse aus der Literaturrecherche dienen zum einen zur wissenschaftlichen Einordnung der Dissertation und zum anderen als Ausgangsbasis für die Konzeptualisierung von SoNBO (vgl. Kapitel 4) und der Entwicklung des SoNBO-Frameworks (vgl. Kapitel 5). Die Literaturanalyse wird anhand von wissenschaftlichen Methoden (vgl. Abschnitt 2.2.2) durchgeführt.

3.2.1 Grundlagen

Für die Ergebnisse der Literaturanalyse und für die Konzeptualisierung sowie Entwicklung des Frameworks sind bestimmte Grundlagen relevant, die im Folgenden aufgeführt werden. Für diese wurden im Rahmen der durchgeführten Literaturrecherche relevante Beiträge identifiziert, die für die Beschreibung der Grundlagen herangezogen werden.

3.2.1.1 Semantische Technologien

Ontologie-basierte Integrationsansätze aus dem Semantic Web verwenden semantische Technologien, die durch das W3C standardisiert wurden (Sabou, 2016). Da der Anteil der Informatik an den Semantic-Web-Technologien sehr hoch ist und diese Konzepte darin auch ihre Stärke haben (indem beispielsweise Reasoning bis hin zur künstlichen Intelligenz ermöglicht werden), greifen auch die meisten

Konzepte auf die Standards zurück. Diese Standards wurden um die Jahrtausendwende geschaffen, wodurch diese in den Konzepten auch erst seit dieser Zeit verwendet werden, auch wenn die Ideen zu Ontologie-basierten Integrationsansätzen bereits vorher existierten. Diese Standardisierung war also ein wichtiger Meilenstein.

3.2.1.1.1 Ontologiebeschreibungssprachen

Für die ersten Ontologien in der Informatik wurden Frames und First-Order Logic verwendet. Die Entwicklung weiterer Sprachen zur Wissenspräsentation erfolgte sukzessive und sie basierten auf Beschreibungslogiken (Description Logics) (Baader, Horrocks, & Sattler, 2004) wie z. B. OIL und DAML+OIL (DARPA Agent Markup Language + Ontology Inference Layer). Es setzte sich schließlich OWL (Web Ontology Language) durch (McGuinness & van Harmelen, 2004; Villazon-Terrazas, Garcia-Santa, Ren, Faraotti, et al., 2017; W3C, 2012), welche ein Standard vom W3C darstellt. Für OWL gibt es drei Ausführungen in verschiedenen Ausdruckstärken: OWL Lite, OWL DL (Basis: Description Logics) und OWL Full. Außerdem existiert die erweiterte Version OWL 2 (Villazon-Terrazas, Garcia-Santa, Ren, Faraotti, et al., 2017; W3C, 2012; Weller, 2014). Nach Weller (2014) geht der aktuelle Trend zu den weniger ausdrucksstarken Varianten wie z. B. OWL Lite und falls dies noch zu komplex sein sollte, werden das einfachere RDF(S) (Resource Description Framework (Schema)) (W3C, 2014a) oder SKOS (Simple Knowledge Organisation System) verwendet (Blumauer, 2014c).

3.2.1.1.2 Abfragesprachen

Query Languages (auch: Abfragesprachen) werden eingesetzt, um die Elemente einer Ontologie zu ermitteln, welche bestimmte Kriterien erfüllen. Diese nutzen die Triple-Struktur (Subjekt, Prädikat, Objekt) von Ontologien. Für jede Ontologiebeschreibungssprache existiert eine Abfragesprache. Für RDF-basierte Ontologien wird SPARQL (SPARQL Protocol and RDF Query Language) verwendet und für OWL existiert OWL-QL (Quilitz & Leser, 2008; Villazon-Terrazas, Garcia-Santa, Ren, Faraotti, et al., 2017; Weller, 2014). Unter der Verwendung dieser Sprachen entstehen Reasoner, mit denen zum einen Modellierungsfehler oder logische Inkonsistenzen identifiziert (ursprünglicher Zweck) und zum anderen Information gesucht und sogar Schlussfolgerungen gezogen werden. Durch letzteres entstehen neue Informationen und Wissen, welches nicht direkt in der Ontologie eingegeben wurde (Weller, 2014).

3.2.1.1.3 Mappingsprachen

Um die Daten in den relationalen Datenbanken in RDF oder OWL zu konvertieren gibt es ebenfalls Standards. 2012 veröffentlichte die RDB2RDF W3C Arbeitsgruppe zwei Empfehlungen zur Standardisierung von Sprachen, um relationale Daten und Datenbankschemata den Ontologiebeschreibungssprachen RDF oder OWL zuzuordnen (Villazon-Terrazas, Garcia-Santa, Ren, Faraotti, et al., 2017). *Direct Mapping* kann verwendet werden, wenn Klarheit über den Input (Daten und das Datenbankschema) existiert, sodass der Output (RDF Version der Daten) relativ schnell erstellt werden kann. Der Vorteil ist, dass es einfach ist. Allerdings gibt es wenig Kontrolle über die Konvertierungseinstellungen. Eine weitere Möglichkeit ist R2RML (RDB to RDF Mapping Language). Dabei erfolgt ein Customising des

Mappings und die RDF Daten entstehen nach einem individuellen Design (Rodríguez-Muro & Rezk, 2015; Villazon-Terrazas, Garcia-Santa, Ren, Faraotti, et al., 2017). Für das Ableiten von XML-Sprachen gibt es ebenfalls einen W3C-Empfehlungsstandard. GRDDL (Gleaning Resource Descriptions from Dialects of Languages) ist eine Standardmethode zum Ableiten von Resource Descriptions (in RDF) aus Dialekten von XML-Sprachen (Villazon-Terrazas, Garcia-Santa, Ren, Faraotti, et al., 2017).

3.2.1.2 Ontologie

Die Bedeutung von Ontologien nahm in der Informatik unter anderem im Bereich der Informationsintegration zu. Sie spielen im SoNBO-Ansatz eine entscheidende Rolle und auch in den existierenden Konzepten aus dem Semantic Web zur Integration von Informationen. Da Ontologien in der Dissertation eine zentrale Rolle einnehmen und sie grundlegend für die Konzepte zur Ontologie-basierten Informationsintegration sind, werden hier zunächst der Ursprung von Ontologien und anschließend der Aufbau und existierende Vorgehen zur Erstellung von Ontologien dargelegt. Der Abschnitt zu Ontologien schließt mit ein paar Beispielen.

3.2.1.2.1 Ursprung von Ontologien

Der Begriff Ontologie stammt ursprünglich aus der Philosophie und existiert in dieser Forschungsdisziplin lediglich im Singular. Er bezeichnet die Lehre des Seienden und beschreibt die Natur und Struktur der Realität (Guarino, Oberle, & Staab, 2009). „Ontos“ bedeutet „seiend“ und „logos“ bezeichnet im altgriechischen „Lehre“ (Lima, Amaral, & Molinaro, 2010). Damit beschäftigte sich schon Aristoteles im Rahmen seiner Metaphysik (Smith & Welty, 2001). In der Philosophie versucht die Ontologie die Strukturen der Wirklichkeit bzw. der Welt zu beschreiben, indem alle existierenden Dinge erfasst werden. In der Informatik soll eine Ontologie ebenfalls die Strukturen beschreiben. Allerdings bezieht sich dort eine Ontologie lediglich auf einen Ausschnitt der Realität, der Domäne genannt wird. Daher existiert der Begriff auch im Plural, da es mehrere Domänen und somit auch mehrere Ontologien gibt (Guarino et al., 2009). Der Unterschied eines Modells (z. B. Flugzeug) zu einer Ontologie besteht darin, dass die Ontologie den Sachverhalt als Ganzes betrachtet (Uschold, 2018). Bei einem Flugzeug würde es sich also nicht nur um die Beschreibung des Modells eines Miniatur-Flugzeuges handeln, sondern auch um weitere Eigenschaften (z. B. Fähigkeit zu fliegen, Zusammensetzung Personal etc.).

Ontologien werden in der Informatik seit Beginn der 1990er Jahre eingesetzt als eine Methode zur Wissensorganisation. Mit der Arbeit an einem Semantic Web erhielten Ontologien seit Anfang 2000 einen besonderen Stellenwert in der Informatik (Weller, 2014). In der Semantic-Web-Forschung wird an den technischen Grundlagen für Ontologien gearbeitet und es werden Ontologiebeschreibungssprachen, dazugehörige Ontologie-Editoren, Abfragesprachen und Reasoner entwickelt (vgl. Abschnitt 3.2.1.1). Ursprünglich bestand die Aufgabe von Reasonern darin, Unstimmigkeiten (Modellierungsfehler, logische Inkonsistenzen) in der Ontologie zu finden. Mittlerweile werden Reasoner eingesetzt, um automatische Schlussfolgerungen auf der Basis der Informationen in der Ontologie ziehen. Auf diese

Weise werden Informationen bereitgestellt, die nicht explizit in die Ontologie eingegeben wurde (Weller, 2014). Der Vorgang, der mit einem Reasoner durchgeführt wird, nennt sich Reasoning.

Das Ziel des Semantic Web besteht darin, dass Computer in der Lage sein sollen, einzelne Informationen herauszufiltern, weiterzuverarbeiten und mit anderen neu zu kombinieren (Weller, 2014). Dazu werden auch Informationen aus mehreren Quellen gesammelt und sinnvoll zusammengestellt. Die Grundlage dafür sind Ontologien, die dem Computer das notwendige Hintergrundwissen eines Fachgebietes (bestehend aus Fachvokabular und Beziehungen zwischen den einzelnen Termen) zur Verfügung stellen (Weller, 2014). Somit ist der Vorteil von Ontologien, dass sie nicht nur von Menschen, sondern auch von Maschinen lesbar sind. Für die Beschreibung der Ontologien wird z. B. RDF genutzt, welches eine Erweiterung von XML darstellt und dadurch flexibler ist (Guarino et al., 2009; Weller, 2014). Mittlerweile gewinnen Ontologien im Bereich Wissensmanagement, Informationsintegration, kooperative Informationssysteme, Information Retrieval und Electronic Commerce zunehmend an Bedeutung. Ein zentraler Anwendungsbereich ist das eben aufgeführte Semantic Web, bei dem Ontologien eine Schlüsselrolle bei der Schaffung einer gemeinsamen Terminologie zwischen Agenten spielen. Dadurch haben verschiedene Agenten ein gemeinsames Verständnis von Begriffen, die im semantischen Markup verwendet werden (Baader et al., 2004).

3.2.1.2.2 Definition und Aufbau von Ontologien

Die Definition einer Ontologie entwickelte sich seit dem Beginn des Einsatzes in der Informatik immer weiter und existiert in verschiedenen Ausprägungen. Diese Entwicklung wird von Guarino et al. (Guarino et al., 2009) folgendermaßen skizziert: Anfang der 1990er definierte Gruber (1993, p. 199) eine Ontologie als „explicit specification of a conceptualization“. Eine Konzeptualisierung ist demnach eine abstrakte, vereinfachte Sicht auf die Welt (oder bestimmte Domäne), die zu einem bestimmten Zweck repräsentiert werden soll (Dietz, 2006; Guarino et al., 2009). Ende der 1990er Jahre definierte Borst (1997, in: Guarino et al. (2009)) eine Ontologie als „formal specification of a shared conceptualization.“. Diese Definition erfordert zusätzlich, dass die Beschreibung der Konzeptualisierung eine gemeinsame Sichtweise zwischen mehreren Parteien zum Ausdruck bringt. Es soll also einen Konsens geben und keine individuelle Sicht. Außerdem soll die Beschreibung der Konzeptualisierung in einem formalen, maschinenlesbaren Format ausgedrückt werden. Studer (1998) fügte diese beiden Definitionen schließlich zu einer zusammen (Guarino et al., 2009):

“An ontology is a formal, explicit specification of a shared conceptualization.”
Studer et al. (1998) in Guarino et al. (2009)

Bereits vor der Definition von Gruber im Jahr (1993) war der Begriff Ontologie in den verschiedenen Teilgebieten der Informatik (knowledge engineering, conceptual modelling, domain modelling) weit verbreitet. Gruber (1993) unternahm dann den ersten Versuch einer Definition und davon ausgehend wurde die heute etablierte Definition von Studer et al. (1998) entwickelt (Smith & Welty, 2001, p. 5f.). Für die Dissertation wird diese Definition herangezogen (auch wenn diese durch die Forderung nach Formalität von Stuckenschmidt (2011) kritisiert wurde), da diese im Umfeld des Semantic Web von

Wissenschaftlern entstand, die in diesem Bereich eine tragende Rolle spielen. Die relevante Gruppe von Wissenschaftlern wurden mit Hilfe der Literaturrecherche identifiziert (vgl. Abschnitt 3.2.1.7). Das Semantic Web und die in diesem Rahmen entstandenen Ontologie-basierten Integrationskonzepte sind für die Arbeit relevant (vgl. Abschnitt 3.2.1.7). Da die hier verwendete Definition nicht die Zusammensetzung einer Ontologie beschreibt, werden im Folgenden der Aufbau bzw. die Komponenten einer Ontologie dargelegt. Eine Ontologie besteht aus Formeln bzw. Regeln (F). Diese werden mit einer Sprache L (z. B. RDF) formuliert. Die Sprache wiederum verwendet für die Formulierung Vokabeln (V). Die zweite Komponente ist das ontological commitment (C), welches aus einer Datenmenge D (z. B. Menge von Klassen bzw. Konzepten) und Beziehungen (R) auf dieser Datenmenge besteht. Außerdem existiert eine totale Funktion $I: V \rightarrow D \cup R$, die jedes V auf die Datenmenge und die Relationen mappt. Damit ist gemeint, dass mit dem Vokabular V die Datenmenge mit den existierenden Relationen ausgedrückt werden kann, um auf diese Weise die Domäne zu beschreiben (Guarino et al., 2009). Taxonomien sind ein Beispiel für kontrollierte Vokabeln und beschreiben die Beziehung zwischen den Ebenen *Konzept* und *Unterkonzept*. Sie sind somit ein Element von Ontologien. Eine Taxonomie besitzt eine Baumstruktur, wodurch ein Unterkonzept zu genau einem Konzept gehört und ein Konzept wiederum beliebig viele Unterkonzepte enthält (Blumauer, 2014a). Die Anzahl der Ebenen ist beliebig. In Abb. 3.9 ist ein Beispiel für eine Taxonomie abgebildet, die aus drei Konzept-Ebenen und einer Instanz-Ebene besteht: Die oberste Ebene ist der Wurzelknoten der Taxonomie (hier: Apollo program). Auf einer weiteren Ebene (hier: lila Knoten) befinden sich Unterkategorien des Wurzelknotens, die wiederum in weitere Unterkategorien (hier: dunkelgrüne Knoten) unterteilt sind. Auf der untersten Ebene befinden sich die Instanzknoten (hier: hellgrün) (Blumauer, 2014a). Für die Taxonomie ist eine einheitliche Terminologie für die Benennung der Knoten grundlegend.

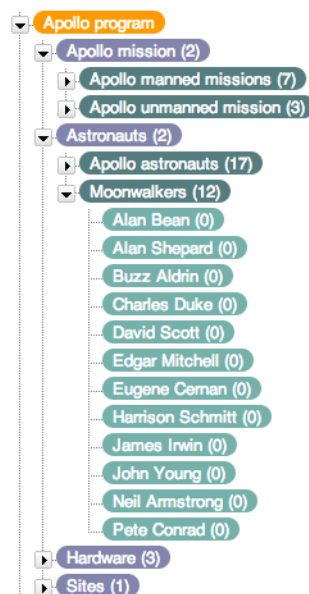


Abb. 3.9: Beispiel einer Taxonomie (Blumauer, 2014a)

Eine Ontologie wird in der Regel als Graph visualisiert, da die Ontologiebeschreibungssprachen vom W3C für das Semantic Web in einer Graphstruktur umgesetzt werden. Daher wird hier konkretisiert, aus welchen Elementen eine Ontologie bei der Visualisierung besteht. Dazu gehören Klassen bzw. Konzepte wie diese im Kontext von Ontologien bezeichnet werden. Mit Hilfe von Konzepten wird in einer Ontologie eine Domäne definiert. Dazu werden die realweltlichen Objekte kategorisiert und diese gebildeten Kategorien sind die Konzepte (Leser & Naumann, 2007; Weller, 2014). Beispielsweise gehören die Kinder (als realweltliche Objekte) Max, Lena, Anni und Fritz zur Kategorie Mensch. Mensch ist dann das Konzept. Konzepte können in Unterkonzepte klassifiziert werden (Weller, 2014). Beispielsweise kann der Mensch in Erwachsener, Jugendlicher und Kind eingeteilt werden. Eine Ontologie soll in der Regel monohierarchisch sein, was bedeutet, dass ein Subkonzept nicht mehrere Oberkonzepte besitzt. In einer Ontologie wird dieser Zusammenhang oft mit „subClassOf“ oder „is_a“ kenntlich gemacht. Außerdem sind Konzepte disjunkt. Es darf also keine Überschneidungen von Objekten geben, die zu verschiedenen Konzepten gehören. Anders formuliert: Jedes Objekt gehört zu genau einem Konzept (Weller, 2014). Für jedes Konzept existieren Attribute bzw. Eigenschaften (engl. Properties), um diese genauer zu spezifizieren. Ein Attribut ist beispielsweise eine Adresse. Diese werden in OWL *Data Properties* genannt und weisen den Konzepten einen Wert zu. Ein weiteres Merkmal ist die Beziehung zwischen zwei Konzepten. In OWL werden diese ebenfalls als Eigenschaft aufgefasst, aber werden von den Data Properties durch die Bezeichnung *Object Properties* unterschieden (Spahn, Kleb, Grimm, & Scheidl, 2008; Staab & Schnurr, 2002; Weller, 2014). Beziehungen bzw. Relationen sind gerichtet, wobei diese in der Regel bidirektional sind. Weitere Bezeichnungen dafür sind wechselseitige oder inverse Relationen (Weller, 2014). Schließlich spielen die Instanzen (Instanzebene), auf die eine Ontologie gemappt wird, eine entscheidende Rolle. Diese sind separat von der Ontologie (Schemaebene) aufzufassen (Weller, 2014). Durch die Beschreibung einer Domäne durch die Ontologie entsteht erst durch das Verknüpfen der Ontologie mit den Instanzen ein Vorteil (Weller, 2014). Schließlich gibt es Axiome bzw. Regeln, die mithilfe der Konzepte, Attribute und Beziehungen die Ontologie spezifizieren (Furrer, 2014; Weller, 2014). Die beschriebenen Elemente werden für die Definitionen des SoNBO-Konzeptes im nächsten Kapitel herangezogen (vgl. Abschnitt 4.1).

Ontologien können anhand ihrer Strukturen und ihrer Verwendung kategorisiert werden. Dazu werden drei Ebenen vorgegeben, in die eine konkrete Ontologie eingeordnet werden kann. Top-Level-Ontologien (auch: foundational ontologies) beinhalten abstrakte und allgemeine Konzepte und sind domänenunabhängig. Diese werden in der Regel nicht für Anwendungen genutzt, sondern dienen als Vorlage bei der Erstellung von weiteren Ontologien. Bekannte Beispiele sind DOLCE und SUMO. Auf der mittleren Ebene befinden sich Domänen- und Aufgabenontologien. In einer Domänenontologie sind Konzepte enthalten, die für eine bestimmte Domäne relevant sind (z. B. bestimmte Krankheiten in der Medizin). Eine Aufgabenontologie beinhaltet Konzepte, die für eine Aufgabe relevant sind (z. B. Behandlung von Symptomen bei bestimmten Krankheiten). Auf der niedrigsten Ebene befinden sich Anwendungsontologien. Diese kombinieren die Domänen- und Aufgabenontologien mit konkreten Konzepten (Grimm, Hitzler, & Abecker, 2007; Guarino, 1997). Je allgemeiner die Konzepte einer Ontologie sind (Top-Level-Ontologie), desto höher ist die Wahrscheinlichkeit für eine Wiederverwendbar-

keit bei der Erstellung einer weiteren Ontologie. Bezogen auf Unternehmen können folgende Beispiele zur Beschreibung dienen: In einer (1) Top-Level-Ontologien beispielsweise werden für Unternehmen allgemeine Informationen festgehalten wie z.B. Person, Dokument, Projekt/Produkt etc. Eine (2) Domänenontologie beschreibt die Struktur einer Domäne wie z.B. Verlags- oder Automobilbranche. Die (3) Anwendungsontologie stellt mithilfe einer Ontologie die Struktur eines konkreten Unternehmens dar. Idealerweise nutzt eine Anwendungsontologie existierende Strukturen (Konzepte, Attribute, Beziehungen, Axiome) der Domänenontologie und die Domänenontologie wiederum bindet Strukturen der Top-Level-Ontologie ein. Eine weitere Möglichkeit der Kategorisierung von Ontologien besteht in dem Grad der Formalität. Dabei werden Ontologien in informal, semi-informal, semi-formal und formal eingeteilt (Gorjatshev, 2019; Uschold & Gruninger, 1996).

3.2.1.2.3 Konstruktion von Ontologien

Die Entwicklung von Ontologien ist in ihrer Komplexität mit der Entwicklung einer komplexen Software vergleichbar. Daher muss der Ontologie-Entwickler nicht nur mit den Ontologieformalismen vertraut sein, sondern auch mit den notwendigen Entwicklungsmethoden, die die Schritte des Prozesses strukturieren (Nagypál, 2007). Es existieren verschiedene Methoden (Gorjatshev, 2019): METHONTOLOGY (Fernández-López, Gómez-Pérez, & Juristo, 1997), Ontology Development 101 (Noy & McGuinness, 2001), UPON (Unified Process for ONtology building) (De Nicola, Missikoff, & Navigli, 2009), UPON lite (De Nicola & Missikoff, 2016) und Methoden zur Erstellung der beiden Ontologien TOVE (TOronto Virtual Enterprise) und EO (Enterprise Ontology) (Gruninger & Fox, 1995; Uschold, 1996) (vgl. Tab. 3.1). Die Ontologie-Erstellung ist ein offener Forschungsbereich (Sequeda & Miranker, 2017) und es ist davon auszugehen, dass die Methoden in den nächsten Jahren weiterentwickelt werden oder neue hinzukommen.

Eine Methode, die auch vom Integrationsansatz OBDA (vgl. Abschnitt 3.2.2.3) aufgegriffen wird, ist die „Skeletal Methodology for Building Ontologies“ von Uschold & Gruninger (1996). Bei dieser Methode muss zunächst der Zweck der Ontologie definiert werden. Anschließend wird die Ontologie gebildet, was in drei Schritten erfolgt: Zuerst werden in der (1) Identifizierungsphase die Schlüsselkonzepte und Beziehungen in der betreffenden Domäne erfasst. Anschließend werden präzise und eindeutige Textdefinitionen für diese Konzepte und Beziehungen erstellt. Außerdem werden Begriffe identifiziert, mit denen sich auf diese Konzepte und Beziehungen bezogen werden kann. Der (2) zweite Schritt umfasst das Coding. Damit ist die explizite Repräsentation der Konzeptualisierung in eine formale Sprache gemeint. Währenddessen wird (3) geprüft, inwiefern existierende Ontologien integriert werden können. Falls möglich, werden diese integriert. Die entstandene Ontologie wird dann evaluiert und für zukünftige Bearbeitungen dokumentiert (Uschold & Gruninger, 1996).

Für die Ontologie-Erstellung gibt es noch keine Standardisierungen wie diese aus der Softwareentwicklung bekannt sind (Nagypál, 2007). Es lässt sich aber festhalten, dass es bei der Ontologie-Erstellung darum geht, in welchem Grad eine existierende Ontologie wiederverwendet werden kann und inwiefern neue Elemente (Konzepte, Attribute, Relationen, Axiome) hinzugefügt werden müssen. Außer-

dem fällt auf, dass es eine konzeptionelle Phase gibt, bei der die Konzepte, Attribute und Beziehungen mit eindeutigen Begriffen festgelegt werden. An diesen Teil schließt sich die Formalisierung der Ontologie an. Diese Eigenschaften werden bei der Entwicklung des SoNBO-Frameworks (vgl. Kapitel 5) wieder aufgegriffen.

3.2.1.2.4 Beispiele für Ontologien (Fokus Unternehmensontologien)

Allgemein lässt sich festhalten, dass der Aufwand zur Erstellung einer Ontologie sehr hoch ist, weshalb die Anzahl bekannter existierender Ontologien überschaubar ist. Folgende Ontologien etablierten sich in der Community: FOAF (Friend Of A Friend) ist eine Ontologie, die zur Personensuche im Semantic Web genutzt wird (Brickley & Miller, 2000). Die Ontologien SUMO (= Suggested Upper Merged Ontology) und DOLCE (= Descriptive Ontology for Linguistic and Cognitive Engineering) sind Top-Level-Ontologien und mit dem Ziel entwickelt worden, formal grundlegend zu sein. In der SUMO-Ontologie werden Konzepte auf hoher Ebene definiert (z. B. Objekt, Prozess, Menge, Beziehung). Außerdem liefert diese Axiome, welche die Eigenschaften dieser Konzepte und Beziehungen zwischen diesen beschreibt. Die DOLCE-Ontologie ist ebenfalls eine formale fundamentale Ontologie, die als grundlegende Ontologie in einem Projekt entwickelt wurde, welches eine große Anzahl europäischer Forschungsgruppen umfasst (Noy, 2004). Da der Fokus in der Dissertation auf Unternehmen liegt, wird hier eine Auswahl von Unternehmensontologien präsentiert.

In einer kürzlich veröffentlichten strukturierten Literaturanalyse (Gorjatschev, 2019) wurden in der Literatur elf Unternehmensontologien identifiziert und beschrieben (vgl. Tab. 3.1).

Tab. 3.1 Übersicht über existierende Unternehmensontologien (Gorjatschev, 2019, p. 28)

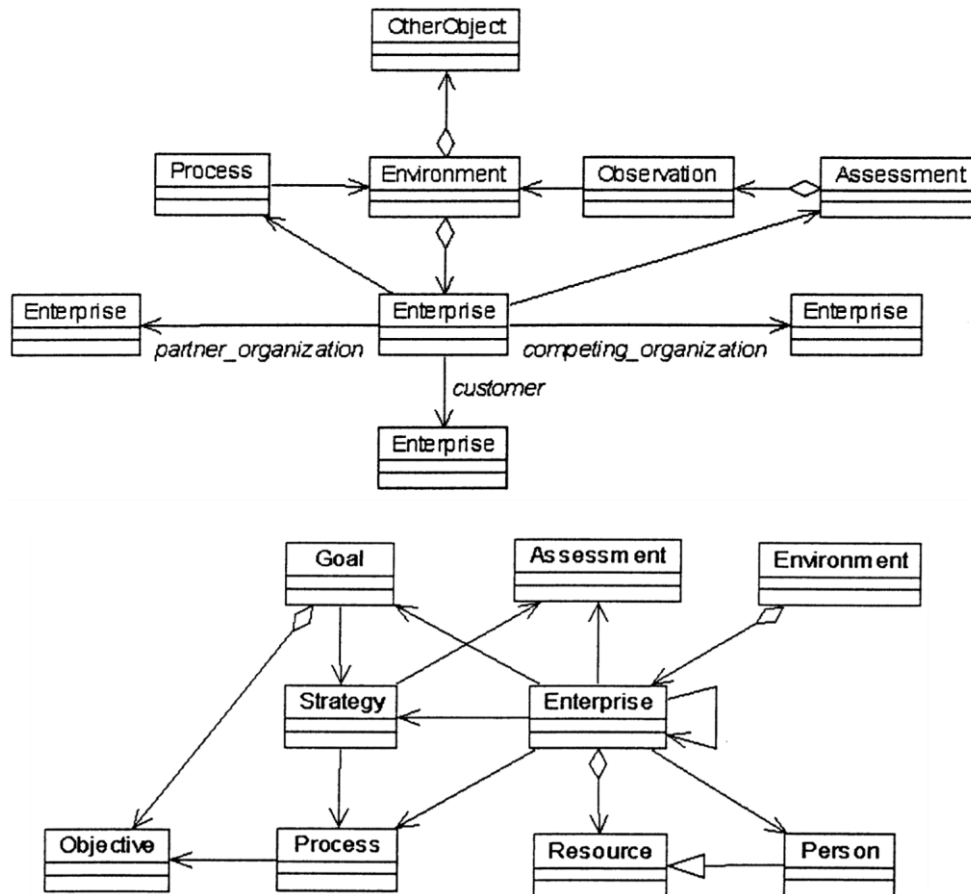
Grau: in der Dissertation beschriebene Ontologien

Name	Jahr	Literatur
Toronto Virtual Enterprise (TOVE)	1992	(Fox, 1992)
The Enterprise Ontology (EO)	1998	(Uschold, King, Moralee, & Zorgios, 1998)
IDEON ontology	2001	(Madni, Lin, & Madni, 2001)
Core Enterprise Ontology (CEO)	2001	(Bertolazzi, Krusich, & Missikoff, 2001)
PSIM ontology	2001	(Pelletier, Goossenaerts, & Szirbik, 2001)
Core Business Ontology (CBO)	2005	(De Nicola, Misceo, & Missikoff, 2005)
Organization ontology	2008	(Sharma & Osei-Bryson, 2008)
GoodRelations	2008	(Hepp, 2008)
gist	2009	(Semantic Arts, 2019)
Organization Ontology (ORG)	2014	(W3C, 2014b)
Business Ontology (BO)	2015	(von Rosing & Laurier, 2015)

Die Zahl *elf* wirkt in diesem Zusammenhang relativ klein und es ist davon auszugehen, dass die Zahl der tatsächlich in der Praxis existierenden Ontologien höher ist. In Tab. 3.1 sind die Unternehmensontologien

tologien, bei denen es sich um Domänenontologien handelt, aufgelistet. Drei davon (IDEON, ORG und BO) werden im Folgenden erläutert und werden zum Teil für die Unternehmensontologie im SoNBO-Framework wiederverwendet (vgl. Abschnitt 5.5.2).

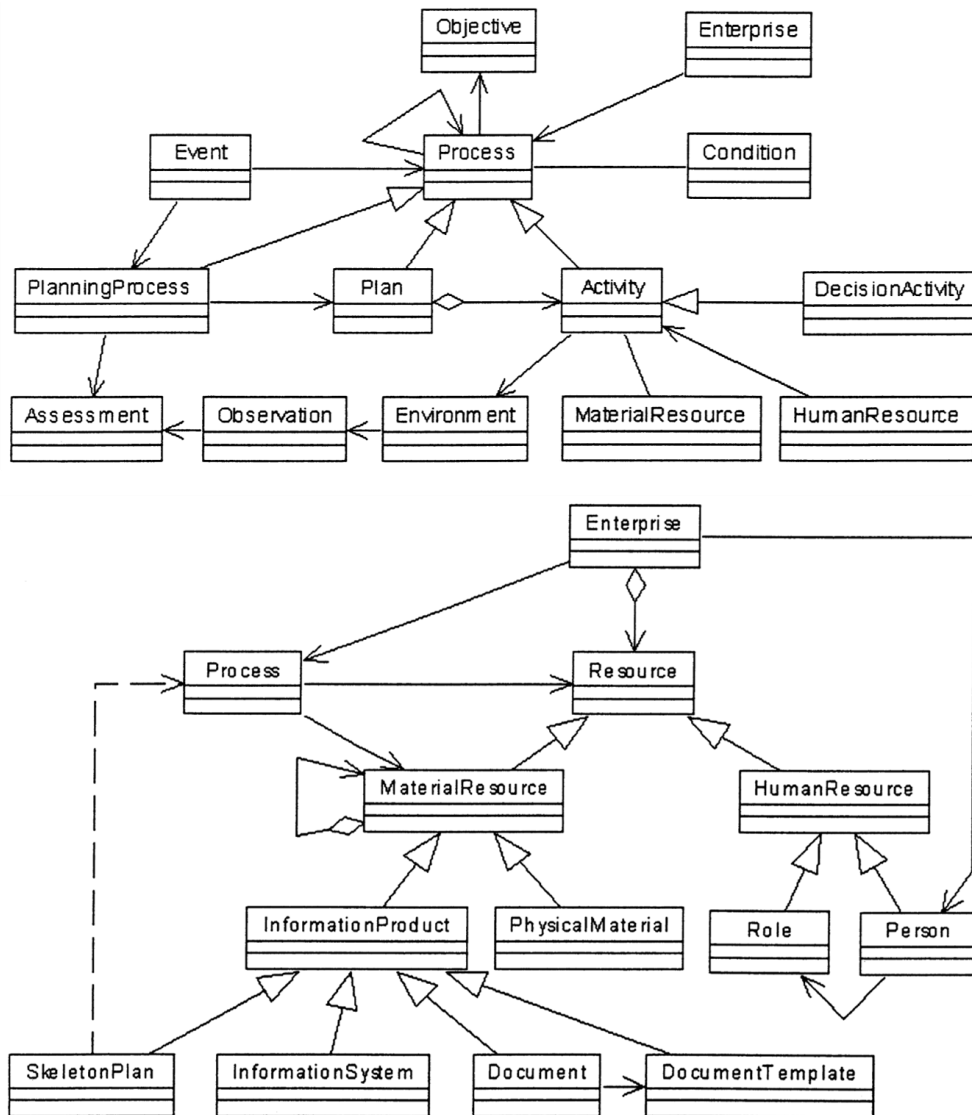
Die **IDEON™** (im Folgenden: IDEON) **Ontologie** ist eine gemeinsame Grundlage für das Entwerfen, Neuerfinden, Verwalten und Steuern von kollaborativen, verteilten Unternehmen (Madni et al., 2001). Diese Unternehmensontologie ist einheitlich und erweiterbar. Sie adressiert die folgenden größten Herausforderungen für Unternehmen: (a) nahtlose Integration von Unternehmensdesign, Management- und Kontrollprozesse sowie Unterstützung von Anwendungen; (b) Gewährleistung der Interoperabilität zwischen neuen und älteren Geschäftsanwendungen und (c) Anpassung der Geschäftsstrategie und des laufenden Betriebs an Veränderungen im externen und internen Umfeld (Erforderung einer integrierten Planung und Ausführung von Unternehmensprozessen) (Madni et al., 2001). Die Ontologie lässt sich durch folgende vier Kriterien charakterisieren: *Neutralität*, *Erweiterbarkeit*, *Komplementarität* und *Interoperabilität*. Die Ontologie ist also so *neutral* wie möglich erstellt, indem sie aus Konzepten und Beziehungen zwischen den Konzepten besteht. Sie ist dabei notationsunabhängig und implementationsneutral. Beispielsweise kann sie mit UML (Unified Modeling Language) beschrieben werden, was in Abb. 3.10 und Abb. 3.11 zu sehen ist. Allerdings kann sie auch mit CORBA IDL (Interface Definition Language der Object Management Group), KIF (Knowledge Interchange Format) oder einer weiteren Modellierungsnotation dargestellt werden. Weiterhin erfüllt die Ontologie den Anspruch der *Erweiterbarkeit*, da der Kern der Ontologie auf weitere Anwendungsbereiche wie Produktdesign, Herstellung, Gesundheitswesen, Finanzdienstleistung sowie militärische Führung und Kontrolle erweiterbar ist. Außerdem erfüllt diese auch die Voraussetzung für die Komptabilität mit anderen relevanten Unternehmensontologien, die einen oder mehrere spezifische Aspekte eines Unternehmens betreffen. Darüber hinaus basiert IDEON auf der Erkenntnis, dass mehrere *komplementäre* Unternehmensperspektiven erforderlich sind, um die verschiedenen Aspekte eines Unternehmens zu modellieren und Unternehmensabläufe zu verstehen, zu entwerfen, zu verwalten und zu steuern. Schließlich ist IDEON *interoperabel*, sodass mit anderen speziellen Unternehmensprozessen zusammengearbeitet oder mit Ontologien und Modellierungsnotationen geplant werden kann. Da komplementäre Unternehmensperspektiven erforderlich sind, wird die Ontologie durch vier komplementäre Sichten (Unternehmenskontextsicht und Unternehmensorganisationssicht in Abb. 3.10 sowie Prozesssicht und Ressourcen-/Produktsicht in Abb. 3.11) dargestellt (Madni et al., 2001).



Unternehmenskontextsicht (oben) und Unternehmensorganisationssicht (unten)

Abb. 3.10: IDEON Ontologie I (Madni et al., 2001, pp. 41, 42)

Die Konzepte *Unternehmen (enterprise)* und *Prozess (process)* werden von allen vier Sichten verwendet. Die Unternehmenskontextsicht fokussiert die Interaktion eines Unternehmens mit seiner Umgebung mit Konzepten wie *Umgebung (environment)* oder *Beobachtung (observation)*. In der Unternehmensorganisationssicht wird die Struktur eines Unternehmens mit Konzepten wie z. B. *Person (person)*, *Ressource (resource)*, *Ziel (goal)* und *Strategie (strategy)* deutlich. Die Prozesssicht konzentriert sich auf den Zyklus von Planung, Ausführung und Steuerung mit Konzepten wie z. B. *Aktivität (activity)*, *Plan (plan)*, *Bedingung (condition)* und *Ereignis (event)*. Mit der Ressourcen-/Produktsicht werden die im Unternehmen vorhandenen Ressourcen beschrieben durch Konzepte wie materielle und personelle Ressourcen (Madni et al., 2001). Während die IDEON Ontologie das Unternehmen als Ganzes betrachtet und erfassen möchte, benötigt SoNBO lediglich eine Teilmenge der Informationen in einem Unternehmen. Daher können Teilaspekte dieser Unternehmensontologie (insbesondere aus der Unternehmensorganisationssicht und Produktsicht) vom SoNBO-Konzept wieder aufgegriffen werden (vgl. Abschnitt 5.5).



Prozesssicht (oben) und Ressourcen-/Produktsicht (unten)

Abb. 3.11: IDEON Ontologie II (Madni et al., 2001, pp. 43, 44)

Die **Organization Ontologie** (ORG) vom W3C (2014b) beschreibt eine zentrale Ontologie für Organisationsstrukturen (einschließlich Regierungsorganisationen), welche die Veröffentlichung von Linked Data von Organisationsinformationen unterstützen soll. Sie soll domänenspezifische Erweiterungen ermöglichen, Organisationen und Rollen klassifizieren und Erweiterungen bereitstellen, um verwandte Informationen wie z. B. organisatorische Aktivitäten zu unterstützen. Diese Ontologie stellt also eine generische und wiederverwendbare Kernontologie dar, die für die Verwendung in konkreten Situationen erweitert oder spezialisiert werden kann (W3C, 2014b). Mit der Ontologie können die organisatorischen Strukturen, die Berichtstruktur, die Standortinformationen und die Geschichte der Organisation dargestellt werden. Diese Informationen sind üblicherweise in Organigrammen enthalten und somit werden diese Informationen nicht als vollständig betrachtet, um eine Organisation zu beschreiben.

An dieser Stelle soll daher nochmal betont werden, dass es sich in dieser Ontologie um Basiskonzepte handelt, die für eine Anpassung auf die Anforderung einer Domäne die Ausgangsbasis bilden. Die Hauptklassen sind in Abb. 3.12 dargestellt und die Ontologie wird dort mit RDF semi-formal beschrieben (W3C, 2014b).

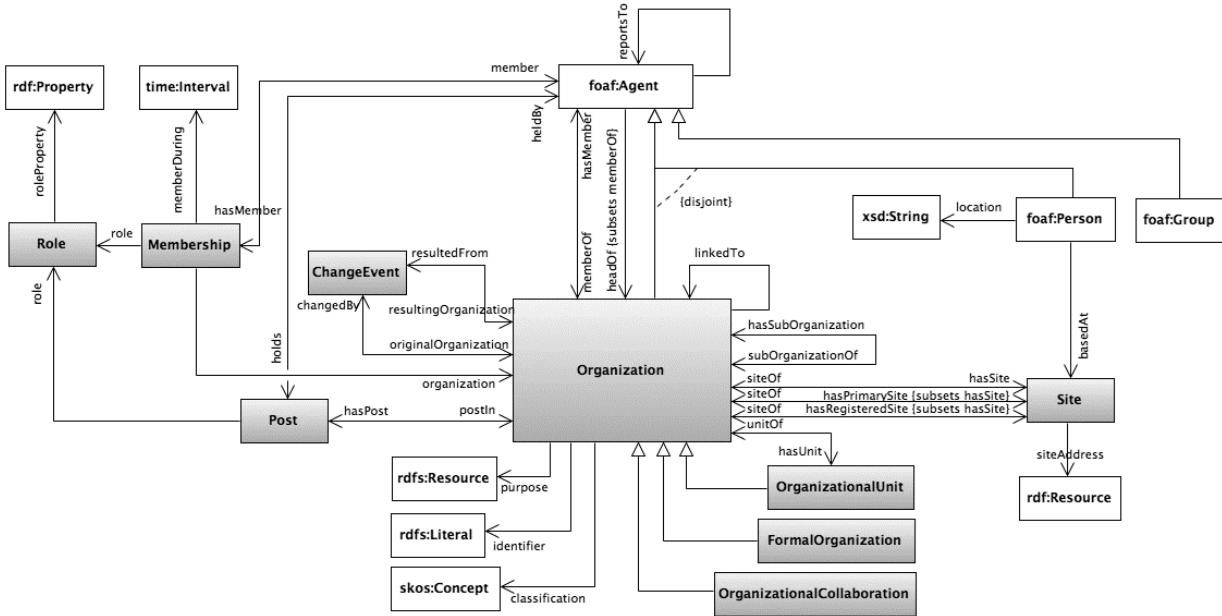


Abb. 3.12: Organization Ontologie (W3C, 2014b)

Auch diese Ontologie beschreibt ein Unternehmen als Ganzes, weshalb für die SoNBO-Ontologie ein Teil dieser Ontologie von Interesse sein kann. Dies wird in Abschnitt 5.5 wieder aufgegriffen.

Die **Business Ontologie** (BO) wurde basierend auf der langjährigen Arbeit der Global University Alliance und ihrer Mitglieder durch von Rosing & Laurier (2015) eingeführt. Diese Ontologie ist umfassend und möchte alle Aspekte des Unternehmens abdecken. Daher sind ihre Begriffe in einer Top-Level-Domäne und in mehreren sich überschneidenden Subdomänen organisiert. In diesen Subdomänen sind klar definierte Modelle (= Sichten mit klar definierten Metaobjekten) enthalten. In Abb. 3.13 sind die Modelle in drei Ebenen (Meta Meta Modell, Meta Modell, Modell) unterteilt abgebildet. Auf der untersten Ebene befindet sich die Objektebene. Innerhalb der Modell-Ebenen werden die Metaobjekte strukturiert, die die Objekte auf der Objektebene beschreiben. Das Metaobjekt definiert u. a. den Typ eines Objekts (auch: Klasse), Beziehungsattribute und Funktionen. Die Meta-Modell-Ebene beschreibt die Objekte innerhalb der Modell-Ebene mit den folgenden acht Gruppen: Purpose & Goal, Business Competency, Business Service, Business Process, Application, Data, Platform und Infrastructure. Sie bilden die Unterklassen des Meta-Meta-Modells. Die Meta-Meta-Modell-Ebene definiert die Business Domain und stellt auch die Beziehungen zu Informationskonzepten her wie z. B. Anwendungen und Technologien. Diese Ebene beschreibt die Meta-Modelle und die darin enthaltenen Beziehungen. Die Beziehungen auf den einzelnen Ebenen sind polymorph. Dies bedeutet, dass sich die Aus-

sagen in den jeweiligen Ebenen auch für die Objekte in den Ebenen darunter gelten. Dies gilt allerdings nicht in die andere Richtung. Bei dieser Ontologie handelt es sich um eine empirische Ontologie, da die Wurzeln in der Praxis liegen (von Rosing & Laurier, 2015).

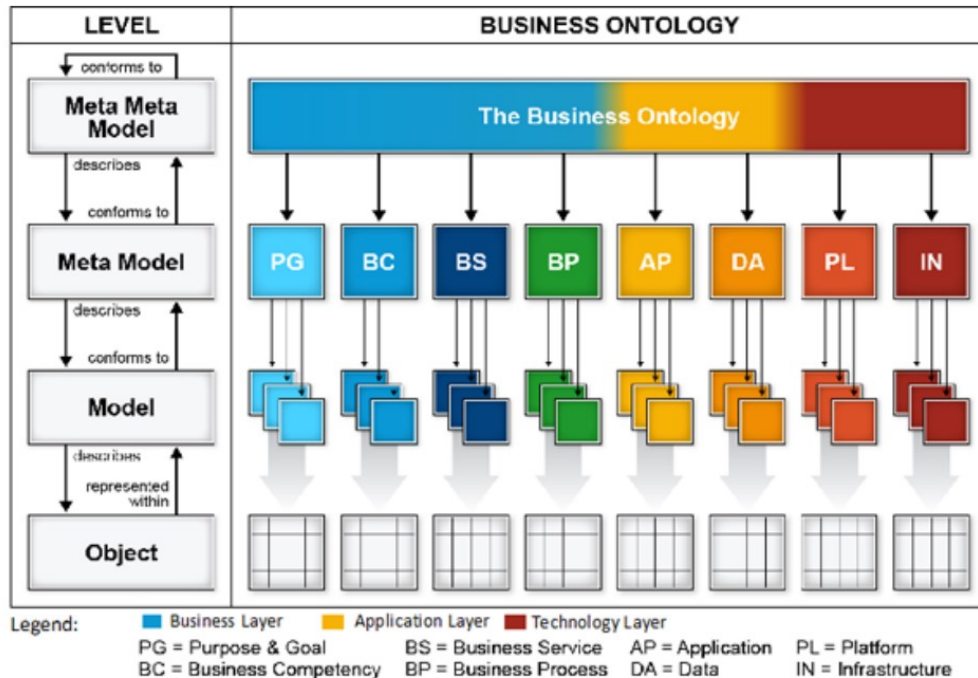


Abb. 3.13: Ebenen der Business Ontologie (von Rosing & Laurier, 2015, p. 37)

Diese Ontologie soll auch ein gesamtes Unternehmen beschreiben. Für die SoNBO-Ontologie (Abschnitt 5.5) sind vor allem die Informationen aus dem Business Layer (blau in Abb. 3.13) relevant. Insbesondere können die Meta-Objekte aus der Gruppe Business Competency wiederverwendet werden. Beispiele dazu sind: Ressource, Akteur, Produkt oder Bericht (von Rosing & Laurier, 2015). Außerdem wird das polymorphe Prinzip mitaufgenommen.

Zusammengefasst lässt sich festhalten, dass die existierenden Unternehmensontologien vor allem aus dem Forschungsumfeld stammen. Über die Anwendung von Ontologien in der Praxis lässt sich keine Aussage treffen, da diese i.d.R. nicht in der wissenschaftlichen Literatur dokumentiert werden (Gorjatshev, 2019). Zudem ist davon auszugehen, dass Unternehmen – sollten sie eine Ontologie einsetzen – diese nicht öffentlich zur Verfügung stellen bzw. Schemata verwenden, die aber nicht als Ontologie bezeichnet werden, sodass diese auch nicht unter diesem Begriff gefunden werden können. Alle Unternehmensontologien haben das Ziel, eine einheitliche Struktur zur Beschreibung eines Unternehmens zu entwickeln. Sie unterscheiden sich in der Fokussierung auf eine Branche (z.B. E-Commerce oder Logistik; diese Beispiele aus Tab. 3.1 wurden hier nicht erläutert) oder auf ein gesamtes Unternehmen (siehe die drei aufgeführten Beispiel-Ontologien: IDEON, ORG, BO). Ein Hauptzweck in der Erstellung von Ontologien für die Praxis ist der strukturierte Informationsaustausch zwi-

schen Unternehmen oder innerhalb eines Unternehmens. Bei der Visualisierung ergibt sich das Problem, dass lediglich ein Ausschnitt für einen Menschen sinnvoll visualisiert werden kann, da eine vollständige Ontologie zwar von einer Maschine gelesen werden kann, aber für einen Menschen in der Regel nicht mehr lesbar ist. In der Historie der Unternehmensontologien wird deutlich, dass es schon seit den frühen 1990er Jahren ein Bedürfnis für eine Unternehmensontologie existierte, die Art der Beschreibung sich allerdings unterschied, da noch keine Standards existierten. Seit der Einführung von Ontologiebeschreibungssprachen durch das W3C werden diese auch zunehmend für die Beschreibung eingesetzt.

3.2.1.3 Graphentheorie

Im SoNBO-Ansatz werden die Ontologie und der Knowledge Graph als soziales Netzwerk und somit als Graph aufgefasst, weshalb hier ein kurzer Überblick über die Graphentheorie gegeben wird. Dies wird in Kapitel 4 bei der Formulierung des SoNBO-Konzeptes wieder aufgegriffen. Die Graphentheorie ist ein Teilgebiet der diskreten Mathematik und geht auf die Arbeit *Königsberger Brückenproblem* von Leonhard Euler zurück, die 1736 erschien. In der zweiten Hälfte des 20. Jahrhunderts gewann dieser Zweig der Mathematik an Bedeutung und wurde ein wichtiges Teilgebiet der Informatik (Brandes, 2010). Die Graphentheorie wird in der Regel dazu verwendet, ein Problem auf das Zusammenspiel von Beziehungen zwischen Objekten herunterzubrechen und somit die Fragestellung unabhängig vom Kontext zu betrachten (Brandes, 2010). Dazu wird ein Graph konstruiert, der von Brandes (2010) folgendermaßen beschrieben wird:

„Ein Graph $G = (V,E)$ ist ein Paar aus einer Menge V von Knoten (engl. vertices, oft auch nodes oder points), welche die Akteure des Netzwerks repräsentieren, und einer Menge E von Kanten (engl. edges, oft auch links, ties oder lines), die Beziehungen zwischen den Akteuren repräsentieren. Wir bezeichnen die Anzahl der Knoten eines Graphen $G = (V,E)$ durchgängig mit $n = |V|$, und die Anzahl seiner Kanten mit $m = |E|$.“ (Brandes, 2010)

Graphen können ungerichtet oder gerichtet sein. Bei einem ungerichteten Graphen besitzen die Kanten keine Richtung und weisen in der Visualisierung keinen Pfeil auf. Weitere Bezeichnungen dafür sind *schlicht* oder *bipartit*. Bei einem gerichteten Graph weisen die Kanten eine Richtung auf und es können auch mehrere Kanten zwischen zwei Objekten bestehen (Brandes, 2010). In der Definition, die Tittmann (2019, p. 12) verwendet, wird hinzugefügt, dass in einem ungerichteten Graph einer Kante zwei (nicht notwendig verschiedenen) Knoten aus V zugeordnet sind. Eine Kante auf den Knoten selbst ist also auch möglich. Dies wird in der Definition nach Brandes (2010) nicht ausgeschlossen, aber es soll hier nochmal betont werden, da dies für das SoNBO-Konzept relevant ist. Ein Netzwerk wird auch mithilfe der Graphentheorie dargestellt. Beispielsweise kann die Verwandtschaft damit dargestellt werden und die Akteure (oder Objekte) sind Personen und die Beziehung ist eine familiäre Beziehung. Der Graphentyp ist dann schlicht und ungerichtet (Brandes, 2010). Ein Beispiel für die Untersuchung von Netzwerken mit der Graphentheorie ist die Erreichbarkeit. Dazu werden die Netzwerke in ihre wesentlichen Bestandteile zerlegt. Es gibt dabei *stark zusammenhängende* und (*schwach*) *zusammen-*

hängende Graphen. Die Unterscheidung ist von der Erreichbarkeit eines Knoten zu einem anderen abhängig. Bei ersterem ist jeder Knoten von jedem anderen aus erreichbar. Bei zweiterem ist dies der Fall, wenn es auf ungerichteten Weg möglich ist (Brandes, 2010).

Die Betrachtung eines Graphen als Netzwerk wird in der sozialwissenschaftlichen Netzwerkanalyse verwendet, indem diese von der Idee geleitet wird, dass Akteure (= Knoten; z. B. Personen, Organisationen, Nationalstaaten) durch soziale Beziehungen (= Kanten) miteinander verbunden sind. Es werden Relationen zwischen Akteuren untersucht (Haas & Malang, 2010).

3.2.1.4 Konzepte für den Informationszugriff

Bei SoNBO ist die Art des Informationszugriffes mithilfe der Navigation durch ein Netzwerk von Bedeutung. Dies wird zum einen in Abschnitt 3.2.2.6 beim Vergleich der Ontologie-basierten Integrationsansätzen wieder aufgegriffen. Außerdem wird dies in Kapitel 4 zur Beschreibung des SoNBO-Konzeptes verwendet. Ter Hofstede, Proper & van der Weide (1996) unterscheiden bei der Art der Informationsgewinnung (Query Formulation) zwischen Query by Construction (QbC) und Query by Navigation (QbN). Während bei ersterem der Suchbegriff von dem Suchenden formuliert werden muss, wird bei letzterem die Informationsbasis dem Nutzer so bereitgestellt, dass der Nutzer durch die Informationen navigieren kann. Ein wichtiges Element dazu ist auch die Facettensuche (Tunkelang, 2009). Dies ist motiviert durch die Problematik, dass die Abfrageformulierung im Kontext großer Schemata eine Herausforderung darstellt. Der Benutzer kann durch die große Menge an Informationen, die im Informationssystem gespeichert sind, überfordert werden. Daher entwickelten ter Hofstede et al. (1996) eine Strategie, um diese Herausforderung zu adressieren. Der Ansatz ist die Verwendung von Navigationsmechanismen, die mit einer geschichteten Hypermedia-Architektur umgesetzt wird. Diese wird verwendet, um die im Informationssystem enthaltenen Informationen auf mehreren Abstraktionsebenen zu beschreiben. Der Benutzer formuliert dabei zunächst eine Reihe einfacher Abfragen (= lineare Pfade in der Informationsstruktur) mithilfe von QbN. Nachdem Benutzer diese linearen Pfade angegeben haben, können sie diese für die Formulierung von komplexen Abfragen kombinieren, sodass QbC ermöglicht wird (ter Hofstede et al., 1996). Dieses Prinzip des Informationszugriffes wird von OBDA und insbesondere von dem Optique-Projekt (vgl. Abschnitt 3.2.2.3) wieder aufgegriffen (Soylu et al., 2017, 2018; Soyly, Giese, Jimenez-Ruiz, Vega-Gorgojo, & Horrocks, 2016).

3.2.1.5 NoSQL-Datenbanken

Die Ontologie-basierten Informationsintegrationsansätze benötigen eine Datenbank, um die Ontologie abzuspeichern. Dazu kann jedes Datenbankmodell verwendet werden, wobei sich NoSQL-Datenbanken (z. B. Dokumentenbasierte-Datenbanken oder Graphdatenbanken) für die Speicherung der Ontologie als Graph durchsetzen. NoSQL steht für *Not only SQL* (Structured Query Language) und die Geschichte dieser Systeme begann parallel zu den Anfängen relationaler Systeme. In den 80er Jahren entstanden mit Systemen wie Lotus Notes die ersten noch heute populären Vorreiter der NoSQL-Systeme (im Vergleich zu heute noch mit geringen Datenmengen). Auch wenn 1998 der Begriff NoSQL

zum ersten Mal bei einer Datenbank von Carlo Strozzi auftauchte, kam der große Durchbruch für NoSQL mit dem Web 2.0 (seit etwa 2000) und dem Ziel, große Datenmengen im Terabyte- oder sogar Petabyte-Bereich mit Web-Scale-Datenbanken zu verarbeiten im Jahr 2009 (Edlich, Friedland, Hampe, & Brauer, 2010). Der NoSQL-Vorreiter war Google gefolgt von Firmen wie z. B. Yahoo, Amazon und später auch von sozialen Netzwerke wie z. B. Facebook oder LinkedIn. Bis 2005 entstanden bereits kleinere Datenbanken, die in vielen Bereichen schon NoSQL-Charakter besaßen (z. B. Neo4j). Zwischen 2006 und 2009 schließlich wurden die heutigen klassischen NoSQL-Systeme wie z. B. CouchDB, Cassandra oder MongoDB entwickelt. Erst im Mai 2009 wurde der Begriff NoSQL in einem Webblog von Eric Evans erwähnt. Datenbanken, die von den relationalen SQL-Strukturen abweichen, existieren also bereits seit mehreren Jahren. Die Etablierung dieser Systeme als Kontrast zu den SQL-Datenbanken entstand aber erst 2009 (Edlich et al., 2010).

Neben der Verarbeitung von großen Datenmengen besitzen Graphdatenbanken als NoSQL-Datenbanken die Eigenschaft, das Graphenmodell effektiv abzuspeichern. Die Verwendung einer relationalen Datenbank für die Traversal eines Graphen mithilfe von Tabellen und Join-Operationen bringt Performanceprobleme mit sich. Somit ist auch eine SQL-Datenbank für die Speicherung des Graphen möglich, wobei die Vorteile bei der Verwendung einer Graphdatenbank überwiegen (Edlich et al., 2010). Das Labeled Property Graph-Modell (LPG-Modell) setzte sich bei allen derzeit auf dem Markt befindlichen Graphdatenbanken durch, obwohl diesem Modell eine wissenschaftliche Fundierung fehlt (Edlich et al., 2010). Dieses Modell basiert auf dem Graphdatenmodell. Während letzteres aus Knoten und Kanten besteht, besitzt das LPG-Modell noch Eigenschaften in den Knoten und Kanten und ist ein gerichteter, multi-relationaler Graph. Die Properties sind Key-Value-Beziehungen (Edlich et al., 2010).

Connection between NYC and SF: 300 USD / 4100 in Km.

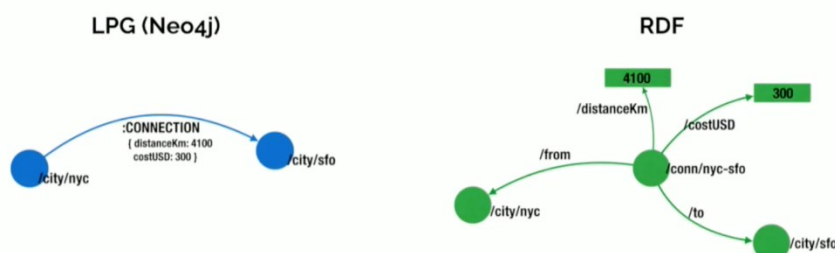


Abb. 3.14: RDF Triple Stores vs. Labeled Property Graphs (Barrasa, 2017)

In Abb. 3.14 wird der Zusammenhang zwischen RDF Triple Stores und LPG deutlich, indem die Verbindung zwischen New York City und San Francisco auf zwei verschiedene Arten visualisiert wird. In beiden Darstellungen werden Knoten und Kanten verwendet. Im LPG-Modell werden zwei Knoten und eine Kante verwendet, indem die Knoten die jeweilige Stadt und die Kante die Verbindung mit der Ei-

genschaft des Preises und der Entfernung sind. Dieses Prinzip wird auch bei SoNBO verwendet. Bei RDF hingegen sind die Städte ebenfalls jeweils ein Knoten, wobei auch die Verbindung als Knoten dargestellt wird. An diesem Knoten wiederum hängen die Eigenschaften dieser Verbindung. Für die Überführung von RDF Daten in das Property Graph-Modell existiert bereits eine Veröffentlichung (Tomaszuk, 2016).

3.2.1.6 Einführung von betrieblichen Anwendungssystemen im Unternehmen

Für die Einführung von betrieblichen Anwendungssystemen werden verschiedene Konzepte in der Literatur vorgeschlagen. Da das SoNBO-Framework ebenfalls bei der Einführung in Unternehmen unterstützen soll, fließen diese existierenden Konzepte bei der Entwicklung in das Framework mit ein (vgl. Kapitel 5). Die in den 1960er Jahren entstandene Soft System Methodology bietet eine Methodik, um eine Problemsituation im Unternehmen zu analysieren. In den 1990er Jahren entstand ARIS, wodurch fünf verschiedene Sichten bereitgestellt werden, mit denen eine Geschäftsprozessmodellierung unterstützt wird. Diese Sichten berücksichtigen neben technischen Aspekten auch die Anforderungen des Unternehmens. ARIS findet z. B. Anwendung in der Einführung von ERP-Standardsoftware. Für die Unterstützung der Einführung von Enterprise Collaboration Systems werden die im IRESS-Konzept existierenden Use Cases und Szenarien vorgeschlagen.

In den vorangegangenen Abschnitten zu den Grundlagen für den Status quo in der Wissenschaft lag der Fokus auf dem Fachwissen, auf welches in der Dissertation wieder zugegriffen wird. In diesem Abschnitt geht es um eine Einführungsmethodik von Software. Aspekte aus den existierenden Konzepten haben Einfluss auf das Framework in dieser Dissertation, da dieses ebenfalls bei der Konfiguration der Standardsoftware SoNBO-Explorer unterstützt (vgl. Kapitel 5).

3.2.1.6.1 SSM – Soft Systems Methodology

Soft Systems Methodology (SSM) wurde in den 1960er Jahren von Peter Checkland (2000) an der Universität Lancaster entwickelt. Mit dieser handlungsorientierten Methode werden Benutzer angeleitet, wie sie sich über eine problematische Situation informieren und Maßnahmen ergreifen können, um diese zu verbessern (Checkland & Poulter, 2010). Diese Methodik kann in jeder Situation angewandt werden, in der überlegtes Handeln von Menschen erforderlich ist (Checkland & Poulter, 2010). SSM lässt sich in verschiedene Teile gliedern (Checkland, 2000): Zunächst muss sich über die Problemsituation informiert werden, indem ein Rich Picture angefertigt wird. Zusätzlich gibt es weitere Frameworks (Analysis One, Two und Three), mit denen die Problemsituation analysiert wird. Darauf aufbauend können dann zielgerichtete Aktivitätsmodelle erstellt werden. Im Folgenden wird ein Verfahren zur Erstellung von Aktivitätsmodellen erklärt: Zunächst sollen (1) mithilfe von Verben als Imperativ die Aktivitäten aufgeschrieben werden. Anschließend werden (2) die Aktivitäten ausgewählt, die sofort ausgeführt werden können, da sie beispielsweise nicht abhängig von anderen sind. Diese werden im dritten Schritt (3) in eine Zeile geschrieben. Darunter werden dann die Aktivitäten aufgeführt, die in Abhängigkeit zu denen in der ersten Zeile stehen. Die weiteren Zeilen werden analog befüllt, bis alle Aktivitä-

ten berücksichtigt sind. Dadurch werden die Abhängigkeiten angezeigt. Diese Visualisierung wird dann in Schritt (4) neu erstellt, um überlappende Pfeile zu vermeiden. Aufbauend auf diesen Aktivitätsmodellen werden dann im letzten Teil von SSM die Situation erkundet und darauf aufbauend Maßnahmen zur Verbesserung der Problemsituation ergriffen (Checkland, 2000). Dieses Prinzip von SSM wird in Abschnitt 5.3 aufgegriffen.

3.2.1.6.2 ARIS – Architektur integrierter Informationssysteme

ARIS steht für *Architektur integrierter Informationssysteme* und wurde von Scheer (1996, 1998, 2002) in den 1990er Jahren entwickelt. ARIS ist ein Rahmenkonzept, um Informationssysteme vom Fachkonzept bis zur Implementierung zu beschreiben (auch: modellieren) und ist damit dem Kern der Wirtschaftsinformatik als Bindeglied zwischen der Betriebswirtschaft und Informationstechnik zuzuordnen. Mit ARIS wird die Komplexität der Geschäftsprozessbeschreibung durch die Strukturierung in Beschreibungssichten reduziert (Scheer, 1998). Es gibt verschiedene Anwendungsbereiche von ARIS. Beispielsweise unterstützt ARIS bei der Geschäftsprozessmodellierung, bei der Entwicklung und/oder bei der Konfiguration von Informationssystemen unter Berücksichtigung der unternehmensspezifischen Anforderungen (Scheer, 2002). Zentral ist bei dem ARIS-Konzept das ARIS-Haus mit den fünf Sichten (vgl. Abb. 3.15): Funktionssicht, Organisationssicht, Datensicht, Leistungssicht und Steuerungssicht (Scheer, 2002).

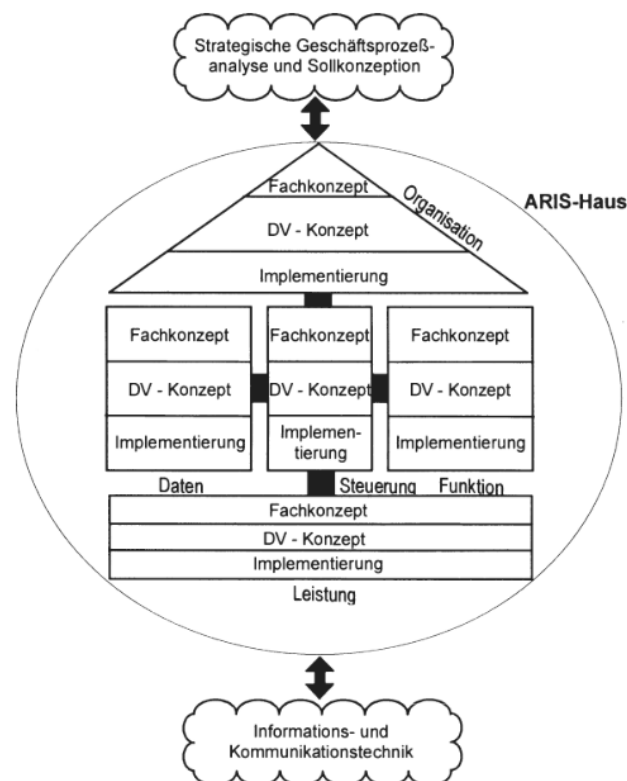


Abb. 3.15: ARIS-Haus (Scheer, 2002, p. 41)

Wie bereits erwähnt ist ein Anwendungsbereich die Einführung von Standardsoftware. Dafür ist ein strukturiertes und methodisches Vorgehen erforderlich. Im Rahmen von Einführungsprojekten von Standardsoftware sind teilweise umfangreiche kundenspezifische Anpassungen notwendig. Ziel der Methode ist es, Projekte schneller und effizienter umzusetzen (Scheer, 1998). Durch den Einsatz des ARIS-Toolset (insbesondere Prozessbeschreibung) kann eine Konfiguration der Standardsoftware methoden- und modellorientiert durchgeführt werden. Dadurch und u. a. durch eine Branchenerfahrung, die Auswirkung auf das ARIS-Toolset hat, können zukünftige Einführungsprojekte effizienter umgesetzt werden (Scheer, 1998).

3.2.1.6.3 IRESS – Identification of Requirements for Enterprise Social Software

Für die Einführung von Enterprise Collaboration Systems, die im Gegensatz zu ERP-Systemen nicht prozessorientiert sind, wird in der Literatur ein anderes Konzept vorgeschlagen: IRESS (vgl. Abb. 3.16) steht für *Identification of Requirements for Enterprise Social Software* und erhebt die Anforderungen für ein ECS in Unternehmen (Glitsch & Schubert, 2017; Schubert & Glitsch, 2015, 2016).

Organization	Use Cases	Customer communication Event management Human resource management Idea and innovation management Inter-employee communication	Internal business communication Knowledge management Management accounting Project organization	Sales opportunity and quotation management Software development Team organization Workshop organization	
	Collaboration Scenarios	Administering documents Alerting to news Conducting a meeting Conducting a poll Conducting a survey Creating meeting minutes and tasks	Discussing topics Documenting information Enriching information Finding an expert Joint authoring Organizing meeting	Posting news Rating information Retrieving information Sharing files Sharing information Solving a problem	
	Actions	Person, Social Business Document, ...	Search Edit Rate Label	Clarify Share Notify ...	
Software Support	Collaborative Features (C⁴)	Communication Asynchronous sent (rich) text message Blogs Broadcast Chat Comments, annotations Discussion forums Message boards Microblogging Unified Communication Video conferencing Voice message asynchronous	Cooperation/Collaboration Markup of changes Ratings, rankings Screen sharing/shared desktop Shared authoring Shared workspaces User profiles Workspace awareness	Content Combination Collecting feedback Content collection Content management Content subscription Data aggregation Data integration Document management Linking Pointers or references to content Search Tagging, Folksonomies Visualization of tag usage	Coordination Document and version control Graphical flow Group calendar, deadline planning Polls and voting Presence awareness Reminders, triggers, alerts Resource planning Roles Shared tasks User directories Workflow support
	Software Components	Workspace Blog Wiki	Forum Tasks Files	Calendar Microblog	

Abb. 3.16: IRESS-Framework (Glitsch & Schubert, 2017, p. 869)

Das IRESS-Framework bietet eine Unterstützung bei der Einführung von ECS, indem die Anforderungen der Organisation (vgl. oberer Bereich in Abb. 3.16) und die Unterstützung durch die Software (vgl. un-

terer Bereich in Abb. 3.16) betrachtet werden. Für die Erhebung der Anforderungen werden Use Cases und Kollaborationsszenarien verwendet, die in Abb. 3.16 aufgelistet sind. Diese Anforderungen werden mit den Funktionalitäten der einzuführenden Software in Verbindung gebracht, die ebenfalls in Abb. 3.16 aufgeführt sind. Auf diese Weise erfolgt die Auswahl und Konfiguration von Kollaborationssoftware und damit verbunden auch die Entscheidung, ob eine integrierte Lösung (ein Enterprise Collaboration System) oder eine Suite (z.B. die Toolsuite von Microsoft) eingeführt wird. Für die Umsetzung von IRESS in Unternehmen unterstützt ein Katalog bestehend aus Use Cases und Kollaborationsszenarien als Webanwendung, der Beispiele enthält und Unternehmen im Evaluationsprojekt bezüglich der Softwareauswahl unterstützen soll (Glitsch & Schubert, 2017).

3.2.1.7 Prinzipien von Social Software

In diesem Abschnitt werden die Eigenschaften von Social Software (im Unternehmenskontext: *Enterprise Social Software*) vorgestellt, die für die Entwicklung des SoNBO-Konzeptes und des SoNBO-Frameworks relevant sind, da bei SoNBO Prinzipien von *Social Software* verwendet werden.

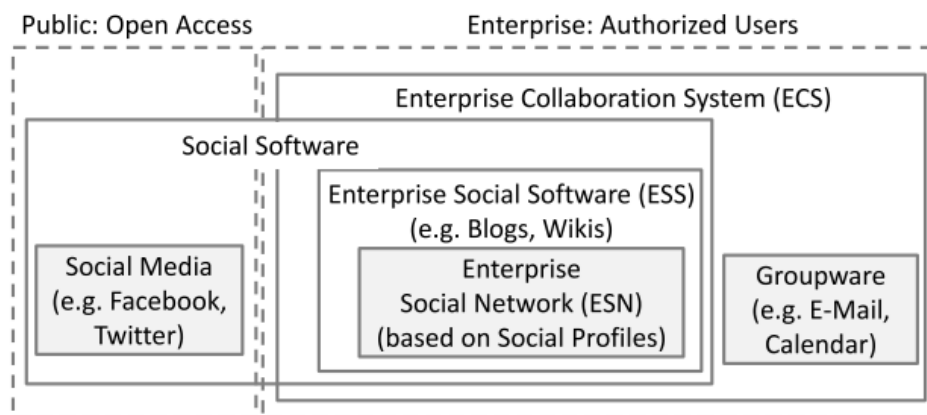


Abb. 3.17: Einordnung von Enterprise Social Software (Schwade & Schubert, 2017, p. 402)

Das hier verwendete Verständnis von *Social Software* folgt der Terminologie (vgl. Abb. 3.17) von Schwade & Schubert (2017). *Social Software* ist eine Softwarekategorie, welche soziale Funktionen (z. B. folgen) bereitstellt. Die Social-Media-Eigenschaften wie z. B. *Social Profile* inspirierten die Entwicklung von spezialisierter *Social Software* für Unternehmen, die *Enterprise Social Software* genannt wird. Mit diesem Begriff wird die Kollaborationssoftware in Unternehmen bezeichnet, die die Social-Media-Eigenschaften besitzt (Schwade & Schubert, 2017). In Unternehmen soll dadurch die Zusammenarbeit (z. B. Informationsaustausch, Koordination zwischen den Mitarbeitern) verbessert und das Bewusstsein der Benutzer für relevante Informationen erhöht werden (Schubert, 2018). Ein zentrales Merkmal ist das *soziale Profil*, welches nach Schubert (2018) ein *Content Type* von Social Software ist. Soziale Profile sind erweiterte Benutzerprofile, welche Informationen (z. B. Fähigkeiten, Ausbildung, Position und Abteilung) enthalten, die traditionell in der Personalakte eines Mitarbeiters verfügbar waren (Schubert, 2018). Diese Informationen charakterisieren diesen Mitarbeiter für das Unterneh-

men. Mit der sozialen Funktion *folgen* (im Folgenden: *follow*) oder *ins eigene Netzwerk einladen*, können Benutzer Beziehungen zwischen ihrem eigenen sozialem Profil und dem sozialen Profil anderer Mitarbeiter herstellen (Schubert, 2018; Schwade & Schubert, 2017). Daraus entsteht ein unternehmensweites soziales Netzwerk (im Folgenden: *Enterprise Social Network*), welches aus Personen besteht. Bei solchen sozialen Netzwerken handelt es sich um Graphstrukturen, die bei den öffentlichen Plattformen (z. B. Facebook) als Graph in einer NoSQL-Datenbank (z. B. Apache Cassandra bei Facebook) abgespeichert werden (Herden & Redenz, 2012). Für die Speicherung solcher Graphstrukturen bietet sich auch insbesondere eine Graphdatenbank an (vgl. Abschnitt 3.2.1.5). Während ein *Enterprise Social Network* in Unternehmen statische Inhalte darstellt, gibt es eine weitere Gruppe von sozialen Merkmalen, die einen dynamischen Informationsaustausch ermöglichen und das Bewusstsein für Themen am Arbeitsplatz schärfen: Funktionen wie *empfehlen* (im Folgenden: *recommend*), *gefällt mir* (im Folgenden: *like*), *erwähnen* (im Folgenden: *@mention*), *kommentieren* (im Folgenden: *comment*) oder *taggen* (im Folgenden: *tag*) werden verwendet, um dem Benutzer zu ermöglichen, Metainformationen zu Inhaltselementen hinzuzufügen (Schubert, 2018). Diese sind für die *Content Types* Microblog, Blog oder Wiki relevant. Ein weiteres Merkmal von Social Software ist der *Activity Stream*, der eine neue Form von Content-Aggregation vornimmt (Schwade & Schubert, 2017). Dieser besitzt insbesondere eine Awareness-Funktion, mit welcher der Benutzer über Beiträge, Änderungen, Empfehlungen oder weiteren Neuigkeiten zu Inhalten auf der Plattform informiert wird. Somit wird das Bewusstsein des Benutzers gegenüber diesen Informationen gesteigert. Diese Funktion bereichert den Arbeitsplatz um Informationen und kann zu *Serendipity* (unerwartete nützliche Informationen) führen (Schubert, 2018). Diese genannten Aktualisierungen eines Objektes können auch in das Social Profile gepostet werden.

Zusammengefasst lässt sich festhalten, dass zwei der charakterisierenden Eigenschaften von Social Software das (1) *Enterprise Social Network* bestehend aus dem Content Type *Social Profile* mit sozialen Funktionen wie z. B. *follow* und (2) der *Activity Stream* mit der Awareness-Funktion sind. Diese Charakteristika haben Einfluss auf den Aufbau des SoNBO-Konzeptes und des SoNBO-Explorers, welche in den späteren Abschnitten wieder aufgegriffen werden (vgl. Abschnitte in Kapitel 4 und 5).

3.2.2 Ontologie-basierte Integrationsansätze

Die Literaturrecherche über Ontologie-basierte Integrationsansätze dient zum einen zur Einordnung der Arbeit in das Fachgebiet und zum anderen werden hier die gewonnenen Erkenntnisse aus diesem Themengebiet für die Konzeptualisierung von SoNBO und das Framework verwendet. Die geeigneten Elemente werden dafür identifiziert (vgl. Abb. 3.46 auf Seite 97). Die Grundlagen dafür wurden im vorangegangenen Kapitel beschrieben.

3.2.2.1 Methodisches Vorgehen bei der Literaturanalyse

Für eine Literaturrecherche schlagen verschiedene Autoren unterschiedliche Vorgehensweisen (vgl. Abschnitt 2.2.2) vor (Torraco, 2005; vom Brocke et al., 2009, 2015; Webster & Watson, 2002). In An-

lehnung an vom Brocke et al. (2009) gliedert sich die Literaturanalyse in fünf Schritte, die in Abb. 3.18 abgebildet sind.

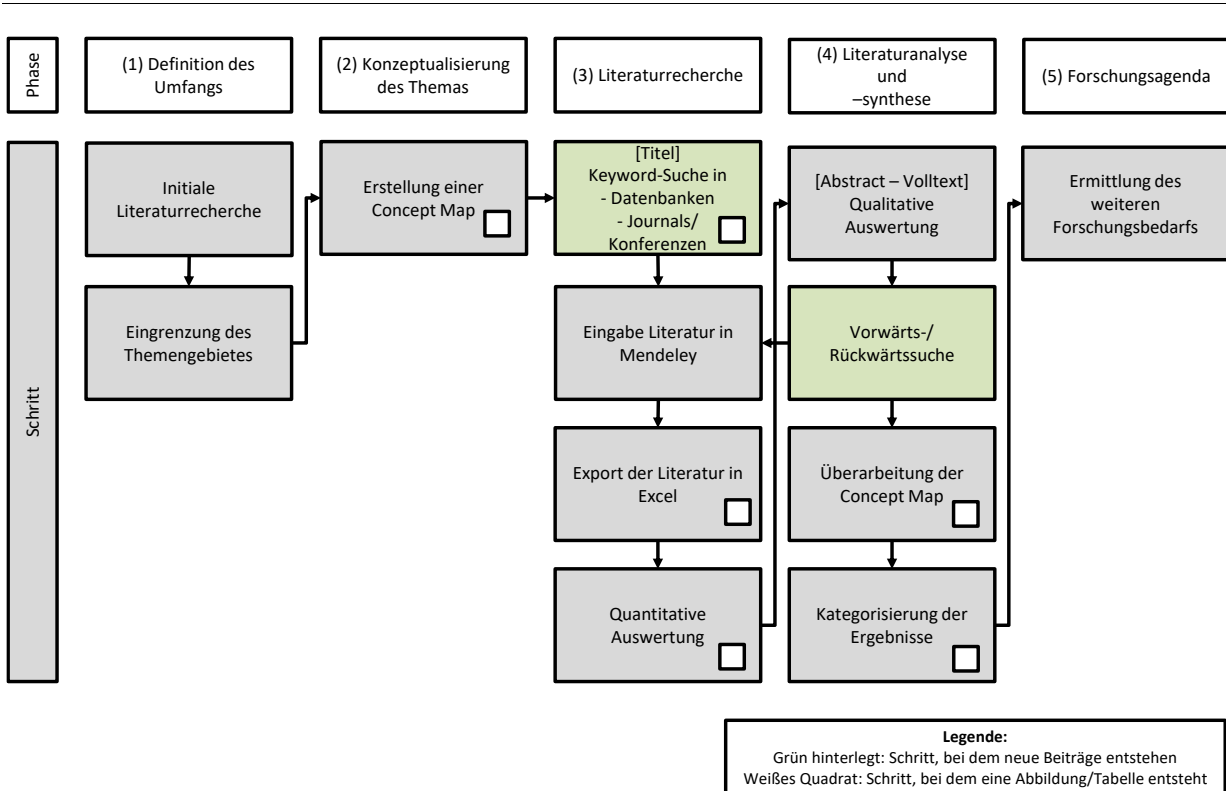


Abb. 3.18: Vorgehen bei der Literaturanalyse - in Anlehnung an vom Brocke et al. (2009)

Zu Beginn der systematischen Literaturrecherche wurde eine initiale Literaturrecherche durchgeführt (Schritt 1), um einen Überblick zu erhalten. Daraufhin wurde das Themengebiet eingegrenzt und der Überbegriff *Ontologie-basierte Integrationsansätze* gewählt. Darunter werden die Ansätze verstanden, die bei der Integration der Informationen die Daten in den Quellsystemen mit einem Schema beschreiben und der Nutzer mithilfe dieses Schemas auf die Informationen zugreift. Diese jüngeren Ansätze sind im Vergleich zu bisherigen Ansätzen (wie z. B. Enterprise Application Integration oder Service-Oriented Architecture) flexibler bei der Integration durch sich ändernde Herausforderungen für Unternehmen und somit auch in der Lage, die immer größer werdenden Datensätze in Unternehmen zu managen. Nach dieser ersten Sichtung der Literatur stellte sich heraus, dass es verschiedene Ontologie-basierte Integrationsansätze gibt. Dazu gehören OBDA (Ontology-Based Data Access), KG (Knowledge Graph), EKG (Enterprise Knowledge Graph), LD (Linked Data) und LED (Linked Enterprise Data). Diese waren bei der Literaturrecherche auch die zentralen Keywords. Die Ergebnisse sind in den Abschnitten 3.2.2.3 bis 3.2.2.5 aufgeführt. Für diese Ansätze sind außerdem weitere Konzepte relevant, die in Abschnitt 3.2.1 erläutert wurden. Dazu gehören beispielsweise die Ontologie, sowie Grundlagen zur Graphentheorie und Konzepte für den Informationszugriff (Query by Navigation; Query by Construction). Im Anschluss an die soeben erwähnte initiale Literaturrecherche wurde eine Concept

Map (vgl. Abb. 3.19) mit dem Tool XMind 8 erstellt (Schritt 2), die eine Übersicht über die Keywords und Suchquellen liefert.

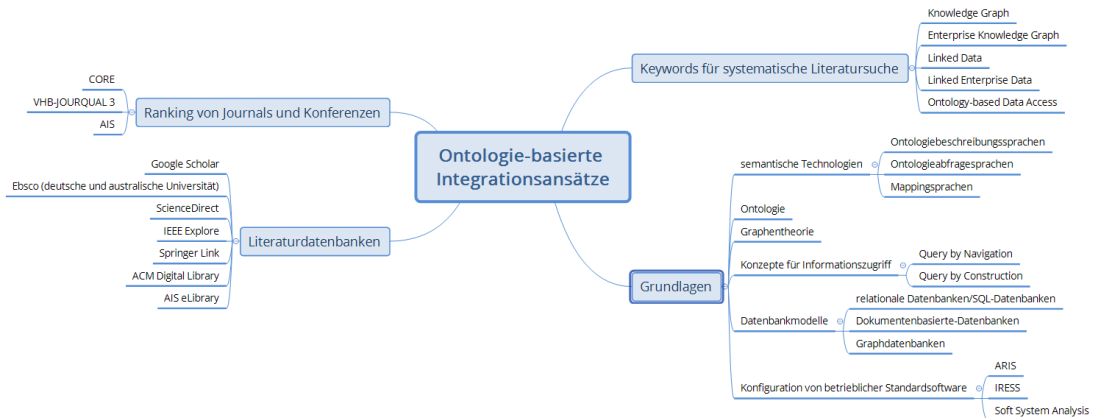


Abb. 3.19: Concept Map

An dieser Concept Map orientiert sich die daran anschließende systematische Literaturrecherche. Die Datengrundlage umfasste alle Datenbanken, die im Juni 2019 (und ergänzend im Januar 2020) im Universitätsnetz der Universität Koblenz verfügbar waren. Die Keywords wurden mit dem Begriff *Integration* in die in Abb. 3.19 aufgeführten Literaturdatenbanken (aus dem Universitätsnetzwerk heraus) eingegeben (Schritt 3). Zuerst wurde die Literaturdatenbank Google Scholar verwendet und jede Keyword-Kombination (also z.B. Enterprise Knowledge Graph Integration) eingegeben. Daran anschließend wurde in ScienceDirect, IEEE Explore, Springer Link, ACM Digital Library, AIS eLibrary und Ebsco recherchiert, bis sich die Beiträge wiederholten. Bei den Beiträgen, die zu diesem Zeitpunkt als zentral identifiziert wurden, wurde zudem noch eine Vorwärts- und Rückwärtssuche (Schneeball) durchgeführt.

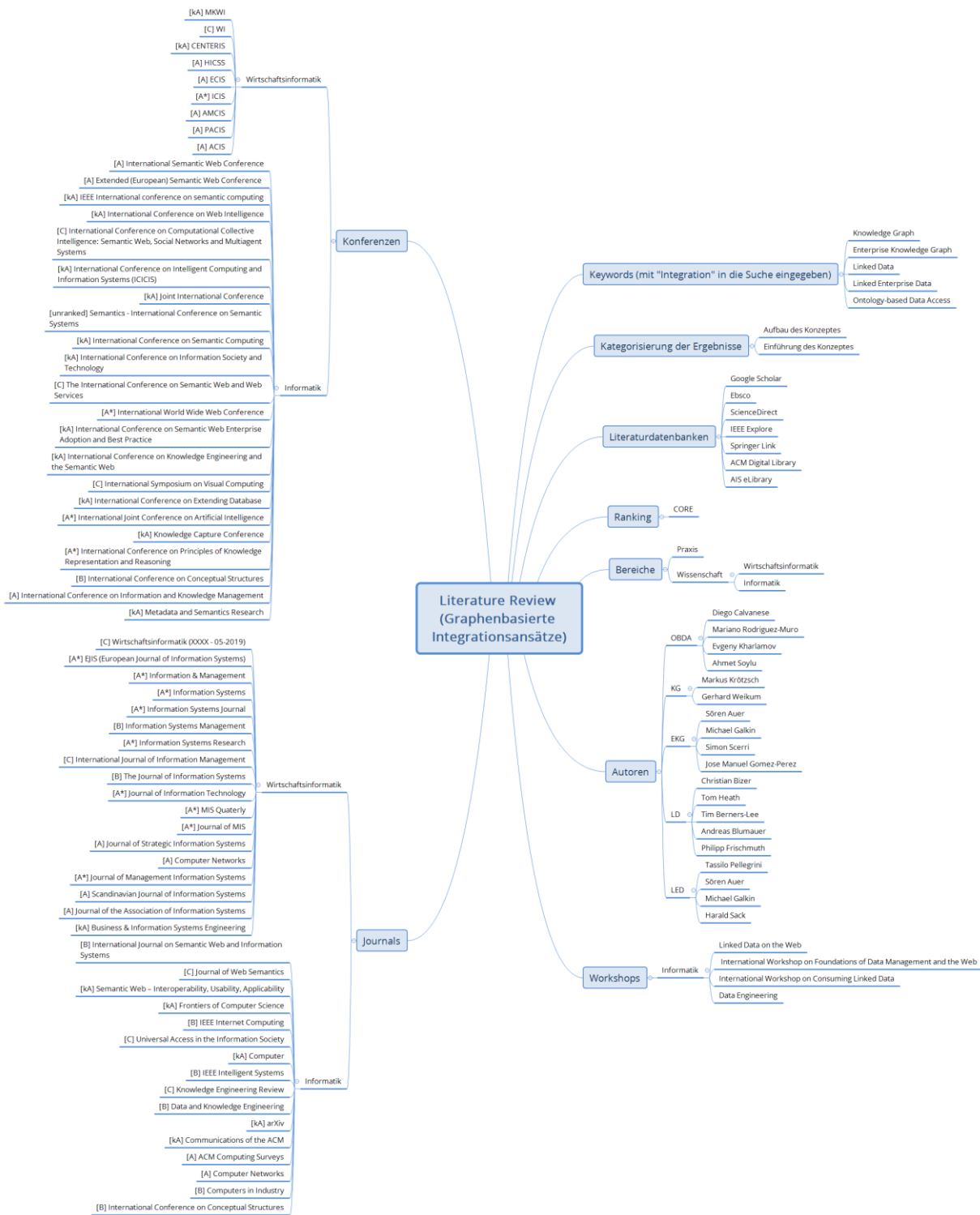


Abb. 3.20: Überarbeitete Concept Map

Die Literatur wurde im ersten Schritt anhand des Titels (204 Beiträge) ermittelt und anschließend in die verwendete Literaturverwaltungssoftware Mendeley eingegeben. Für die quantitative Auswertung wurden die Publikationseinträge nach Excel exportiert, um Analysen wie z. B. Anzahl der Paper bezogen auf die Autoren, häufige Konferenzen und Journals oder Entwicklung des Forschungsgebietes über

die Jahre hinweg durchzuführen. Dafür wurden die einzelnen Beiträge mit verschiedenen Codes versehen, indem in Excel weitere Spalten ergänzt wurden: Zum einen wurde bei den peer-reviewed Beiträgen (also Paper von Konferenzen oder Journals) das Ranking nach CORE (Computing Research and Education) ermittelt. Außerdem wurde eingetragen, ob der Beitrag in den Bereich Wissenschaft (Informatik, Wirtschaftsinformatik) oder Praxis eingeordnet werden kann. Weiterhin wurde die Zuordnung des Beitrages zu dem Keyword (OBDA, EKG, LED) eingetragen. Die Ergebnisse dieser quantitativen Analyse werden im nächsten Abschnitt vorgestellt. Für die zentrale qualitative Analyse (Schritt 4) wurden die 204 Paper anhand des Abstracts (105 Beiträge) und schließlich des Volltextes (51 Beiträge) weiter eingegrenzt (vom Brocke et al., 2009). Mit dieser Auswahl wurde auch eine Vorwärts- und Rückwärtssuche (Wohlin, 2014) durchgeführt, woraufhin diese Paper wiederum in Mendeley eingegeben (vgl. Pfeil in Abb. 3.18 von „Vorwärts-/ Rückwärtssuche“ nach „Eingabe Literatur in Mendeley“) und die quantitative sowie qualitative Auswertung erneut mit dieser Auswahl vorgenommen wurde. Anschließend wurde die eingangs aufgestellte Concept Map mit den neuen Ergebnissen überarbeitet (vgl. Abb. 3.20) und schließlich das Herzstück des Kapitels aufgestellt: Die Kategorisierung der Ergebnisse (vgl. Abschnitt 3.2.2.6). Die Ergebnisse wurden mit Prof. Dr. Steffen Staab (Universität Stuttgart) diskutiert, der ein Experte auf dem Gebiet zu Ontologien ist und damit verbunden auch Einblick in die dazugehörigen wissenschaftlichen Integrationskonzepte hat. Er veröffentlichte auch zahlreiche Beiträge in diesem Bereich (Guarino et al., 2009; Maedche, Staab, & Studer, 2001; Staab, 2002; Staab et al., 2000; Staab, Studer, Schnurr, & Sure, 2001; Staab & Schnurr, 2002; Volz, Handschuh, Staab, Stojanovic, & Stojanovic, 2004). Abschließend wurde ein weiterer Forschungsbedarf ermittelt (Schritt 5).

Im Folgenden werden die Ergebnisse im Detail erläutert. Der Abschnitt schließt mit einer Zusammenfassung der Ergebnisse aus der Literaturanalyse.

3.2.2.2 Hintergründe und Entwicklung des Fachgebietes

In diesem Abschnitt werden die Ergebnisse der quantitativen Analyse vorgestellt, um einen Überblick über das Fachgebiet zu erhalten. Dabei werden auch die Beiträge aus dem SoNBO-Forschungsprojekt mitaufgenommen, die während des Forschungsprozesses entstanden (Gebel-Sauer & Schubert, 2019b, 2019a; Gewehr et al., 2017; Götz & Gebel-Sauer, 2018).

3.2.2.2.1 Zeitliche Entwicklung der Beiträge

Für die historische Entwicklung des Fachgebietes wird zunächst die zeitliche Entwicklung der Beiträge ermittelt und visualisiert. In Abb. 3.21 wird die Entwicklung der Anzahl aller Beiträge anhand der schwarz gestrichelten Linie ersichtlich. Die Beiträge im Bereich der Ontologie-basierten Informationsintegration nehmen seit den 1990er Jahren sukzessive zu, wobei der größte Zuwachs seit etwa 2008 zu verzeichnen ist. Differenziert man noch zwischen den Beiträgen der einzelnen Themengebiete (vgl. die farbigen Linien in Abb. 3.21) ist zu erkennen, dass bereits in den 1970er Jahren der Grundstein für den Knowledge Graphen gelegt wurde. Zu dieser Zeit wurde damit allerdings noch keine Informations-

integration verbunden, sondern die Strukturierung von Wissensdomänen, welche später vom Semantic Web wieder aufgegriffen wurde. Ende der 1990er Jahre sind dann die ersten Veröffentlichungen zu OBDA und Linked Data zu erkennen. Dies erfolgte vermutlich im Zuge der zunehmenden Herausforderung des Managements mit der immer größer werdenden Datenmenge im Internet und Unternehmen. Es ist anzunehmen, dass die Standardisierung der semantischen Technologien durch das W3C auch einen Anteil daran hat, da die Entwicklung von Konzepten (z. B. OBDA) durch das Zurückgreifen von technisch komplexen Standards nun im Vordergrund stehen kann. Kurz darauf entstehen Beiträge zu Knowledge Graph, die sich auf die Theorien dazu aus den 1970er Jahren stützen und die Standardisierungen vom W3C verwenden. Daran anschließend erfahren die Beiträge zu Linked Data und Knowledge Graphen einen Zuwachs und etwas zeitverzögert entstehen auch die Beiträge für die Unternehmenswelt: Linked Enterprise Data und Enterprise Knowledge Graph. In den letzten Jahren schließlich nahmen die Beiträge zu SoNBO zu und somit kam das Thema auch in der Wirtschaftsinformatik an.

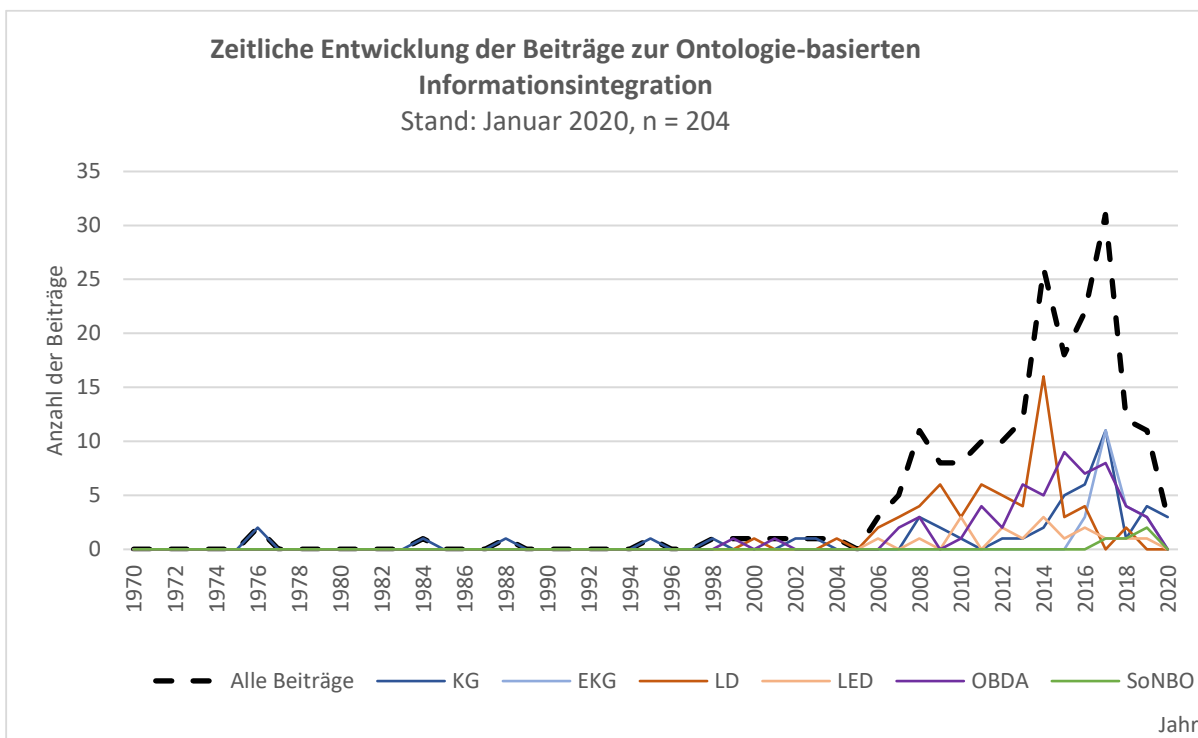


Abb. 3.21: Zeitliche Entwicklung der Beiträge zur Ontologie-basierten Informationsintegration

Zusammenfassend lässt sich festhalten, dass die Herausforderung, eine große Menge von verteilten Daten zu managen und darauf zuzugreifen, als erstes für das Internet durch die Semantic-Web-Community adressiert wurde und schließlich auf Unternehmen übertragen wurde. Es kann angenommen werden, dass die Standardisierung von semantischen Technologien für den sprunghaften Anstieg der Konzepte ausschlaggebend war.

3.2.2.2 Verteilung der Beiträge nach Fachgebiet, Themengebiet und CORE Ranking

Im Folgenden werden die relevanten Beiträge nach Fachgebiet, nach Themengebiet und nach CORE Ranking dargestellt.

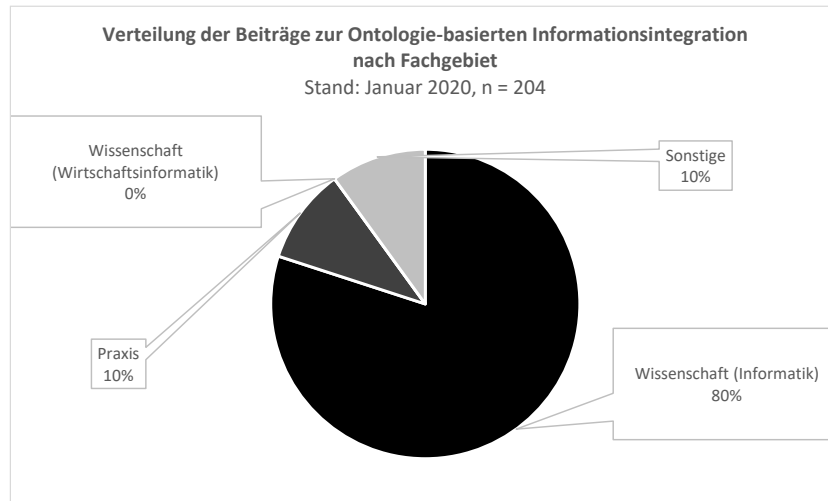


Abb. 3.22: Verteilung der Beiträge nach Fachgebiet

Der überwiegende Teil der Beiträge stammt aus der Informatik (vgl. Abb. 3.22). Zu gleichen Teilen stammen die Beiträge aus der Praxis und fachübergreifenden Gebieten (wie z. B. Biologie/Medizin) (da für diesen Bereich Beispielumsetzungen veröffentlicht wurden). Fast keine Beiträge stammen aus der Wirtschaftsinformatik. Dieser Teil ist mit fünf Beiträgen relativ klein und wird in dem Diagramm nicht angezeigt (vgl. Abb. 3.22).

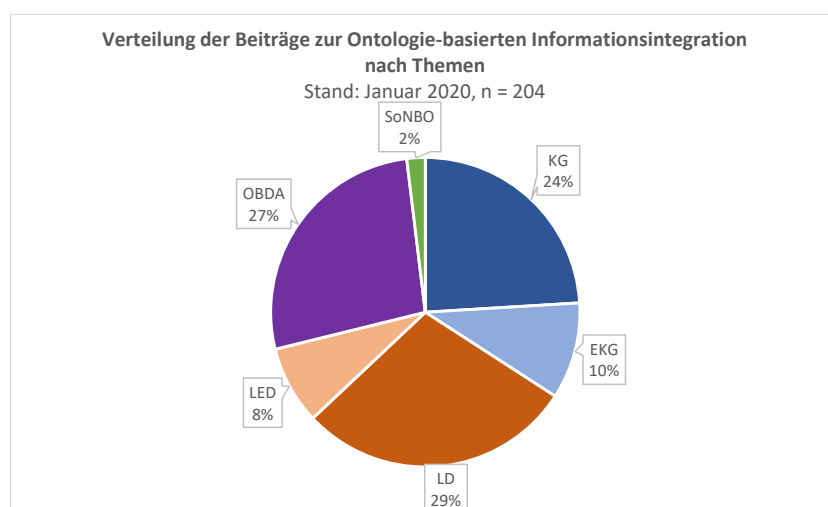


Abb. 3.23: Verteilung der Beiträge nach Themen

Die meisten Beiträge sind dem Themengebiet Linked (Enterprise) Data zuzuordnen (vgl. Abb. 3.23). Etwa ein Drittel der Beiträge sind im Bereich (Enterprise) Knowledge Graph veröffentlicht worden. Der Anteil der Veröffentlichungen im Bereich OBDA liegt etwas darunter.

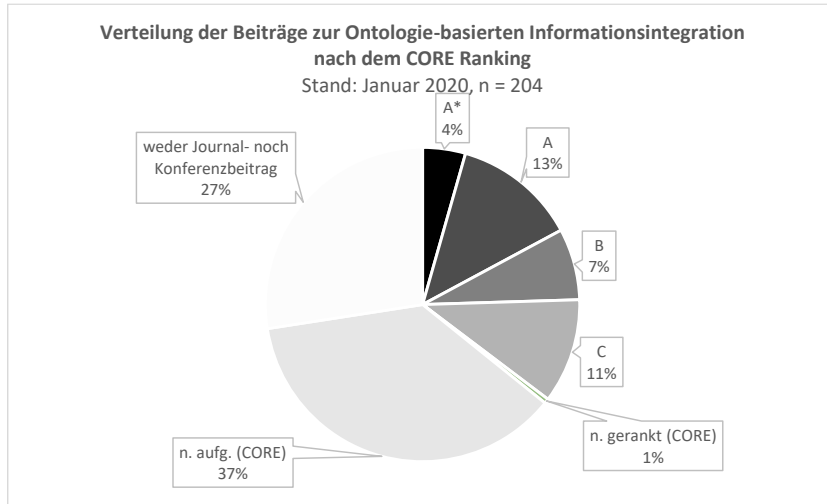


Abb. 3.24: Verteilung der Beiträge nach CORE Ranking

Abb. 3.24 zeigt die Verteilung der Beiträge nach dem CORE Ranking. Etwa 3/4 der Beiträge sind begutachtete Artikel (peer-reviewed), die in einem Journal oder auf einer Konferenz veröffentlicht wurden. Um die Qualität zu beurteilen, wurde das CORE Ranking (Computing Research and Education) gewählt, welches insbesondere in der Informatik verwendet wird. Fast die Hälfte der Konferenzen oder Journals werden von CORE nicht gerankt oder sind dort nicht aufgeführt. Bei den gerankten Bereichen überwiegen die Beiträge, die als A oder C bewertet werden. Dies ist dicht gefolgt von B und die wenigsten Publikationen sind A*. Etwa ein Drittel der Beiträge ist weder ein Journal- noch ein Konferenzbeitrag, sondern beispielsweise ein Buch oder ein Buchkapitel.

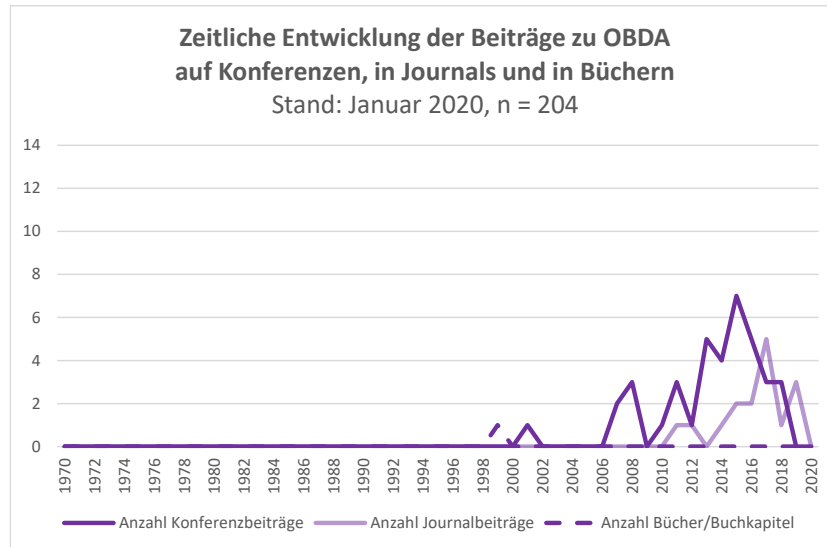


Abb. 3.25: Detaillierte zeitliche Entwicklung der Beiträge zu OBDA

Abschließend wird noch die zeitliche Entwicklung der Beiträge in den einzelnen Fachgebieten bezüglich der Publikationstypen (Konferenz, Journal, Buch) betrachtet (vgl. Abb. 3.25, Abb. 3.26, Abb. 3.27). Im Themengebiet OBDA wurden zunächst Beiträge auf Konferenzen und anschließend in Journals veröffentlicht (vgl. Abb. 3.25). Dieses Vorgehen ist typisch, wenn sich ein Forschungsgebiet neu entwickelt. Bei OBDA liegt der Ursprung in einem Forschungsteam in Italien von wo es sich weiter ausbreitete.

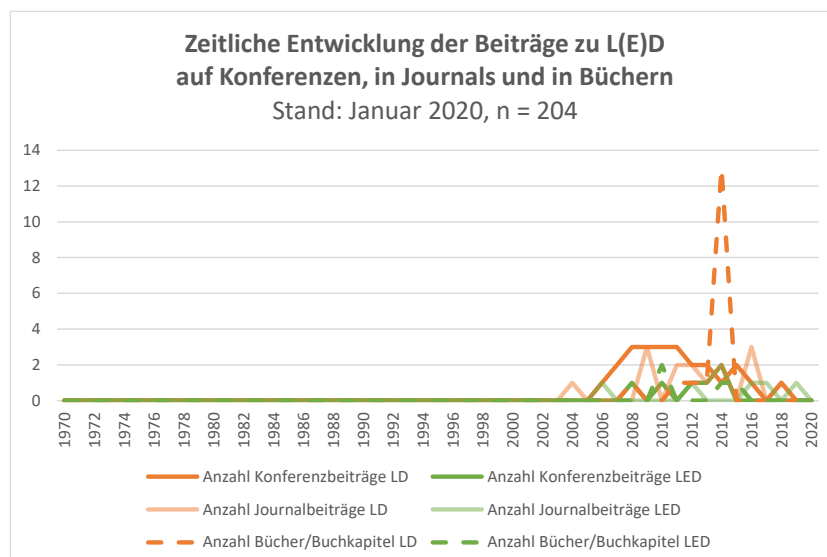


Abb. 3.26: Detaillierte zeitliche Entwicklung der Beiträge zu L(E)D

Eine andere zeitliche Entwicklung der Beiträge lässt sich für L(E)D und (E)KG erkennen (vgl. Abb. 3.26, Abb. 3.27). Es kann keine zeitliche Abfolge der Veröffentlichungen auf Konferenzen hin in Journals identifiziert werden. Bei L(E)D beginnen die Veröffentlichungen nach der Jahrtausendwende und bei (E)KG beginnen diese in den 1970er Jahren und treten über die Jahre hinweg immer wieder auf.

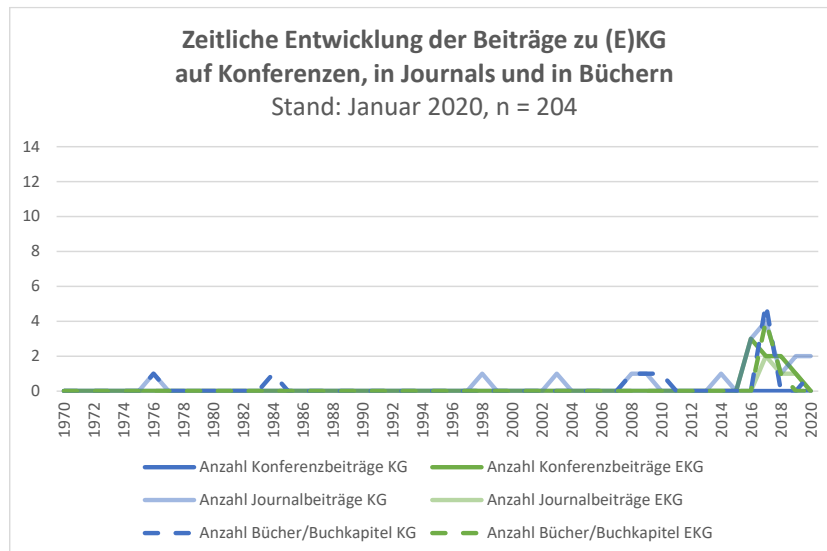


Abb. 3.27: Detaillierte zeitliche Entwicklung der Beiträge zu (E)KG

Dies lässt sich vermutlich darauf zurückführen, dass diese beiden Konzepte von Forschern aus verschiedenen Bereichen unabhängig voneinander entwickelt und immer wieder aufgegriffen wurden. Durch die Standardisierung von Technologien durch das W3C wurde das Erforschen durch verschiedene Forschergruppen auch möglich.

3.2.2.2.3 Zentrale Journals und Konferenzen

Die Beiträge verteilen sich auf zahlreiche verschiedene Konferenzen und Journals. Dennoch konnten einige zentrale Publikationsorte identifiziert werden: Auf der A*-Konferenz International World Wide Web Conference wurden vier Beiträge veröffentlicht und auf der A-Konferenz International Semantic Web Conference zwölf. Auf einer weiteren Konferenz, die nicht im CORE Ranking berücksichtigt wird (International Conference on Semantic Computing), wurden vier Beiträge publiziert. Mehrfach auftretende Journals waren zum einen das IEEE Intelligent Systems-Journal (B) mit fünf und das Journal of Web Semantics-Journal (C) mit neun Beiträgen. Das Semantic-Web-Journal, das nicht durch CORE gerankt ist, veröffentlichte neun Beiträge. Auffallend ist auch hier, dass es sich insbesondere um informatiknahe Konferenzen und Journals im internationalen Bereich handelt.

3.2.2.2.4 Zentrale Autoren

Für die Identifizierung der zentralen Autoren nach Fachgebiet wurde die Anzahl der Beiträge je Autor für jedes Fachgebiet ermittelt. Da sich die Themengebiete inhaltlich zum Teil überschneiden, treten

einige Autoren mehrmals auf. Im Folgenden werden die zentralen Autoren zu den jeweiligen Themengebieten aufgeführt.

Tab. 3.2: Autoren zu Ontology-Based Data Access

Name	Institution	Anzahl der Beiträge
Calvanese, Diego	Free University of Bozen-Bolzano	15
Rodríguez-Muro, Mariano	Free University of Bozen-Bolzano, Google	10
Kharlamov, Evgeny	Bosch Center for AI, University of Oslo	6
Soylu, Ahmet	SINTEF (unabhängige Forschungsorganisation Norwegens) Digital, Norwegian University of Science and Technology (NTNU)	8

Im Bereich des Themengebietes OBDA (Ontology-Based Data Access) konnten vier zentrale Autoren identifiziert werden (vgl. Tab. 3.2). Die Autoren stammen zum einen aus der Wissenschaft, sind aber auch in Kooperationen mit der Praxis aktiv. Diego Calvanese, Professor an der Universität Bozen, forscht im Bereich Intelligent Integration und Datenzugriff und beschäftigt sich in diesem Zusammenhang mit OBDA, worin mithilfe der Literaturrecherche 15 Beiträge ermittelt wurden. Ein Doktorand dieser Universität, Mariano Rodriguez-Muro, forscht ebenfalls in diesem Bereich und arbeitet mit Google zusammen. Außerdem wurden zwei Professoren von norwegischen Universitäten ermittelt (Evgeny Kharlamov, und Ahmet Soylu), die ebenfalls in Kooperationen mit der Praxis in diesem Bereich forschen und dementsprechend publizieren.

Tab. 3.3: Autoren zu Linked Data

Name	Institution	Anzahl der Beiträge
Bizer, Christian	Universität Mannheim	10
Heath, Tom	Talis (UK)	5
Berners-Lee, Tim	Universität Oxford, MIT, W3C	5
Blumauer, Andreas	Semantic Web Company GmbH, PoolParty Semantic Suite	3
Frischmuth, Philipp	Universität Leipzig	4

Zeitlich relativ parallel (vgl. Abb. 3.21) entwickelte sich das Forschungsgebiet der Linked Data für die Informationsintegration. Dies lässt sich anhand der steigenden Publikationszahlen festmachen, die in Abb. 3.21 dargestellt sind. Auch die Autoren zu Linked Data lassen sich sowohl in der Wissenschaft als auch in der Praxis verorten und stammen aus dem Forschungsgebiet des Semantic Web (vgl. Tab. 3.3). Der wohl bekannteste Autor ist Tim Berners-Lee als Entwickler von HTML, der damit die Grundlage für das World Wide Web schaffte. Hier wird nochmal deutlich, dass die Herausforderung der Informationsintegration und die Adressierung dieser durch Technologien auch im Internet zu finden ist, da dort eine steigende Datenmenge verteilt auf mehreren Webseiten vorhanden ist. Weitere Autoren in diesem Bereich stammen entweder aus der Wissenschaft (z. B. Christian Bizer oder Philipp Frischmuth) oder entwickelten Technologien in Unternehmen (z. B. Andreas Blumauer).

Tab. 3.4: Autoren zu Linked Enterprise Data

Name	Institution	Anzahl der Beiträge
Pellegrini, Tassilo	FH St. Pölten	1
Auer, Sören	Universität Hannover	16
Galkin, Michael	Universität Bonn	5
Sack, Harald	KIT	2

Nach der Entstehung als Forschungsgebiet erhielt Linked Data den Einzug in die Unternehmenswelt (vgl. Tab. 3.4). Dies wurde maßgeblich unterstützt durch die Wissenschaft, indem Tassilo Pellegrini, Harald Sack und Sören Auer einen umfassenden Sammelband zu Linked Enterprise Data herausgaben. Außerdem existieren zahlreiche Beiträge von Auer und Galkin. Sören Auer sticht mit 16 Beiträgen zu Linked Data in Unternehmen besonders hervor.

Tab. 3.5: Autoren zu Knowledge Graph

Name	Institution	Anzahl der Beiträge
Kröttsch, Markus	TU Dresden	3
Weikum, Gerhard	Max-Planck Institut for Informatics; Universität Saarland	1
Villazon-Terrazas, Boris	Artificial Intelligence Research Division	3
Alexopoulos, Panos	Head of Ontology at Textkernel	4

Im Bereich des Knowledge Graphen sind die zwei Professoren, Markus Kröttsch und Gerhard Weikum, auffällig (vgl. Tab. 3.5). Diese sind auch eng mit der Praxis verbunden. Gemeinsam verfassten die Autoren das Editorial zu einer Ausgabe des *Journal of Web Semantics* (CORE Ranking: C) im Jahr 2016 (Kröttsch & Weikum, 2016), in der es einen speziellen Bereich zu Knowledge Graphen gab. Die anderen beiden Autoren (Boris Villazon-Terrazas und Panos Alexopoulos) trugen in einem Sammelband mit dem Titel *Exploiting Linked Data and Knowledge Graphs in Large Organisations* zahlreiche Beiträge zusammen und veröffentlichten diese.

Tab. 3.6: Autoren für Enterprise Knowledge Graph

Name	Institution	Anzahl der Beiträge
Auer, Sören	Universität Hannover	16
Galkin, Mikhail	Universität Bonn	5
Scerri, Simon	Fraunhofer IAIS	4
Gomez-Perez, Jose Manuel	Expert Systems, Madrid	3

Bei der Übertragung des Knowledge Graphen auf Unternehmen stechen drei Autoren heraus, die zum Teil auch bereits bei LED aufgeführt waren: Sören Auer, Michael Galkin und Simon Scerri. Galkin und Scerri gehören dem Forschungszentrum Smart Data Analytics (SDA) an, deren Fokus auf der Erforschung von Machine-Learning-Techniken liegt, indem sie strukturiertes Wissen („smart data“) nutzen.

SDA arbeitet eng mit der Universität Bonn und dem Institut für angewandte Informatik Leipzig zusammen. Die Gruppe hat das Ziel, das volle Spektrum der Forschung abzudecken, welches Theorie, Algorithmen, Prototypen und Unternehmensanwendungen beinhaltet. Ein weiterer Autor (Jose Manuel Gomez-Perez) fiel in diesem Themengebiet auf, der mit seinen Beiträgen nicht so dominant ist wie die zuvor genannten, aber auch relevante Beiträge veröffentlichte.

Tab. 3.7 gibt einen Gesamtüberblick über die genannten zentralen Beiträge.

Tab. 3.7: Zentrale Beiträge

Forschungsgebiet	Autoren & Jahr	CORE Ranking	Art des Beitrags
OBDA	D. Calvanese, B. Cogrel, S. Komla-Ebri, D. Lanti, M. Rezk, and G. Xiao, "How to Stay Ontop of Your Data: Databases, Ontologies and More," in <i>The Semantic Web: ESWC (European Semantic Web Conference) 2015 Satellite Events</i> , 2015, pp. 20–25. (Calvanese et al., 2015)	A	Konferenzbeitrag
OBDA	E. Kharlamov et al., "OPTIQUE: Ontology-Based Data Access Platform," in <i>Proceedings of the International Semantic Web Conference 2015 Posters & Demonstrations Track</i> , 2015. (Kharlamov, Jimenez-Ruiz, et al., 2015)	A	Konferenzbeitrag
OBDA	H. Wache et al., "Ontology-Based Integration of Information — A Survey of Existing Approaches," in <i>International Joint Conference on Artificial Intelligence (Workshop: Ontologies and Information Sharing)</i> , 2001, pp. 108–117. (Wache et al., 2001)	A*	Konferenzbeitrag
LD	C. Bizer, "The Emerging Web of Linked Data," <i>IEEE Intell. Syst.</i> , vol. 24, no. 5, pp. 87–92, 2009. (Bizer, 2009)	B	Journalbeitrag
LD	P. Frischmuth et al., "Linked Data in Enterprise Information Integration," <i>Semant. Web</i> , 2012. (Frischmuth et al., 2012)	k. A.	Journalbeitrag
LD	T. Heath and C. Bizer, <i>Linked Data: Evolving the Web into a Global Data Space</i> . Morgan & Claypool, 2011. (Heath & Bizer, 2011)	/	Buch
LED	S. Auer, R. Petzsch, and J. Unbehauen, "Datenintegration im Unternehmen mit Linked Enterprise Data," in <i>Linked Enterprise Data</i> , T. Pellegrini, H. Sack, and S. Auer, Eds. Berlin, Heidelberg: Springer, 2014, pp. 85–101. (Auer et al., 2014)	/	Buchkapitel
LED	M. Galkin, S. Auer, and S. Scerri, "Enterprise Knowledge Graphs: A Backbone of Linked Enterprise Data," in <i>IEEE/WIC/ACM International Conference on Web Intelligence</i> , 2016, pp. 497–502. (Galkin, Auer, & Scerri, 2016)	k. A.	Konferenzbeitrag
LED	B. Hu and G. Svensson, "A Case Study of Linked Enterprise Data," in <i>9th International Semantic Web Conference</i> , 2010, pp. 129–144. (Hu & Svensson, 2010)	A	Konferenzbeitrag
KG	M. Krötzsch and G. Weikum, "Editorial," <i>J. Web Semant.</i> , vol. 37–38, pp. 53–54, Mar. 2016. (Krötzsch & Thost, 2016)	A	Konferenzbeitrag
KG	M. Krötzsch and V. Thost, "Ontologies for Knowledge Graphs: Breaking the Rules," in <i>15th International Semantic Web Conference</i> , 2016, pp. 376–392. (Krötzsch & Weikum, 2016)	C	Journalbeitrag
KG	J. Pujara, H. Miao, L. Getoor, and W. Cohen, "Knowledge Graph Identification," in <i>12th International Semantic Web Conference</i> , 2013, pp. 542–557. (Pujara, Miao, Getoor, & Cohen, 2013)	A	Konferenzbeitrag
EKG	M. Galkin, S. Auer, H. Kim, and S. Scerri, "Integration Strategies for Enterprise Knowledge Graphs," in <i>IEEE Tenth International Conference on Semantic Computing (ICSC)</i> , 2016, pp. 242–245. (Galkin, Auer, Kim, et al., 2016)	k. A.	Konferenzbeitrag
EKG	M. Galkin, S. Auer, and S. Scerri, "Enterprise Knowledge Graphs: A Backbone of Linked Enterprise Data," in <i>IEEE/WIC/ACM International Conference on Web Intelligence</i> , 2016, pp. 497–502. (Galkin,	k. A.	Konferenzbeitrag

Forschungsgebiet	Autoren & Jahr	CORE Ranking	Art des Beitrags
	Auer, & Scerri, 2016)		
EKG	M. Galkin, S. Auer, M.-E. Vidal, and S. Scerri, "Enterprise Knowledge Graphs: A Semantic Approach for Knowledge Management in the Next Generation of Enterprise Information Systems," in <i>19th International Conference on Enterprise Information Systems</i> , 2017, pp. 88–98. (Galkin, Auer, Vidal, & Scerri, 2017)	C	Konferenzbeitrag
EKG	J. M. Gomez-Perez, J. Z. Pan, G. Vetere, and H. Wu, "Enterprise Knowledge Graph: An Introduction," in <i>Exploiting Linked Data and Knowledge Graphs in Large Organisations</i> , J. Z. Pan, G. Vetere, J. M. Gomez-Perez, and H. Wu, Eds. Cham: Springer, 2017, pp. 1–15. (Gomez-Perez, Pan, Vetere, & Wu, 2017)	/	Buchkapitel

3.2.2.2.5 Zusammenfassung

Zusammenfassend lässt sich anhand der quantitativen Auswertung der Literatur zu den verschiedenen Forschungsgebieten zur Informationsintegration festhalten, dass im Zusammenhang mit der immer größer werdenden Datenmenge in der Informatik Technologien entwickelt wurden, um auf diese „Big Data“ zuzugreifen. Dazu werden zum einen neue Datenbank-Technologien (wie z. B. Graphdatenbanken) eingesetzt, um die Daten beispielsweise mittels Linked Data oder Knowledge Graphen so abzuspeichern, dass der Datenzugriff erleichtert wird. Außerdem werden neue Technologien eingesetzt, um auf die Daten mittels Ontologien (z. B. OBDA) zuzugreifen. Auf diese Weise werden Ansätze wie EAI (Enterprise Application Integration) oder SOA (Service-Oriented Architecture), die statische Lösungen für Datenzugriffe für einen vorgegebenen Anwendungsfall bieten, durch flexiblere Ansätze für situativen Datenzugriff ergänzt. Die Entwicklung von Standards vom W3C war ein wichtiger Meilenstein. Die Semantic-Web-Technologien erhielten in den letzten Jahren Einzug in die Unternehmenswelt, da dort ein Bedürfnis für einen flexiblen Informationszugriff vorliegt, was sich daran erkennen lässt, dass die Publikationen zu Linked *Enterprise* Data und *Enterprise* Knowledge Graphen anstiegen. Die angesprochenen Ansätze stammen zwar aus der Wissenschaft, sind aber direkt in der Praxis anwendbar. Dies macht sich auch bei den Autoren bemerkbar, die oft in beiden Bereichen aktiv sind.

Im Folgenden werden die angesprochenen Themengebiete auf der Basis der quantitativen Analyse inhaltlich dargestellt und die Entwicklung skizziert. Abschließend werden die Gebiete miteinander in Zusammenhang gesetzt und es wird eine Kategorisierung vorgenommen, die für die Dissertation relevant ist (vgl. Abschnitt 3.2.2.6).

3.2.2.3 OBDA – Ontology-Based Data Access

Unter OBDA wird eine Überlagerung der üblichen Datenschicht eines Informationssystems mit einer konzeptionellen Schicht verstanden, die im Client zur Verfügung gestellt werden soll. Eine solche Schicht ermöglicht es dem Nutzer, eine konzeptionelle Sicht auf die Informationen im System zu erhalten. Diese Sicht abstrahiert die Art und Weise, wie diese Informationen in der Datenschicht des Sys-

tems selbst verwaltet werden. Während Ontologien für die Realisierung der konzeptionellen Schicht geeignet sind, werden relationale Datenbankmanagementsysteme in der Regel für die Verwaltung der Datenschicht verwendet (Calvanese et al., 2007). Es gibt allerdings auch jüngere Beiträge, die das Konzept OBDA auf NoSQL-Datenbanken (z.B. Mongo-DB) ausweiten (Araujo, Avena, Braghetto, & Wassermann, 2017).

Im Folgenden wird zunächst der Ursprung von OBDA dargestellt und anschließend der Aufbau des Konzeptes erklärt. Daraufhin wird die Konstruktion eines OBDA-Systems in der Literatur aufgeführt und abschließend werden konkrete Beispiele gelistet.

3.2.2.3.1 Ursprung von OBDA

Der semantische bzw. Ontologie-basierte Zugriff auf Datenbanken hat eine lange Geschichte und entstand in frühen Stadien der Entwicklung der Datenbanktechnologie (Gagnon, 2007; Kogalovsky, 2012; Wache et al., 2001). Erst mit der Schaffung von Standards für Ontologiebeschreibungssprachen durch das W3C wurde eine neue Aktivitätswelle bei der Entwicklung von Werkzeugen für die Systeme des semantischen Zugriffes auf Datenbanken sowie eine neue Klasse von Datenbanksystemen (=OBDA-Systeme) ausgelöst. In diesen Systemen wird die Ontologie als konzeptionelles Schema der jeweiligen Domäne und als Grundlage der Benutzerschnittstelle für SQL-Datenbanken verwendet (Kogalovsky, 2012). Der Begriff OBDA schließlich stammt von einer Gruppe von italienischen Forschern, die diesen Begriff mit der Publikation von Calvanese et al. (2007) einführten. Sie entwickelten eine Familie von Beschreibungslogiken (DL-Lite) für die Ontologiebeschreibung, wobei eine von ihnen (DL-Lite_R) dem OWL2-Profil unterliegt, und führten in diesem Zusammenhang OBDA ein. DL-Lite wird zum einen zur Beschreibung von Ontologien und zum anderen zur Umsetzung von Datenintegration und von OBDA-Systemen verwendet (Calvanese et al., 2007). Für diese Art der Integration werden auch folgende Begriffe verwendet: *accessing data mediated by an ontology*, *ontology-driven information systems* und *ontology based data management* (Kogalovsky, 2012). OBDA ist mittlerweile in der Theorie weitgehend erforscht. Allerdings gibt es noch Bedarf an Konzepten, wie OBDA-Systeme effektiv in der Praxis eingesetzt werden können (Sequeda, 2017).

3.2.2.3.2 Aufbau von OBDA

Im OBDA-Paradigma definiert eine Ontologie ein globales Schema und stellt ein Vokabular für Benutzerabfragen bereit. Dadurch wird der Nutzer von der Struktur der Datenquellen (z.B. relationale Datenbanken, Triplestore, Datenprotokoll-Engines usw.) isoliert. Das OBDA-System wandelt Benutzerabfragen (auf die Ontologie) in das Vokabular der Daten um und delegiert dann die eigentliche Abfrageauswertung an die Datenquellen (in den Quellsystemen). Der Zugriff auf die Daten wird also für den Nutzer erleichtert, indem dieser von den Rohdatenquellen mithilfe einer Ontologie getrennt wird, die eine benutzerorientierte Sicht der Daten bietet und somit ein Kenntnis der Daten in den Datenbanken nicht nötig ist. Ein traditionelles Datenbanksystem wird durch OBDA in eine Wissensbasis mit einer mehrschichtigen Architektur umgewandelt, die dem Prinzip der Datenunabhängigkeit entspricht

(Kontchakov, Lutz, Toman, Wolter, & Zakharyashev, 2011; Rodríguez-Muro, Kontchakov, & Zakharyashev, 2013).

Die allgemeine Definition von OBDA besagt, dass bei einer relationalen Quelldatenbank, einer Zielontologie und einer Zuordnung von der relationalen Datenbank zur Ontologie (= Mapping) das Ziel darin besteht, mithilfe dieser drei Komponenten Fragen über die Zielontologie zu beantworten (Sequeda, 2017). Der Ontologie-basierte Datenzugriff (OBDA) ist also ein Ansatz zum Abfragen relationaler Daten über einen einheitlichen semantischen Zugriffspunkt. Dieser wird von einer Ontologie gesteuert, die über Zuordnungen (= Mapping) mit den zugrundeliegenden Datenbanken (auch: Quelldatenbanken) verbunden ist (Kharlamov, Jimenez-Ruiz, et al., 2015). Abb. 3.28 zeigt die Architektur in dem OBDA Paradigma, die von den italienischen Forschern vorgeschlagen wurde, die den Begriff 2007 prägten. Auf der linken Seite werden die beiden Ebenen ersichtlich, indem die Quellsysteme als *Data Layer* und die Ontologie als *Conceptual Layer* bezeichnet werden. In der rechten Abbildung wird die Abbildung um das Mapping ergänzt, das bei OBDA außerdem erforderlich ist.

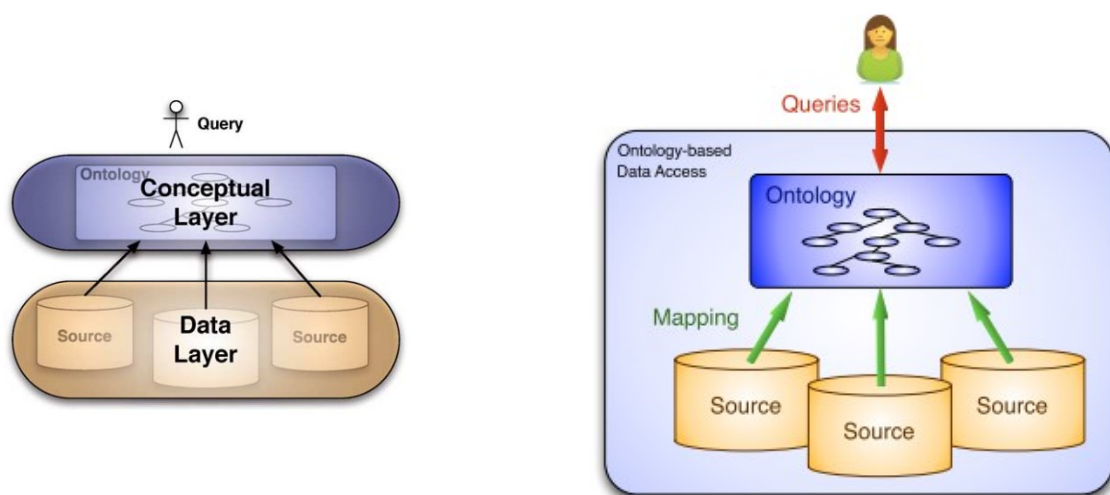


Abb. 3.28: Ontology-Based Data Access (Calvanese et al., 2011, p. 43, 2015, p. 2)

Die Motivation für OBDA besteht in der Bereitstellung von High-level-Services für den Nutzer des Informationssystems. Der wichtigste Dienst ist die Beantwortung von Fragen. Anwender stellen ihre Anfragen an die Ontologie und das System übersetzt die Anfrage mithilfe eines zuvor angefertigten Mappings in für das Quellsystem passende Queries (Poggi et al., 2008). Üblicherweise werden OBDA-Systeme auf der Grundlage von zwei Ansätzen implementiert: Materialisierung oder Virtualisierung. Bei der *Materialisierung* werden die Quelldatenbanken, die Zielontologie und das Mapping verwendet, um neue Fakten als RDF abzuleiten, welche anschließend in einem Triplestore gespeichert werden. Somit handelt es sich um eine redundante Datenspeicherung, auch wenn das Format sich unterscheidet. SPARQL-Abfragen werden dann im Dreifachspeicher ausgewertet. Beim *Virtualisierungsansatz* werden SPARQL-Abfragen unter Berücksichtigung des Mappings und der Zielontologie in SQL-

Abfragen umgeschrieben. Somit handelt es sich hier um eine *nicht* redundante Datenspeicherung. Wenn mehrere Quelldatenbanken beteiligt sind, können die Abfragen zusammengefasst werden. Außerdem können OBDA-Systeme auch in einer Mischung aus Materialisierungs- und Virtualisierungsansatz implementiert werden (Sequeda & Miranker, 2017).

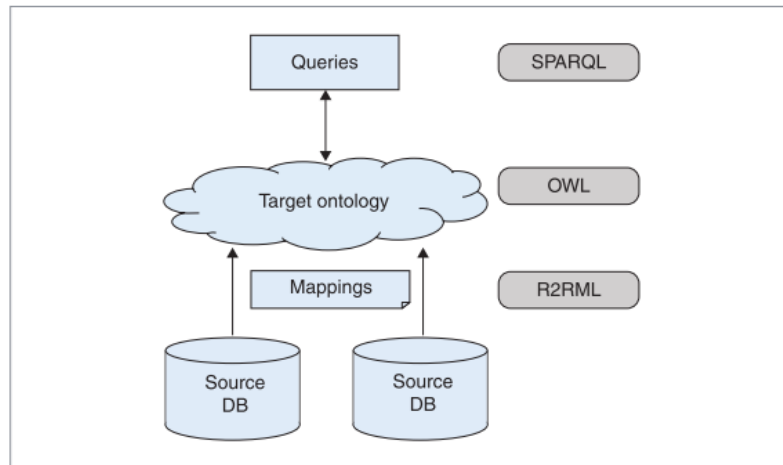


Abb. 3.29: OBDA Paradigma (Sequeda & Miranker, 2017, p. 93)

Abb. 3.29 zeigt das OBDA-Paradigma mit den verwendeten Sprachen. Die Queries werden in SPARQL (vgl. Abschnitt 3.2.1.1.2) auf die Ontologie abgesetzt, die wiederum beispielsweise mit der Ontologiebeschreibungssprache OWL (vgl. Abschnitt 3.2.1.1.1) beschrieben ist. Das Mapping wird mit einem R2RML-Mapping erstellt (vgl. Abschnitt 3.2.1.1.3) und die Query wird auf diese Weise beispielsweise in eine SQL-Abfrage übersetzt.

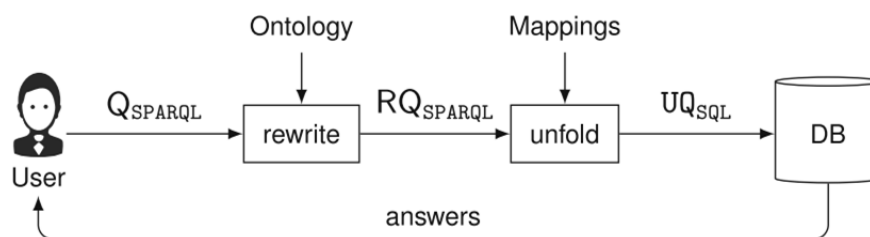


Abb. 3.30: Query processing in OBDA (Kharlamov, Hovland, et al., 2015, p. 97)

In Abb. 3.30 ist der Abfrageprozess detaillierter abgebildet. Hier befindet sich auf der linken Seite der Nutzer und auf der rechten Seite die Quelldatenbank, auf welche der Nutzer über die Ontologie und das Mapping zugreift. Weitere Ausführungen für den Datenzugriff sind bei Soylu et al. (2017, 2016) beschrieben. Außerdem gibt es noch zahlreiche weitere Artikel zu den entwickelten Sprachen

(Calvanese et al., 2007; Kontchakov et al., 2011; Kontchakov, Rodríguez-Muro, & Zakharyashev, 2013; Rodríguez-Muro & Calvanese, 2011).

3.2.2.3.3 Konstruktion von OBDA

Wie bereits erwähnt, ist das Konzept theoretisch umfangreich erforscht, wobei es noch immer Bedarf an Umsetzungen in der Praxis gibt (Sequeda, 2017). Dazu gehört insbesondere die Einführung solcher Systeme in Unternehmen. Bei der Nutzung eines OBDA-Systems geht es vor allem darum, zwei der drei Komponenten eines OBDA-Systems (relationalen Quelldatenbank, Zielontologie und Mapping = Zuordnung von der relationalen Datenbank zur Ontologie) (Sequeda, 2017) auf das jeweilige Unternehmen anzupassen: Zielontologie und Mapping (Zuordnung der Informationen in der Ontologie zu Informationen in der Quelldatenbank). Der erste Schritt bei der Erstellung eines OBDA-Systems ist dabei nach Kogalovsky (2012) die Entwicklung der Ontologie der Subjektdomäne (z. B. konkretes Unternehmen), deren Informationen beispielsweise im SQL-Datenbanksystem⁸ gespeichert sind. Im nächsten Schritt wird dann das Mapping vorgenommen.

Die Herausforderungen der Ontologie-Erstellung sind bereits in der klassischen Ontologie-Erstellung beschrieben (Uschold & Gruninger, 1996). Dort wird vorgeschlagen, vorhandene Ontologien zu verwenden, da der Aufwand der Erstellung einer Ontologie sehr hoch ist (vgl. Abschnitt 3.2.1.2.3). Existierende Ontologien sind beispielsweise Good Relations für E-Commerce (Hepp, 2008), FIBO für Finanzen (Zaino, 2017), Gist für allgemeine Geschäftskonzepte (Semantic Arts, 2019) oder schema.org für allgemeine Informationen (Nogales, Sicilia, García-Barriocanal, & Sánchez-Alonso, 2013). Eine weitere Möglichkeit, eine erste Ontologie als Ausgangsbasis zu erstellen, sind *putative ontologies*. Dabei wird mithilfe eines Bootstraps aus dem Quelldatenbankschema eine Ontologie abgeleitet. Diese kann dann in eine Zielontologie umgewandelt werden (Sequeda, 2017). Für den nächsten Schritt – das Mapping zwischen der Zielontologie und der Datenbank – existieren verschiedene Standards. Beispielsweise kann mit dem W3C Direct Mapping Standard das Mapping gebootet werden. Außerdem gibt es die R2RML-Zuordnungssprache (vgl. Abschnitt 3.2.1.1.3), bei welcher der Benutzer angeben kann, welche Elemente der Quelldatenbank mit der Zielontologie verbunden sind ohne Code zu erstellen. Trotz der Tools ist ein standardisiertes Vorgehen zur Einführung eines OBDA-Systems in der Praxis noch weitestgehend unerforscht. Auch nach Sequeda (2017) wirft das theoretische Konstrukt von OBDA aus praktischer Sicht die Frage auf, woher die Zielontologie und die Mappings kommen und schlägt dafür eine Vorgehensweise (Pay-as-you-go Methodology, vgl. Abb. 3.31) vor (Sequeda, 2017; Sequeda &

⁸ In der Literatur werden insbesondere relationale Datenbanken (oder auch: SQL-Datenbanken) aufgeführt, weshalb diese hier genannt werden. Es gibt auch jüngere und vereinzelt Beiträge, die auf andere Arten von Datenbanken zugreifen. Diese stellen aber nach wie vor nur einen kleinen Teil dar.

Miranker, 2017). Die Vorgehensweise wurde in einem als B geranktes⁹ Journal (IEEE Internet Computing) veröffentlicht.

Bei der Pay-as-you-go-Methodology geht es um eine Vorgehensweise, um die Zielontologie und das Mapping zu erstellen. Dazu werden Geschäftsfragen und auf dieser Basis dann die Zielontologie und das Mapping erstellt, welche die Antworten auf die zu Beginn erstellten priorisierten Geschäftsfragen liefern. Nachdem eine kleine Anzahl von Geschäftsfragen erfolgreich modelliert, zugeordnet und beantwortet wurde, wird die nächste Gruppe von Fragen behandelt. Die neuen Fragen wiederum können ggf. die Zielontologie erweitern und neue Zuordnungen werden sukzessive hinzugefügt. Auf diese Weise werden die Zielontologie und das Mapping in einem iterativen Prozess entwickelt. Es handelt sich also um eine agile Methode, Daten mithilfe von OBDA zu integrieren, da der Endanwender frühzeitig miteingebunden wird (Sequeda & Miranker, 2017). In den Prozess sind drei Akteure involviert: Der (1) *Business User* ist der Fachexperte, der über Kenntnisse der Geschäftsprozesse verfügt und die Liste der priorisierten Geschäftsfragen identifizieren kann. Der (2) *IT-Entwickler* ist eine Person, die Kenntnisse über die Datenbanken besitzt und weiß, wie die Daten miteinander verbunden sind. Der (3) *Wissensingenieur* ist eine Person, die als Kommunikationsbrücke zwischen den Endanwendern und den IT-Entwicklern fungiert und über Erfahrung in der Modellierung von Daten mithilfe von Ontologien verfügt (Sequeda & Miranker, 2017).

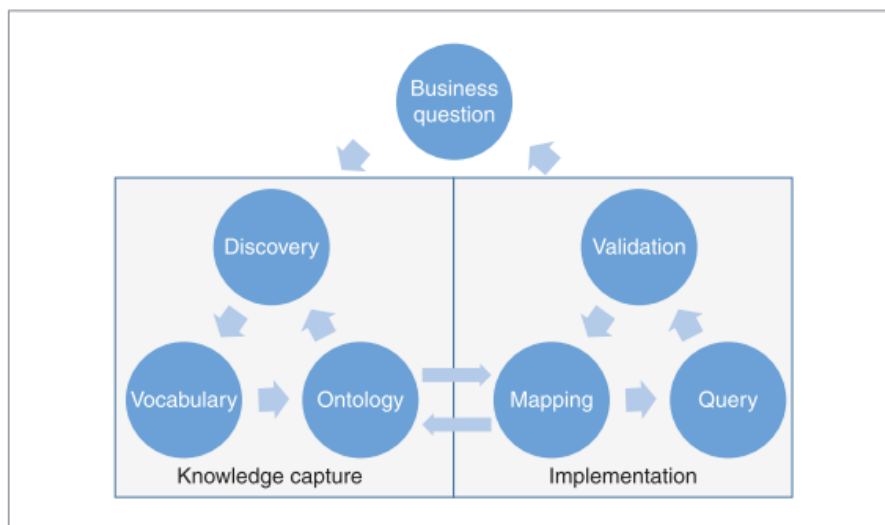


Abb. 3.31: Pay-as-you-go Methodology für OBDA (Sequeda & Miranker, 2017, p. 94)

Die Methodology ist in zwei Phasen unterteilt: In der (1) ersten Phase (knowledge capture) geht es zum einen um die Identifizierung der Schlüsselkonzepte und Beziehungen aus den erhobenen Ge-

⁹ CORE Ranking

schäftsfragen. Der Wissensingenieur arbeitet dabei mit den Business Usern zusammen. Außerdem besteht das Ziel in der Ermittlung der notwendigen Daten in den Quelldatenbanken für die erhobenen Konzepte und Beziehungen. Der Wissensingenieur ermittelt auf der Basis der Geschäftsfragen und den identifizierten Konzepten und Beziehungen gemeinsam mit dem IT-Entwickler, welche Tabellen und Attribute erforderlich sind. Diese beiden Ziele werden durch die drei Schritte *discovery*, *vocabulary* und *ontology* erreicht. Im Discovery-Schritt wird die Ermittlung der Konzepte und Beziehungen auf Basis der Geschäftsfragen durchgeführt und es wird untersucht, wie die Konzepte und Beziehungen in den Datenbanken verbunden sind. Durch den Vocabulary-Schritt wird ermittelt, welche Bezeichnungen und Definitionen für die Konzepte und Beziehungen genutzt werden (können). Im Ontology-Schritt wird die Ontologie in OWL übersetzt, sodass die Geschäftsfragen darüber beantwortet werden können. Ein von Sequeda & Miranker (2017) gegebenes Beispiel verdeutlicht die erste Phase: Beispielsweise hat der Wissensingenieur aus den Geschäftsfragen das Konzept *Bestellung* erhoben. In Zusammenarbeit mit den IT-Entwicklern identifiziert er in der Datenbank das *Order Management* als maßgebliche Quelle der Daten, die für die Bestellung notwendig sind. Innerhalb dieser Tabelle können die Daten (insbesondere die Attribute), die sich auf Bestellungen beziehen, identifiziert werden. Mithilfe des Vokabulars wird dann die Ontologie erstellt (Sequeda & Miranker, 2017).

In der (2) zweiten Phase (Implementation) soll die Beantwortung der Geschäftsfragen ermöglicht werden. Dafür wird die Ontologie mit den Daten verknüpft, indem der Wissensingenieur die Erkenntnisse aus den vorherigen Schritten nutzt und das Mapping in R2RML implementiert. Die Geschäftsfragen werden als SPARQL-Abfragen mit der in der Zielontologie definierten Geschäftsterminologie implementiert. Für das R2RML-Mapping erfolgt die Eingabe in ein OBDA-System, das die Ausführung der SPARQL-Abfragen ermöglicht. Anschließend werden die Ergebnisse mit den Business Usern validiert. Das Mapping ist in drei Schritte unterteilt: *mapping*, *query* und *validation*. Im Mapping-Schritt wird das Mapping durch R2RML implementiert auf der Basis der vorher erstellten Ontologie. Dies wird dann zum Einrichten des OBDA-Systems verwendet. Mit dem Query-Schritt werden die Geschäftsfragen als SPARQL-Abfragen unter Verwendung der Terminologie der Zielontologie implementiert. Die Antworten auf diese Fragen sind SPARQL-Ergebnisse. Im Validation-Schritt wird sichergestellt, dass die SPARQL-Abfragen die richtigen Antworten zurückgeben. Für die Erklärung der Schritte wird wieder das Beispiel der Bestellung herangezogen. Nach der ersten Phase erhob der Wissensingenieur beispielsweise, dass die Business User eine Bestellung als „Bestellung“ betrachten, sobald sie verschickt wurde. Der Wissensingenieur (also die Person, die das Mapping R2RML schrieb) identifizierte diese Anforderung an eine Bestellung als Tupel in der Tabelle „Masterorder“, wobei der Bestellstatus dafür gleich 2 oder 3 ist. Deshalb existiert eine R2RML-Zuordnung von der Ontologie zu der SQL-Abfrage `SELECT*FROM MASTERORDER WHERE Bestellstatus IN (2,3)`. Für die Ermittlung des Nettoumsatzes einer Bestellung ist eine mathematische Formel notwendig, die auch in einer SQL-Abfrage dargestellt werden kann. Wenn die Validierung erfolgreich war, wird der iterative Prozess von neuem angestoßen, indem neue Geschäftsfragen formuliert werden. Bei einer nicht erfolgreichen Validierung werden die Ontologie und das Mapping nochmal überarbeitet, bis das gewünschte Ergebnis erreicht ist (Sequeda & Miranker, 2017). Elemente dieses Vorgehens (z. B. Einbezug der Business User, Erstellung

von Konzepten und Beziehungen aus den Erhebungen bei den Business Usern) werden für das SoNBO-Framework (vgl. Kapitel 5) verwendet.

Nach den Autoren dieser Pay-as-you-go-Methodology ist das Erstellen der Ontologie und des Mappings ein offener Forschungsbereich. Die R2RML-Zuordnungen können in RDF dargestellt und somit in einem Triple-Store gespeichert und abgefragt werden. Außerdem kann damit auch Reasoning (vgl. Abschnitt 3.2.1.2.1) durchgeführt werden. Somit kommen einem OBDA-System und damit verbundenen Methoden, die die Einführung unterstützen, eine relevante Rolle zu. Darüber hinaus sind Tools erforderlich, die die Modellierung, Steuerung und Integration von Daten mithilfe von OBDA unterstützen (Sequeda & Miranker, 2017).

3.2.2.3.4 Beispiele für OBDA

Es gibt bereits verschiedene Beispielumsetzungen für OBDA. Die Forschergruppe, die den Begriff mit ihrer Publikation 2007 maßgeblich prägte (Calvanese et al., 2007), entwickelte u.a. das System Ontop für eine softwaretechnische Umsetzung von OBDA (Bagosi et al., 2014; Calvanese et al., 2015, 2017; Kharlamov, Hovland, et al., 2015; Rodríguez-Muro et al., 2013). Eine weitere Umsetzung erfolgte im Rahmen des von der EU finanzierten Projekts Optique (Giese et al., 2015; Kharlamov et al., 2013; Kharlamov, Jimenez-Ruiz, et al., 2015; Soylu et al., 2015, 2018, 2016; Soylu & Giese, 2016). Weitere Beispiele, auch zum Teil von vor 2007, sind Ontomongo (Araujo et al., 2017), Mastro (Calvanese et al., 2011) und ontobroker (Decker, Erdmann, Fensel, & Studer, 1999). Im Folgenden werden Ontop und Optique vorgestellt.

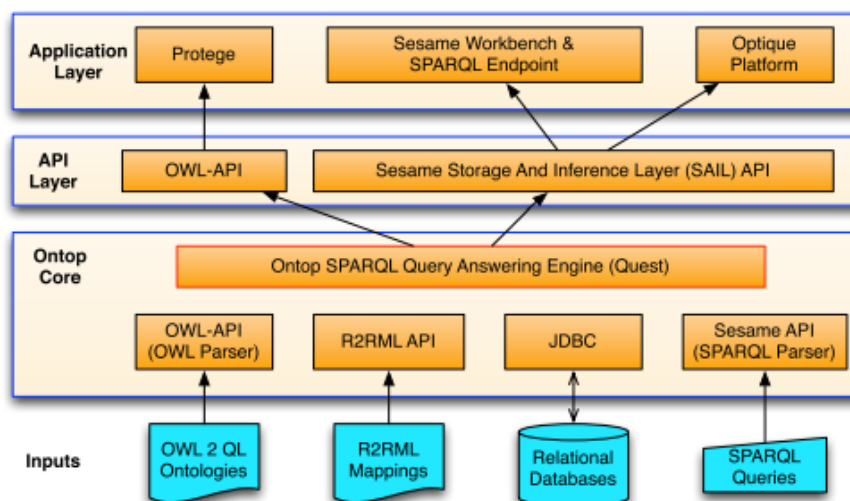


Abb. 3.32: Ontop (Calvanese et al., 2015, p. 2, 2017, p. 3)

Ontop ist eine Plattform, die das OBDA-Paradigma umsetzt. Benutzer können relationale Datenbanken als virtuelle RDF-Graphen mithilfe von SPARQL über eine Domänenontologie abfragen. Diese ist im

OWL-2-QL-Fragment der OWL-2-Ontologiebeschreibungssprachen ausgedrückt (Calvanese et al., 2015). Diese Elemente lassen sich auch in der Abbildung Abb. 3.32 finden. Im unteren Bereich befindet sich das Quellsystem als relationale Datenbank. Weitere Inputs sind OWL 2 QL zur Beschreibung der Zielontologie, R2RML zur Erstellung des Mappings und SPARQL zur Erstellung der Queries auf die Ontologie. Diese befinden sich im Ontop Core. Im API Layer wird *Sesame Storage And Inference Layer (SAIL) API* verwendet, sodass im Application Layer über die Optique Plattform auf die Daten in den relationalen Datenbanken zugegriffen werden kann.

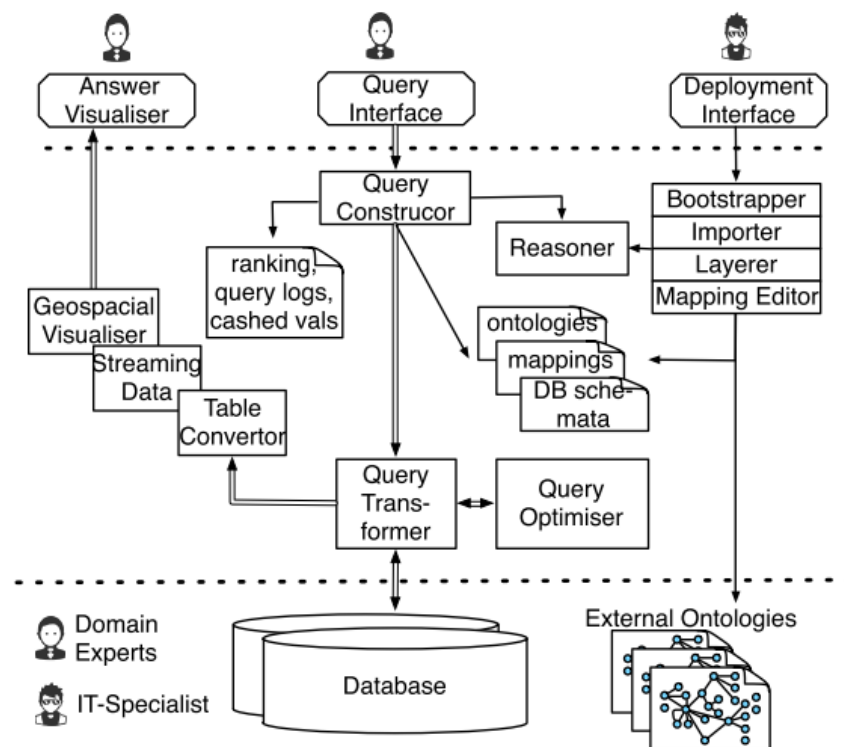


Abb. 3.33: Optique (Kharlamov, Jimenez-Ruiz, et al., 2015, p. 2)

Die Drei-Schichten-Architektur bei Optique ist in Abb. 3.33 zu sehen. Doppelte Pfeile repräsentieren einen Datenfluss und einfache Pfeile kennzeichnen eine Abhängigkeit zwischen Komponenten. Bei Optique wird das Einbinden von *externen* Ontologien zugelassen, die im unteren Bereich neben den Quelldatenbanken abgebildet sind. Dies wird vom IT-Spezialisten durchgeführt, welcher auch für die Ontologie-Erstellung, das Mapping und das Einrichten des Reasoners zuständig ist. Der Domänenexperte nutzt das Query Interface, um über den Abfragekonstruktor eine Query zu formulieren. Mithilfe der Ontologie wird diese Abfrage auf die Quelldatenbank gesetzt und die Antwort wird im *Answer Visualiser* angezeigt (Kharlamov, Jimenez-Ruiz, et al., 2015).

3.2.2.4 LED – Linked Enterprise Data

Linked Data basiert auf RDF und OWL und ist ein gemeinsames Framework zum Veröffentlichen und Freigeben von Daten über verschiedene Anwendungen hinweg. Im Linked-Data-Paradigma können RDF-Graphen mithilfe von Zuordnungen miteinander verknüpft werden. Diese Zuordnung geschieht sowohl auf Schemaebene (subClassOf) als auch auf Objektebene (sameAs) (Gomez-Perez et al., 2017). Linked Enterprise Data besitzt zwei Vorteile, die auch im Unternehmenskontext wirken: Zum einen kann damit eine Datenintegration umgesetzt werden. Dabei wird das Linked-Data-Paradigma aus dem Web herangezogen, sodass ein unternehmensinternes Daten-Intranet dazu beitragen kann, die Brücke zwischen semi- und unstrukturierten Informationen (wie z. B. Dokumente, Wikis, Portale) zu schlagen (Auer, 2016; Auer et al., 2014; Frischmuth et al., 2013; Heath & Bizer, 2011; Hu & Svensson, 2010). Zum anderen können externe Daten z. B. aus der Linked Open Data Cloud integriert werden (Auer et al., 2014). Im Folgenden wird die historische Entwicklung von Linked Data skizziert. Anschließend wird das Konzept und die Konstruktion von Linked Data erklärt. Abschließend werden Beispiele aufgeführt.

3.2.2.4.1 Ursprung von Linked (Enterprise) Data

Seit der Entstehung des World Wide Webs war über die folgenden Jahre eine stetige Weiterentwicklung der zugrundeliegenden Standards und darauf aufbauender Technologien und Anwendungen zu erkennen. Während Anfang der 1990er noch insbesondere statische Webseiten existieren, entstehen einige Jahre später dynamische Webanwendungen, Content Management Systeme und E-Commerce-Systeme, die Inhalte aus Datenbanken und anderen strukturierten Datenquellen beziehen. Mit dem Web 2.0 um die Jahrtausendwende gewinnen vor allem soziale Funktionen in Web-Technologien und Web-Anwendungen an Bedeutung (Auer et al., 2014). Mit der Publikation von Tim Berners-Lee (2006) zu Linked Data und schließlich zum Semantic Web (Berners-Lee, 2008) kann ein Wandel zum Web der Daten beobachtet werden. Die bis dahin üblichen HTML-Seiten sind zwar für den Menschen lesbar, die darin enthaltenen Daten aber maschinell in dieser Form kaum nutzbar. Die Idee neuerer Ansätze ist, dass Daten in *strukturierter* Form mittels RDFa, Microdata oder RDF/Linked Data veröffentlicht werden, was die darin enthaltenen Daten maschinell auslesbar macht. Dieser Wandel kann anhand verschiedener Beispiele festgemacht werden wie z. B. der Initiative schema.org (von Google, Bing, Yandex und Yahoo vorangetrieben), der Linked Open Data Cloud, Googles Knowledge Graph oder Facebooks Open Graph (Auer et al., 2014).

Dieser Idee folgend existiert seit etwa 2010 (Allemang, 2010) ein Trend, der die Vorteile eines Semantic Web und somit von Linked Data nicht nur für komplexe und offene Systeme wie dem Web sondern auch für spezifische Unternehmensanwendungen nutzt. Dabei sind schlecht formalisierte, aber stark miteinander verknüpfte Daten, die für Nachschlage- und Zugehörigkeitsoperationen besser geeignet sind, beliebter als hoch formalisierte, aber isolierte Datensätze. Die derzeitige Anwendungslandschaft, die sich mehr an Echtzeit und Mobilgeräten orientiert, verstärkt diesen Trend (Gomez-Perez et al., 2017).

Im Folgenden wird das Konzept sowie damit verbunden der Aufbau von Linked Data und anschließend die Möglichkeiten für eine Konstruktion erläutert.

3.2.2.4.2 Aufbau von Linked (Enterprise) Data

Grundsätzlich handelt es sich bei Linked Data um ein gemeinsames Framework, um die Inhalte von Webseiten standardisiert in einer gemeinsamen Sprache (z. B. RDF) zu veröffentlichen und Verlinkungen zu weiteren Webseiten zu ermöglichen. Um dies zu erreichen, nennt Berners-Lee (2006) mit seiner grundlegenden Veröffentlichung über Linked Data und somit für das Semantic Web vier Prinzipien. Blumauer (2014b) gibt dazu jeweils ein Beispiel, welches hier zur Erläuterung der vier Prinzipien mitaufgeführt wird.

Mit dem ersten Prinzip *Use URIs (Uniform Resource Identifiers) as names for things* wird jedem Ding zur Identifikation eine URI als eindeutiger (identifizierender) Name zugeordnet. Beispielsweise hat Tim Berners-Lee (unter anderem) die URI http://dbpedia.org/resource/Tim_Berners-Lee. Als zweites Prinzip sollen für URIs gültige HTTP-Adressen benutzt werden, damit ein Agent (Mensch oder Software) diese Namen im WWW benutzen kann. Beispielsweise können die eben genannte URI und damit zahlreiche weitere Fakten zur entsprechenden Entität von einer Softwareanwendung oder auch mit einem einfachen Browser aufgerufen werden. Durch das dritte Prinzip wird sichergestellt, dass die Standards (RDF*, SPARQL) beim Erstellen der Linked Data verwendet werden, sodass beim Aufrufen der URI nützliche Informationen bereitgestellt werden. Beispielsweise ist das Ergebnis dadurch maschinenlesbar und standardisiert. Außerdem werden relevante Fakten auf eine strukturierte Art und Weise wiedergegeben: Tim Berners-Lee wurde am 08.06.1955 in London geboren. Tim Berners-Lee wird auch TimBL genannt. Tim Berners-Lee ist der Direktor des W3C. Mit diesen ersten drei Prinzipien stellt das Framework also das Veröffentlichen (auch: Übersetzen) des Inhaltes einer Webseite in eine standardisierte Sprache sicher. Durch das vierte Prinzip werden Verlinkungen zu anderen Datensätzen ermöglicht, indem Links zu anderen URIs eingesetzt werden, damit weitere Linked Data entdeckt werden können. Beispielsweise hat das W3C ebenfalls eine URI (http://dbpedia.org/resource/World_Wide_Web_Consortium), welche mit der URI von Tim Berners-Lee verknüpft ist.

Diese Idee des Standardisierens und Verknüpfens von Daten wurde in den letzten Jahren auch in Unternehmen eingesetzt, indem im Intranet erste Versuche gestartet wurden, Informationen als Linked Data zu veröffentlichen. Das Konzept war 2016 allerdings noch immer eher theoretisch (Galkin, Auer, & Scerri, 2016).

In Abb. 3.34 ist die Vision von Auer et al. (2014) dargestellt, bei der die Informationen aus den verschiedenen Systemen als Linked Data veröffentlicht und zu einem internen Linked-Data-Web verknüpft werden. Dies umfasst eine Enterprise KB (= Knowledge Base), die Wortschatz-Definitionen, Kopien von relevanten Linked Open Datasets sowie interne und externe Linksets zwischen Datensätzen enthält. Interne Daten sind vor externem Zugriff geschützt, wobei die Daten aus der öffentlichen Linked Open Data Cloud wiederverwendet werden können. Dies wird durch die gestrichelten Linien visualisiert. Die durchgezogenen Linien verdeutlichen, wie die IT-Systeme verbunden sind.

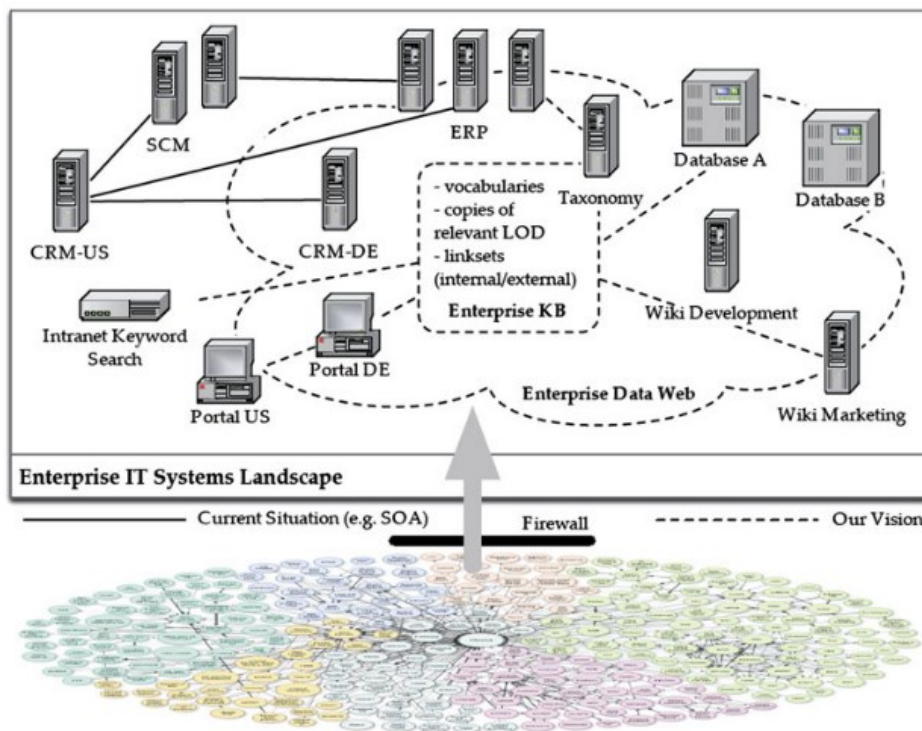


Abb. 3.34: Vernetzung von Daten und Informationen im Intranet (Auer et al., 2014, p. 91)

Neben den beiden Merkmalen von Linked Data (standardisiertes Veröffentlichen und Verlinken) für das Web als verteiltes und dezentrales Informationsmedium ohne eine zentrale Kontrolle, können diese um folgende *Enterprise-Data-Prinzipien* erweitert werden (Auer et al., 2014): Einerseits müssen bestehende Thesauri, Taxonomien, Wikis und Master-Data-Management-Systeme in Unternehmenswissensbasen und Wissenshubs für die Etablierung eines Linked Enterprise Data weiterentwickelt werden. Außerdem muss ein unternehmensweites URI-Schema eingeführt werden und die bestehenden Informationssysteme im eigenen Intranet um Linked-Data-Schnittstellen erweitert werden. Weiterhin müssen Verlinkungen zwischen im Zusammenhang stehenden Informationen hergestellt werden.

Wie bereits erwähnt ist Linked Data im Unternehmenskontext noch ein theoretisches Thema und Frischmuth et al. (2012) identifizierten sechs Herausforderungen, denen sich Unternehmen stellen müssen. Diese Herausforderungen sind in Abb. 3.35 visualisiert. Die (1) Unternehmenstaxonomien werden derzeit geschützt, zentralisiert und ohne Beziehungen zwischen den Begriffen abgespeichert. Außerdem gibt es voneinander unabhängige Terminologien, wobei kontrollierte Vokabulare (Idee ist älter als die moderne Informatik) diese Problematik adressiert (Allemang, 2010). Der Vorteil durch Linked Data wäre die Verwendung von offenen Standards (z. B. SKOS), wodurch die Informationen verteilt, hierarchisch, multilingual und in anderen Szenarien wiederverwendbar wären. Für den Datenaustausch wird derzeit (2) ein XML-Schema verwendet, welches keine Integration zu anderen Dokumenten bietet. Durch Linked Data könnten Beziehungen zwischen Entitäten von verschiedenen Schemata hergestellt werden. (3) Wikis oder intern genutzte Enzyklopädien werden in Unternehmen für Arbeits-

gruppen meist in Textform erstellt. Durch Linked Data könnten die strukturierten Daten in den Datenwikis durch andere Anwendungen wiederverwendet werden. Außerdem könnten Verlinkungen mit anderen Datenquellen (z. B. Taxonomien) eingerichtet werden. Im Bereich der (4) Web-Portal- und Intranet-Suche wird derzeit eine Schlüsselwortsuche im Textinhalt durchgeführt. Der Vorteil von Linked Data wäre hier die Möglichkeit der Einrichtung eines anspruchsvollen Suchmechanismus, indem implizites Wissen der verschiedenen Datenquellen verwendet wird. Für die (5) Datenbankintegration werden derzeit Data Warehouse, Schema Mediation und Query Federation verwendet. Durch Linked Data könnte eine einfache Datenintegration durch eine RDF-Schicht durchgeführt werden. Außerdem ist der (6) Enterprise-Single-Sign-on (SSO) im Bereich der Datenintegration eine Herausforderung. Durch die Verwendung von Linked Data wird ein Zugriff ohne die wiederholte Eingabe des Passwortes durch anspruchsvolle Zugangskontrollmechanismen möglich (Frischmuth et al., 2012).

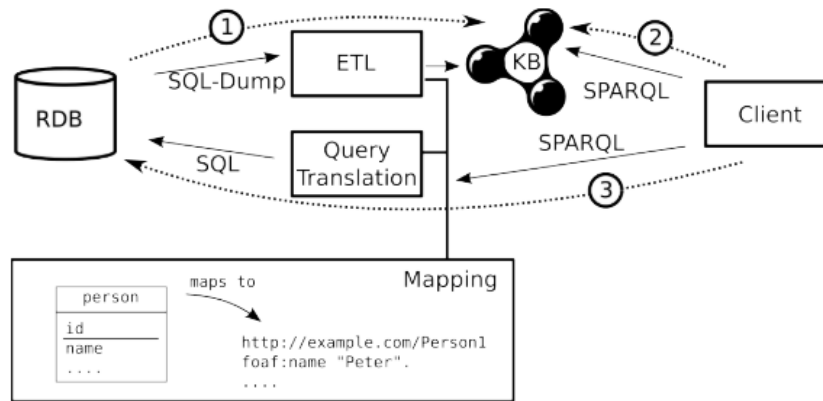
3.2.2.4.3 Konstruktion von Linked (Enterprise) Data

Für ein standardisiertes Vorgehen bei der Erstellung von Linked Data können zum einen die vier Prinzipien betrachtet werden, die bereits im vorherigen Abschnitt für die Erklärung des Konzeptes beschrieben wurden. Frischmuth et al. (2012) und Auer et al. (2014) schlagen einen Linked-Data-Lebenszyklus vor, der in Abb. 3.35 dargestellt ist. Die Phasen sind die acht Kreise und in der Mitte der Abbildung befinden sich die Herausforderungen, die im Abschnitt zuvor erläutert wurden. Der Linked-Data-Lebenszyklus unterstützt die wichtigen Herausforderungen bei der Datenintegration in Unternehmensumgebungen. Jede der Herausforderungen kann sich auf mehr als eine Lebenszyklusphase beziehen. Mit (1) *Storage/Querying* wird die Technologie für das RDF-Datenmanagement bereitgestellt. Im (2) zweiten Schritt (*Manual revision/authoring*) geht es um die Erstellung und Bearbeitung der RDF-Daten. Der (3) nächste Schritt (*Inter-linking/Fusing*) beinhaltet das (teil-)automatisierte Erstellen und Verwalten von Verknüpfungen zwischen verschiedenen Wissensbasen. Im (4) vierten Schritt (*Classification/Enrichment*) werden die Instanzdaten aus dem Web und Unternehmen auf Taxonomien, Vokabulare und Ontologien gemappt. Anschließend wird (5) die Datenqualität durch Herkunft, Kontext und Abdeckung beurteilt (Quality Analysis). Falls (6) Änderungen und Modifikationen an Wissensbasen, Vokabularen und Ontologien vorgenommen werden, sollten diese in diesem Schritt erkennbar gemacht werden (*Evolution/Repair*). Anschließend müssen (7) für Anwender die Daten und Beziehungen zwischen diesen Daten sichtbar gemacht werden. Dafür können verschiedene Such-, Browsing-, Explorations- und Visualisierungstechniken für verschiedene Arten von Linked Data genutzt werden (*Search/Browsing/Exploration*). (8) Abschließend erfolgt eine Gewinnung von neuen Informationen (Extraction) (Auer et al., 2014; Frischmuth et al., 2012). Da es sich um einen Zyklus handelt gibt es streng genommen keinen Anfang und kein Ende. Für die Erläuterung des Zyklus wurde allerdings ein Anfang gewählt. Durch diesen Zyklus können im Unternehmen flexibel immer mehr Linked Data veröffentlicht werden. Da im Unternehmen auch immer neue Daten hinzukommen, ist dieser Zyklus daher auch für die Veröffentlichung von neu dazukommenden Daten geeignet.



Abb. 3.35: Die Phasen des Linked Data-Lebenszyklus (Auer et al., 2014, p. 93; Frischmuth et al., 2012, p. 4)

Berners-Lee (in Sack, 2014, p. 50) erstellte als Urheber des Web of Data mit seinem 5-Sterne-Plan für die Veröffentlichung von Linked Data einen Vorschlag zur Qualitätsbewertung. Dabei werden fünf unterschiedliche Qualitätsstufen bezüglich der Veröffentlichung unterschieden: Für die Veröffentlichung von Linked Data sollen folgende Merkmale gelten: Eigene Daten als (2) *maschinenlesbare, strukturierte Daten* in (4) *W3C-standardisierten*, (3) *offenen Formaten* (1) *unter offener Lizenz frei verfügbar* machen und (4) *URIs zur Identifikation* verwenden. Das bedeutet, die eigenen Daten sollen (2) maschinenlesbar und strukturiert sein. Diese sollen (3) offene Formate nutzen. Diese offenen Formate sollen (4) W3C-standardisiert sein und URIs zur Identifikation verwenden. Außerdem sollen zu den eben genannten Voraussetzungen die eigenen Daten mit (5) bereits bestehenden Daten des Web of Data verlinkt werden.



Mapping-basierte Datenbankintegration mit Abfrageumsetzung und Extract Transform Load (ETL)

Abb. 3.36: Datenbankintegration (Auer et al., 2014, p. 96; Frischmuth et al., 2012, p. 14)

Abb. 3.36 zeigt eine Mapping-basierte Datenbankintegration. Bei der Veröffentlichung einer relationalen Datenbank als Linked Data können wiederum die in Abschnitt 3.2.2.3.2 zu ODBA beschriebenen zwei Vorgehensweisen unterschieden werden: *Materialisierung* (hier: 1 und 2) oder *Virtualisierung* (hier: 3) (Auer et al., 2014). Bei der Materialisierung der relationalen Datenbank als RDF wird unter Zuhilfenahme eines Mappings ein Export erstellt, der ähnlich zu einem Extract-Transform-Load (ETL) ist (1). Dieser wird in einer RDF-Datenbank gespeichert (KB). Auf diese Weise kann eine Bereitstellung als SPARQL-Endpoint oder eine Dereferenzierung per HTTP erfolgen. Über den Client wird auf den materialisierten Graphen zugegriffen (2). Bei der Erstellung eines virtuellen Graphen (3) wird eine SPARQL- oder HTTP-Abfrage in eine oder mehrere SQL-Abfragen übersetzt. Dies erfolgt durch ein Mapping. Die relationale Datenbank sendet das Ergebnis dann an den Client zurück (Auer et al., 2014). Der Vorteil eines materialisierten RDF-Graphen besteht in der Nutzung der speziellen Indizes der RDF-Datenbank. Bei der Virtualisierung liegt der Vorteil im direkten Zugriff auf die Daten und somit der Vermeidung von Redundanzen und ggf. Inkonsistenzen in den redundanten Speichern.

3.2.2.4.4 Beispiele für Linked (Enterprise) Data

Für Linked Data gibt es zahlreiche praktische Umsetzungen: Schema.org ist eine Initiative, die von Google, Bing, Yandex und Yahoo vorangetrieben wird. Außerdem sind Googles Knowledge Graph, Facebooks Open Graph oder die Linked Open Data Cloud zu nennen (Auer et al., 2014). Die Linked Open Data Cloud beinhaltet beispielsweise OpenCorporates (<http://opencorporates.com/>), welche eine Wissensdatenbank mit Informationen zu mehr als 50.000 Unternehmen weltweit ist. Außerdem gehört dazu die Product Ontology (<http://www.productontology.org/>), welche eine detaillierte Klassifikation und Informationen über mehr als 1 Million Produkte umfasst. Damit kann sich ein Unternehmen beispielsweise vernetzen, wenn dieses Linked Data verwendet (Auer et al., 2014). DBpedia („semantische Schwester“ von Wikipedia) gehört zum Web of Data, was eine Vielzahl unterschiedlicher Datenquellen über einen einfachen standardisierten Zugriffsmechanismus (HTTP-Protokoll) mit einem

einheitlichen Datenmodell (RDF) zur Verfügung stellt (Sack, 2014). Ein erfolgreiches Beispiel, dieses Web of Data zu nutzen, ist der britische BBC, der einer der größten öffentlich-rechtlichen Rundfunksender weltweit ist. BBC reicherte bereits früh das eigene Radioprogramm mit weiteren Informationen zu Künstlern, deren Werken und dazugehörigen Konzerten an. Da sich diese Informationen schnell ändern und die manuelle Pflege dazu bei der großen Menge an Daten hoch ist, wurden diese zusätzlichen Informationen über Linked Data aus öffentlich verfügbaren Datensätzen wie z. B. DBPedia bezogen. Daraufhin wurden die Daten bei BBC von weiteren Teilbereichen mit Linked-Data-Mechanismen miteinander verknüpft (Sack, 2014). Nach Allemang (2010) ist der Vorteil vom Einsatz der Linked Data in Unternehmen, dass diese schrittweise eingeführt werden können. Er führt neben der Regierung noch die Verlagsbranche auf, die sich zu Linked Data verpflichtete. In der Verlagsbranche geht es um das Verwalten von zahlreichen Medien, die zunehmend elektronisch angeboten werden. Damit verbunden muss die Organisation des veröffentlichten Materials verbessert werden, um eine praktikable Suche zu ermöglichen. Das Verlagswesen ist also eine Branche, die die Notwendigkeit und das Verwenden von Tagging-Vokabularen versteht und es werden bereits umfangreiche Standardvokabulare verwendet. SKOS wird in diesem Zusammenhang beispielsweise verwendet, um die verwendeten Vokabulare abzustimmen. Ein konkreter Verlag setzt dann eindeutige Verweise auf die Begriffe (Allemang, 2010; Sack, 2014). Ein weiteres Beispiel ist das Humboldt-Projekt. Dabei wird dem Nutzer eine Oberfläche (Humboldt Browser) bereitgestellt, in der Linked Data entdeckt werden können. Ausgehend von einem konkreten Objekt werden dem Nutzer Filtermöglichkeiten für die umliegenden Knoten angezeigt (Kobilarov & Dickinson, 2008). Bei dem Projekt Parallax und Companion wird dem Benutzer ermöglicht, die Daten im Web auf effiziente Weise zu durchqueren, indem dieser von einem „set of things“ zu einem „related set of things“ anstelle von einer einzelnen Website zu einer weiteren alleinstehenden Website navigieren kann (Huynh & Karger, 2009). Ein weiteres Beispiel ist VisNav, das eine Anwendung darstellt, die dem Nutzer Datensuche und -navigation ermöglicht, indem sie folgende Abfrageoperationen anbietet: Schlüsselwortsuche, Objektnavigation, Facet Selection, Path Traversal, Projektion und Sortierung (Harth, 2009). All diese Beispiele zeigen, dass es bereits zahlreiche Anwendungen im Web gibt, Daten und Informationen aus verteilten Quellen mithilfe von Semantic-Web-Technologien zu integrieren und dem Nutzer zur Verfügung zu stellen.

Linked Enterprise Data ist in Unternehmen noch nicht etabliert, da Unternehmensdaten andere Merkmale aufweisen als öffentliche Daten im Internet. Diese Unterschiede zeigen sich in technischer und betrieblicher Hinsicht. Ein offensichtlicher Vorteil der Nutzung semantischer Technologien ist allerdings die Integration verteilter Datensätze und es existieren vereinzelte (prototypische) Umsetzungen. In einer Pilotstudie wurde in einer internationalen Organisation Linked Data eingeführt, um eine Integration bereitzustellen (Hu & Svensson, 2010). In Abb. 3.37 ist die Architektur eines sogenannten Customer Information Portals dargestellt.

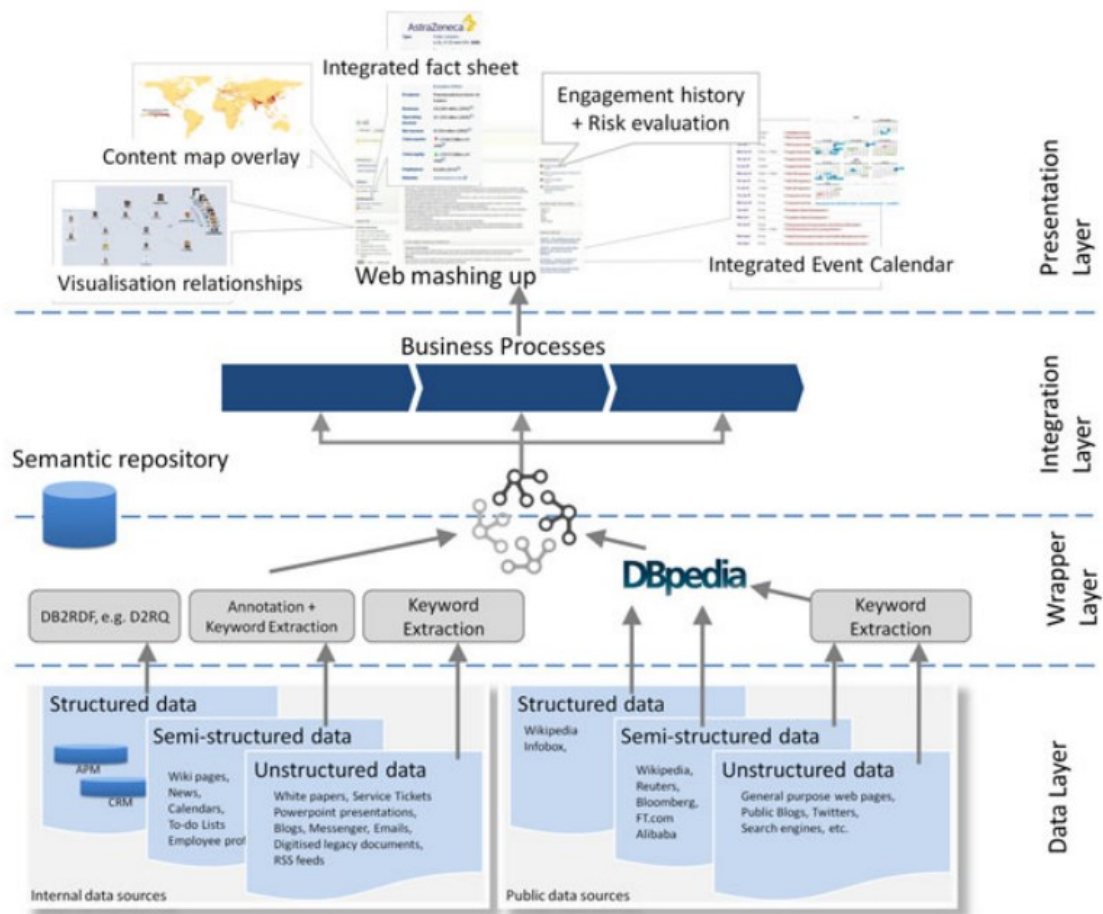


Abb. 3.37: Customer Information Portal Architecture (Hu & Svensson, 2010, p. 135)

Im unteren Bereich befinden sich die für das Unternehmen relevanten internen und externen Datenquellen. Diese unterscheiden sich jeweils in strukturierte, semi-strukturierte und unstrukturierte Daten. Strukturierte interne Daten beispielsweise werden direkt auf die Ontologie im Unternehmen gemappt. Diese übersetzten Daten werden im semantischen Repository gespeichert, von dort weiterverwendet und in die Unternehmensprozesse integriert. Dazu werden die Daten im *integrated fact sheet* angezeigt. Beispielsweise unterstützt dies bei der Kundengewinnung, indem potenzielle Kundenstammsätze in der unternehmensinternen Datenbank mit externen Informationen aus dem Internet verknüpft werden können. Auf diese Weise werden Veranstaltungen von Kunden identifiziert, welche dann mit den Kalendern der Vertriebsmitarbeiter abgeglichen werden. Die Endbenutzer können aus mehreren Visualisierungsoptionen auswählen: Liste der nächsten Ereignisse, Kartenüberlagerung der Ereignisorte aber auch herkömmliche Kalender (Hu & Svensson, 2010).

3.2.2.5 EKG – Enterprise Knowledge Graph

Trotz zahlreicher Paper zu Linked (Enterprise) Data und zu konkreten Vorschlägen, wie semantische Technologien in Unternehmens-IT-Umgebungen integriert werden können, ist LED in der Praxis noch

wenig verbreitet und existiert somit eher als abstrakte Idee (Galkin, Auer, & Scerri, 2016). Enterprise Knowledge Graph kann als eine Verkörperung des Linked-Data-Paradigmas betrachtet werden, indem das Management der Unternehmensinformationen auf eine semantische Ebene gehoben wird (Galkin, Auer, & Scerri, 2016). Im Folgenden wird das Konzept Enterprise Knowledge Graph vorgestellt und es werden bestehende standardisierte Vorgehen für das Design in Unternehmen dargelegt. Abschließend werden konkrete Beispiele aufgeführt. In Abschnitt 3.2.2.6 erfolgt ein Vergleich der hier aufgeführten Ontologie-basierten Integrationsansätze und insbesondere auch ein Vergleich von Linked Enterprise Data und Enterprise Knowledge Graph.

3.2.2.5.1 Ursprung von (Enterprise) Knowledge Graph

Informationen als Graph darzustellen und somit Entitäten (oder Business Objects) miteinander zu verknüpfen entspricht eher dem Denken der Menschen als die Informationen in Silos (wie es bei relationalen Datenbanken der Fall ist) zu betrachten (Hoede, 1995). Dies griff das Wissensmanagement auf und in diesem Bereich stieg der Bedarf an neuen Technologien in den letzten Jahren (Hislop, 2005).

Nach Gomez-Perez et al. (2017) führte Quillian als Erster 1968 semantische Netze in seine Modelle für semantische Speicher ein. Dies ist einer der ersten computergestützten Ansätze zur Wissensrepräsentation, der Wissen in Form von miteinander verbundenen Knoten und Bögen/Kanten darstellt. Knoten stellen dabei Objekte, Konzepte oder Situationen dar. Kanten visualisieren die Beziehungen zwischen den Knoten einschließlich „is-a“ (z. B. Ein Stuhl ist von der Art Möbel) und ein-Teil-von (z. B. Ein Sitz ist Teil eines Stuhls) (Gomez-Perez et al., 2017). Dieses semantische Gedächtnis bezieht sich auf allgemeines Wissen (Fakten, Konzepte und Beziehungen) z. B. über einen Stuhl. Es unterscheidet sich von einer anderen Art des Langzeitgedächtnisses (episodisches Gedächtnis), das sich auf bestimmte Ereignisse bezieht (z. B. Bewegungen eines Stuhls). Nach Quillian wurden verschiedene Varianten semantischer Netzwerke vorgeschlagen (Gomez-Perez et al., 2017). Semantische Netzwerke sind im Vergleich zu formalen Wissensrepräsentations- und Argumentationsformalismen wie Prädikatenlogik relativ einfach zu verwenden und zu warten. Allerdings leiden sie unter einigen Einschränkungen: Beispielsweise gibt es keine formale Syntax und Semantik für Quillians semantisches Netzwerk (wie z. B. is-a-Beziehung). Dieser Ansatz ist zwar flexibel, wird aber auch kritisiert, da es schwierig ist, semantische Netzwerke zu integrieren und ihre ursprüngliche Bedeutung beizubehalten. Außerdem können in semantischen Netzwerken die Bedeutungen von Bezeichnungen auf Knoten und Bögen von Benutzern nicht definiert werden (Gomez-Perez et al., 2017).

Mit dem Ziel, Wissen in einem Graphen darzustellen, entstanden in den 1980er Jahren im Anschluss an die semantischen Netze zwei unabhängig voneinander entwickelte Ansätze: Conceptual Graphs (Sowa, 1984) und Theory of Knowledge Graphs (Stokman & de Vries, 1988). Bei dem Conceptual Graph geht es darum, mit Klassen, Individuen und gewichteten Beziehungen das Wissen in einem Graph zu repräsentieren, der dann intuitiv durch den Menschen zu verstehen ist. Die Basis für das Logik-System ist der Existential Graph und dadurch existiert eine Standardisierung für die grafische Repräsentation (Peirce, 1885a, 1885b). Der Conceptual Graph wird in der künstlichen Intelligenz und im

Natural Language Processing angewendet und wurde in den letzten Jahrzehnten weiterentwickelt (Chein & Mugnier, 2009; Sowa, 1976, 1984, 2008). Die Grundlage des Semantic Web mit seinem Verständnis von Daten und Informationen in einem Graph übernimmt ausgewählte Aspekte dieser Theorie *Conceptual Graph* (Berners-Lee, 2008). Die *Knowledge Graph Theorie* wurde 1982 durch den diskreten Mathematiker Hoede der Universität Twente (Niederlande) und den mathematischen Soziologen Stokman der Universität Groningen (Niederlande) initiiert (Nurdiati & Hoede, 2008). Zu Beginn bestand das Ziel darin, mithilfe von Graphen das Wissen in Form von Expertensystemen darzustellen. Ende der 1980er Jahre wurden solche Expertensysteme in Medizin und Sozialwissenschaften erfolgreich unter der Verwendung der Knowledge Graph Theorie entwickelt (Nurdiati & Hoede, 2008; Stokman & de Vries, 1988). In dieser werden zwei Arten von Wissensrepräsentationen unterschieden. Zum einen geht es um die *prozedurale* Repräsentation, bei der Vorschriften und Regeln dargestellt werden, und zum anderen wird die *deklarative* Repräsentation eingesetzt, um Aussagen über ein Thema auszudrücken (Stokman & de Vries, 1988). Die Aussagen in dem Graph (als semantisches Netz) werden für formale Schlussfolgerungen verwendet. In der darauffolgenden Phase wurde das ursprüngliche Ziel erweitert und Knowledge Graphen werden zur Darstellung der natürlichen Sprache verwendet. Seitdem kann diese Theorie als eine der Methoden für NLP (natural language processing) betrachtet werden (Hoede, 1995; Nurdiati & Hoede, 2008).

Während in der ursprünglichen Konzeptionierung des Semantic Web von Berners-Lee (2008) zwar schon von *Conceptual Graphs* die Rede ist, wird der Begriff Knowledge Graph zunächst nicht verwendet. Dies erfolgt erst durch die Einführung des Googles Knowledge Graph durch Singhal (2012). Durch diese Erfolgsgeschichte inspiriert, gewinnen Knowledge Graphen im World Wide Web zunehmend an Bedeutung. In den letzten Jahren wurde das Knowledge-Graph-Konzept sowohl im akademischen als auch im industriellen Umfeld weiterentwickelt und erweitert und durch die Übertragung des Knowledge-Graph-Konzeptes aus dem Semantic Web in den Unternehmenskontext etablierte sich der Begriff Enterprise Knowledge Graph (Galkin, Auer, Kim, et al., 2016; Galkin, Auer, & Scerri, 2016; Gomez-Perez et al., 2017; Humm, Ege, & Reibold, 2015; Kim, Lee, & Yang, 2015; Miao, Meng, & Zhang, 2015). Es wurden zusätzliche Funktionen, neue Erkenntnisse und verschiedene Anwendungen eingeführt (Gomez-Perez et al., 2017). Das Konzept EKG wurde also nicht erst in diesem Zusammenhang erfunden. Unternehmen sowie semantische Gemeinschaften fanden noch keinen formalen umfassenden Rahmen für die Gestaltung solcher Graphen im Unternehmenskontext (Galkin, Auer, & Scerri, 2016). In den nächsten Abschnitten wird der derzeitige Stand in diesem Bereich skizziert.

3.2.2.5.2 Aufbau von (Enterprise) Knowledge Graphen

Bei der gegenwärtigen Definition eines Knowledge Graphen gibt es unterschiedliche Vorschläge und eine uneinheitliche Verwendung der Begrifflichkeiten (Blumauer, 2014a; Färber, Bartscherer, Menne, & Rettinger, 2018; Krötzsch & Weikum, 2016; McCusker, McGuinness, Erickson, & Chastain, 2016; Paulheim, 2017; Pujara et al., 2013).

Krötzsch & Weikum (2016) schreiben, dass der Knowledge Graph ein mächtiges Instrument für Suche, Analyse, Empfehlungen und Datenintegration darstellt. Zur Erläuterung dieses Konzeptes wird hier der Vorschlag zur Definition von Krötzsch & Weikum (2016) herangezogen:

“Knowledge graphs (KGs) are large networks of entities, their semantic types, properties, and relationships between entities.” (Krötzsch & Weikum, 2016)

Die Definition von Gomez-Perez et al. (2017) umfasst ähnliche Elemente und nimmt noch die Ontologie mit auf:

“A knowledge graph consists of a set of interconnected typed entities and their attributes.” “A knowledge graph has an ontology as its schema defining the vocabulary used in the knowledge graph.” (Gomez-Perez et al., 2017)

Ein Knowledge Graph besteht also aus einer Ontologie und einer Faktenbasis, indem die Faktenbasis mithilfe einer Ontologiebeschreibungssprache bzw. einem vorgegebenen Schema (z.B. OWL, RDF, SKOS) beschrieben und untereinander verknüpft wird. Die Ontologiebeschreibungssprache (wie z. B. RDF) ist eine formale, explizite Spezifikation einer gemeinsamen Konzeptualisierung und besteht aus einem Vokabular und Regeln zu seiner Zusammensetzung. Diese zeichnet sich durch eine hohe semantische Ausdruckskraft aus, die für eine erhöhte Komplexität erforderlich ist (Feilmayr & Wöß, 2016; Guarino et al., 2009; Suárez-Figueroa, Gómez-Pérez, Motta, & Gangemi, 2012). Das Ergebnis (= Knowledge Graph) ist dann maschinenlesbar. Ein solches Schema bestimmt z. B., dass es die Klassen Gemälde und Museum gibt und dass ein Gemälde in einem Museum hängt. Für Webseiten über Mona Lisa und Louvre wird dann jeweils festgelegt, dass es sich um ein Gemälde und um ein Museum handelt (Gradmann, 2013). Mithilfe dieser Wissensbasis kann dann beispielsweise geschlussfolgert werden, dass Mona Lisa im Louvre hängt. Die Klassen besitzen Eigenschaften, haben Beziehungen untereinander und sind unter Zuhilfenahme einer Ontologie so typisierbar, dass eine maschinelle Weiterverarbeitung dieses Beziehungsgeflechts möglich ist (Weller, 2014). Das Veröffentlichen der Daten und Verknüpfen zu einem Knowledge Graph kann mit den Linked-Data-Prinzipien umgesetzt werden (vgl. Abschnitt 3.2.2.4). Ein Knowledge Graph wird in der Regel in Graphdatenbanken redundant gespeichert, sodass Suchabfragen effektiver sind und Schlussfolgerungen gezogen werden können. Allerdings gibt es auch im Knowledge-Graph-Konzept Virtualisierungen solcher Graphen (Galkin et al., 2017).

Knowledge Graphen prägen das Semantic Web (Dengel, 2012; Krötzsch & Weikum, 2016). Mit diesem Konzept sind bereits zahlreiche Dokumente und Informationen im Web auf der Grundlage einer vorgegebenen Ontologie so maschinenlesbar gemacht worden, dass diese in einem Graph dargestellt werden können. Konkrete Knowledge Graphen unterscheiden sich vom Semantic Web darin, dass diese eine Domäne (wie z. B. Google) haben und es keinen umfassenden Knowledge Graphen für das gesamte Web gibt. Ein Knowledge Graph, der das gesamte Web umfasst, könnte als in sich abgeschlossenes Semantic Web interpretiert werden (Ehrlinger & Wöß, 2016). Es geht also beim Semantic Web um die Transformation der Hypertext-Anwendung WWW in einen weltumspannenden Wissensraum (Gradmann, 2013).

Knowledge Graphen werden auch für die Integration von Informationen in Unternehmen angewandt und beschreiben dann die Objekte der jeweiligen Domäne des betreffenden Unternehmens. Die Definition eines Enterprise Knowledge Graphen unterscheidet sich nicht wesentlich von der eines Knowledge Graphen:

„By EKG we refer to a semantic network of concepts, properties, individuals and links representing and referencing foundational and domain knowledge relevant for an enterprise.“ (Galkin, Auer, Kim, et al., 2016)

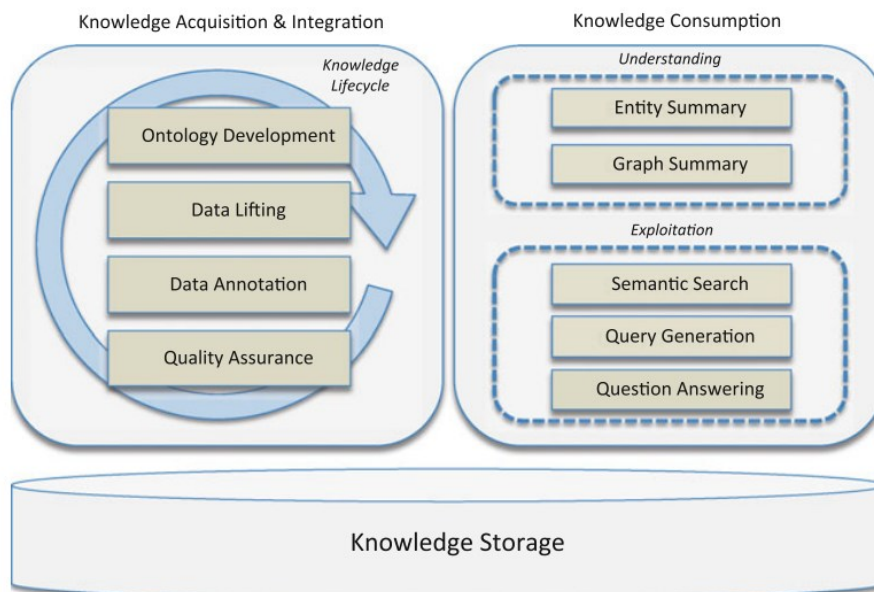
Verschiedene Autoren schlagen vor, Konzepte des Semantic Webs und somit des Knowledge Graphen auf Unternehmen zu übertragen (Galkin, Auer, Kim, et al., 2016; Galkin, Auer, & Scerri, 2016; Gomez-Perez et al., 2017; Humm et al., 2015; Kim et al., 2015; Miao et al., 2015). Allerdings ist der Ansatz Enterprise Knowledge Graph für Unternehmen noch nicht so sehr verbreitet wie im Web. Beispielsweise kann ein Enterprise Knowledge Graph Informationen über Produkte, Kunden, Absatzkanäle oder Mitbewerber enthalten. Ein zentrales Element ist dabei die Verlinkung von Elementen unter Einbezug der verschiedenen Datenquellen (intern und extern). Es handelt sich also um die Verwaltung und Integration von Wissen, eine Form der Wissensrepräsentation bis hin zu einer detaillierten Sicht im Zusammenhang von Textanalyse (Gomez-Perez et al., 2017; Kim et al., 2015; Miao et al., 2015).

Villazon-Terrazas et al. (2017) vergleichen den Einsatz des Knowledge Graphen im Web und im Unternehmen anhand von Funktionalitäten (vgl. Abb. 3.38). Beispielsweise sind die Datenquellen im Web verteilt, während diese in einem Unternehmen zwar auch verteilt, aber verglichen zum Web zentralisiert sind. Die Datenmengen sind im Unternehmen groß, im Vergleich zu den Datenmengen im Web (durch Big Data) sind diese allerdings geringer. Demgegenüber ist die Qualität der Daten im Web geringer und im Unternehmen hoch. Da es sich bei einem Unternehmen um eine übersichtliche Domäne verglichen zum Web handelt, ist die Ontologiebeschreibungssprache im Web eher einfach und im Unternehmen ausdrucksstärker gehalten. Das Wissen im Web ist eher allgemeines Wissen während es sich im Unternehmen um *domänenspezifisches* Wissen handelt.

Features	KG for Web searching	KG for enterprise
Data source	Distributed	usu. centralised
Openness of data	Open to public	Private
Size of data	Huge	Big
Data acquisition	Harder	Easier
Quality of data	Low	High
Ontology language	Simple	Likely to be more expressive
Knowledge	Generic knowledge	Domain-specific knowledge

Abb. 3.38: Knowledge Graph im Vergleich (Villazon-Terrazas, Garcia-Santa, Ren, Faraotti, et al., 2017, p. 55)

Ein auf Knowledge Graphen basierendes Informationssystem bildet üblicherweise ein Ökosystem, das folgende Hauptkomponenten umfasst (vgl. Abb. 3.39): Aufbau/Konstruktion (Knowledge Acquisition & Integration), Speicherung (Knowledge Storage) und Verbrauch (Knowledge Consumption) (Gomez-Perez et al., 2017). Dazu werden verschiedene Technologien eingesetzt: Für die (1) Wissensrepräsentation und Argumentation werden Sprachen, Schemata und Standardvokabulare eingesetzt. Für (2) den Wissensspeicher werden Graphdatenbanken und Repositories verwendet. Das (3) Knowledge Engineering nutzt Methoden, Editoren und Entwurfsmuster. Bei diesen ersten drei Punkten stammen die Technologien oftmals aus den Bereichen Datenbanken, Ontologien und Semantic Web (Gomez-Perez et al., 2017). Schließlich werden noch für (4) das automatische Wissenserlernen (einschließlich Schemalernen und Population) Technologien eingesetzt. Dafür werden in der Regel Frameworks und Technologien aus Data Mining, Natural Language Processing und Machine Learning verwendet (Gomez-Perez et al., 2017).



ARA = Abstract Reference Architecture

Abb. 3.39 ARA für Knowledge Graph Management in Organisationen (Denaux et al., 2017, p. 58)

Wie bereits erwähnt gibt es drei Hauptaufgaben im Zusammenhang mit der Verwendung von Knowledge Graphen: Konstruktion, Speicherung und Verbrauch. Diese Aufgaben sind die Hauptschichten in der Architektur (vgl. Abb. 3.39). Im *Knowledge Acquisition and Integration*-Layer wird der Knowledge Graph erstellt. Dies kann bottom-up (beginnend mit den Daten, die bereits vorhanden sind), top-down (beginnend mit der Analyse der Use Cases) oder middle-out (Kombination der beiden Ansätze) erfolgen und wird ausführlich im nächsten Abschnitt 3.2.2.5.3 erläutert. Auf dem *Knowledge Storage* Layer wird zweitens der Knowledge Graph so abgespeichert, dass dieser im Laufe der Zeit weiterentwickelt werden kann (z. B. Hinzufügen neuer Arten von Wissen). Außerdem muss effizient auf das im Graphen

kodierte Wissen zugegriffen werden können. Diese Aufgabe entspricht dem Knowledge Storage Layer. Schließlich muss drittens im *Knowledge Consumption* Layer das im Knowledge Graph verschlüsselte Wissen für die Nutzung in der Organisation verwendet werden, um die Effizienz im Unternehmen zu verbessern. Diese Aufgabe entspricht diesem Layer (Denaux et al., 2017).

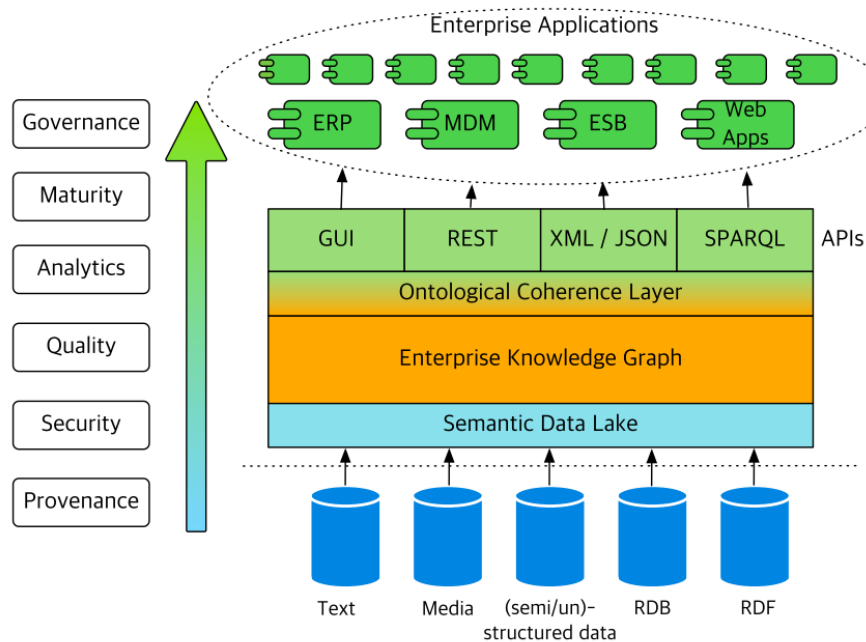


Abb. 3.40: Position eines EKG in der Enterprise Information System Architektur (Galkin et al., 2017, p. 90)

In Abb. 3.40 ist die Position eines Enterprise Knowledge Graphen in der Enterprise Information System Architektur abgebildet. Es wird eine Strukturansicht präsentiert, die eine statische Orchestrierung von Informationssystemen in einem Unternehmen darstellt. Ein EKG nimmt eine Vermittlungsposition zwischen der Speicherung von Rohdaten des Unternehmens und zahlreichen Diensten ein, während die Kohärenzschicht eine einheitliche Sicht auf die Daten bietet (Galkin et al., 2017). EKGs besetzen somit eine einzigartige Nische im Ökosystem von Unternehmensanwendungen. In Abb. 3.40 ist zu erkennen, dass ein EKG ein Verbraucher von heterogenen Rohdaten ist und gleichzeitig ein Anbieter von Wissen für Unternehmensanwendungen. Zahlreiche strukturierte und unstrukturierte Datenquellen bilden einen Semantic Data Lake (SDL), welcher eine neue Form eines Data Lake darstellt. Ein SDL kann sowohl die eigenen Daten des Unternehmens als auch extern bezogene Daten in den Open Access integrieren (z. B. die Verknüpfung der Open Data Cloud). In der ontologischen Kohärenzschicht befinden sich eine Reihe von high-level und domänenspezifischen Ontologien, die unterschiedliche semantische Sichten zu den EKG-Inhalten bieten. Eine organisatorische Ontologie definiert beispielsweise eine Geschäftsabteilung in einem Unternehmen, während eine Sicherheitsontologie den Zugriff auf die Daten beschränkt, die dieser Abteilung zur Verfügung stehen. Außerdem legen einige Supply-Chain-Ontologien

die Rolle der Abteilung in einem Produktlebenszyklus fest. Eine solche Granularität erhöht die Flexibilität der Wissensrepräsentation und stellt sicher, dass externe Anwendungen eine standardisierte, Ontologie-basierte Sicht auf die erforderlichen Daten aus dem EKG verwenden. Die API-Schicht stellt Kommunikationsschnittstellen zur Verfügung (z. B. REST, XML, JSON oder SPARQL), um in EKGs codiertes Wissen für Unternehmensanwendungen bereitzustellen. Mögliche Nutznießer der EKG-Technologie sind: ERP-Systeme (Enterprise Resource Planning), MDM-Systeme (Master Data Management), ESB (Enterprise Service Buses), E-Commerce und die Webanwendungen eines Unternehmens. Ein EKG spezifiziert eine Reihe von domänenunabhängigen Merkmalen (z. B. Provenienz, Governance oder Sicherheit), die den Betrieb und die Wartung des gesamten Systems unterstützen (Galkin et al., 2017). Die Autoren Galkin et al. (2017) argumentieren, dass diese Unternehmensmerkmale einen Enterprise Knowledge Graph von einem allgemeinen Knowledge Graph unterscheiden.

3.2.2.5.3 Konstruktion von (Enterprise) Knowledge Graphen

Wie bereits erwähnt entwickelten (noch) weder Unternehmen noch die semantische Community einen formalen umfassenden Rahmen für die Gestaltung von Knowledge Graphen (Galkin, Auer, & Scerri, 2016). Allerdings schlagen Villazon-Terrazas et al. (2017) einen generischen Lebenszyklus zum Erstellen und Verwalten von Wissen in Knowledge Graphen vor (vgl. Abb. 3.41). Wie bereits erwähnt, stammen die Informationen in einem Knowledge Graphen aus verschiedenen Quellsystemen, die unter einem gemeinsamen Schema oder Vokabular extrahiert, integriert und erweitert werden müssen. Der dafür erforderliche Aufwand ist nicht trivial und entspricht beispielsweise dem Aufwand zur Veröffentlichung vorhandener Daten als Linked Data (z. B. Konvertierung unstrukturiertes Wissen in Wikipedia als DBpedia). Daher ist ein systematisches Vorgehen von Bedeutung. Das von den Autoren vorgeschlagene Vorgehen folgt einem iterativen und inkrementellen Lebenszyklus, basierend auf einem vorhandenen Linked-Data-Lebenszyklusmodell, das bereits in realen Szenarien angewendet wurde (Villazon-Terrazas, Garcia-Santa, Ren, Srinivas, et al., 2017).

Im (1) ersten Schritt (Specification) erfolgt eine detaillierte Erhebung und Spezifikation der Anforderungen. Damit wird die Grundlage für die Vereinbarung zwischen Kunden und Entwicklern gelegt, was die Anwendung leisten soll. Außerdem kann damit auch eine Grundlage für die Schätzung von Kosten und Zeitplänen gelegt werden. Neben diesen Grundanforderungen sind auch die domänenspezifischen Anforderungen wichtig. Daher werden folgende Aufgaben in dieser Phase identifiziert: Identifizierung und Analyse der Datenquellen sowie das URI-Design (Villazon-Terrazas, Garcia-Santa, Ren, Srinivas, et al., 2017). Nach der ersten Phase, in der die Datenquellen identifiziert, ausgewählt und analysiert wurden, wird in der (2) nächsten Phase (Modeling) die Ontologie bestimmt, die zur Modellierung der Domäne dieser Datenquellen verwendet werden soll. Wichtig ist, verfügbare Vokabeln so weit wie möglich wiederzuverwenden. Diese Wiederverwendung beschleunigt die Entwicklung der Ontologie und spart Zeit, Aufwand und Ressourcen. Folgende Aufgaben fallen hier an: Suche nach geeigneten Vokabeln zur Wiederverwendung (z. B. in den Repositories Swoogle und LOV.2), Erstellung von noch nicht vorhandenen Vokabeln (auf Basis folgender Ressourcen: <http://semic.eu/>) und Erstellung von

neuem Vokabular (mit NeOn Methodology) (Villazon-Terrazas, Garcia-Santa, Ren, Srinivas, et al., 2017).

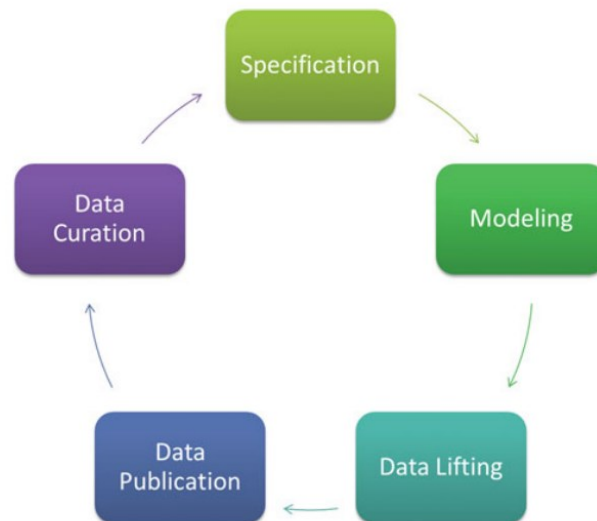


Abb. 3.41: Knowledge construction and maintenance lifecycle (Villazon-Terrazas, Garcia-Santa, Ren, Srinivas, et al., 2017, p. 80)

Im (3) nächsten Schritt (Data Lifting) werden die in der Spezifikationsaktivität ausgewählten Datenquellen genommen und gemäß dem in der Modellierungsaktivität erstellten Vokabular in RDF (Standarddatenmodell für den Datenaustausch und Format für den Knowledge Graph) umgewandelt. Dazu sind zwei Teilschritte notwendig: Transformation und Verknüpfung. Bei der Transformation erfolgt eine vollständige Konvertierung, indem alle Abfragen, die in der ursprünglichen Quelle möglich sind, auch in der RDF-Version möglich sein sollten. Außerdem sollten die erzeugten RDF-Instanzen die Zielontologiestruktur so genau wie möglich widerspiegeln bzw. die RDF-Instanzen müssen mit der bereits verfügbaren Ontologie/Vokabular übereinstimmen. Dies entspricht dem Mapping. Es gibt mehrere Tools, die diese Aufgaben technologisch unterstützen. Dazu gehört z. B. RDB2RDF oder GRDDL. Der weitere Teilschritt (Verknüpfung) entspricht dem vierten Linked-Data-Schritt (vgl. Abschnitt 3.2.2.4.2). Dabei sollen Verknüpfungen zwischen dem hier erstellen Knowledge Graphen und (optional) externen Knowledge Graphen hergestellt werden. Dazu müssen die Beziehungen zwischen Datenelementen ermittelt werden. Diese Links können manuell (aber zeitaufwendig) oder unter Einbezug von automatischen Tools (ggf. nicht sauber) erstellt werden. Falls Knowledge Graphen verknüpft werden sollen, müssen diese identifiziert (z. B. aus Linked Data-Repository CKAN – ComprehensiveKerbal Archive Network) und die Beziehungen identifiziert sowie geprüft werden. Die Überprüfung erfolgt durch Domänenexperten (Villazon-Terrazas, Garcia-Santa, Ren, Srinivas, et al., 2017). Der (4) vierte Schritt (Data Publication) besteht aus Knowledge Graph Publication und Metadata Publication. Mit dem ersten Teilschritt werden die in RDF transformierten Daten in einem Triple Store gespeichert und veröffentlicht. Dazu gibt es verschiedene Tools zum Speichern von RDF-Datensätzen (z. B. Virtuosos Universal Server,

Jena, Sesam, 4Store, YARS27, OWLIM). Einige dieser Tools haben bereits einen SPARQL-Endpunkt und ein Linked-Data-Frontend. Nach der Veröffentlichung des Knowledge Graphen müssen die Metadateninformationen darüber eingefügt werden. Dazu gibt es Vokabulare (z. B. VoID), mit denen Metadaten zu RDF-Datensätzen ausgedrückt werden können. Diese umfassen allgemeine Metadaten, Zugriffsmetadaten, strukturelle Metadaten und die Beschreibung von Verknüpfungen zwischen Knowledge Graphen (Villazon-Terrazas, Garcia-Santa, Ren, Srinivas, et al., 2017). Der (5) letzte Schritt (Data Curation) dient zur Bereinigung und Pflege der Daten, damit diese später wiederverwendet werden können (Villazon-Terrazas, Garcia-Santa, Ren, Srinivas, et al., 2017). Das Paradigma des Erzeugens, Publizierens und Auswertens von Knowledge Graphen führte unweigerlich zu Problemen. Beispielsweise werden zum Teil Anwendungen daran gehindert, die strukturierten Informationen, die einem Knowledge Graphen zu Grunde liegen, effektiv zu nutzen. In diesem Schritt sollen diese Fehler bereinigt werden. Dazu müssen die Fehler identifiziert und anschließend dauerhaft bereinigt werden. Häufige Fehler sind: Probleme auf HTTP-Ebene, Argumentationsprobleme (z. B. Namensraum ohne Vokabular) oder fehlerhafte/inkompatible Datentypen (Villazon-Terrazas, Garcia-Santa, Ren, Srinivas, et al., 2017). Darüber hinaus gibt es weitere Schemata zum Erstellen von Knowledge Graphen (Alexopoulos, Zhao, Pan, & Zhu, 2017; Galkin, Auer, Kim, et al., 2016; Jetschni & Meister, 2017; Ruan et al., 2016; Sabou et al., 2018; Song et al., 2017). Ein weiteres Beispiel ist OSMoSys (Psyllidis, 2015).

3.2.2.5.4 Beispiele für (Enterprise) Knowledge Graphen

Durch die Einführung des Google Knowledge Graphen im Jahr 2012 (Singhal, 2012) wurde der Begriff Knowledge Graph in der Semantic Web Community etabliert. Weitere Umsetzungen, die zu dieser Anwendungsfamilie gehören, sind DBpedia, YAGO, Freebase, Wikidata, Spark, Googles Knowledge Vault, Microsofts Satori und Facebooks Entity Graph, deren gemeinsamer Nenner die Verwendung von Linked Data ist (Ehrlinger & Wöß, 2016; Färber et al., 2018). DBpedia wurde durch Freebase gestartet und stellt eine Wikipedia Wissensextraktion dar. Nach der Übernahme von Freebase durch Google wurde der Knowledge Graph zu Googles zentraler Wissensbasis ausgebaut. Neben allgemeinen Daten enthält dieser Knowledge Graph auch sehr spezifische Themenbereiche wie Musikstücke, Bücher oder Filme (Auer et al., 2014). Die Ergebnisse von Googles Suchmaschine wird mithilfe von semantischen Suchinformationen aus einer Vielzahl von Quellen verbessert (Gomez-Perez et al., 2017). Google nutzt den sogenannten Google Knowledge Graph (Singhal, 2012) nicht nur für eine durch Semantik verbesserte Suche, sondern stellt dem Nutzer auch Informationen integriert aus verschiedenen Quellen zur Verfügung. Wird beispielsweise der Begriff Mona Lisa in der Google-Suche eingegeben, erscheinen rechts neben der Auflistung der Suchergebnisse gebündelte Informationen über das Objekt wie z. B. Künstler: Leonardo da Vinci, Standort: Louvre (seit 1797) und Geschaffen: 1503. Einige Informationen wie z. B. Louvre sind verlinkt und leiten den Nutzer direkt weiter auf diesen Knoten. Das Ergebnis ist das gleiche, wie wenn der Nutzer den Suchbegriff Louvre eingegeben hätte. Auf diese Weise wird man durch die Informationen des Google Knowledge Graphen ausgehend von einem konkreten Knoten geführt.

Unternehmen und öffentliche Verwaltungen beschäftigen sich derzeit damit, das Knowledge-Graph-Konzept auf Unternehmensebene zu nutzen, um ihre Daten zu beschreiben, sie durch Verknüpfung mit anderen Wissensdatenbanken innerhalb und außerhalb ihrer Umgebung anzureichern und die Entwicklung von wissensbasierten Daten zu beleben (Gomez-Perez et al., 2017). Es gibt bereits einige Produkte von Unternehmen, die dies für andere Unternehmen ermöglichen: Cognitum Ontorion, Semantic Web Company PoolParty, KnowledgeStore, Metaphacts, Smartlogic Semaphore 4, Cambridge Semantics Anzo Smart Data Platform, RAVN Applied Cognitive Engine, SindiceTech CloudSpace Platform, Synptica Enterprise Taxonomy Management Software oder SAP HANA SPS 11 (Galkin et al., 2017).

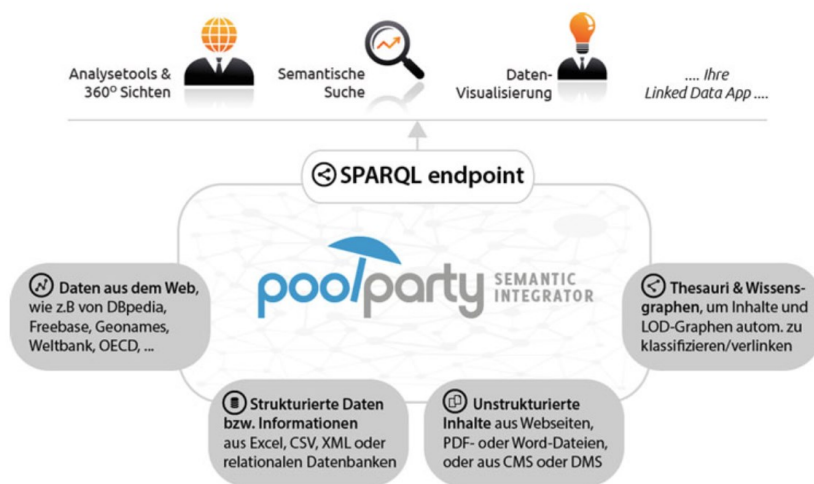


Abb. 3.42: High-Level Architecture eines Linked Data Warehouses (Blumauer, 2014b, p. 11)

Im Folgenden werden zwei dieser aufgeführten Beispiele erläutert. Der PoolParty Semantic Integrator (vgl. Abb. 3.42) ist ein Linked Data Warehouse, welches sowohl Daten aus relationalen Datenbanken als auch Daten und Fakten aus beliebigen Arten von Texten erfasst (Blumauer, 2014b). Auf diesen „Pool“ an Informationen kann über einen SPARQL-Endpunkt zugegriffen werden, indem die Daten entweder visualisiert werden, eine 360°-Sicht und Analysemöglichkeiten genutzt werden oder eine semantische Suche ausgewählt wird. Die Motivation dafür ist, dass in Unternehmen der Großteil des entscheidungsrelevanten Wissens in unstrukturierten Informationen (z. B. E-Mails, Pressemitteilungen oder Gesprächsprotokollen) enthalten ist. Auf diese Weise können beispielsweise Fakten aus Produktdatenbanken mit den eben erwähnten unstrukturierten Daten verknüpft werden, sodass tiefgreifende Analysen zur Wettbewerbssituation oder zur aktuellen Marktentwicklung möglich werden (Blumauer, 2014b).

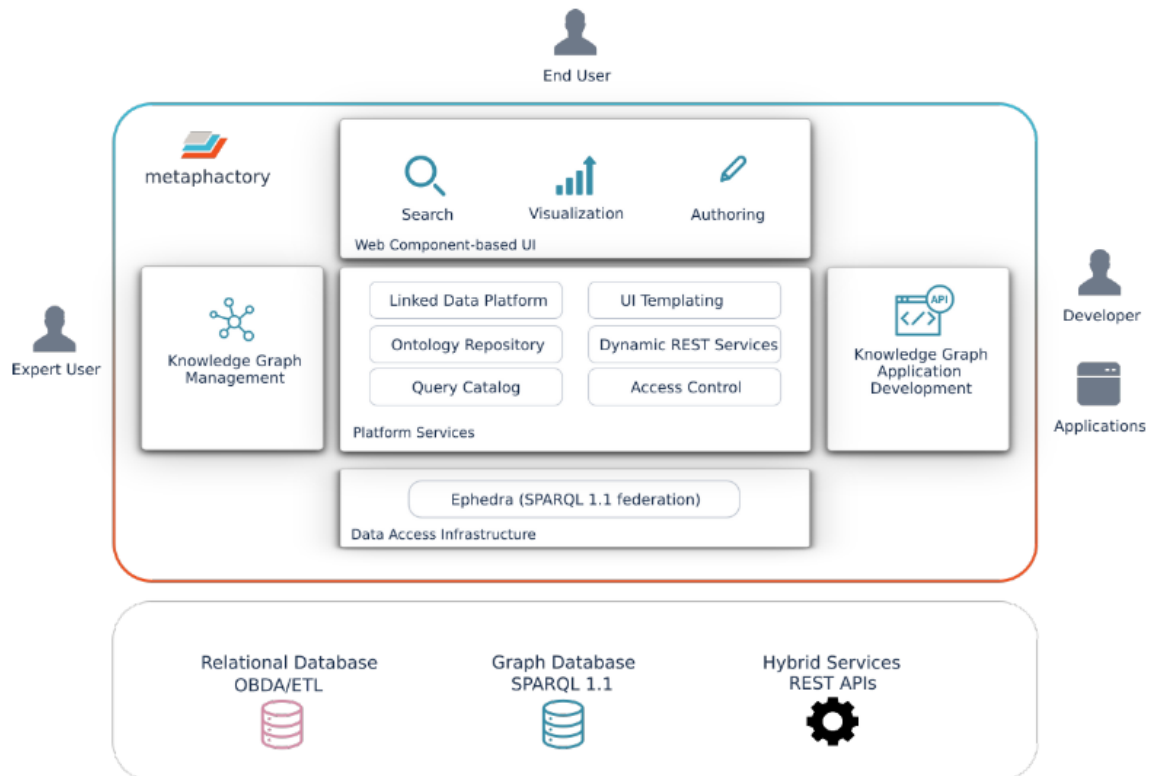


Abb. 3.43: Metaphactory Systemarchitektur (Haase, Herzig, Kozlov, Nikolov, & Trame, 2019, p. 3)

Metaphacts GmbH ist ein Unternehmen, das anderen Unternehmen die Plattform metaphactory für die Erstellung und Nutzung von Enterprise Knowledge Graphs anbietet. Dazu werden relevante Dienste für die Verwaltung von Knowledge Graphen realisiert, eine umfangreiche und anpassbare Benutzeroberfläche bereitgestellt und die schnelle Erstellung von anwendungsfallspezifischen Anwendungen ermöglicht. Die Plattform bietet Funktionen zur Unterstützung des gesamten Lebenszyklus von Knowledge Graphen (Haase, 2017). Metaphactory zielt darauf ab, verschiedene Kategorien von Benutzern (Expert User, End User, Developer) innerhalb des Unternehmens bei der Erstellung und/oder Nutzung des Knowledge Graphen zu unterstützen (vgl. Abb. 3.41). Die Plattform wird in verschiedenen Bereichen eingesetzt. Beispielsweise etablierten sich Knowledge-Graph-Technologien im Kontext des kulturellen Erbes. Mithilfe der Ontologie CIDOC-CRM (<http://www.cidoc-crm.org/>) wurde ein Standard geschaffen, um Informationen über kulturelles Erbe als Linked Data bereitzustellen. Die metaphactory Plattform wird im Rahmen des ResearchSpace-Projektes (<https://www.researchspace.org/>) genutzt, um den Knowledge Graphen des British Museum zu managen und die Forscher zu unterstützen, die Metadaten über die Artefakte (historischer Kontext, Zusammenhänge zu geographischen Orten, Ersteller, vergangene Besitzer etc.) im Museum zu analysieren. Außerdem soll dadurch die Möglichkeit geschaffen werden, diese Metadaten in Zusammenarbeit zu erweitern, indem Anmerkungen, Erzählungen mit semantischen Referenzen und Argumentationen erstellt werden (Haase et al., 2019).

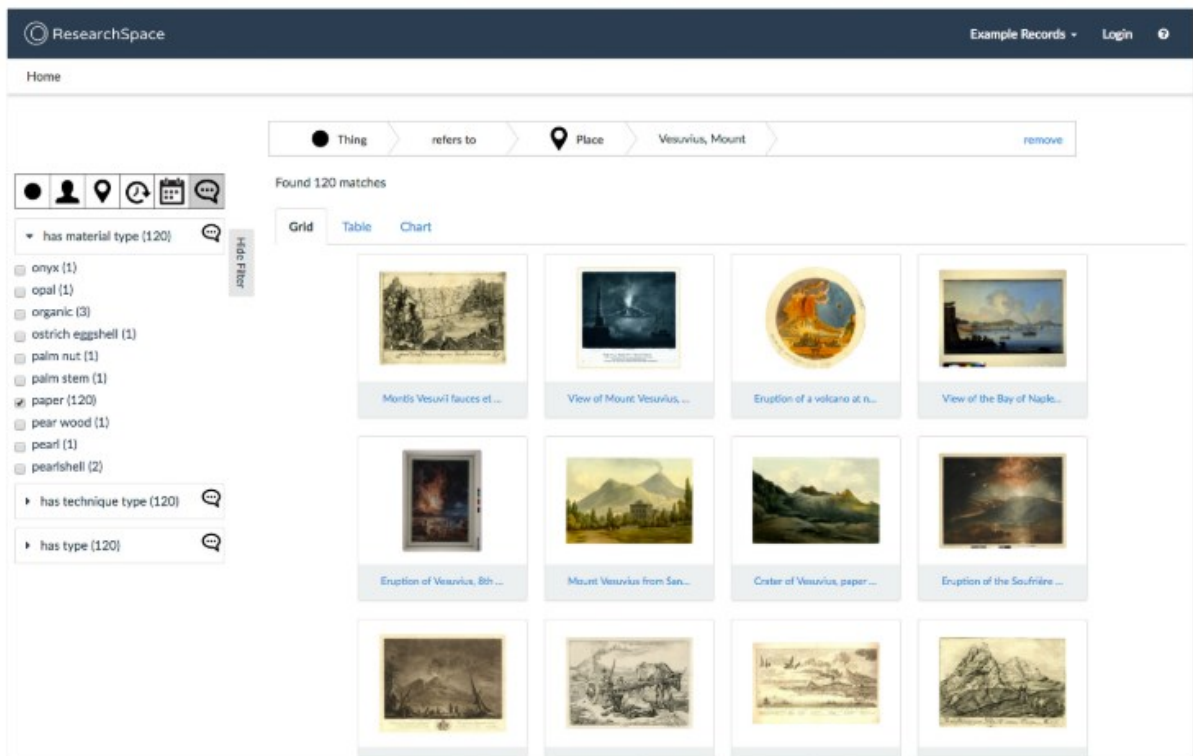


Abb. 3.44: Strukturierte semantische Suche in den Artefakten des British Museums (Haase et al., 2019, p. 12)

In Abb. 3.44 ist das Frontend mit den verschiedenen Artefakten des Museums zu sehen und wie der Nutzer auf diese Informationen zugreifen kann. Ein entscheidendes Merkmal dabei ist die Bereitstellung einer strukturierten Suche, bei der der Nutzer verschiedene Kriterien kombinieren kann. Beispielsweise kann folgende Abfrage zusammengesetzt werden: „give me all bronze artifacts created in Egypt between 2500BC and 2000BC“. Dies wird in SPARQL übersetzt und mithilfe der Daten im Knowledge Graphen beantwortet (Haase et al., 2019).

3.2.2.6 Vergleich der Ontologie-basierten Integrationsansätze

Im Rahmen der strukturierten Literaturanalyse konnten eine Reihe an Ontologie-basierten Integrationsansätzen aus dem Semantic Web identifiziert werden, die Ähnlichkeiten zu dem in dieser Dissertation betrachteten Ansatz (SoNBO) aufweisen. Die wichtigsten davon wurden in diesem Abschnitt vorgestellt. Im folgenden Abschnitt erfolgt jetzt ein Vergleich der Ansätze. Dafür wird zunächst noch einmal an das fundamentale Konzept hinter dem Semantic Web erinnert, das für den Vergleich relevant ist.

Das Semantic Web entstand mit der Idee, die Informationen im Web zu standardisieren und zu verlinken, sodass diese maschinenlesbar sind. Dadurch können die Informationen der unterschiedlichen Webseiten integriert betrachtet werden und neben einem simplen Informationszugriff für den Nutzer

(z. B. QbC, QbN) ist auch Reasoning (Schlussfolgern) möglich. Eine zentrale Rolle dabei spielen zum einen semantische Standards, die durch das W3C entwickelt wurden, und darauf aufbauende Ontologien. Mit diesen Konzepten können also die Informationen von Webseiten in eine einheitliche Sprache (RDF, OWL) transformiert und daraufhin untereinander verlinkt werden. Das Transformieren und Verlinken erfolgt weiterhin durch den Menschen, indem der Mensch die Konzepte, Attribute und Relationen, die die Webseite charakterisieren, auswählt. Allerdings gibt es Tools, die bei der Übersetzung und beim Verlinken unterstützen. Linked Data beschreibt mit seinen vier Prinzipien dieses Vorgehen der Transformation und Verlinkung. Idealerweise existiert eine einheitliche Ontologie, die für die Informationen im gesamten Web genutzt werden kann, sodass sukzessive ein standardisierter Graph entsteht. Dies wäre dann ein einheitlicher Knowledge Graph, der aus einer Ontologie und einer Faktenbasis besteht. In der Realität allerdings bildeten sich durch die Dezentralisierung des Webs und der unterschiedlichen Anbieter unterschiedliche Ontologien und verschiedene Knowledge Graphen heraus, die allerdings in der Regel die Standards des W3C nutzen. Die Linked Open Data Cloud ist die Initiative, die der ursprünglichen Idee am nächsten kommt. Ein weiteres prominentes Beispiel ist Googles Knowledge Graph. Diese Ansätze wurden in den letzten Jahren auf Unternehmen übertragen (EKG und LED) und stoßen dort im Bereich der Informationsintegration auf großes Interesse.

Die Basis von Linked (Enterprise) Data sind die vier Prinzipien zum Veröffentlichen und Verlinken von Linked Data. Die Unterschiede zum Knowledge Graphen sind dabei subtil aber vorhanden (Gomez-Perez et al., 2017).

Features	Pure RDF datasets	Linked Data	Knowledge Graph
Machine readability	Y	Y	Y
Human readability	NN	NN	Y
Data distribution	N	Y	NN
Inter-dataset linkage	L	Y	Y
Data integration	NN	NN	Y
Data consistency	NN	NN	Y
Reliability	NN	NN	Y
High quality	NN	NN	Y

Y: Yes, N: No, NN: Not necessary, L: Limited

Abb. 3.45: RDF-Datensätze vs. Linked Data vs. Knowledge Graph (Villazon-Terrazas, Garcia-Santa, Ren, Faraotti, et al., 2017, p. 52)

Villazon-Terrazas et al. (2017) vergleichen Linked Data und Knowledge Graph mit reinen RDF-Datensätzen (vgl. Abb. 3.45). RDF-Datensätze sind Datensammlungen, bei denen die Daten in einem RDF-Format (Subjekt, Prädikat, Objekt) dargestellt werden. Linked Data sind mehrere RDF-Datensätze, die unabhängig voneinander entwickelt, verwaltet und verteilt und dennoch miteinander verbunden sind. Diese Verbindungen entstehen, wenn beispielsweise auf dieselbe Person mit derselben URI in

unterschiedlichen Datensätzen Bezug genommen wird. Ein Knowledge Graph ist ein strukturierter Datensatz, der mit einem RDF-Datenmodell kompatibel ist und eine Ontologie als Schema hat (was für Linked Data nicht zwingend notwendig ist). Dabei muss ein Knowledge Graph nicht mit externen Knowledge Graphen verknüpft sein. Es wird erwartet, dass Knowledge Graphen zuverlässig, von hoher Qualität, hoher Zugänglichkeit und Endbenutzer-orientierte Informationsdienste sind (Villazon-Terrazas, Garcia-Santa, Ren, Faraotti, et al., 2017). Der für die Dissertation interessanteste Aspekt des Vergleichs in Abb. 3.45 ist die Datenintegration. Während diese beim Knowledge Graphen vorhanden ist, ist sie mit Linked Data durchaus möglich, allerdings nicht zwingend vorhanden. Galkin, Auer, & Scerri (2016) schreiben in einem Paper dazu, dass Linked Enterprise Data als Framework für Enterprise Knowledge Graphen betrachtet werden und somit der EKG die LD-Prinzipien nutzen kann. Daher hat dieses Konzept (Linked Data) in dieser Dissertation seine Berechtigung (da es auch zur Integration genutzt werden kann), wird allerdings nicht weiter separat betrachtet, da die Ontologie dort nicht im Vordergrund steht und dem Knowledge Graph-Ansatz sehr ähnlich ist. Für den Vergleich der Ontologie-basierten Ansätze (vgl. Tab. 3.8) wird daher Linked (Enterprise) Data nicht aufgeführt.

In den Ontologie-basierten Integrationsansätzen existieren einige Schlüsselkonzepte, die in den letzten Abschnitten erklärt wurden und die für das SoNBO-Konzept relevant sind. Diese sind in Abb. 3.46 dargestellt. Diese Schlüsselkonzepte werden auch für die folgenden Vergleiche der Integrationsansätze OBDA, EKG und SoNBO wieder aufgegriffen.

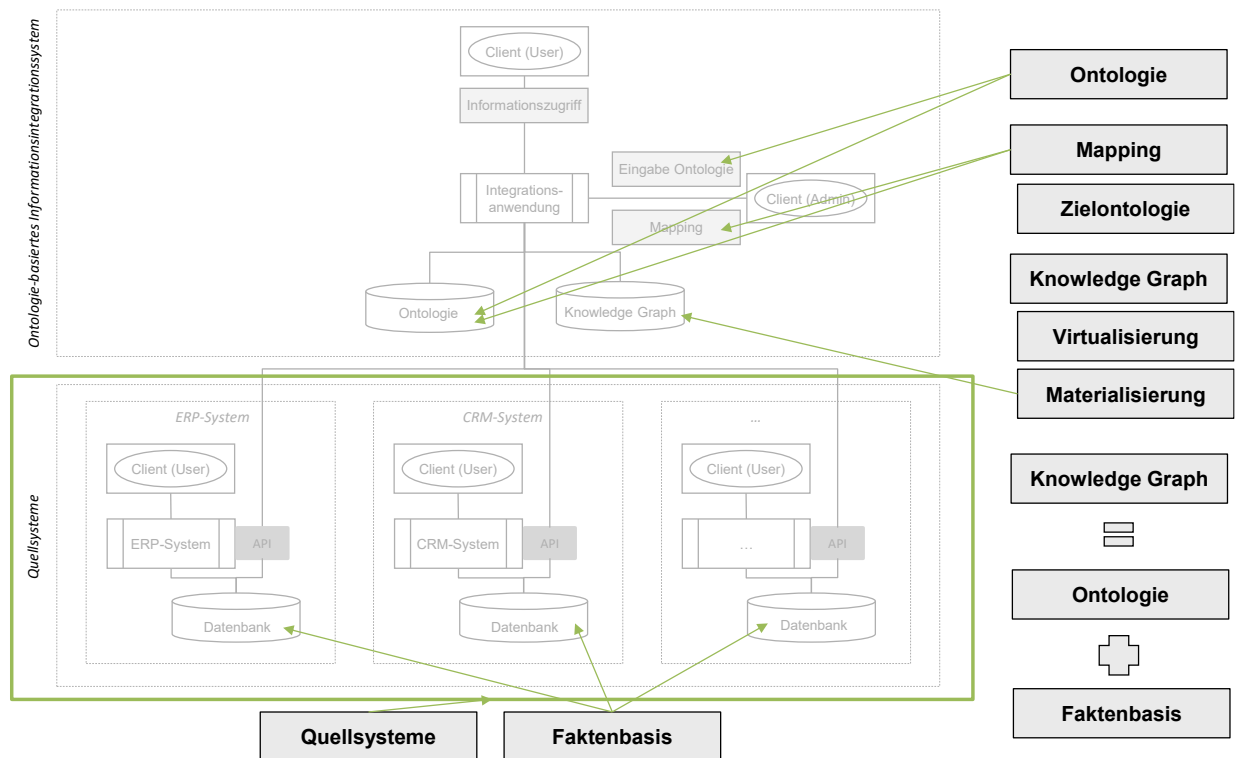


Abb. 3.46: Schlüsselkonzepte der Ontologie-basierten Integrationsansätze aus dem Semantic Web

Enterprise Knowledge Graph und Social Network of Business Objects¹⁰ weisen gewisse Gemeinsamkeiten und Unterschiede auf. Die Gemeinsamkeit besteht insbesondere in dem gemeinsamen Verständnis eines Graphen als Integrationsschicht. Dies macht sich in der Definition eines Knowledge Graphen bemerkbar. Allerdings ist die ursprüngliche Idee beim Knowledge Graphen, dass der Knowledge Graph redundant abgespeichert wird (Materialisierung), während bei SoNBO der Graph virtualisiert wird und somit nur zur Laufzeit generiert wird. Es gibt mittlerweile auch Knowledge Graphen, bei denen der Graph virtualisiert wird. Allerdings ist prinzipiell noch der Fokus auf die Materialisierung zu erkennen. Bei beiden Ansätzen spielen Ontologien für die Erstellung des Graphen eine tragende Rolle. Das Hauptziel bei der Erstellung eines Knowledge Graphen ist historisch bedingt Reasoning. Allerdings ermöglicht der Knowledge Graph auch QbC und QbN. Bei SoNBO liegt der Fokus auf QbN. Vor allem diese intuitive Art des Informationszugriff, die an die Navigation in sozialen Netzwerken (mit sozialen Funktionalitäten wie Activity Stream) angelehnt ist, ist ein wesentlicher Unterschied zum Knowledge Graph. Für die technische Umsetzung werden beim Knowledge Graphen semantische Technologien (standardisiert durch das W3C) benutzt, während diese in der ursprünglichen SoNBO-Applikation nicht verwendet werden. Daher wird für die Umsetzung bei SoNBO kein technisches Knowhow in diesem Bereich benötigt, wobei dadurch aber auch nicht die Vorteile wie z. B. Reasoning genutzt werden können. Da aber (wie bereits erwähnt) der Fokus bei SoNBO auf der Navigation durch das Netzwerk liegt, stellt dies kein Nachteil dar. Außerdem sei zu erwähnen, dass EKGs aus der Wissenschaft in die Praxis übertragen wurden, während SoNBO in der Praxis entstand und von der Wissenschaft aufgegriffen wurde.

Ontology-Based Data Access (OBDA) besitzt ebenfalls Gemeinsamkeiten und Unterschiede zu SoNBO. Die Gemeinsamkeit ist die Virtualisierung des Graphen, indem zur Laufzeit mithilfe der Ontologie die Faktenbasis entsprechend aufgerufen wird¹¹. Dies erfolgt durch ein Übersetzen der Datenbankabfragen (SPARQL in SQL) und auf diese Weise werden die Informationen in der Ontologie auf die Daten in den Quellsystemen gemappt. Die Daten bleiben also in beiden Ansätzen in den Quellsystemen in ihrer ursprünglichen Struktur und eine Ontologie wird als zusätzliche Ebene mit geeigneten Mechanismen zur Virtualisierung eingebunden (Soylu et al., 2017). Dadurch erfolgt auch eine nicht-redundante Speicherung der Faktenbasis, wodurch es sich um Live-Zugriffe handelt. Allerdings ist die Art des Datenzugriffes unterschiedlich. Bei OBDA erfolgt der Zugriff über QbC, während bei SoNBO der Fokus auf QbN liegt. Ein weiterer Unterschied ist auch hier die technische Umsetzung. OBDA nutzt semantische Standards, die SoNBO-Applikation wurde mit proprietärer Technologie realisiert.

¹⁰ Hier wird auf den Begriff „Social Network of Business Objects“ (SoNBO) Bezug genommen, wie er in der Tiefenfallstudie Vössing identifiziert und beschrieben wird. Die wissenschaftliche Konzeptualisierung erfolgt im nächsten Kapitel.

¹¹ Auch hier ist eine Materialisierung möglich. Dies steht allerdings bei der Idee nicht im Vordergrund.

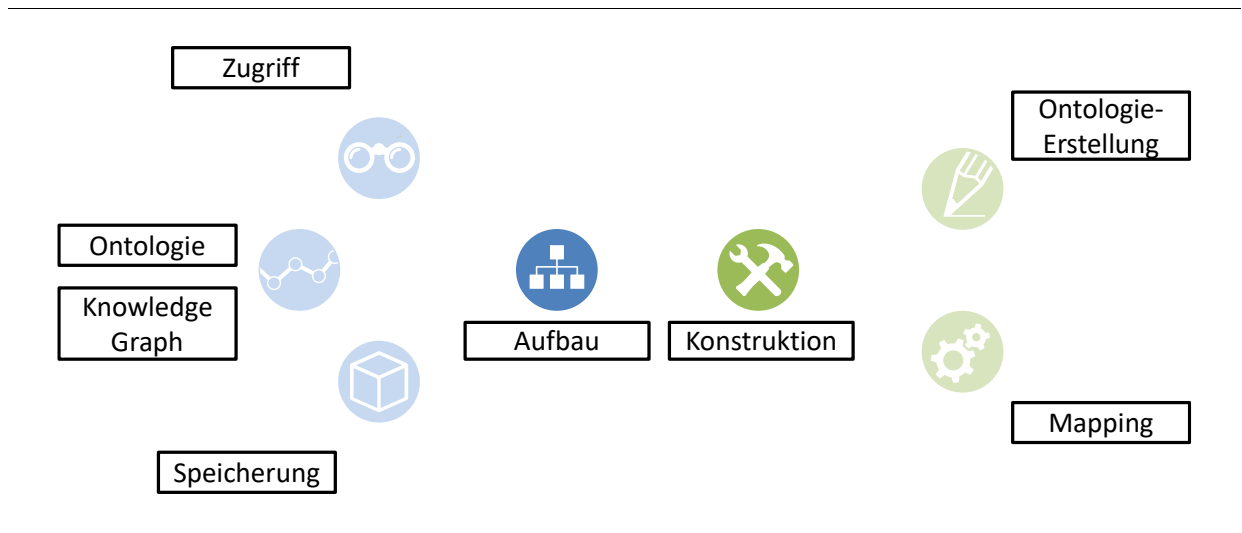


Abb. 3.47: Kategorien für den Vergleich von OBDA und EKG

In Abb. 3.47 sind die Kategorien abgebildet, anhand derer die Ansätze verglichen werden. Für die Dissertation sind zum einen der Aufbau und insbesondere die Architektur der Konzepte relevant. Diese werden nochmal in die Art des Datenzugriffes, in den Aufbau der Ontologie sowie des Knowledge Graphen und in die Art der Speicherung der Faktenbasis (im Falle einer Materialisierung) unterschieden. Bei der Konstruktion geht es um die Erstellung eines Ontologie-basierten Systems. Dazu sind die Schritte Ontologie-Erstellung und Mapping notwendig.

In Tab. 3.8 sind diese Kategorien aufgelistet und die drei Ansätze OBDA, EKG und SoNBO werden anhand dieser Kriterien miteinander verglichen. Die ausgewählten Kriterien und Attribute für den Vergleich wurden so ausgewählt, dass ein Vergleich mit SoNBO möglich ist. Bei OBDA dominiert beispielsweise der Datenzugriff durch QbC, da die Suchabfrage durch den Endbenutzer eingegeben wird. Bei Umsetzungen mithilfe des Knowledge Graphen wird für den Datenzugriff sowohl QbC als auch QbN ermöglicht. Bei SoNBO steht QbN im Vordergrund. Bei den technischen Umsetzungen sticht heraus, dass bei OBDA und EKG semantische Technologien verwendet werden (Abfragesprache, Beschreibungssprache, Mapping), während bei SoNBO diese nicht verwendet werden. Ein Hauptunterscheidungsmerkmal ist der Aufbau der Ontologie und damit verbunden der Knowledge Graph. Dies ist bei SoNBO ein soziales Netzwerk, was dem Endbenutzer auch verdeutlicht wird. Bei OBDA und EKG wird ein Netzwerk nicht explizit ausgeschlossen, doch bisher wurde diese Form in der Literatur nicht beschrieben. Außerdem existieren verschiedene Vorgehensweisen für die Konstruktion eines OBDA oder EKG, was in den Abschnitten 3.2.2.3.3, 3.2.2.4.3 und 3.2.2.5.3 ausgeführt wurde. Für SoNBO waren vor der Erstellung dieser Dissertation (noch) keine Vorgehensweisen vorhanden.

Tab. 3.8: Vergleich der Ontologie-basierten Integrationsansätze

Kriterium	Subkriterium	Komplementäre Attribute	OBDA	(E)KG	SoNBO ¹²
Beginn		Jahreszahl	Anfang 2000	1970er & 2012	2016
Zielgruppe		[Mensch in Organisation]; [Mensch im Web]	[Mensch in Organisation]	[Mensch in Organisation] → [Mensch im Web] → [Mensch in Organisation]	[Mensch in Organisation]
Aufbau	Datenzugriff konzeptionell	[QbC]; [QbN]; [Reasoning]	[QbC]	[QbC]; [QbN]; [Reasoning]	[QbN]
	Datenzugriff technisch	[SPARQL]; [kein SPARQL]	[SPARQL]	[SPARQL]	[kein SPARQL]
	Ontologie/Knowledge Graph konzeptionell	[Netzwerk]; [kein Netzwerk]	[kein Netzwerk]	[Netzwerk]; [kein Netzwerk]	[Netzwerk]
	Ontologie/Knowledge Graph technisch (Beschreibungssprache)	[RDF(S); [OWL]; [Sonstige SW-Technologien]; [LPG]	[RDF(S); [OWL]; [Sonstige SW-Technologien]	[RDF(S); [OWL]	[LPG]
	Speicherung des Knowledge Graphen konzeptionell	[redundant/materialisiert]; [nicht-redundant/virtualisiert]	[nicht-redundant/virtualisiert]; [redundant/materialisiert]	[redundant/materialisiert]; [nicht-redundant/virtualisiert]	[nicht-redundant/virtualisiert]
	Speicherung des Knowledge Graphen technisch	[Graph-DB]; [Dok-DB]; [SQL-DB]	[Graph-DB]	[Graph-DB]	[Dok-DB]
Konstruktion	Ontologie-Erstellung konzeptionell	[ja]; [nein]	[ja]	[ja]	[nein]
	Ontologie-Erstellung technisch	[ja]; [nein]	[ja]	[ja]	[nein]
	Mapping konzeptionell	[ja]; [nein]	[ja]	[ja]	[nein]
	Mapping technisch	[R2RML]; [GRDDL]; [kein Standard]	[R2RML]	[R2RML]; [GRDDL]	[kein Standard]

3.2.2.7 Zusammenfassung

Zusammengefasst besitzt SoNBO Gemeinsamkeiten sowohl mit OBDA (Virtualisierung und keine Materialisierung des Graphen im Vordergrund) als auch mit dem EKG (Definition, Verständnis und Bedeu-

¹² In dieser Tabelle wird unter SoNBO die Umsetzung in dem Ingenieurbüro Vössing verstanden. Die wissenschaftliche Konzeptualisierung erfolgt erst im nächsten Kapitel.

tung des Graphen). In allen drei Ansätzen kommt der Ontologie eine tragende Rolle zu. Allerdings unterscheidet sich SoNBO insbesondere durch die technische Umsetzung (keine semantischen Technologien) und die spezielle Art des Informationszugriffes, indem der Graph als Netzwerk aufgefasst wird und der Nutzer durch die Informationen des Unternehmens navigiert wie dies aus den sozialen Netzwerken bekannt ist. Somit betrachtet der Nutzer die aus verschiedenen Systemen integrierten Informationen aus der Sicht eines konkreten Objektes (Social Business Object) und erhält somit einen schnellen Zugriff und eine umfassende und konsistente Sicht auf die Informationen.

Für die Konzeptualisierung von SoNBO können daher gewisse Aspekte (z. B. Konzept, Attribut, Beziehung, Quellsystem, Materialisierung, Virtualisierung, ...) herangezogen werden, um SoNBO zu beschreiben (vgl. Kapitel 4). Für die Entwicklung des Frameworks zur Einführung eines SoNBO-Explorers in einem beliebigen Unternehmen können zudem Elemente von OBDA oder EKG (z. B. Mapping, Zielontologie, Startontologie, ...) angepasst für das SoNBO-Framework genutzt werden. Welche Elemente im Framework aus welchen Bereichen herangezogen wurden, wird in Kapitel 5 dargelegt.

3.3 Zusammenfassung des Status quo in der Praxis und Wissenschaft

In diesem Kapitel wurde die Basis für das Ziel der Dissertation gelegt, indem die bei Vössing identifizierte SoNBO-Anwendung vorgestellt wurde, die mithilfe einer Tiefenfallstudie aufgenommen wurde. Außerdem wurde der Status quo in der wissenschaftlichen Literatur zu Ontologie-basierten Integrationsansätzen dargelegt und somit das Thema der Dissertation wissenschaftlich verortet. Mit dieser Basis werden in den nächsten Kapiteln die Konzeptualisierung von SoNBO und die Entwicklung des Frameworks erstellt, indem auf den Key Case und die mit SoNBO verwandten wissenschaftlichen Konzepte zurückgegriffen wird. Dazu gehören OBDA (Ontology-Based Data Access) und (EKG) Enterprise Knowledge Graph, welcher Linked Enterprise Data nutzen kann.

4 SoNBO-Konzept

In diesem Abschnitt wird das SoNBO-Konzept (Ergebnis 1) als erweiterter Ontologie-basierter Integrationsansatz SoNBO (Social Network of Business Objects) beschrieben, welches aus der Synthese der vorangegangenen Ergebnisse (vgl. Kapitel 3) entstand. Somit fließen sowohl die Erkenntnisse aus dem Status quo der Wissenschaft (Literaturanalyse) also auch aus dem Status quo in der Praxis (Tiefenfallstudie) in die Konzeptualisierung mit ein. Außerdem wurde im Rahmen des SoNBO-Forschungsprojektes im Anschluss an die Erhebung des Key Case Vössing ein SoNBO-Explorer mit Studierenden im Rahmen von Abschlussarbeiten entwickelt (Götz, 2018; Riedle, 2018) und für die FG BAS (= Forschungsgruppe der Universität Koblenz-Landau) konfiguriert (Konfigurations-Projekt 1 und 2). Diese Ergebnisse wurden bereits veröffentlicht und die Erkenntnisse aus diesen Projekten fließen auch mit in die Konzeptualisierung ein (Gebel-Sauer & Schubert, 2019b; Götz & Gebel-Sauer, 2018). Im Anhang befindet sich eine Übersicht über die Dokumentation des Forschungsprozesses (vgl. A.2). Dieses Kapitel ist grundlegend für das Framework in Kapitel 5.

Im Folgenden werden zunächst die Definitionen und Eigenschaften, die zur Beschreibung des SoNBO-Konzeptes notwendig sind, dargelegt. Anschließend werden ein generisches Benutzerfrontend für einen SoNBO-Explorer sowie eine generische Anwendungssicht vorgestellt, um SoNBO zu erklären.

4.1 Definitionen und Eigenschaften

Für die Erklärung des SoNBO-Konzeptes sind bestimmte Begriffe von Bedeutung, die im Folgenden definiert werden.

4.1.1 Social Business Object

Für das SoNBO-Konzept spielt das Business Object eine tragende Rolle, weshalb zunächst die Bedeutung dieses Begriffes in der wissenschaftlichen Literatur identifiziert wird. Die Ausführungen in diesem Abschnitt wurden in Teilen bereits in einer begutachteten (peer-reviewed) Arbeit veröffentlicht (Gebel-Sauer & Schubert, 2019a).

Der Begriff *Business Object* wird in verschiedenen Teilgebieten der (Wirtschafts-)Informatik verwendet: Softwareentwicklung, Softwareeinführung und Softwareorchestrierung (vgl. Tab. 4.1). Die OMG (Object Management Group) entwickelte das Business-Object-Konzept für die objektorientierte Anwendungsentwicklung und es stellt somit ein logisches Abbild eines betriebswirtschaftlichen Sachverhaltes dar (Schmidt-Volkmar, 2008). Das Konzept wurde von ARIS für die Beschreibung von Softwaresystemen aufgegriffen und beispielsweise in Softwareeinführungsprojekten in Unternehmen genutzt. Die Softwareorchestrierung benutzt ebenfalls das Konzept des Business Objects.

Tab. 4.1: Verwendung von Business Objects für betriebliche Anwendungssysteme
(Gebel-Sauer & Schubert, 2019a, p. 205)

Fachgebiet	Teilgebiet	Business Object	Beispiele	Literatur
Informatik	(1) Softwareentwicklung	„A business object is a representation of a thing active in the business domain, including at least its business name and definition, attributes, behavior, relationships, rules, policies and constraints.“ (Sutherland, Casanave, Miller, Patel, & Hollowell, 1995)	Mitarbeiter, Produkt, Rechnung, Auftrag, Adresse, Konto, LKW	(Eicker & Nietsch, 1999; Frank, 1999; Seubert, 1998; Sims, 1995; Sutherland et al., 1995; Weske, 1999; Zimmermann, 1999)
Wirtschaftsinformatik	(2) Softwareeinführung (z. B. ARIS)	„Eine Anwendung besteht aus mehreren Business Objects, die von einem für die Anwendung definierten Geschäftsprozess gesteuert wird.“ (Scheer, 1998)	Auftrag, Artikel, Lieferant, Kunde	(Nüttgens & Zimmermann, 1998; Scheer, 1998; Scheer, Nüttgens, & Zimmermann, 1997)
Wirtschaftsinformatik	(3) Softwareorchestrierung	„Ein Geschäftsvorfall (Gesamtprozess) benötigt für seine Abwicklung immer Informationen aus mehreren BO.“ (Bleiholder & Schmid, 2008)	Kunde, Produkt, Mitarbeiter, Dienstreise-Antrag, Genehmigung, Ablehnung	(Bleiholder & Schmid, 2008; Dietzsch & Goetz, 2005; Schmidt, Fleischmann, & Gilbert, 2009)

Allen drei Bereichen ist die Strukturierung und Kategorisierung von Unternehmensinformationen in den Informationssystemen gemeinsam, die für den Entwickler oder Administrator vorgenommen wird. Nach Casanave (1995) ähneln Business Objects zunächst möglicherweise Tabellen in einem relationalen Datenbankmanagementsystem (DBMS), da Tabellen auch Geschäftsinformationen darstellen (= Kategorisierungsmöglichkeit). In einigen einfacheren Fällen kann auch eine direkte Entsprechung zwischen einem Business Object und einer DBMS-Tabelle bestehen. Allerdings implementieren die Business Objects in den meisten Fällen Regeln und Prozesse, welche über die Fähigkeiten eines DBMS hinausgehen. Sie können also auch mehrere Tabellen kombinieren, die Verteilung verwalten oder Informationen verwalten, die nicht in einem DBMS gespeichert sind (wie Online-Aktienkursnotierungen). Business Objects repräsentieren also mehrere Tabellen, Prozesse und Regeln auf einer höheren Ebene als die DBMS-Tabelle (Casanave, 1995). Unabhängig davon, um welche Kategorisierung es sich bei den Business Objects handelt, wird diese Strukturierung für den Endanwender nicht sichtbar gemacht. Dadurch sind der Aufbau und die Definition in den einzelnen Disziplinen ähnlich, wobei sie sich im jeweiligen Zweck (bedingt durch die verschiedenen Teilgebiete) unterscheiden. Außerdem werden verschiedene Unterscheidungen zwischen den Ebenen Klasse, Business Object und Attribut vorgenommen. Während in einigen Beiträgen zwischen diesen Ebenen unterschieden wird, indem eine Klasse (z. B. Rechnung) die Klassifikation von ähnlichen Business Objects (z. B. konkreten Rechnungen) ist und Attribute die Business Objects spezifizieren (z. B. Rechnungsdatum, Rechnungsbetrag), wird in einigen Beiträgen insbesondere aus der Informatik der Begriff Business Object sowohl als Klasse als auch für ein Attribut oder auch als Überbegriff für die Klasse und das Objekt benutzt (Eicker & Nietsch, 1999; Frank, 1999; Seubert, 1998; Sims, 1995; Sutherland et al., 1995; Weske, 1999;

Zimmermann, 1999). Dies stellt den funktionalen Zweck für ein Business Object in der Softwareentwicklung in den Vordergrund, während in den anderen beiden Bereichen auch die Bedürfnisse des Unternehmens fokussiert werden. Da das SoNBO-Konzept auch die Unternehmenssicht miteinbezieht, wird bei einem Business Object zwischen der Klasse, Business Object (Objekt/Instanz) und Attributen klar unterschieden, wie es in den Teilgebieten der Wirtschaftsinformatik bereits gemacht wird.

Das Verständnis eines Business Objects, das für die weitere Definition im SoNBO-Konzept zu Grunde gelegt wird, ist in Tab. 4.2 beschrieben. Da sich die Business Objects in den zu integrierenden Quellsystemen befinden, werden diese auch in Bezug zu den Quellsystemen gesetzt.

Tab. 4.2: Voraussetzung (Quellsystem, Business Object, Klasse)

(a) Jedes Informationssystem (betriebliches Anwendungssystem wie z. B. ERP-System, Vertriebssystem, oder Produktivsystem) in einem Unternehmen kann für das SoNBO-Konzept als Quellsystem Q dienen. In einem beliebigen Quellsystem Q befinden sich endlich viele Business Objects BO_i . Ein BO setzt sich aus endlich vielen Attributen a_j und deren Attributwerten v_{jk} zusammen, welche das Business Object spezifizieren. Attribute sind Einträge in den Datenbanken der Quellsysteme.

Sei $i \in \{1, \dots, n\}$; $j \in \{1, \dots, m\}$; $k \in \{1, \dots, l\}$; $i, j, k, n, m, l \in \mathbb{N}$

Es gilt: $Q = \{BO_1, BO_2, \dots, BO_n\}$ und $BO_i = \{(a_1 | v_{11}, v_{12}, \dots, v_{1l}), (a_2 | v_{21}, v_{22}, \dots, v_{2l}), \dots, (a_m | v_{m1}, v_{m2}, \dots, v_{ml})\}$

(b) Jedes Business Object BO_i gehört zu genau einer Klasse C_g . Zu einer Klasse C_g in einem Quellsystem Q gehören endliche viele Business Objects derselben Kategorie. Eine Klasse setzt sich aus endlich vielen Attributen a_j zusammen.

Sei $g \in \{1, \dots, h\}$, $g, h, d \in \mathbb{N}$

Es gilt: $Q = \{C_1, C_2, \dots, C_h\}$, $C_g = \{BO_1, BO_2, \dots, BO_d\}$ und $C_g = \{a_1, a_2, \dots, a_m\}$

(c) insbesondere gilt:

- $h < n$ (= die Anzahl der Klassen ist echt kleiner als die Anzahl der Business Objects)

- wenn $BO_i \in C_g \rightarrow a_j$ sind identisch

Für das SoNBO-Konzept werden die Informationssysteme, deren Informationen integriert werden sollen, als Quellsystem bezeichnet wie dies auch im OBDA-Ansatz gemacht wird (vgl. Abschnitt 3.2.2.3). Ein Informationssystem ist in der Regel ein betriebliches Anwendungssystem wie z. B. das ERP-System SAP ERP oder das CRM-System Salesforce. Es kann sich dabei auch um ein Produktivsystem handeln, wobei hier auf betriebliche Anwendungssysteme als Quellsystem fokussiert wird. In diesen Informationssystemen befinden sich die Informationen in Form von endlich vielen Business Objects (n : Anzahl der Business Objects je Quellsystem). Ein Business Object setzt sich aus endlich vielen Attributen (m : Anzahl der Attribute je Business Object) und deren Attributwerten zusammen. Je Attribut kann es mehrere Attributwerte (l : Anzahl der Attributwerte je Attribut) geben.

Tab. 4.3: Beispiel zu Quellsystem, Klasse, Business Object, Attribut und Attributwert

Quellsystem	Klasse	Business Object	Attribute	Attributwert(e)
ERP-System (z. B. SAP ERP)	Rechnung-Kunde	Rechnung 1725	Artikel	Herd, Küchenaufbau
			Rechnungsbetrag	399,99 Euro
			Kundenkontakt	Anja Schmidt
			Status	offen (nicht bezahlt)
			Sachbearbeiter	Peter Müller
		
ERP-System (z. B. SAP ERP)	Rechnung-Kunde	Rechnung 1754	Artikel	Kühlschrank, Arbeitsplatte
			Rechnungsbetrag	439,90 Euro
			Kundenkontakt	Anja Schmidt
			Status	abgeschlossen (bezahlt)
			Sachbearbeiter	Peter Müller
		
CRM-System (z. B. Salesforce)	Kundenkontakt	Anja Schmidt	E-Mail	anjasch- midt@company.com
			Telefonnummer	0123/456789
		
ECS (z. B. HCL Connections)	Mitarbeiter	Peter Müller	E-Mail-Adresse	pmueller@enterprise.com
			Bild	
			Position	Sachbearbeiter
		
HR-System (z. B. Datev Personal- Managementsystem classic)	Mitarbeiter	Peter Müller	Personalnummer	254
			Geburtsdatum	22.06.1986
		

Ein Business Object ist beispielsweise die Rechnung 1725 mit den Attributen Artikel und Rechnungsbetrag (vgl. Tab. 4.3). Die Attributwerte zu Artikel sind z. B. Herd und Küchenaufbau und zu Rechnungsbetrag: 399,90 Euro. Endlich viele Business Objects derselben Kategorie werden zu einer Klasse kategorisiert. Beispielsweise gehören verschiedene Rechnungen an Kunden zu der Klasse *Rechnung – Kunde* oder verschiedene Kundenkontakte zu der Klasse *Kundenkontakt*. Die Attribute der Business Objects, die zu derselben Klasse gehören, und die Attribute der zugehörigen Klasse sind gleich. Dieser Aufbau von Klasse, Business Object und Attribute sind in den Quellsystemen durch den Anbieter des Informationssystems vorgegeben.

Ein Social Business Object (vgl. Tab. 4.4) besteht aus der Teilmenge von Attributen und Attributwerten eines oder mehrerer Business Objects. Außerdem wird ein Social Business Object um soziale Funktionen (z. B. follow) erweitert (vgl. Abschnitte 3.2.1.7 und 4.2). Diese sozialen Funktionen werden durch

ausgewählte Attribute erzeugt. Beispielsweise wird ein Attribut ausgewählt, um die Beziehung zwischen zwei Social Business Objects zu erzeugen. Dies wird auch bei Ontologien auf diese Weise gemacht (vgl. Abschnitt 3.2.1.2.2).

Tab. 4.4: Definition (Social Business Object)

<p>Jedes Business Object BO in einem Quellsystem Q kann zu einem Social Business Object SBO werden. Jedes SBO gehört somit zu einem BO aus Q.</p> <p>(a) Ein SBO besteht aus einer Teilmenge aus Attributen a_j und Attributwerten v_{jk} des zugehörigen BO (= Ursprungs-BO). Ein SBO kann verschiedene Ursprungs-BO aus verschiedenen Quellsystemen haben.</p> <p>Sei $i \in \{1, \dots, n\}$; $j \in \{1, \dots, m\}$; $k \in \{1, \dots, l\}$ $a_j, v_{jk} \subset BO_i$; $i, j, k, n, m, l \in \mathbb{N}$</p> <p>Es gilt: $SBO_i = \{(a_1 v_{11}, v_{12}, \dots, v_{1l}), (a_2 v_{21}, v_{22}, \dots, v_{2l}), \dots, (a_m v_{m1}, v_{m2}, \dots, v_{ml})\}$</p> <p>(b) Das SBO wird um soziale Funktionen erweitert. Die sozialen Funktionen entstehen durch die Attribute und Attributwerten aus den Ursprungs-BO.</p> <p>(c) Insbesondere gilt folgende Eigenschaft: $SBO \leq BO$ (die Anzahl der SBO ist kleiner gleich die Anzahl der BO)</p>
--

Die Auswahl von Attributen und Attributwerten sowie die Erweiterung um soziale Funktionen hängen von den Anforderungen des Unternehmens ab. Damit wird das Business-Object-Konzept dem Endanwender verdeutlicht. Die auszuwählenden Attribute müssen bzw. sollen gewisse Kriterien erfüllen (vgl. Tab. 4.5).

Tab. 4.5: Spezifizierung der auszuwählenden Attribute

<p>Die aus den Business Objects stammenden Attribute <i>a</i> müssen folgende Kriterien erfüllen:</p> <ul style="list-style-type: none"> - ID: Ein Attribut muss als eindeutig zuordenbare ID verwendet werden können (z. B. E-Mail-Adresse). - Kategorisierung: Ein Attribut muss zur Kategorisierung des SBO verwendet werden können. - Name: Ein Attribut muss für die Benennung des SBO verwendet werden können. - Charakterisierende Attribute: Beliebige viele Attribute müssen zur Charakterisierung des SBO verwendet werden können. - Beziehung: Ein Attribut muss für die Beziehung (falls vorhanden) zwischen zwei SBO verwendet werden. <p>Die aus den Business Objects stammenden Attribute <i>a</i> sollen folgende Kriterien erfüllen:</p> <ul style="list-style-type: none"> - Notification: Durch das Ändern eines Attributwertes soll das Generieren einer Notification ausgelöst werden können. - Bild: Ein Attribut soll ein Bild beinhalten. - Filterbarkeit: Beliebige viele Attribute sollen für eine Filterung verwendet werden können. - Sortierbarkeit: Beliebige viele Attribute sollen für eine Sortierung verwendet werden können.

Die Kriterien, anhand derer die Attribute ausgewählt werden müssen bzw. sollen sind für die Umsetzung des SoNBO-Konzeptes durch eine Anwendung relevant. Diese werden in den nächsten Abschnitten wieder aufgegriffen.

4.1.2 Social Concept und Social Subconcept

Wie bereits erwähnt, werden zueinander ähnliche Business Objects zu Klassen kategorisiert. Social Business Objects werden daher ebenfalls zu Klassen kategorisiert. Eine Klasse wird Social Concept genannt in Anlehnung an die Terminologie („Concept“) von Ontologien aus der Informatik (vgl. Abschnitt

3.2.1.2.2). Ein Social Concept kann in Social Subconcepts klassifiziert werden. Die Social Concepts orientieren sich dabei an den Klassen in den Quellsystemen, wobei auch davon abgewichen werden kann, wenn es technisch möglich und es für den Endanwender sinnvoll ist. Durch diese Hierarchisierung handelt es sich dabei um eine Taxonomie (vgl. Abschnitt 3.2.1.2.2).

Tab. 4.6: Definition (Social Concept und Social Subconcept)

<p>(a) Sinnvoll zusammengehörende Social Business Objects werden zu Social Concepts SC und ggf. weiter in Social Subconcepts SSC kategorisiert. Jedes SBO gehört zu genau einem SC und (falls vorhanden) zu genau einem SSC. Ein SC (oder falls vorhanden ein SSC) besteht aus den Attributen a_j, aus dem auch das zugehörige Social Business Object besteht.</p> <p>Sei $g, d, m \in \mathbb{N}$ und SC_g ein beliebiges Social Concept</p> <p>Es gilt: $SC_g = \{SBO_1, SBO_2, \dots, SBO_d\}$ und $SC_g = \{a_1, a_2, \dots, a_m\}$</p> <p>(analog: SSC)</p> <p>(b) Die Kategorisierung in SC und SSC orientiert sich dabei an den Klassen in den Quellsystemen. Sie können aber auch neu definiert werden, falls die Anforderungen des Unternehmens dies erfordert und es technisch möglich ist.</p> <p>(c) Ein SC besteht aus einer Teilmenge von Attributen aus den Quellsystemen und wird um soziale Funktionen erweitert. Die sozialen Funktionalitäten entstehen durch die Attribute aus den Ursprungs-BO bzw. der Ursprungs-Klasse.</p>

Social Concepts kategorisieren also die Unternehmensinformationen in den Quellsystemen und präsentieren diese Kategorisierung auch dem Endanwender. In Tab. 4.7 ist eine Übersicht über die Verwendung der Begrifflichkeiten in den Quellsystemen und im SoNBO-Konzept dargestellt.

Tab. 4.7: Vergleich der Ebenen und verwendeten Begriffe

Bezeichnung im Quellsystem	Bezeichnung im SoNBO-Konzept	Weitere Bezeichnungen
Klasse	Social Concept	Konzept, Typ, Kategorie
(Subklasse)	Social Subconcept	Subkonzept, Subtyp, Subkategorie
Business Object	Social Business Object	Instanz, Objekt
Attribut	Attribut	
Attributwert	Attributwert	

Die *Social Business Objects* und deren Kategorisierung in *Social Concepts* sind also nicht von den Entwicklern der Quellsysteme vorgegeben, sondern werden in Abhängigkeit der Anforderungen des Unternehmens und den technischen Gegebenheiten gebildet. Zusammengefasst besteht ein Social Concept und damit ein Social Business Object aus einer Teilmenge von Attributen der ursprünglichen Klasse bzw. des Business Objects und wird um soziale Funktionen erweitert, indem ausgewählte Attribute dafür verwendet werden.

4.1.3 Social Network of Concepts und Social Network of Subconcepts

Durch die Erweiterung um soziale Funktionen wie z. B. follow, ist ein Social Concept ein Knoten in einem Netzwerk, das eine Graphstruktur aufweist. Der Graph besteht aus Knoten und Kanten und die Definition von Graphen (vgl. Abschnitt 3.2.1.3) wird auch hier für die Definition verwendet. Dieser

Graph wird Social Network of (Sub-)Concepts genannt und ist eine Ontologie. Die Ontologie im SoN-BO-Konzept wird anhand der Daten in den Datenbanken erstellt und ist auf die Bedürfnisse der Endanwender ausgerichtet. Diese wird in Tab. 4.8 definiert.

Tab. 4.8: Definition (Social Network of Concepts und Social Network of Subconcepts)

<p>(a) Ein Social Concept ist ein Knoten in einem Schema für ein soziales Netzwerk. Dies wird <i>Social Network of Concepts</i> genannt. Das Schema für das soziale Netzwerk weist eine Graphstruktur auf und es besteht somit aus Knoten (= Social Concept) und Kanten (= Beziehungen). Die Kanten besitzen keine Richtung. Sei $j \in \{1, \dots, m\}$; $a_j, \subset BO$; $j, m \in \mathbb{N}$ Es gilt: SoNC = (SC, a_j) (= Graph bestehend aus SC als Knoten und a_j als Kanten) a_j = Kantenattribut</p> <p>(b) Das Social Network of Concepts beschreibt die Strukturen der Informationen in den Quellsystemen als Schema für ein soziales Netzwerk. Es handelt sich um eine Ontologie.</p> <p>(c) Die Attribute aus Tab. 4.5 werden für die Spezifizierung der Social Concepts und somit der Ontologie Social Network of Concepts verwendet. Dadurch werden die sozialen Funktionen (z. B. follow) hinzugefügt. Die Auswahl dieser Attribute für jedes Social Concept wird semantisches Mapping genannt.</p> <p>(d) Die Formulierung der Queries für die benötigten Attribute wird technisches Mapping genannt.</p> <p>Analog: Social Network of Subconcepts</p>
--

Nach der Erstellung der Ontologie wird ein Mapping zwischen den in der Ontologie enthaltenen Informationen in Form von Attributen und den Attributen in den Quellsystemen vorgenommen. Auf diese Weise werden die Attribute aus den Quellsystemen für neue Zwecke (vgl. Tab. 4.5) verwendet. Beispielsweise wird die Kante zwischen zwei Knoten durch ein Attribut des jeweiligen Social Concepts definiert (vgl. Tab. 4.5), die eine Beziehung untereinander haben. Auf diese Weise entsteht das Social Network. Mit diesem Attribut wird also die soziale Funktion *follow* umgesetzt, wodurch diese Funktion nicht hinzuprogrammiert werden muss. In der Rechnung 1725 (vgl. Tab. 4.3) ist dieses Attribut beispielsweise das Feld „Sachbearbeiter = Peter Müller“, wodurch die Rechnung 1725 mit Peter Müller verknüpft wird. Weiterhin werden soziale Funktionen hinzugefügt, indem Attribute in den Quellsystemen ausgewählt werden, die für das Erzeugen von Notifications (vgl. Abschnitt 3.2.1.7) notwendig sind. Bei Notifications handelt es sich um Aktualisierungen eines *Social Business Objects*, die durch die Änderung eines Attributes in den Quellsystemen ausgelöst werden. Durch die Änderung eines Attributes (z. B. das Attribut *Status* der Rechnung 1725 ändert sich von *nicht bezahlt* zu *bezahlt*) wird eine solche *Notification* („Rechnung 1725 wurde von Kunde XY bezahlt.“) vom System generiert, die im Profil (Bereich: *Recent Events*) des Social Business Objects angezeigt wird. Außerdem wird diese Notification in dem Activity Stream der Enterprise Social Software angezeigt, in die der SoNBO-Explorer integriert wird. Dadurch wird die Awareness des Benutzers gegenüber Informationen aus den transaktionalen Systemen gesteigert.

Die Ontologie kann flexibel um weitere Knoten (Social Concepts/Social Subconcepts) erweitert werden. Dadurch existiert bei der Ontologie-Erstellung zunächst eine Zielontologie (wie aus der Pay-as-you-go-Methodology von OBDA bekannt, vgl. Abschnitt 3.2.2.3.3), aus der sich dann eine Startontologie bestehend aus einer Teilmenge der Zielontologie ableitet, die sukzessive um weitere Social Con-

cepts erweitert wird, bis dann die Zielontologie erreicht ist. Die Zielontologie ist dabei nicht statisch zu sehen und kann auch agil angepasst werden, falls sich neue Anforderungen ergeben. Dies wird für das SoNBO-Framework (vgl. Kapitel 5) wieder aufgegriffen.

4.1.4 Social Network of Business Objects

Das Social Network of Business Objects entsteht durch die neue Beschreibung der Daten mithilfe der Ontologie *Social Network of Concepts* und den Daten in den Quellsystemen. Dazu ist ein Mapping der Informationen in der Ontologie zu den Daten in den Quellsystemen notwendig. Das Social Network of Concepts ist also das instanziierte soziale Netzwerk, in dem die ausgewählten Business Objects mit den ausgewählten Attributen zu einem Social Business Object als Knoten in diesem sozialen Netzwerk werden.

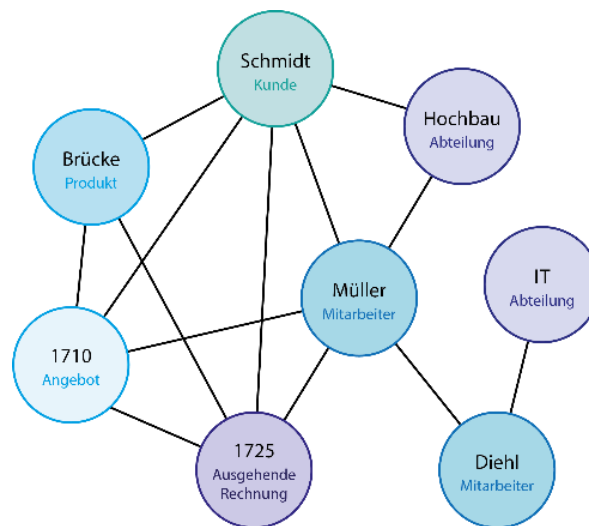


Abb. 4.1: Social Network of Business Objects

Die Social Business Objects werden dann dem Nutzer mit sozialen Prinzipien aus der Social Software präsentiert: Die ausgewählten Attribute werden dem Endanwender als Social Profile angezeigt. Außerdem erhält das Social Business Object soziale Funktionen, indem es Notifications erzeugen und eine Beziehung zu einem anderen Social Business Object haben kann. Dem Endanwender wiederum wird dadurch ermöglicht, durch das Netzwerk von Knoten zu Knoten (also von Social Business Object zu Social Business Object) zu navigieren und die für ihn autorisierten Informationen individuell angezeigt zu bekommen. Dabei sieht der Endanwender immer die Informationen aus der Sicht des jeweiligen *Social Business Objects*. Da bei SoNBO auch die Berechtigungen mitübergeben werden, erhält der Endanwender allerdings auch nur Zugriff auf die Informationen, die dieser lesen darf. Somit entsteht für jeden Nutzer ein *nutzerspezifisches soziales Netzwerk*.

Tab. 4.9: Definition (Social Network of Business Objects)

(a) Ein Social Business Object ist ein Knoten in einem sozialen Netzwerk. Dies wird Social Network of Business Objects genannt. Das soziale Netzwerk weist eine Graphstruktur auf und es besteht somit aus Knoten (= Social Concept) und Kanten (= Beziehungen). Die Kanten besitzen keine Richtung.
 Sei $j \in \{1, \dots, m\}$; $k \in \{1, \dots, l\}$ $a_j, v_{jk} \subset BO_i$; $j, m, k, l \in \mathbb{N}$
 Es gilt: $SoNBO = (SBO, (a_j | v_{jk}))$ (= Graph bestehend aus SBO als Knoten und $(a_j | v_{jk})$ als Kanten)
 a_j = Kantenattribut; v_{jk} = Kantenattributwert
 (b) Das Social Network of Business Objects ist entweder ein virtualisierter oder materialisierter Knowledge Graph, der durch die Beschreibung der Struktur durch das Social Network of Concepts und den Daten aus den Quellsystemen entsteht.

Dieser Graph wird entweder materialisiert oder virtualisiert (Unterscheidung aus Ontologie-basierten Ansätzen bekannt vgl. Abschnitt 3.2.2).

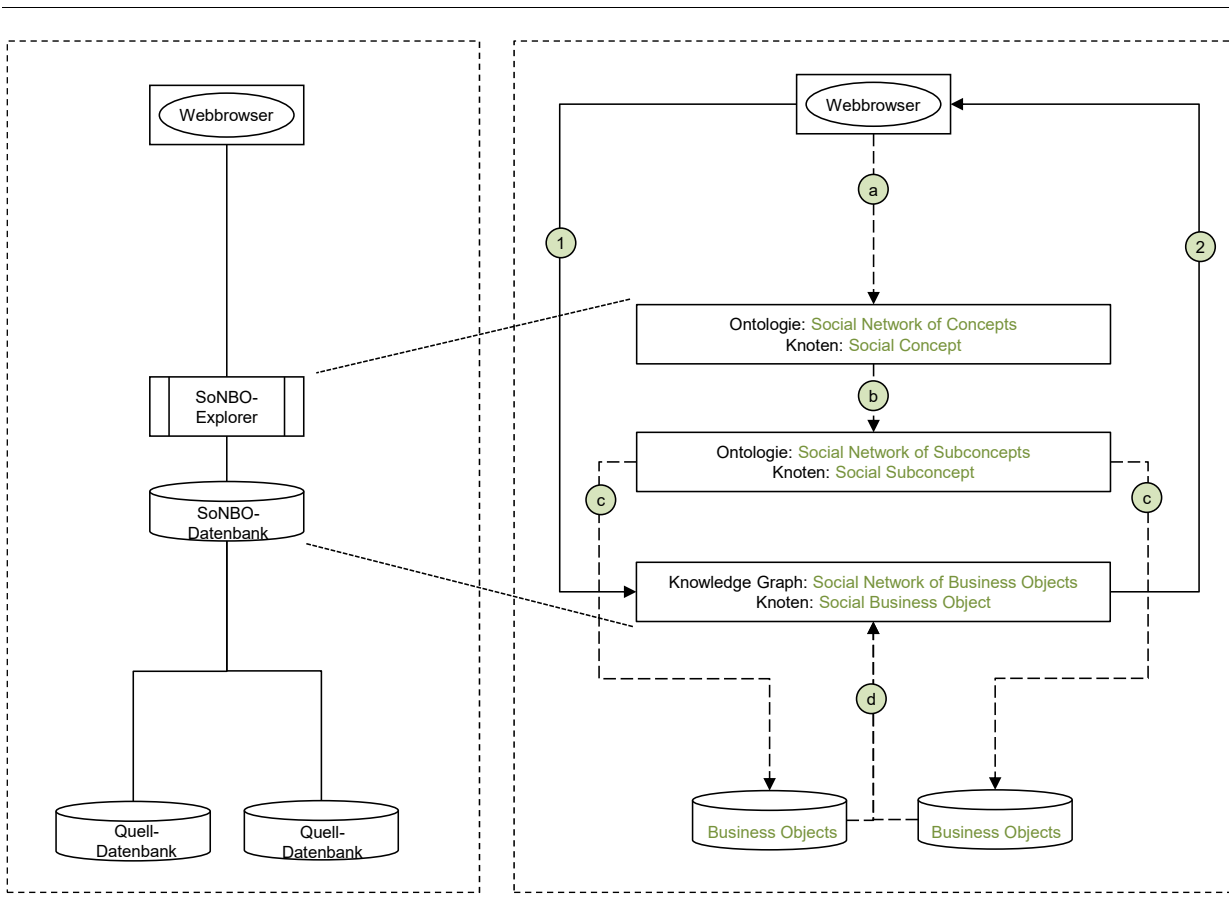


Abb. 4.2: Datenzugriff bei SoNBO durch Materialisierung

Bei einer Materialisierung (vgl. Abb. 4.2) wird der Graph in einer Datenbank abgespeichert und in bestimmten Intervallen mit den Daten aus den Quellsystemen synchronisiert. Der Graph wird im Vorfeld erzeugt, indem durch die Ontologie und den Daten aus den Quellsystemen der Knowledge Graph entsteht (vgl. a, b, c, d in Abb. 4.2). Dieser Knowledge Graph wird dann bei einem Datenzugriff genutzt (vgl. 1, 2 in Abb. 4.2). Der Nachteil ist, dass der Zugriff auf die Daten im Social Network of Business Ob-

ject nicht live erfolgt und die Daten dadurch ggf. ein paar Stunden alt sind. Der Vorteil ist, dass eine Navigation durch das Netzwerk schneller möglich ist.

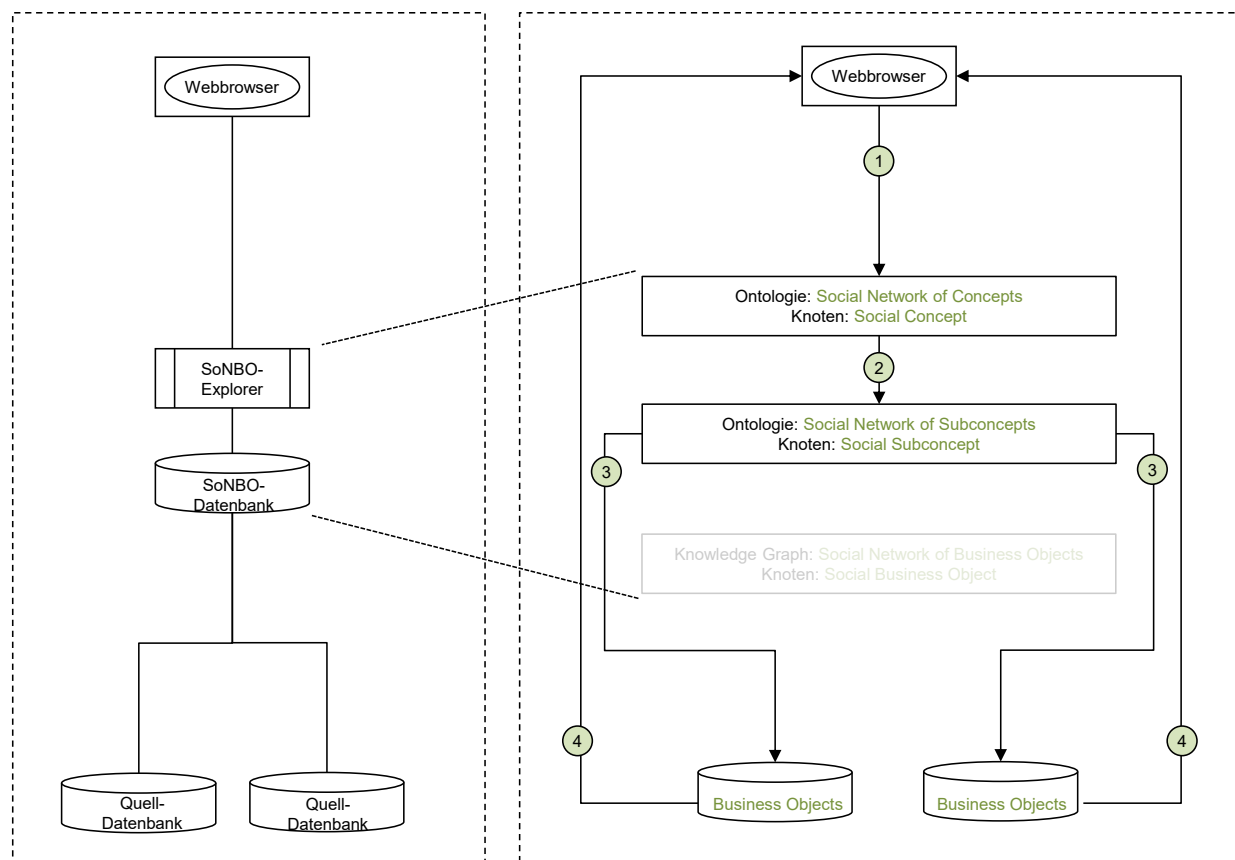


Abb. 4.3: Datenzugriff bei SoNBO durch Virtualisierung

Bei einer Virtualisierung (vgl. Abb. 4.3) wird das Social Network of Business Objects nicht abgespeichert, sondern es werden zur Laufzeit die Informationen aus den Quellsystemen in Verbindung mit der Ontologie als Schema (vgl. 1, 2, 3, 4 in Abb. 4.3) geladen. Der Vorteil ist, dass es sich immer um Live-Daten handelt. Der Nachteil ist, dass der Informationszugriff in der Regel langsamer ist. Außerdem besteht eine Mischung zwischen diesen beiden Möglichkeiten. Zur Laufzeit werden die Knoten, die benötigt werden, zwischengespeichert (Caching), sodass auf diesen Cache zurückgegriffen wird, wenn der Endanwender auf diese Knoten zurücknavigiert. Der Nachteil ist hier, dass das erste Laden lange dauert. Der Vorteil besteht darin, dass sie in der Regel aktueller sind, als wenn sie in bestimmten Intervallen synchronisiert und dann für den Zugriff abgespeichert werden. Die Ontologie zur Beschreibung der Graphstruktur wird in allen Fällen abgespeichert.

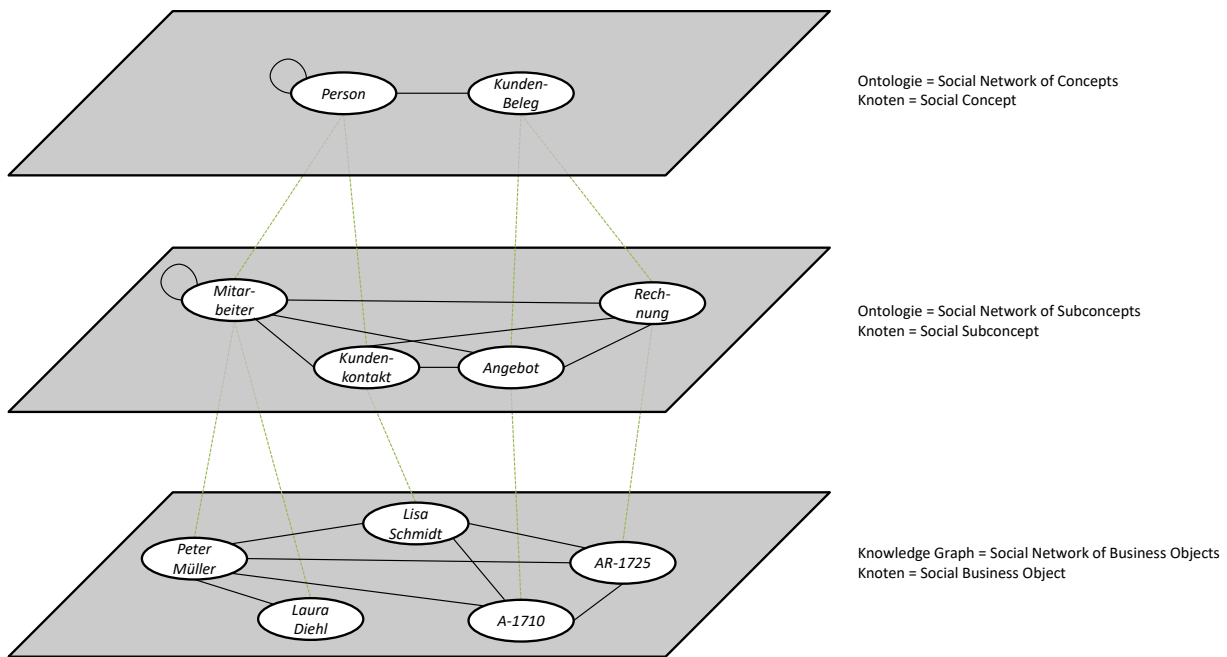


Abb. 4.4: Zusammenhang zwischen den drei Graphen bei SoNBO

Der Zusammenhang zwischen den drei Graphen *Social Network of Concepts*, *Social Network of Subconcepts* und *Social Network of Business Objects* ist in Abb. 4.4 visualisiert (in Anlehnung an die drei Ebenen von Wissen nach Bertalozzi et al. (2001)). Auf der obersten Ebene befindet sich das Social Network of Concepts. Darunter befindet sich das Social Network of Subconcepts, wobei jeder Knoten zu genau einem Social Concept gehört. Auf der untersten Ebene befinden sich die Social Business Objects und jedes Social Business Object gehört zu genau einem Social Subconcept. In Tab. 4.10 ist die Übersicht der hier verwendeten Terminologie abgebildet und die Zuordnung zu den Begriffen aus dem Semantic Web und der Graphentheorie.

Tab. 4.10: Übersicht über die verwendete Terminologie

Semantic Web	SoNBO	Graphentheorie
Ontology (Knoten: Concept)	Social Network of Concepts (Knoten: Social Concept)	Graph
	Social Network of Subconcepts (Knoten: Social Subconcept)	Graph
Knowledge Graph	Social Network of Business Objects (Knoten: Social Business Object)	Graph

Der Knowledge Graph könnte bei SoNBO auch „Social Network of *Social* Business Objects“ heißen. Dadurch, dass das erste “social” allerdings schon deutlich macht, dass es sich um ein soziales Netzwerk handelt, was sich dann auch auf die Business Objects auswirkt, wird bei SoNBO auf das zweite “social” aus Gründen der Einfachheit verzichtet. Bei der Knotenbezeichnung wird allerdings *Social Business Ob-*

ject verwendet, um sich von den *Business Objects* abzugrenzen. Dasselbe gilt für die Bezeichnung des *Social Network of Concepts/Subconcepts*.

4.2 Prinzipien von Social Software im SoNBO-Konzept

In den vorherigen Abschnitten wurden die Prinzipien von Social Software und soziale Funktionen verwendet, um das SoNBO-Konzept zu beschreiben. Die Prinzipien wurden bereits in Abschnitt 3.2.1.7 eingeführt. In diesem Abschnitt werden die Prinzipien von (Enterprise) Social Software mit denen in einer generischen SoNBO-Anwendung verglichen und die Unterschiede und Gemeinsamkeiten aufgezeigt (vgl. Tab. 4.11). Dabei werden lediglich die für SoNBO relevanten Prinzipien aufgeführt. Der Aufbau einer generischen SoNBO-Anwendung wird in den nächsten Abschnitten durch die Beschreibung eines Benutzerfrontends, eines Administrationsfrontends und einer Anwendungssicht vorgestellt.

Tab. 4.11: Vergleich von *Social Software* und einer generischen *SoNBO-Anwendung*

Zweck	Funktionalität	Enterprise Social Software (ESS)	SoNBO-Anwendung
Enterprise Social Network (ESN)	Knoten: Social Profile	Erstellt durch Benutzer; existiert ausschließlich für <i>Persone</i> ; beschreibt eine Person	Erstellt durch Administrator; existiert für <i>Persone</i> ODER weitere Dinge wie z. B. Rechnung XY, Angebot XY, ...; beschreibt eine Person ODER ein Ding
	Kante: Soziale Funktion follow im Social Profile	Erstellt durch Benutzer	Erstellt durch Administrator
	Nutzung: Query by Navigation	Informationszugriff erfolgt durch die Navigation entlang der Social Profiles	Informationszugriff erfolgt durch die Navigation entlang der Social Profiles
Awareness	Activity Stream (= Notifications)	Erzeugt durch System; Teil der ESS	Erzeugt durch System; Teil der ESS, in das der SoNBO-Explorer integriert ist
	Recent Events (= Notification im Social Profile)	/	Erzeugt durch System

Sowohl in ESS als auch in einer SoNBO-Anwendung werden die Prinzipien in der Form von Funktionalitäten verwendet, um ein *Enterprise Social Network* (ESN) zu erzeugen und die *Awareness* der Benutzer zu steigern (vgl. Tab. 4.11). Für das Erzeugen des ESN existieren in beiden Systemen *Social Profiles*, die mit der sozialen Funktion *follow* zu dem ESN verknüpft werden. Für jeden Knoten gibt es ein Social Profile, welches den dahinter stehenden Knoten charakterisiert, und die Kante wird durch *follow* erzeugt. Der Unterschied zwischen den beiden Systemen besteht darin, *wer* diese Elemente für *wen* oder *was* erstellt: Bei ESS wird sowohl das Social Profile als auch die Kante durch den Akteur *Benutzer* erzeugt und das Social Profile existiert auch nur für den Akteur *Benutzer*. Bei SoNBO wird das Social Profile durch den Akteur *Administrator* erstellt und durch die Attribute aus den Quellsystemen befüllt.

Auch die Kante zwischen zwei Knoten wird durch den Akteur Administrator im Rahmen der Ontologie-Erstellung und des Mappings (vgl. Abschnitt 4.1.3 und Kapitel 5) erzeugt. Das Social Profile existiert dabei nicht nur für die *Benutzer* (also für Menschen), sondern auch für *Dinge* (z.B. Rechnung XY). Während die *Erzeugung* der Knoten und Kanten unterschiedlich sind, ist die *Nutzung* des daraus resultierenden Netzwerkes für den Informationszugriff in beiden Systemen ähnlich: In beiden Fällen erfolgt der Informationszugriff durch das Navigieren entlang der Social Profiles nach dem Vorbild *Query by Navigation* (vgl. Abschnitt 3.2.1.4).

Für den zweiten Zweck *Awareness* wird in beiden Systemen der Activity Stream verwendet, der die Notifications der Knoten (bei ESS: Menschen, bei SoNBO: Menschen oder Dinge) aus dem ESN anzeigt. Diese werden in beiden Systemen durch den Akteur *System* automatisch erzeugt, wenn Neuigkeiten bei einem Knoten eintreten. In einer ESS befindet sich der Activity Stream oftmals auf der Startseite (als zentraler Einstiegspunkt) und ist somit Teil der ESS. Eine SoNBO-Anwendung besitzt keinen eigenen Activity Stream, sondern nutzt den Activity Stream der ESS, in das die SoNBO-Anwendung integriert wird. Diese Notifications werden bei SoNBO auch in das Profil des betreffenden Knotens in den Bereich *Recent Events* gepostet, was bei ESS nicht vorgesehen ist.

Prinzipiell können in beiden Systemen auch weitere Funktionen wie *recommend*, *comment* oder *like* (in der Tabelle nicht aufgeführt) Anwendung finden, wenn die Notification im Activity Stream angezeigt wird und der Benutzer diese dann beispielsweise kommentieren kann (z. B. Kommentar von Projektleiter an die Notification „Rechnung wurde bezahlt“: @*Sachbearbeiter* Peter Mueller: Haben Sie das gesehen? Das Geld ist eingegangen und die Rechnung kann ausgebucht werden.)

Abschließend lässt sich festhalten, dass der Akteur (Benutzer, Administrator oder System), der für die Erzeugung des ESN jeweils verantwortlich ist, in den beiden Systemen unterschiedlich ist. Die Idee des Netzwerkes bestehend aus Knoten und Kanten (ESN ist bei SoNBO das Social Network of Business Objects), die Art des Informationszugriffes durch die Navigation und die Steigerung der Awareness ist gleich.

4.3 Benutzerfrontend

Wie bereits erwähnt dient die beschriebene Graphstruktur (Social Network of Business Objects) aus Abschnitt 4.1.4 dem Endbenutzer als soziales Netzwerk zur Navigation von Knoten zu Knoten, um an die Informationen aus den Quellsystemen zu gelangen. In diesem Abschnitt wird der Aufbau eines Benutzerfrontends für eine SoNBO-Anwendung als Mockup vorgestellt. Die Struktur des Benutzerfrontends (vgl. die Bereiche 1 bis 7 in Abb. 4.5) ist für jedes Social Business Object immer gleich, da das jeweilige Social Business Object in Anlehnung an ein Social Profile präsentiert wird. Navigiert der Nutzer also auf einen weiteren Knoten, werden die jeweiligen Bereiche mit den Informationen des jeweiligen Social Business Objects befüllt. Diese Art der Navigation für den Informationszugriff kann als QbN eingeordnet werden (vgl. Abschnitt 3.2.1.4).

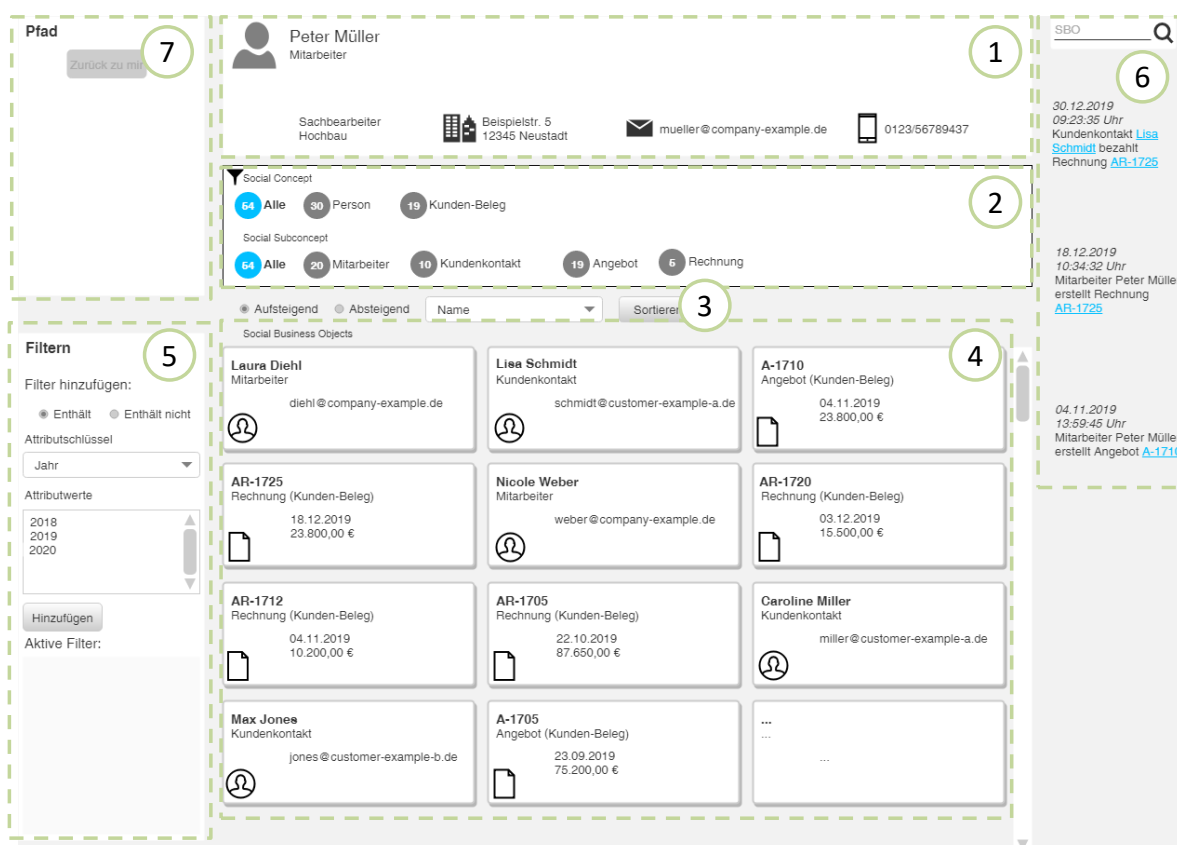


Abb. 4.5: Benutzerfrontend eines SoNBO-Explorers als Mockup

Das Benutzerfrontend lässt sich in folgende Bereiche unterteilen (vgl. Abb. 4.5): Im (1) oberen Bereich („Eigenschaftenbereich“) befindet sich der Name des Social Business Objects (Peter Müller) und das zugehörige Social Concept (Mitarbeiter). Außerdem befinden sich dort die Attribute, die das Social Business Object charakterisieren, welches eine Teilmenge von Attributen des zugehörigen Business Objects ist. Im (4) unteren Bereich („Ergebnisbereich“) befinden sich die mit diesem Social Business Object verknüpften Social Business Objects. Die Verknüpfung kommt zu Stande, wenn im Business Object Rechnung beispielsweise das Attribut Sachbearbeiter mit dem Attributwert auf ein anderes Business Object verweist (vgl. Tab. 4.3). In diesem Fall wird also dieses Attribut für die Herstellung der Beziehung verwendet (= Kantenattribut) (vgl. Tab. 4.5). Im (2) mittleren Bereich („Filterbereich I“) werden die Kategorisierungen der Social Business Objects (also die Social Concepts und Social Subconcepts) aus dem Ergebnisbereich angezeigt. Die obere Reihe zeigt die Social Concepts und die untere Reihe die Social Subconcepts an. Der Nutzer kann jeweils ein Social (Sub-)Concept auswählen und der Ergebnisbereich zeigt nur noch die Social Business Objects an, die zu diesem Social (Sub-)Concept gehören. Dies dient also auch gleichzeitig zur Filterung der Social Business Objects. Der Ergebnisbereich kann auch nach Attributwerten sortiert werden (Bereich 3, „Sortierbereich“). Im (5) linken Bereich („Filterbereich II“) gibt es weitere Filtermöglichkeiten für den Ergebnisbereich. In diesem Beispiel kann nach

Jahr gefiltert werden. Auf der (6) rechten Seite („Recent-Events-Bereich“, vgl. Abschnitt 3.2.1.7) werden Änderungen, die das Social Business Object *Peter Müller* betreffen, als Notification gepostet. Die Notification wird nach einem vorgefertigten Schema erzeugt, wenn sich zuvor ausgewählte Attributwerte in den zugehörigen Business Objects ändern (vgl. Tab. 4.3). Beispielsweise kann sich der Status einer Rechnung von offen (also nicht bezahlt) zu abgeschlossen (also bezahlt) ändern. Diese Information wird dann in den rechten Bereich des Profils der Rechnung, des Kunden und des Mitarbeiters gepostet und ist öffentlich (jeder, der auf das Social Business Object navigiert und die Berechtigung besitzt, kann dies sehen):

- Profil Kunden-Rechnung: „tt.mm.jjjj: Kunden-Rechnung XY wurde von Mitarbeiter XY freigegeben.“
- Profil Kunde: „tt.mm.jjjj: Kunden-Rechnung XY wurde von Mitarbeiter XY freigegeben.“
- Profil Mitarbeiter: „tt.mm.jjjj: Mitarbeiter XY gibt die Kunden-Rechnung XY frei.“

Das Benutzerfrontend kann auch in das ECS (Enterprise Collaboration System) des Unternehmens (z. B. HCL Connections) integriert werden, das die Funktionalität einer *Enterprise Social Software* umfasst (vgl. Abb. 3.17 auf S. 52). Wenn das Enterprise Collaboration System einen Activity Stream (vgl. Abschnitt 3.2.1.7) besitzt, in das die Neuigkeiten der Nutzer gepostet werden, können diese generierten Notification auch in den Activity Stream des jeweiligen Mitarbeiters gepostet werden (z. B. dass die Rechnung bezahlt wurde):

- Activity Stream des Mitarbeiter XY:
 - „tt.mm.jjjj: Kunden-Rechnung XY wurde von Kunde XY bezahlt.“
 - Oder: „tt.mm.jjjj: Kunden-Rechnung XY wurde von Mitarbeiter XY freigegeben.“

Diese Notifications sind dann privat und können nur von dem eingeloggtten Mitarbeiter eingesehen werden. Außerdem kann dieses Frontend an sich auch in das ECS integriert werden. Auf diese Weise gelangen die Informationen aus den transaktionalen Systemen (z.B. ERP-System) in ein Kollaborationssystem. Dadurch kann auch automatisch die Benutzerauthentifizierung für den nutzerspezifischen Knowledge Graphen durchgeführt werden, weil die SoNBO-Anwendung den Benutzer durch das Einloggen auf der Plattform direkt erkennt. Es soll nochmal betont werden, dass der Activity Stream sich *nicht* im Frontend des SoNBO-Explorers befindet, sondern Teil des *Enterprise Collaboration Systems* ist (vgl. Abschnitt 4.2), in das der SoNBO-Explorer integriert wird. Zwischen diesen beiden Softwaretypen fand in Unternehmen in der Vergangenheit in der Regel keine Integration statt.

An dieser Stelle sei nochmal erwähnt, dass jeder Nutzer dieses Benutzerfrontend sieht (sofern es die Berechtigungen zulassen) und somit auch jeder Mitarbeiter, der nicht Peter Müller ist, diese Informationen aus seiner Perspektive auf diese Weise einsehen kann. Dem Nutzer wird nun nach der Sichtung der Informationen über Peter Müller und die Informationen, die ihn betreffen, ermöglicht, auf einen weiteren Knoten aus dem Ergebnisbereich zu navigieren. Nach einem Klick darauf (z. B. ausgehende Rechnung 1725) werden dem Nutzer die Informationen mit derselben Struktur aus der Sicht der Rechnung angezeigt und der Navigationspfad wird in Bereich (7) angezeigt („Pfadbereich“). Neben dem Le-

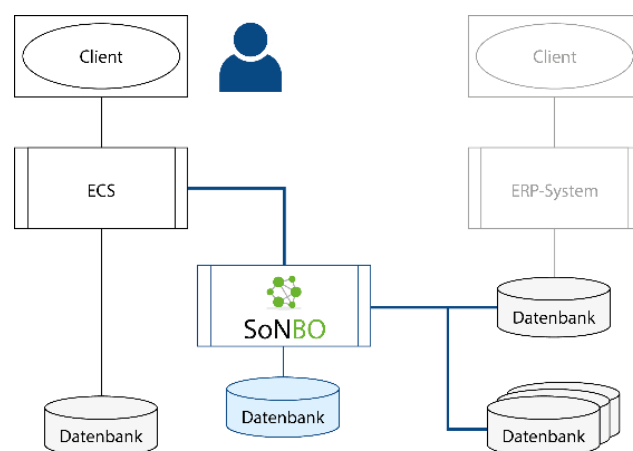
se-Zugriff werden auch CRUD-Funktionalitäten ermöglicht. CRUD steht für create, read, update und delete. Im Vordergrund steht die read-Funktionalität, die immer vorhanden ist. Außerdem wird dem Nutzer auch das Erstellen, Aktualisieren und Löschen der Datensätze bedingt ermöglicht. Im Vordergrund steht bei SoNBO allerdings die Informationsbereitstellung und es sind lediglich kleinere Änderungen bzw. Veränderungen der Inhalte einzelner Attribute (z. B. Rechnung genehmigen) möglich. Für die Dateneingabe (z. B. von einer Lieferantenrechnung, von einem Kundenkontakt, ...) sind nach wie vor die Clients der Quellsysteme notwendig.

4.4 Administrationsfrontend

Für die Erzeugung des Social Network of Business Objects, durch das der Benutzer navigieren kann, ist wie bereits erwähnt die Ontologie *Social Network of Concepts* notwendig. Diese Ontologie wird zu Beginn angelegt und kann flexibel um weitere Knoten erweitert werden. Dies erfolgt über das Administrationsfrontend. Über dieses Frontend legt der Administrator Social Concepts und Social Subconcepts an und gibt die relevanten Attribute (vgl. Tab. 4.5) aus der zugehörigen Klasse an. Anschließend erfolgt das semantische Mapping, indem die Attribute, die für die Ontologie notwendig sind, mit den Attributen der zugehörigen Klasse verknüpft werden. Außerdem werden die Datenbankabfragen dazu hinterlegt (technisches Mapping). Dies wird detailliert in Kapitel 5 (SoNBO-Framework) beschrieben.

4.5 Anwendungssicht

Das SoNBO-Konzept ist technologieagnostisch. Im Rahmen des SoNBO-Forschungsprojektes wurde allerdings ein SoNBO-Explorer mit einer konkreten Technologie entwickelt, welcher in Kapitel 5 vorgestellt und erklärt wird. In diesem Abschnitt soll die Anwendungssicht ohne einen konkreten Technologiebezug beschrieben werden. Der SoNBO-Explorer integriert die Informationen aus den Quellsystemen (vgl. Abb. 4.6 und Abb. 4.7).



ECS = Enterprise Collaboration System ERP-System = Enterprise Resource Planning System

Abb. 4.6: Anwendungssicht (einfache Darstellung)

Dieser stellt zum einen ein Administrationsfrontend bereit, mit dem die Ontologie, also das Social Network of Concepts, konfiguriert werden kann (dieser Client ist nicht in der Abbildung ersichtlich). Außerdem stellt dieser das Benutzerfrontend bereit, das eine Webpage sein kann. Beispielsweise stellt HCL Notes/Domino die Technologie XPage dafür bereit. Darin werden lediglich die Bereiche (wie in Abb. 4.5 und in dem zugehörigen Abschnitt beschrieben) definiert, sodass beim Aufrufen des jeweiligen Knotens die Informationen dieses Social Business Objects in die jeweiligen Bereiche geladen werden. Der Datenzugriff erfolgt entweder über den materialisierten Graphen, in welchem sich die Daten aus den Quellsystemen bereits als Social Network of Business Objects befinden, oder über den virtualisierten Graphen, der zur Laufzeit entsteht, indem mit der hinterlegten Ontologie als Schema auf die Daten in den Quellsystemen zugegriffen wird.

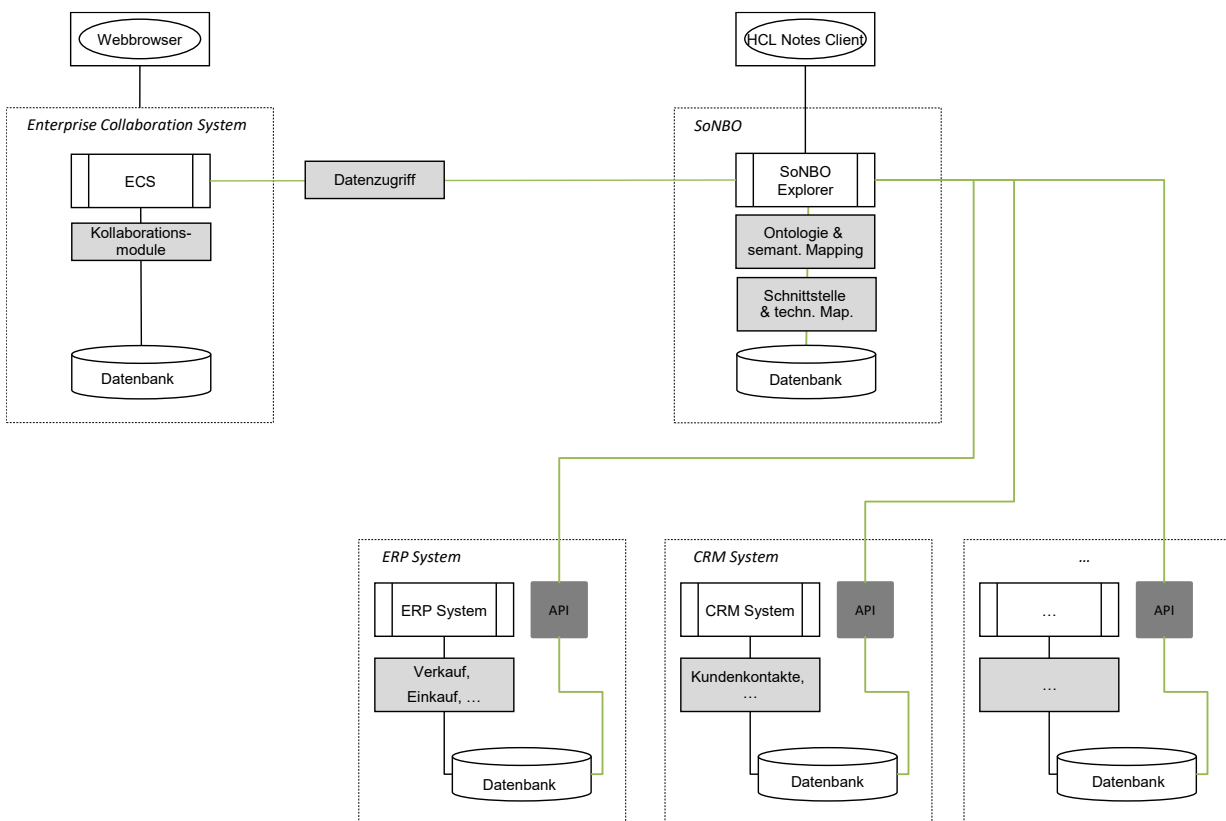


Abb. 4.7: Anwendungssicht (detaillierte Darstellung)

Die Webapplikation kann als eigenständige Anwendung von den Mitarbeitern im Unternehmen genutzt werden. Diese kann aber auch als iFrame in das Kollaborationssystem des Unternehmens (Enterprise Collaboration System wie z. B. HCL Connections) integriert werden, sodass die Mitarbeiter aus diesem System heraus, das in vielen Unternehmen auch als Portal und somit als Einstieg in die täglichen Informationen dient, auf die Informationen in den transaktionalen Systemen zugreifen können. Durch diese Integration kann der SoNBO-Explorer auch die Funktionalitäten der Kollaborationsplattform verwenden. Beispielsweise kann der Activity Stream des jeweiligen Kollaborationssystems ge-

nutzt werden. In diesen können die Social Business Objects dann vordefinierte Änderungen als Notification posten, sodass die Awareness gegenüber den Informationen in den Quellsystemen gesteigert wird. Ein weiterer Vorteil bei der Integration des SoNBO-Explorers in das Kollaborationssystem besteht in der Identifikation des Nutzers durch die Anmeldung an das Kollaborationssystem, sodass idealerweise eine Anmeldung an die Quellsysteme nicht mehr notwendig ist, da im Hintergrund die Anmeldung für den Nutzer vorgenommen wird. Dieser Single-Sign-on (SSO) ist eine der technischen Herausforderungen bei der Integration. Eine weitere technische Herausforderung ist die Einrichtung des Zugriffs auf die Quellsysteme. Der SoNBO-Explorer integriert die Informationen aus den Quellsystemen idealerweise mithilfe einer API. Web Services sind beispielsweise geeignet für die Datenabfrage (Tabellen und Queries) und für die Manipulation geeignet. Dies erfolgt z. B. über die Abfrage mit dem Open Data Protocol (OData). Allerdings werden keine tiefergehenden Operationen unterstützt, was aber für SoNBO erstmal nicht notwendig ist. Falls eine API nicht vorhanden ist, kann auch direkt auf die Datenbank zugegriffen werden. Alternativ können die Daten, die für SoNBO benötigt werden (also die Attribute) manuell in einem Datenformat bereitgestellt werden, sodass diese von SoNBO verarbeitet werden können. Dafür kommen beispielsweise JSON-Strings (JavaScript Object Notation) in Frage, wodurch die für SoNBO relevanten Attribute als Key-Value-Pärchen übersetzt und somit in ein einheitliches Datenformat übertragen werden. Der Vorteil beim Benutzen einer existierenden API besteht darin, dass auch die Berechtigungen auf die Daten bei der Anmeldung des Endanwenders mit übergeben werden. Dadurch wird sichergestellt, dass dieser auch nur die Informationen sieht, die dieser sehen darf. Beim direkten Zugriff auf die Datenbanken muss dies noch zusätzlich eingerichtet werden. Diese Berücksichtigung der Berechtigungen ist eine weitere technische Herausforderung für ein Unternehmen. Für den Zugriff auf die Daten ist eine korrekte Autorisierung notwendig, die die Berechtigungen des aktuellen Nutzers widerspiegelt. Eine ideale Lösung wäre OAuth (Open Authorization), das eine standardisierte und sichere API-Autorisierung für Anwendungen ermöglicht. Allerdings funktioniert dies nur in der Microsoft-eigenen Azure-Umgebung. Für On-Premises-Betriebssysteme gibt es zwei weitere Möglichkeiten: Mittels eines Keystores können die Berechtigungen mitübergeben und die Passwörter abgespeichert werden. Dies ist allerdings sicherheitstechnisch bedenklich. Eine weitere Option ist Basic Auth, wobei die Berechtigungen ebenfalls übergeben werden, sich der Nutzer allerdings für jede Session neu am Quellsystem (hier: NAV) anmelden muss.

Zusammengefasst müssen für die technische Integration folgende Aspekte berücksichtigt werden: Bereitstellung der Daten in einem einheitlichen Datenformat, Abfragen der Daten mit dem Schema der Ontologie, Single-Sign-on, Übergabe der Berechtigungen und ggf. Speicherung des Knowledge Graphen.

4.6 Vergleich SoNBO zu Ontologie-basierten Ansätzen aus dem Semantic Web

Ausgehend von der Definition eines Business Objects wird in diesem Abschnitt das SoNBO-Konzept für den weiteren Teil der Dissertation beschrieben: Das Business-Object-Konzept kommt aus der Informatik (Softwareentwicklung von betriebswirtschaftlichen Anwendungssystemen). Insbesondere SAP griff

dieses Konzept auf. In der Wirtschaftsinformatik wird das Konzept zudem für die Softwareeinführung und -orchestrierung verwendet. Im SoNBO-Ansatz wird ebenfalls auf das Business-Object-Konzept zurückgegriffen. Diesen vier Ansätzen ist das grundlegende Verständnis eines Business Objects als Mittel zur Strukturierung der Unternehmensinformationen gemein. Bei den ersten drei Ansätzen erfolgt die Strukturierung transparent (für den Nutzer im Sinne der Informatik) im Backend, während bei SoNBO die Struktur der Objekte für den Nutzer beim Zugriff auf die Daten deutlich wird.

SoNBO (Social Network of Business Objects) ist ein Ontologie-basierter Integrationsansatz, bei dem ein materialisierter oder virtualisierter Knowledge Graph aus der Ontologie und den Daten aus den Quellsystemen erzeugt wird. Dieser Knowledge Graph besteht aus den Business Objects der Quellsysteme, die um soziale Funktionen angereichert werden und somit zu Social Business Objects werden. Ein Knoten in dem Graph entspricht einem Social Business Object. Dem Mitarbeiter wird auf diese Weise ermöglicht, durch den Knowledge Graph, der wie ein soziales Netzwerk aufgebaut ist, zu navigieren und die Informationen aus der Sicht des jeweiligen Knotens zu betrachten.

Durch die Beschreibung von SoNBO in diesem Kapitel wurden die Gemeinsamkeiten zu den existierenden Semantic-Web-Konzepten deutlich: Bei SoNBO werden die Daten in den Quellsystemen ebenfalls als Graph betrachtet, der in Verbindung mit einer Ontologie entsteht (= Enterprise Knowledge Graph). Dieser Graph entsteht entweder virtualisiert, indem die Abfragen auf die Quellsysteme übersetzt werden (z. B. wie bei OBDA) oder er wird materialisiert, indem er in einer Datenbank (in der Regel Graphdatenbank z. B. wie bei EKG) abgespeichert wird. Die Ontologie wird bei SoNBO in zwei Ebenen (Social Network of Concepts und Social Network of Subconcepts) dargestellt. Bei den Konzepten aus dem Semantic Web und den existierenden Ontologie-Editoren werden diese in einem Graph visualisiert. Der Vorteil bei der Visualisierung in einem Graph ist, dass dieser besser für Maschinen lesbar ist. Bei SoNBO werden die beiden Ebenen genutzt, um die Ontologie mit den Mitarbeitern zu entwickeln und hat somit diesen anderen Zweck.

SoNBO weist weitere Merkmale von diesen Ontologie-basierten Ansätzen auf. Der bei SoNBO entstehende Knowledge Graph besitzt die Merkmale eines Enterprise Social Network mit der entsprechenden Funktionalität von Social Software (z. B. follow). Dadurch werden die Informationen als Social Business Objects auch mithilfe des Konzepts von sozialen Profilen beschrieben und dem Nutzer zur Verfügung gestellt. Dadurch soll ein intuitiver Informationszugriff für den Anwender entstehen, da diese Art des Informationszugriffes durch die existierenden sozialen Netzwerke (z. B. Facebook) bereits bekannt ist. Außerdem kann SoNBO dadurch nahtlos in das existierende Kollaborationssystem und somit in den täglichen Informationszugriff integriert werden. Ein weiteres Merkmal von SoNBO ist die Entwicklung der Ontologie *Social Network of Concept*, da es sich dabei um ein Customising handelt, bei dem die Anforderungen des Unternehmens durch diese explizite Vorgabe zusätzlich zu den technischen Voraussetzungen (Aufbau der Quellsysteme etc.) miteingebunden wird. Dadurch wird eine größtmögliche Anpassung an die Nutzer erreicht wie es in der Wirtschaftsinformatik gefordert wird. Der entstehende Knowledge Graph als Social Network of Business Objects ist zudem auch für jeden Nutzer einzigartig (benutzerspezifisch), da sich das Netzwerk durch die übergebenen Berechtigungen

an jeden Benutzer anpasst und orientiert (Voraussetzung: es erfolgt eine Benutzerauthentifizierung). Dies wird zum einen bei der Navigation des Nutzers durch das Netzwerk deutlich, aber auch bei der Sicht auf die Informationen aus der Perspektive des jeweiligen Social Business Objects. Der in der Literatur beschriebene (vgl. Abschnitt 3.2.2.5) Enterprise Knowledge Graph ist demgegenüber nicht benutzerspezifisch, da in der Regel eine Ontologie und der damit verbundene Knowledge Graph in Abhängigkeit der existierenden Quellsysteme erzeugt werden und die Sicht des Mitarbeiters kaum Beachtung findet. Somit sind die Semantic-Web-Konzepte vor allem in der Informatik angesiedelt, während mit SoNBO diese Art des Ontologie-basierten Integrationskonzeptes in die Wirtschaftsinformatik übertragen wurde. Darin liegt auch ein weiterer Unterschied begründet, welcher die verwendete Technologie betrifft. Während die existierenden Ansätze aus dem Semantic Web immer in Verbindung mit den dort verwendeten Technologien (RDF, OWL, SPARQL, ...) betrachtet werden und sich die Modelle, Methoden und Tools vor allem auf diese Techniken beziehen, liegt ein weiterer Fokus bei SoNBO auch auf dem Einbezug der Mitarbeiter im Unternehmen, wodurch das SoNBO-Konzept ein Framework bestehend aus Modellen, Methoden und Tools bereitstellt, das dies ermöglicht. Dies wird in dieser Dissertation entwickelt und vorgestellt (vgl. Kapitel 5). Dadurch kann das SoNBO-Konzept zunächst auch technologieagnostisch (wie in diesem Kapitel) betrachtet werden. Bezüglich der technischen Umsetzung kann dies dann mit existierenden Semantic-Web-Technologien (z. B. Graphdatenbanken für die Speicherung der Ontologie) oder mit weiteren Technologien (z. B. Dokumentendatenbanken von HCL für die Speicherung der Ontologie) umgesetzt werden.

4.7 Zusammenfassung

In diesem Kapitel wurde das SoNBO-Konzept als ein Ontologie-basierter Integrationsansatz beschrieben, der um soziale Funktionen erweitert wird. Außerdem berücksichtigt SoNBO bei der Entwicklung der Ontologie nicht nur die Struktur der Daten in den Quellsystemen, sondern auch die Anforderungen der Nutzer (= *unternehmensspezifische Ontologie*). Durch die Ontologie wird (1) eine sinnvoll gewählte Teilmenge von Daten aus den zu integrierenden Quellsystemen beschrieben. Dadurch erfolgt bei der Integration eine Konzentration auf wesentliche, für den Nutzer ausgerichtete Informationen. Außerdem erfolgt (2) durch die Ontologie eine Aufbereitung der Daten als soziales Netzwerk. Dadurch wird ein für den Nutzer intuitiver Informationszugriff und eine zusammenhängende und konsistente Sicht auf die Informationen ermöglicht. Der daraus resultierende Knowledge Graph ist für jeden Benutzer einzigartig (= *userspezifischer Knowledge Graph*). Somit unterscheidet sich SoNBO von den existierenden Ontologie-basierten Ansätzen aus dem Semantic Web und mit diesem Abschnitt ist der Grundstein für die Entwicklung des Frameworks gelegt.

5 SoNBO-Framework

In diesem Kapitel wird das zweite Ergebnis¹³ dieser Dissertation vorgestellt: Das *SoNBO-Framework* unterstützt mit seinen Modellen, Methoden und Tools einen SoNBO-Einführungsberater¹⁴ (entspricht *SoNBO-Experte* in Tab. 5.1 auf S. 131), das Customising einer SoNBO-Standardsoftware (*SoNBO-Explorer*) für ein Unternehmen durchzuführen. Dabei werden die Anforderungen des Unternehmens und die technischen Gegebenheiten berücksichtigt. Das Customising umfasst die Entwicklung einer auf das jeweilige Unternehmen angepassten Unternehmensontologie, das Mapping der Informationen in der Ontologie auf die Informationen in den Quellsystemen (Faktenbasis) und die anschließende Konfiguration im SoNBO-Explorer. Für die Durchführung dieser Schritte unterstützen Modelle, Methoden und Tools. Das Framework ist somit als Vorgehensempfehlung für den SoNBO-Experten im Rahmen eines SoNBO-Einführungsprojektes konzipiert.

Das Framework wurde entwickelt mit Hilfe der Erkenntnisse aus dem *Key Case Vössing* (vgl. Abschnitt 3.1) sowie den Ergebnissen der *Literaturanalyse* zu Ontologie-basierten Integrationsansätzen (vgl. Abschnitt 3.2) und nutzt die *Konzeptualisierung* des SoNBO-Integrationsansatzes (vgl. Kapitel 4). Außerdem fließen die Erfahrungen aus zwei *Konfigurations-Projekten* für universitätseigene SoNBO-Explorer (CRM- und ERP-System) mit ein und die Ergebnisse dieser Projekte wurden in begutachteten Publikationen beschrieben (Gebel-Sauer & Schubert, 2019b; Götz & Gebel-Sauer, 2018). Die beiden *Konfigurations-Projekte* wurden inspiriert durch das Vorbild aus dem *Key Case*. Im Rahmen von zwei Qualifikationsarbeiten wurde jeweils ein Testsystem entwickelt und unter der Leitung der Autorin der Dissertation konfiguriert. Als Quellsystem diente zuerst ein CRM-System (*Konfigurations-Projekt 1*) und anschließend ein ERP-System (*Konfigurations-Projekt 2*). In den entsprechenden Abschnitten, in denen die Inhalte des Frameworks detailliert ausgeführt werden, wird erklärt, aus welchen dieser Bereiche (Key Case Vössing, Literaturanalyse, Konzeptualisierung SoNBO oder Konfigurations-Projekt) die Erkenntnisse für das SoNBO-Framework stammen. Diese Bereiche sind auch zeitlich sortiert im Anhang als Dokumentation des Forschungsprozesses aufgelistet (vgl. A.2).

Das Kapitel ist folgendermaßen aufgebaut: Zunächst werden die Begriffe *Modell*, *Methode* und *Tool* definiert, da diese im Kontext vom SoNBO-Framework verwendet werden. Anschließend wird das Framework zunächst als Ablaufschema und anschließend im Detail mit den Modellen, Methoden und Tools vorgestellt. Das SoNBO-Framework wurde bereits in der Praxis evaluiert. Die Ergebnisse und Erkenntnisse aus der Evaluation werden in Kapitel 6 vorgestellt.

¹³ Ergebnis 1 = SoNBO-Konzept (Kapitel 4); Ergebnis 2 = SoNBO-Framework (Kapitel 5)

¹⁴ Für SoNBO-Projekte gelten durch den Einsatz einer Standardsoftware ähnliche Voraussetzungen wie für ERP-Projekte: Die Entwicklung der individuellen Unternehmensontologie und das Customising der Anwendung sollte von spezialisierten Einführungsberatern begleitet werden (Scheer, 1998).

5.1 Definition Modell, Methode und Tool

In diesem Abschnitt werden die im SoNBO-Framework enthaltenden Artefakte basierend auf der wissenschaftlichen Literatur allgemein definiert. Die konkrete Verwendung im SoNBO-Framework wird in den entsprechenden Abschnitten 5.4, 5.5 und 5.6 erklärt.

Modell

Modelle stellen etwas (ausgehend von einem Original) dar und haben einen Zweck (Bretzke, 1980; O. Thomas, 2005). Ein Zweck ist in der Regel das Ableiten von Handlungsempfehlungen. Diese Handlungen sollen dann auf eine Realität einwirken und diese verändern. In der Wirtschaftsinformatik werden Modelle bzw. die modellhafte Untersuchung häufig eingesetzt, um die Komplexität eines Untersuchungsgegenstandes beherrschbar zu machen (O. Thomas, 2005). Neben dem Zweck von Modellen sind auch die Zielgruppe und die Zeit relevant. Diese dreifach-pragmatische Relativierung des Modellbegriffes führt dazu, dass es bei der Betrachtung eines Modells nicht nur darum geht, was es darstellen soll, sondern auch wozu, für wen und wann es entstand (Stachowiak, 1973; O. Thomas, 2005).

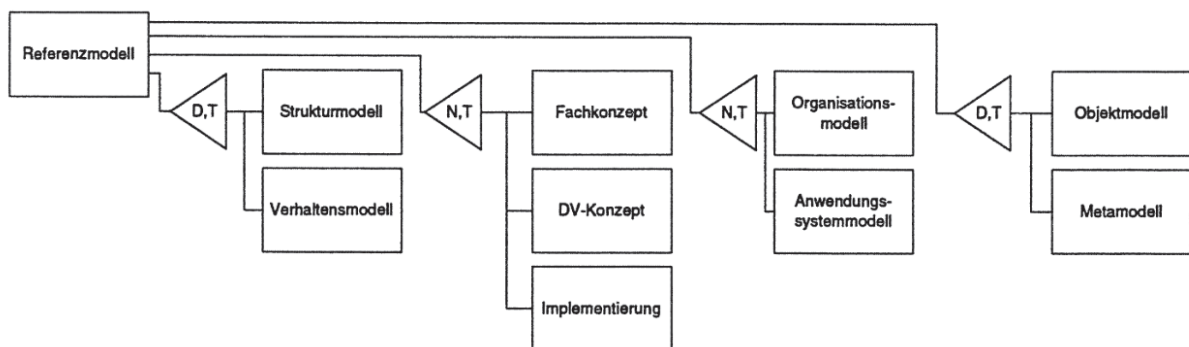


Abb. 5.1: Klassifikation von Referenzmodellen (Schütte (1998, p. 183) in Rosemann & Schütte (1998, p. 24))

Referenz-Informationsmodelle, die auch verkürzt als Referenzmodelle bezeichnet werden, haben das Ziel, allgemeingültige betriebliche Sachverhalte zu repräsentieren. Dieser Anspruch auf Allgemeingültigkeit wird in der Regel für eine Klasse von Unternehmen erhoben. Referenzmodelle können in eine Klassifikation von Informationsmodellen eingeordnet werden (vgl. Abb. 5.1). Dabei werden die wichtigsten Klassifikationsmöglichkeiten von Referenzmodellen herausgehoben (Rosemann & Schütte, 1998). Im Folgenden wird das Referenzmodell der Einfachheit halber als Modell bezeichnet.

Methode

Allgemein wird unter einer Methode die Art und Weise des Vorgehens verstanden, die sich als ein Mittel zur Zielerreichung durch eine bestimmte Auswahl von Instrumenten auszeichnet (Chmielewicz, 1994; Wilde & Hess, 2006). In der Wirtschaftsinformatik werden zwischen Forschungsmethoden (vgl. Abschnitt 2.2) und Entwicklungsmethoden unterschieden. Während Forschungsmethoden ein Instru-

ment zur Erkenntnisgewinnung sind und daher in dieser Dissertation in Abschnitt 2.2 behandelt wurden, sind Entwicklungsmethoden (oder auch: Methoden der Informationssystemgestaltung) auch Untersuchungsgegenstand der Wirtschaftsinformatik (Wilde & Hess, 2006). In dem Framework handelt es sich um die Entwicklung von Methoden zur Informationssystemgestaltung (hier: SoNBO-Explorer), die im Folgenden der Einfachheit halber nur als *Methode* bezeichnet werden.

Tool

Software-Tools sind nach Hildebrand (1990) „Computerprogramme, die die Entwicklung von Software unterstützen“. Ein Software-Tool ist also kein eigenständiges betriebswirtschaftliches Anwendungssystem, sondern wird dazu verwendet, bei der Entwicklung eines solchen zu unterstützen. In der Dissertation geht es nicht um die Entwicklung einer Individualsoftware, sondern um die Unterstützung des Konfigurationsprozesses einer Standardsoftware. Die Definition *Software-Tool* (im Folgenden der Einfachheit halber nur *Tool*) wird in diesem Framework auf die Unterstützung in diesem Konfigurationsprozess übertragen. Es wird also unter Tool die Software verstanden, die bei diesem Konfigurationsprozess unterstützt.

5.2 SoNBO-Explorer

In diesem Abschnitt wird der Aufbau des SoNBO-Explorers vorgestellt, der mithilfe des SoNBO-Frameworks für ein beliebiges Unternehmen konfiguriert werden soll. Außerdem werden die Erfahrungen aus den vergangenen *Konfigurations-Projekten 1* und *2* vorgestellt, die für die Entwicklung des SoNBO-Frameworks grundlegend waren.

5.2.1 Hintergrund

Der *SoNBO-Explorer* ist eine Standardsoftware, die ein Benutzerfrontend und ein Konfigurationsfrontend (auch: Administrationsfrontend) beinhaltet. Das SoNBO-Konzept ist technologieagnostisch, aber da das Framework in der Praxis evaluiert werden sollte, ist ein Technologiebezug notwendig und es wurde eine konkrete Anwendung (*SoNBO-Explorer*) entwickelt. Die für die Evaluation zur Verfügung stehende Anwendung wurde in HCL Notes im Rahmen von zwei Qualifikationsarbeiten (Götz, 2018; Riedle, 2018) erstellt. Die Ereignisse sind im Anhang (vgl. Dokumentation des Forschungsprozesses in A.2) aufgelistet. Die Anwendungen wurden in Zusammenarbeit mit der Autorin der Dissertation für ein CRM-System und anschließend für ein ERP-System konfiguriert. Exemplarische Resultate aus diesen *Konfigurations-Projekten* befinden sich im Anhang (vgl. Anhang A.3 und A.4). Die Projektergebnisse wurden an Konferenzen vorgestellt und in den entsprechenden Tagungsbänden veröffentlicht (Gebel-Sauer & Schubert, 2019b; Götz & Gebel-Sauer, 2018).

5.2.2 Aufbau des SoNBO-Explorers

Mit der ersten Masterarbeit wurde die Logik für alle Komponenten (vgl. Kapitel 4) im Frontend entwickelt, bis auf die *Recent Events* (Ereignisse, die kürzlich zum gerade angezeigten Knoten aufgetreten

sind, z.B. eine Statusänderung, vgl. auch Abschnitte 3.2.1.7, 4.1.3 und 4.2) und die Integration der Aktualisierung der *Social Business Objects* in den *Activity Stream* des Kollaborationssystems. Wie bereits in Abschnitt 4.2 und 4.3 erklärt, sind die *Recent Events* Teil des Benutzerfrontends des SoNBO-Explorers, während der *Activity Stream* Teil des Kollaborationssystem ist, in das der SoNBO-Explorer integriert ist und in den die Statusänderungen des SoNBO-Explorers als Notification gepostet werden (vgl. Abschnitt 3.2.1.7). In der zweiten Masterarbeit wurde die Logik um die *Recent Events* und den *Activity Stream* erweitert. Ausgehend von der ersten Version erfolgte unter der Leitung der Autorin dieser Dissertation eine Überarbeitung des Designs und für die Benutzung des SoNBO-Explorers in einem Unternehmen wurde der SoNBO-Explorer überarbeitet, indem beispielsweise die Installation auf einem weiteren Server vereinfacht erfolgt und nicht nur die Virtualisierung, sondern auch die Materialisierung des *Social Network of Business Objects* ermöglicht wird. Die Speicherung der Ontologie erfolgt in einer NoSQL-Datenbank (vgl. Abschnitt 3.2.1.5). Eine weitere technische Umsetzung mit beispielsweise Semantic-Web-Technologien unter der Verwendung von Graphdatenbanken wäre ebenfalls denkbar, wurde aber bisher noch nicht ausprobiert. Da im Key Case HCL-Notes-Technologie verwendet wurde und diese Technologie auch in der Forschungsgruppe eingesetzt wird, wurde der SoNBO-Explorer mit dieser Technologie umgesetzt. Bei der für diese Dissertation entwickelten Anwendung wurde der Fokus auf die Umsetzung der Funktionalität gelegt und weniger auf Usability oder Design, da sie primär für Forschungszwecke entwickelt wurde und nicht zur kommerziellen Vermarktung.

Das Framework geht von einem SoNBO-Explorer mit dem folgenden Aufbau aus: Das Benutzerfrontend orientiert sich an dem Mockup, das im Rahmen der Konzeptualisierung von SoNBO basierend auf dem Key Case Vössing entwickelt wurde (vgl. Abb. 4.5 auf S. 115), und das Administrationsfrontend orientiert sich an der Beschreibung in Abschnitt 4.4. Da der SoNBO-Explorer vor der Evaluation noch keine aussagekräftigen Daten enthielt, wurden Screenshots der Anwendung aus der Evaluationsphase (vgl. Kapitel 6) erstellt. Der Fokus liegt in diesem Abschnitt auf der Funktionsweise, also der Logik des SoNBO-Explorers (vgl. Abb. 5.2). Über das Administrationsfrontend wird die Ontologie *Social Network of Concepts* mit den die Ontologie spezifizierenden *Attributen* angelegt (vgl. das Konfigurationsschema in Abb. 5.10 auf S. 154). Außerdem wird die Verknüpfung zu den Datenbanken in den Quellsystemen eingerichtet (*Mapping*). Für den Zugriff auf die Informationen in den Quellsystemen (vgl. Abb. 5.2) navigiert der Nutzer über den Client beispielsweise auf den Knoten *Prof. Schubert*. Dazu wird folgender Prozess ausgelöst (im Falle einer Virtualisierung des Netzwerks, vgl. Abb. 5.2), um das *Social Business Object* zusammensetzen: Die *XPage* (*main.xsp*) ruft den *SoNBOManager* auf, der wiederum den *NoDeService* aufruft. Dadurch wird beispielsweise das Social Business Object *Prof. Schubert* dem SSC *Professor* zugeordnet und die dazu existierenden Konfigurationen als Schema oder Teil der Ontologie werden durch den *ConfigService* aus der SoNBO-Datenbank geladen. Mit diesem Schema wird mit dem *QueryService* auf die Daten in den Quellsystemen zugegriffen und der Knoten *Prof. Schubert* kann mit der *Node*-Klasse zusammengesetzt und dem Nutzer zurückgegeben werden.

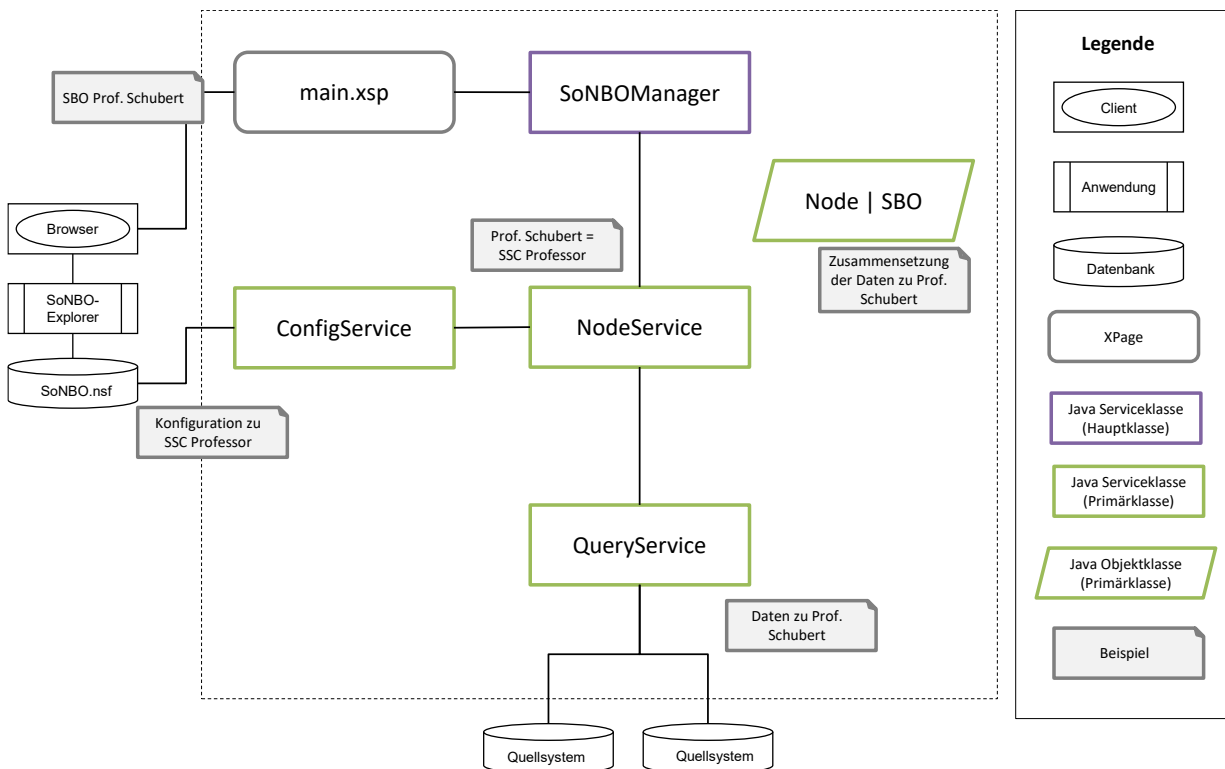


Abb. 5.2: Logik des SoNBO-Explorers (Virtualisierung) – verändert nach Götz & Gebel-Sauer (2018, p. 2002)

Bei einer Materialisierung funktioniert dieses Vorgehen analog. Der einzige Unterschied besteht darin, dass alle *Social Business Objects* mit den Kanten (also das Social Network of Business Objects) auf diese Weise im Vorfeld geladen und gespeichert werden. Der Nutzer greift dann während des Informationszugriffs auf diesen Graphen zu. Wie in Abschnitt 3.2.2.3.2 ausgeführt, liegt der Vorteil bei der *Virtualisierung* darin, dass die Daten immer aktuell sind. Allerdings wurde dort auch erwähnt, dass dadurch die Navigation durch den Knowledge Graph für den Mitarbeiter in der Regel langsamer wird. Im Rahmen der erfolgten Umsetzung von SoNBO in eine Applikation konnte in der Tat nachgewiesen werden, dass die Navigation bei der *Materialisierung* durch den beschleunigten Datenzugriff schneller und somit performanter ist, aber die Speicherung des Graphen gesondert Zeit und Speicherplatz in Anspruch nimmt. Außerdem sind die Daten dann ggf. nicht immer aktuell. Daher sollte der Graph in kurzen Zeitintervallen aktualisiert werden.

Bei der Einrichtung der Schnittstellen im SoNBO-Explorer (für das Zugreifen auf die Quellsysteme) sind folgende Herausforderungen zu beachten, die im Vorfeld (vgl. Abschnitt 4.5) bereits angesprochen wurden: Zunächst muss identifiziert werden, ob eine API vorhanden ist und wie diese genutzt werden kann. Falls keine API existiert, muss eine Schnittstelle (z. B. JSON-Strings für ein einheitliches Datenformat) eingerichtet werden. Der Nachteil besteht in solchen Veränderungen des Quellsystems, dass bei einem Software-Update diese nachträglich eingerichtete Schnittstelle manuell aktualisiert werden muss. Ein weiterer Nachteil bei einer nicht vorhandenen API besteht darin, dass die Berechtigungen

der Nutzer in der Regel nicht mitübereben werden. In diesem Fall kann also ggf. ein Benutzer mehr Daten sehen als er berechtigt ist. Dafür muss dann wiederum eine gesonderte Lösung entwickelt werden. Weiterhin ist zu überlegen, inwiefern es ein (lizenzrechtliches) Problem darstellen könnte, wenn ein Endbenutzer keine Lizenz für das Quellsystem (z. B. SAP ERP) hat, aber über den SoNBO-Explorer trotzdem auf SAP-Daten zugreift. Weiterhin sollte ein SSO (Single-Sign-on) für den Nutzer eingerichtet werden, damit dieser sich bei dem Informationszugriff nicht an jedem Quellsystem anmelden muss. Dazu gibt es verschiedene Möglichkeiten, bei denen die Verantwortlichen für die IT-Sicherheit in das Projekt einbezogen werden müssen (vgl. dazu auch die Ausführungen in Abschnitt 4.5). Diese Erkenntnisse stammen aus dem *Key Case Vössing* und den *Konfigurations-Projekten*.

5.2.3 Konfiguration des SoNBO-Explorers (= Konfigurations-Projekt)

Die Konfiguration wurde in Zusammenarbeit mit der Autorin der Dissertation durchgeführt und erfolgte für das CRM-System *GEDYS IntraWare 8*, das für die Verwaltung der forschungsgruppeninternen CRM-Informationen verwendet wird (= *Konfigurations-Projekt 1*), sowie für das ERP-System *Microsoft Dynamics NAV*, das für die Lehre eingesetzt wird (= *Konfigurations-Projekt 2*). Beide Projekte erfolgten auf der Basis des *Key Case Vössing* und wurden intuitiv und noch ohne strukturiertes Vorgehen durchgeführt. In beiden Projekten wurde eine Taxonomie bestehend aus Social Concepts/Subconcepts erstellt (vgl. Tab.-A 1 und Abb.-A 2) sowie eine Ontologie (Social Network of Concepts/Subconcepts, vgl. Abb.-A 4, Abb.-A 5, Abb.-A 6 und Abb.-A 7). Daraufhin wurde ein Mapping der Informationen in der Ontologie auf die Daten in den Quellsystemen durchgeführt, was dann anschließend im SoNBO-Explorer konfiguriert wurde. Ein Teil der Ergebnisse aus den Konfiguration-Projekten wurden bereits veröffentlicht (Gebel-Sauer & Schubert, 2019b; Götz & Gebel-Sauer, 2018). Nach beiden Projekten erfolgten eine Analyse und Ableitung der notwendigen Schritte für ein *Konfigurations-Projekt* in einem *beliebigen Unternehmen*, welches strukturiert durchgeführt werden soll. Die Erfahrung aus diesen Konfigurations-Projekten haben somit Einfluss auf das SoNBO-Framework.

5.3 Framework: Phasen und Akteure

In diesem Abschnitt werden die Phasen des Frameworks vorgestellt, die für den Prozess des Customings eines SoNBO-Explorers vorgeschlagen werden. Die Erkenntnisse stammen aus den in der Universität durchgeführten *Konfigurations-Projekten* (Gebel-Sauer & Schubert, 2019b; Götz & Gebel-Sauer, 2018), deren Grundlage wiederum der *Key Case* (vgl. Abschnitt 3.1) war. Im Anschluss an die beiden Projekte konnten notwendige Schritte abgeleitet werden, die ein weiteres Konfigurations-Projekt in einem *Unternehmen* von Beginn an strukturieren sollen. Diese sechs Schritte wurden bereits veröffentlicht (Gebel-Sauer & Schubert, 2019b) und sind in Abb. 5.3 verändert dargestellt, da dieses sogenannte *SoNBO-Prozessmodell* im Laufe des fortgeschrittenen Forschungsprozesses weiter angepasst wurde. Das Prozessmodell empfiehlt durch eine (1) *Geschäftsanalyse* und eine (2) *Systemanalyse* die Business-Sicht und technische Sicht zu erheben, um einen Überblick über die geschäftsrelevanten Informationen und die eingesetzten Informationssysteme (als Quellsysteme) zu erhalten. Darauf auf-

bauend wird eine (3a) *Zielontologie* entwickelt, woraufhin eine Startontologie und somit eine (3b) *implementierte Ontologie* abgeleitet wird. Anschließend erfolgt ein (4) *Mapping*, bei dem die Informationen in der Ontologie auf die Informationen in den Quellsystemen gemappt werden. Die Startontologie wird daraufhin erweitert, sodass sich die (3b) *implementierte Ontologie* ändert und das (4) *Mapping* ergänzt wird. Dies wiederholt sich so lange, bis die Zielontologie erreicht wurde oder bis eine Ontologie erreicht wurde, welche die Anforderungen der Nutzer abdeckt. Falls bei diesem Vorgang ein Fehler in der Zielontologie entdeckt wird, kann auch nochmals zu Schritt (3b) gegangen und diese angepasst werden. Nach dem Mapping erfolgt eine (5) *Evaluation* der konfigurierten SoNBO-Anwendung und falls diese zufriedenstellend ist, wird das (6) *Deployment* durchgeführt. Falls die Evaluation nicht zufriedenstellend ist, muss die (3a) *Zielontologie* oder (3b) *implementierte Ontologie* angepasst werden und somit zu diesem Schritt zurückgegangen werden. Nach dem Deployment kann die Ontologie im Laufe der Zeit um weitere Knoten angereichert werden, sodass die Ontologie und somit die SoNBO-Anwendung auf die geänderten Bedürfnisse eines Unternehmens flexibel reagieren kann. Bei diesem Prozessmodell handelt es sich um eine agile Methode. Die Idee der Start- und Zielontologie wurde aus der Pay-as-you-go-Methodology übernommen, die bei der Konstruktion eines OBDA-Systems unterstützt (vgl. Abschnitt 3.2.1.2.3). Dabei wird die Ontologie sukzessive unter Einbezug von Business Usern entwickelt (Sequeda & Miranker, 2017).

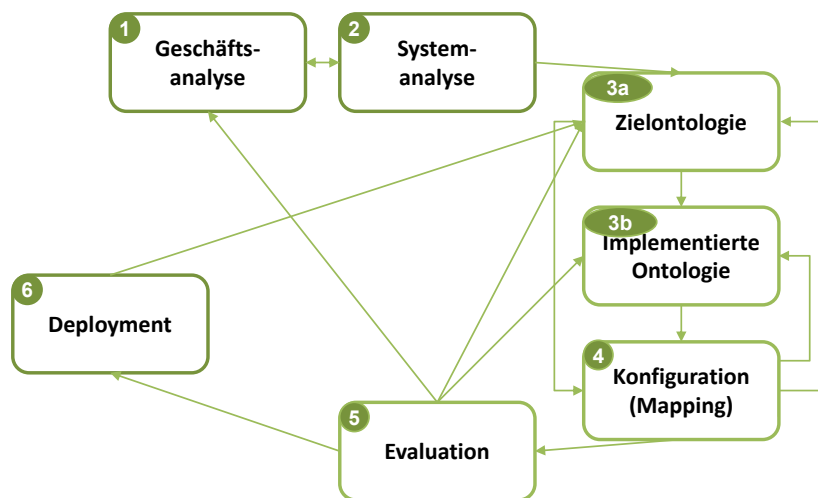


Abb. 5.3: SoNBO-Prozessmodell - verändert nach Gebel-Sauer & Schubert (2019b, p. 4)

Die Erfahrung aus den Konfigurations-Projekten zeigte, dass die größte Herausforderung für ein SoNBO-Einführungsprojekt die Entwicklung der Ziel- und Startontologie darstellt. Diese Herausforderung wurde auch in der wissenschaftlichen Literatur beschrieben (vgl. Abschnitt 3.2.1.2). Deshalb wird sowohl für den Schritt 3 der Ontologierstellung als auch für die Schritte davor (1 *Geschäfts-* und 2 *Systemanalyse*) und danach (4 *Mapping*) in einem detaillierten Framework eine Struktur durch Modelle, Methoden und Tools vorgeschlagen, die einen SoNBO-Experten bei den notwendigen Erhebungen und Entwicklungen unterstützt. Auch wenn beide Visualisierungen den Prozess des Customisings beschrei-

ben, besitzen sie jeweils Stärken und Schwächen. Während das SoNBO-Prozessmodell den gesamten Zyklus des Einführungsprojektes abdeckt und somit in die *Breite* geht, greift das SoNBO-Framework die für einen SoNBO-Experten relevanten Aspekte auf und schlägt Modelle, Methoden und Tools für die einzelnen Phasen vor. Es geht somit in die *Tiefe*. Beispielsweise fällt das Deployment nicht in den Aufgabenbereich des SoNBO-Beraters. Dies ist allerdings im gesamten Einführungsprojekt ein wichtiger Schritt, weshalb dieser im SoNBO-Prozessmodell aufgeführt wird. Außerdem kann im SoNBO-Prozessmodell die Dynamik durch die sich wiederholenden Schleifen zwischen den einzelnen Schritten aufgezeigt werden.

Das Framework besteht aus drei Phasen (vgl. Abb. 5.4) und diese Einteilung stammt ebenfalls aus den Erkenntnissen, die nach den in der Universität durchgeführten *Konfigurations-Projekten* (Gebel-Sauer & Schubert, 2019b; Götz & Gebel-Sauer, 2018) entstanden. Außerdem wurde die wissenschaftliche Literatur zur Konstruktion von *OBDA-Systemen* berücksichtigt (vgl. Abschnitt 3.2.2.3.3). Für OBDA wird empfohlen, dass zunächst eine Ontologie und anschließend ein Mapping erstellt wird. Außerdem wird bei der Ontologie die Idee der Erstellung einer Zielontologie verwendet.

Die Herleitung der einzelnen Modelle, Methoden und Tools wird in den entsprechenden Abschnitten (5.4 bis 5.6) erläutert. Als Phasen erwiesen sich für zukünftige Projekte folgende als sinnvoll: In der (1) *ersten Phase* wird der *Status quo* im Unternehmen durch ein teilstandardisiertes Leitfadent-Interview erhoben. Dadurch werden die relevanten und geschäftskritischen Informationen ermittelt, die für das tägliche Arbeiten relevant sind. Im SoNBO-Prozessmodell lassen sie sich der Geschäfts- und Systemanalyse zuordnen. In der (2) *darauffolgenden Phase* werden auf dieser Basis und mit weiteren Datenerhebungsmethoden die *Social Concepts* und das *Social Network of (Sub-)Concepts* (= Unternehmensontologie) entwickelt, wofür ein Modell die Ausgangsbasis ist, welches für das SoNBO-Framework entwickelt wurde. Dies erfolgt zum einen durch Workshops und Fragebögen mit den Mitarbeitern (= Business-Sicht) und zum anderen durch Analysen der Business Objects in den anzubindenden Datenbanken der IT-Landschaft (= technische Sicht). Das in dieser Phase unterstützende Tool ist der *SoNBO-Visualiser* (Meier, 2019; Meier, Gebel-Sauer, & Schubert, 2020), der die Ontologie visualisiert. Im SoNBO-Prozessmodell lässt sich diese Phase den Schritten *Zielontologie* und *implementierte Ontologie* zuordnen. In der (3) *dritten und letzten Phase* werden die Informationen in der Ontologie auf die Daten in den Quellsystemen *gemappt* und im SoNBO-Explorer *konfiguriert*. Dabei unterstützt das Tool *SoNBO-Mapper*, welches ein selbst erstelltes Excel-Dokument darstellt, in dem das Mapping strukturiert wird. Im SoNBO-Prozessmodell entspricht dies dem Schritt *Konfiguration (Mapping)*.

Customising des SoNBO-Explorers				
Phase	1 Status quo	2 Ontologieentwicklung		3 Mapping
Ziel	Verständnis über geschäftsrelevante Informationen im Unternehmen	Identifizierung von SBO & SC im Unternehmen	Identifizierung von SoNBO & SoNC im Unternehmen	Konfiguration von SoNC durch Mapping im SoNBO-Explorer
Beteiligte Akteure	SoNBO-Experte & Domain-Experte	SoNBO-Experte & Domain-Experte		SoNBO-Experte & Domain-Experte & IT-Experte
Modell		Taxonomie (= SC)	Ontologie (= SoNC)	
Methode	Teilstandardisiertes Leitfaden-Interview	Business Sicht: World-Café & Fragebogen		Semant. Mapping
		Technische Sicht: Systemlandschaft-Analyse		Techn. Mapping
Tool		SoNBO-Visualiser		SoNBO-Mapper

Legende:
 SBO = Social Business Object; SC = Social Concept; SoNBO = Social Network of Business Objects; SoNC = Social Network of Concepts

Abb. 5.4: SoNBO-Framework

Für die einzelnen Phasen im Framework sind verschiedene Akteure von Bedeutung, die in Tab. 5.1 dargestellt sind. Diese werden in den Kapiteln zu den drei Phasen (vgl. Abschnitte 5.4, 5.5 und 5.6) wieder thematisiert und die Funktion der Akteure erklärt. Die Erkenntnisse für die Differenzierung in die unterschiedlichen Akteure in einem SoNBO-Einführungsprojekt stammen sowohl aus dem *Key Case Vössing* als auch aus der Erfahrung aus den *Konfigurations-Projekten*. Sie wurden im Anschluss mit der Überlegung abgeleitet, ein SoNBO-Projekt in einem beliebigen Unternehmen durchzuführen. Die Notwendigkeit der Unterscheidung in die verschiedenen Akteure ergibt sich durch folgenden Unterschied zwischen den vergangenen und zukünftigen SoNBO-Projekten: Im *Key Case Vössing* wurde nicht explizit zwischen den verschiedenen Akteuren unterschieden, da das IT-Team implizit das Wissen besaß bzw. sich aneignete, um eine SoNBO-Applikation zu entwickeln und zu konfigurieren. Dies war bei den Konfigurations-Projekten in der Universität ebenfalls der Fall. In zukünftigen, strukturiert durchgeführten SoNBO-Einführungsprojekten verteilt sich dieses Wissen auf verschiedene Akteure (z. B. SoNBO, Domain, IT), da der SoNBO-Experte in der Regel ein (aus Unternehmenssicht) extern eingesetzter Berater ist, um einen SoNBO-Explorer für das Unternehmen zu konfigurieren. Somit überschneidet sich das Wissen, das für das Customising notwendig ist, nicht, sondern verteilt sich auf ver-

schiedene Personen. Ein Idealzustand wäre zwar für zukünftige Projekte, wenn der SoNBO-Experte das Wissen von Domain-Experten und IT-Experten besitzen würde. Da dies allerdings in der Realität selten der Fall sein wird, ist ein korrekter Informationsaustausch zwischen den drei Akteuren für eine erfolgreich konfigurierte Software von großer Bedeutung, weshalb hier vorgeschlagen wird, zwischen diesen Akteuren zu unterscheiden. Eine ähnliche Einteilung nimmt in der wissenschaftlichen Literatur die *Pay-as-you-go-Methodology* (vgl. Abb. 3.31 auf S. 72) für die Erstellung eines OBDA-Systems vor (Sequeda & Miranker, 2017): Business User, IT-Entwickler und Wissensingenieur. Bei SoNBO entspricht der Business User dem Domain-Experten, der IT-Entwickler dem IT-Experten und der Wissensingenieur dem SoNBO-Experten.

Tab. 5.1: Charakterisierung der einzelnen Akteure

Akteur	Charakterisierung des Akteurs	Anwendung, mit welcher der Akteur interagiert
Endbenutzer	Person, die mit dem Benutzerfrontend des SoNBO-Explorers auf die Informationen in den Quellsystemen Ontologie-basiert zugreift.	SoNBO-Explorer (Benutzerfrontend)
SoNBO-Experte	Person, die Experte im Themengebiet der Ontologie-basierten Integrationsansätze (insbesondere SoNBO) ist.	SoNBO-Explorer (Administrationsfrontend), SoNBO-Visualiser, SoNBO-Mapper
Domain-Experte	Person im Unternehmen, die einen Überblick über geschäftsrelevante Informationen und/oder die eingesetzten betrieblichen Anwendungssysteme hat. Diese Person ist mit hoher Wahrscheinlichkeit gleichzeitig ein künftiger Endbenutzer des SoNBO-Explorers für den Informationszugriff (Benutzerfrontend).	SoNBO-Explorer (Benutzerfrontend), SoNBO-Visualiser
IT-Experte	Person (typischerweise aus der IT-Abteilung), die im Unternehmen für den SoNBO-Explorer zuständig sein wird.	SoNBO-Explorer (Administrationsfrontend), SoNBO-Visualiser

Neben diesen Phasen wurden noch weitere Erkenntnisse aus den beiden *Konfigurations-Projekten* abgeleitet, die in der wissenschaftlichen Literatur bereits im Bereich der Methoden zur Einführung von Unternehmenssoftware beschrieben sind und den Aufbau des Prozessmodells sowie des Frameworks charakterisieren: Bei *SSM* (Soft Systems Methodology) wird als erster Schritt eine gründliche Analyse der Problemsituation/Ist-Zustand empfohlen, worauf aufbauend dann Handlungsempfehlungen ausgesprochen werden (vgl. Abschnitt 3.2.1.6.1). In den Konfigurations-Projekten erwies sich beispielsweise eine gründliche Analyse des CRM-Systems mit der Dokumentation des Systemaufbaus und der Art der Nutzung als Produktivsystem durch die Forschungsgruppe der Universität als sehr hilfreich. Dies erfolgte im Konfigurations-Projekt zunächst aus Zeitgründen nicht, da die Entwicklung des SoNBO-Explorers im Vordergrund stand. Allerdings trat die Situation ein, dass durch das mangelnde Wissen über den Ist-Zustand des Quellsystems kein weiterer Fortschritt bei der Konfiguration erzielt werden konnte, sodass diese Analyse dann zu einem späteren Zeitpunkt erfolgte. Darauf aufbauend kann

te dann die Ontologie entwickelt werden, was bei SSM die aus dem Status quo abgeleitete Handlungsempfehlung darstellt. Außerdem wird von ARIS (vgl. Abschnitt 3.2.1.6.2) der Gedanke übernommen, dass der Berater bei der Einführung von Standardsoftware als Bindeglied zwischen Betriebswirtschaft und Informationstechnik fungiert. Durch ARIS werden bei der Einführung von Standardsoftware keine kundenspezifischen Individuallösungen angestrebt, sondern Methoden für ein strukturiertes Vorgehen bei der Einführung bereitgestellt. Dieses Ziel verfolgt auch das Framework, indem es Werkzeuge für das Customising der Standardsoftware *SoNBO-Explorer* zur Verfügung stellt, die sowohl die Betriebswirtschaft als auch die Informationstechnik miteinbindet. In den Konfigurations-Projekten erwies es sich beispielsweise als sinnvoll, die Ontologie aus der Sicht der Nutzer zu modellieren (top-down) und die darin enthaltenden Informationen auf die Informationen im Quellsystem (ERP-System) zu mappen. Dadurch war der Aufwand zum Teil hoch, da die Anforderungen der Nutzer nicht mit den technischen Gegebenheiten übereinstimmten. Aber bei SoNBO soll die Nutzersicht im Vordergrund stehen, sodass diese beiden Bereiche kombiniert werden sollen. Für die Entwicklung der Ontologie wird auf der Basis der Erkenntnisse aus den Projekten daher auch ein Modell als Ausgangsbasis und eine Methode zur Entwicklung im Framework zur Verfügung gestellt. Solche Modelle und Methoden gibt es analog auch bei ARIS, die dann auf ERP-Systeme ausgerichtet sind. Im Gegensatz zu einem ERP-System handelt es sich bei SoNBO allerdings nicht um ein prozessorientiertes System, weshalb Aspekte der Einführungsmethoden für Kollaborationssoftware durch *IRESS* (vgl. Abschnitt 3.2.1.6.3) zum Tragen kommen. Auch bei SoNBO müssen keine Prozesse modelliert werden, sondern es geht eher um das Erheben von Nutzungsszenarien bezüglich der Informationen, die aus den Quellsystemen benötigt werden. Auf dieser Basis wird dann durch den SoNBO-Experten die Unternehmensontologie entwickelt.

In den folgenden Abschnitten werden die entwickelten Modelle, die Methoden und die Tools sowie die jeweilige Herleitung im Detail vorgestellt.

5.4 Phase 1: Status quo

Das Ziel in dieser Phase ist die Erhebung des Status quo im Unternehmen bezüglich der geschäftsrelevanten Informationen und der eingesetzten betrieblichen Anwendungssysteme. Sie bildet somit die Grundlage für den nächsten Schritt (Ontologieentwicklung, vgl. Abschnitt 5.5). Durch die Erhebung des Status quo erhält der SoNBO-Experte einen Überblick über die Informationen, die Systeme und ggf. Besonderheiten in dem betreffenden Unternehmen. Die Notwendigkeit dieser Phase zeigte die Erfahrung aus den *Konfigurations-Projekten*.

5.4.1 Beteiligte Akteure

In dieser Phase sind vor allem zwei Akteure von Bedeutung: Der *SoNBO-Experte* erarbeitet mit ausgewählten *Domain-Experten* die benötigten Informationen. Die Identifikation der Domain-Experten im Unternehmen durch den SoNBO-Experten stellt eine Herausforderung dar, was die Diskussion des Key Case Vössing mit Praxisvertretern an IndustryConnect06 im Jahr 2017 (vgl. A.2) zeigte (vgl. Abschnitt 2.2.2). Daher wird vorgeschlagen, dass der SoNBO-Experte die ersten Interviews mit Mitarbeitern in

Schlüsselposition (z. B. Abteilungsleiter IT) führt, und auf diese Weise im Laufe dieser Gespräche ggf. weitere Experten identifizieren kann.

5.4.2 Methode: Teilstandardisiertes Leitfaden-Interview

Für das Erreichen des Ziels in dieser Phase (Erhebung Status quo) wird die Methode *teilstandardisiertes Leitfaden-Interview* vorgeschlagen. Sowohl im *Key Case Vössing* als auch in den *Konfigurations-Projekten* war (wie bereits angesprochen) diese Phase nicht explizit vorhanden, da dieses Wissen bei den Beteiligten vorhanden war und nicht explizit erhoben werden musste. Mit anderen Worten: Der SoNBO-Experte besaß auch das Wissen des Domain-Experten. In einem SoNBO-Einführungsprojekt in einem beliebigen Unternehmen besitzt der SoNBO-Experte allerdings nicht das Domain-Wissen, weshalb es notwendig ist, dass der SoNBO-Experte durch eine Befragung des Domain-Experten an dieses Wissen gelangt. Für die Erhebung von Informationen von Menschen existieren geschlossene, halbgeschlossene/halboffene oder offene Methoden. Für die erste Phase im Framework wurde sich für das teilstandardisierte Leitfaden-Interview entschieden, welches eine (halb)offene Methode darstellt, da der SoNBO-Experte auf diese Weise auch an Wissen gelangen kann, welches nicht antizipiert werden kann. Ein Leitfaden bestehend aus Fragen gibt Impulse für das Ziel (Erhebung Status quo). Es ist aber nicht ausgeschlossen, dass weitere Methoden existieren, die an dieser Stelle sinnvoll eingesetzt werden können. Falls das Unternehmen bereits eine Prozesslandkarte (oder Teile davon) oder weitere Dokumentationen über Unternehmensinformationen erstellte, sollten diese unbedingt miteinbezogen werden.

Es wird vorgeschlagen, dass sich das Interview für die *Geschäftsanalyse* an folgenden Fragen orientiert:

- Welche sind die geschäftsrelevanten Prozesse/Informationen?
- Gibt es eine auftragsbezogene Fertigung oder erfolgt eine Lagerfertigung?
- Welche betrieblichen Anwendungssysteme werden für die Speicherung und das Abrufen von relevanten Informationen verwendet?

Bei der ersten Frage ist das Ziel keine Modellierung der Geschäftsprozesse, sondern vielmehr das Erheben der geschäftsrelevanten Informationen, welche auch aus den Prozessen abgeleitet werden können. Da durch SoNBO der ad hoc entstandene Informationsbedarf befriedigt werden soll, ist SoNBO nicht prozessorientiert. Während des Interviews sollen die Mitarbeiter möglichst frei reden, sodass auch die Bereiche angesprochen werden, die zuvor nicht antizipiert wurden. Während des Gesprächs oder zum Abschluss sollten die Systeme, insbesondere das Frontend, miteinbezogen werden, mit denen die Mitarbeiter täglich arbeiten. Im Anschluss oder während dieser Fragen kann gemeinsam das Frontend der häufig verwendeten betrieblichen Anwendungssysteme eingesehen werden.

Des Weiteren werden die betrieblichen Anwendungssysteme ermittelt (*Systemanalyse*), in denen sich voraussichtlich die zu integrierenden Informationen befinden und somit für die SoNBO-Anwendung als

Quellsysteme dienen können. Dazu wird ebenfalls ein Interview geführt. Es wird vorgeschlagen, sich an folgenden Fragen zu orientieren:

- In welchen betrieblichen Anwendungssystemen befinden sich die zu integrierenden Informationen?
- In welchen Datenbanktabellen/Views befinden sich die Informationen?
- Welche Datenbankmodelle werden verwendet?
- Welche Schnittstellen (API) für den Zugriff auf die Informationen sind in den betrieblichen Anwendungssystemen vorhanden?

Falls die IT-Abteilung eine Auflistung der Systeme bereits z.B. im Rahmen einer entwickelten IT-Strategie besitzt, sollte darauf zurückgegriffen werden. Während des Gesprächs oder zum Abschluss sollte auch das Backend der Systeme miteinbezogen werden, um einen Eindruck über die Systeme zu erhalten. Beispielsweise zeigte der *Key Case Vössing*, dass in den Informationssystemen Klassifikationen verwendet werden, die dann auch für die Identifizierung der Social Concepts relevant sein können.

5.5 Phase 2: Ontologieentwicklung

Das Ziel in dieser Phase ist die Entwicklung der *Social Concepts* und das darauf basierende *Social Network of Concepts* (entspricht der *Ontologie*). Dafür fließen Aspekte aus dem *Key Case Vössing*, Teile von existierenden *Unternehmensontologien* (Abschnitt 3.2.1.2.4) und *Konstruktionen von Ontologien* (Abschnitt 3.2.1.2) sowie die Erfahrungen aus den *Konfigurations-Projekten* mit ein, was in den Abschnitten konkret benannt wird. Nachdem sich der SoNBO-Experte in der Phase zuvor bereits einen Überblick über die geschäftskritischen Informationen des Unternehmens verschaffte (Status quo), ist es in dieser Phase von Bedeutung, das Wissen zur Entwicklung des *Social Network of Concepts* zu erhalten. Dazu verwendet der SoNBO-Experte zwei Quellen: Mit ausgewählten *Domain-Experten* wird die Ontologie, also im Fall von SoNBO das *Social Network of Concepts*, ausgehend von einem Referenzmodell entwickelt. Außerdem wird die *IT-Landschaft* analysiert und die für SoNBO relevanten Quellsysteme werden identifiziert. Anschließend werden die Anforderungen der Business-Sicht und die technischen Gegebenheiten zusammengeführt, um auf diese Weise ein *Social Network of Concepts* zu erstellen, das den Anforderungen der Mitarbeiter gerecht wird und sich an den technischen Gegebenheiten orientiert.

5.5.1 Beteiligte Akteure

In dieser Phase wird vorgeschlagen, dass folgende Akteure für ein SoNBO-Einführungsprojekt von Bedeutung sind: Der *SoNBO-Experte* verfügt über das notwendige Wissen zur Ontologie-Erstellung und muss die Mitarbeiter identifizieren, die im Unternehmen an Schlüssel-Positionen sitzen und einen Überblick über die Unternehmensinformationen besitzen. Diese Mitarbeiter (*Domain-Experten*) sollten außerdem die Fähigkeit besitzen, sich in neue Konzepte einzudenken, sodass sie ein Verständnis

über *Social Concepts* entwickeln und somit bei der Ontologieentwicklung aktiv unterstützen können. Ein zentraler Untersuchungsgegenstand ist dafür die *Systemlandschaft*, welche der SoNBO-Experte parallel zu der Entwicklung der Ontologie analysiert, um auf diese Weise die Anforderungen des Unternehmens und die technischen Gegebenheiten zu vereinen. Diese Erkenntnisse für die Notwendigkeit dieser Akteure in dieser Phase basieren auf der Analyse des *Key Case* und die Erfahrung aus den *Konfigurations-Projekten*.

5.5.2 Modell: Unternehmensontologie

In diesem Abschnitt wird ein *Referenzmodell* als *Ontologie* (entspricht *Social Network of Concepts/Subconcepts*) vorgeschlagen, welches unternehmensübergreifend formuliert ist und somit für ein konkretes Unternehmen als Ausgangsbasis bei der Ontologieentwicklung dienen soll. Bei der Konstruktion von Ontologien im allgemeinen wird in der wissenschaftlichen Literatur vorgeschlagen, existierende zu verwenden und anzupassen, um den Aufwand bei der Ontologie-Erstellung gering zu halten (vgl. Abschnitt 3.2.1.2.3) (Uschold & Gruninger, 1996). Eine weitere Möglichkeit auf eine bestehende Ontologie als Vorlage zurückzugreifen sind *putative ontologies*. Dazu wird aus dem Quelldatenbankschema mithilfe eines Bootstraps eine Ontologie abgeleitet (vgl. Abschnitt 3.2.1.2.3) (Sequeda, 2017).

Das hier vorgeschlagene Referenzmodell besteht zum einen aus einer *Taxonomie* (vgl. Abschnitt 3.2.1.2.2), welche im SoNBO-Kontext den *Social Concepts/Social Subconcepts* entspricht (vgl. Tab. 5.3). Zum anderen besteht es aus einer Verknüpfung dieser Objekte zu einer Ontologie (oder auch: *Social Network of Concepts* in Abb. 5.5 bzw. *Social Network of Subconcepts* in Abb. 5.6).

Tab. 5.2: Vergleich der hier verwendeten Begriffe (eigene Darstellung)

Semantic Web	SoNBO	Wirtschaftsinformatik
Taxonomie	Social Concept/Social Subconcept	Referenzmodell
Ontologie	Social Network of Social (Sub)Concepts	

Die Differenzierung zwischen den Begriffen *Taxonomie*, *Ontologie* und *Modell* wurde in Abschnitt 3.2.1.2 mithilfe der wissenschaftlichen Literatur erläutert und wird in Tab. 5.2 mit den SoNBO-Begriffen verglichen: Aus der SoNBO-Perspektive sind Social (Sub)Concepts (z. B. Person – Mitarbeiter; Kundenbeleg – ausgehende Rechnung) für den Mitarbeiter sinnvoll kategorisierte Business Objects aus den Quellsystemen (vgl. Kapitel 4). Durch die Benennung der Social (Sub)Concepts (z. B. Person – Mitarbeiter; Kundenbeleg – ausgehende Rechnung) ergibt sich die Taxonomie für die zu entwickelnde Ontologie. Die Ontologie wird bei SoNBO (wie in Kapitel 4 erklärt) *Social Network of (Sub)Concepts* genannt. Aus der Perspektive der Wirtschaftsinformatik handelt es sich bei der in diesem Framework vorgeschlagenen Ontologie *Social Network of (Sub)Concepts* um ein *Referenzmodell*, da diese als Ausgangsbasis zur Anpassung auf ein beliebiges Unternehmen dienen soll. *Referenzmodell* wird der Einfachheit halber in dieser Dissertation auch nur als Modell bezeichnet wird (vgl. Abschnitt 5.1).

Im Folgenden wird zunächst eine generische Taxonomie und die daraus resultierende Ontologie vorgestellt, die für die Identifizierung der unternehmensspezifischen Ontologie relevant ist. Die Ergebnisse stammen aus dem *Key Case Vössing* (vgl. Abschnitt 3.1), aus der *Literaturanalyse* (vgl. Abschnitt 3.2, insbesondere Beispiele zu Unternehmensontologien in Abschnitt 3.2.1.2.4) und aus den beiden Konfigurationen (*Konfigurations-Projekte*) eines SoNBO-Explorers (Gebel-Sauer & Schubert, 2019b; Götz & Gebel-Sauer, 2018). Dies wird in den folgenden Abschnitten jeweils konkret hergeleitet.

5.5.2.1 Taxonomie (= Social Concepts und Social Subconcepts)

Bereits im *Key Case* wurden im Interview sowie in einem vom IT Leiter zur Verfügung gestellten Foliensatz relevante Social Subconcepts¹⁵ genannt, die im Anhang aufgeführt sind (vgl. Abb.-A 1 auf S. 213). Beispiele dafür sind *BANF* (= Bedarfsanforderung), *Lieferantenrechnung*, *Mitarbeiter* oder *Rechnung*. Eine weitere Basis für das Ergebnis in diesem Abschnitt ist die wissenschaftliche Literatur zu Unternehmensontologien: In der *IDEON Ontologie* existieren u. a. die Unternehmensorganisationssicht (vgl. Abb. 3.10 auf S. 42) und die Ressourcen-/Produktsicht (vgl. Abb. 3.11 auf S. 43), aus der z. B. das Social Concept *Person* und *Dokument* übernommen wurde. Außerdem wurden in den beiden *Konfigurations-Projekten* ein CRM- und ein ERP-System analysiert und daraus eine Taxonomie abgeleitet (vgl. im Anhang Tab.-A 1 auf S. 214 und Abb.-A 2 auf S. 215), die ebenfalls Einfluss auf die hier entwickelte Taxonomie hat.

Diese für das Framework vorgeschlagene Taxonomie befindet sich in Tab. 5.3. In dieser Taxonomie wird zwischen den zwei Ebenen *Social Concepts* (erste Spalte) und *Social Subconcepts* (zweite Spalte) unterschieden. Der Unterschied ist in Abschnitt 4.1.2 erklärt. In der dritten Spalte befinden sich synonyme Bezeichnungen für die vorangegangenen Social Subconcepts, da sich diese von Unternehmen zu Unternehmen unterscheiden. Dies liegt daran, dass sich diese Bezeichnungen in der Regel an den verwendeten betrieblichen Anwendungssystemen und somit an den vorgegebenen Benennungen der Hersteller orientieren, welche die Mitarbeiter im Sprachgebrauch adaptieren. Diese Spalte ist nicht vollständig, da dies nicht der Kern dieser Dissertation ist. Es wird aber vorgeschlagen, dass dies durch den SoNBO-Berater bei dem Domain-Experten im Rahmen eines Einführungsprojektes vollständig erhoben wird, sodass zwischen dem SoNBO-Experten und den Mitarbeitern dieselbe Sprache verwendet wird. Auf dieser Basis kann dann entschieden werden, welche Begriffe im SoNBO-Explorer benutzt werden, was dann wiederum Einfluss auf die Taxonomie haben kann. Es wird allerdings empfohlen, sich an die hier vorgeschlagene Taxonomie zu halten, sodass sich auch unternehmensübergreifend eine einheitliche Begriffssystematik durchsetzt. In der letzten Spalte ist die Datenquelle teilweise angegeben.

¹⁵ Im *Key Case* wurden diese nicht „Social Concepts“, sondern „Knoten“ oder „Business Objects“ genannt. Allerdings wird an dieser Stelle der im Rahmen dieser Dissertation konzeptualisierte Begriff verwendet.

Tab. 5.3: Taxonomie bestehend aus Social Concepts und Subconcepts (eigene Darstellung)

Social Concept	Social Subconcept	Synonyme Bezeichnung	Datenquelle
Kunden-Beleg	Angebot	Verkaufsangebot	ERP-System
	Auftrag	Verkaufsauftrag	ERP-System
	Lieferschein		ERP-System
	Rechnung	Verkaufsrechnung, Ausgangsrechnung	ERP-System
	Mahnung		ERP-System
Lieferanten-Beleg	Bestellung		ERP-System
	Lieferschein		ERP-System
	Rechnung	Einkaufsrechnung, Eingangsbuchung	ERP-System
	Mahnung		ERP-System
Interner Beleg	Bedarfsanforderung		ERP-System
	Vertrag		ERP-System
Person	Mitarbeiter		HR-System
	Kundenkontakt		CRM-System
	Lieferantenkontakt		SRM-System
Institution	Kundenunternehmen		
	Lieferantenunternehmen		
	Öffentliche Einrichtung		
	Verein		
	Universität		
Niederlassung	Niederlassung		
Gebäude	Gebäude		
Abteilung	Abteilung		
Lager	Lagerort		ERP-System
	Lagerplatz		ERP-System
Projekt	Projekt		Projektmanagement-System
Produkt	Rohmaterial		ERP-System
	Artikel		ERP-System
	Position		ERP-System
	Dienstleistung		ERP-System
	Stückliste		ERP-System
Sachkonto	Bestandskonto		ERP-System
	Erfolgskonto		ERP-System
Inventar	Laptop		
	Handy		
Fuhrpark	PKW		
	LKW		
Arbeitszeitorganisation	Stundenkonto		HR-System
	Urlaubskonto		HR-System
	Urlaubsantrag		HR-System

In dieser Tabelle liegt der Fokus auf Social Concepts aus betrieblichen Anwendungssystemen. Dies kann auf weitere Quellsystemtypen wie z. B. Produktivsystem (Maschinen als SC durch IoT) oder ECS (Social Documents als SC wie Blog, Forum, Wiki, ...) erweitert werden.

Die genannten Social Concepts sollen die Informationen in einem Unternehmen als Referenzmodell möglichst vollständig beschreiben. Diese Übersicht dient als Grundlage bei der Identifikation der Social Concepts des betreffenden Unternehmens und wird auf die Bedürfnisse des jeweiligen Unternehmens angepasst. Dazu kann zum einen die Hierarchisierung der Social Concepts zu den Social Subconcepts angepasst werden. Beispielsweise kann die Abteilung dem Gebäude untergeordnet werden. Zum anderen kann eine *Ziel-* und eine *Starttaxonomie* definiert werden: Die Zieltaxonomie umfasst dann alle Social Concepts, die nach der Erhebung identifiziert wurden. Anschließend wird der Fokus bei der weiteren Konfiguration auf eine sinnvolle Auswahl (z. B. Belege und Mitarbeiter) gelegt, was dann die Starttaxonomie ist. Die Starttaxonomie wird daraufhin in mehreren Zyklen um weitere Social (Sub-)Concepts erweitert, die dann ebenfalls konfiguriert werden. Das Unternehmen muss bei der Auswahl der Starttaxonomie miteinbezogen werden. Falls im Laufe der Zeit die Zieltaxonomie um weitere Social Concepts ergänzt wird, können diese auch nachträglich konfiguriert werden. Auf diese Weise kann mit dem SoNBO-Konzept auf Änderungen der Anforderungen im Unternehmen flexibel reagiert werden.

Diese Hierarchie ist relevant für die Konfiguration des SoNBO-Explorers. Sie wird dem Benutzer im Frontend visualisiert, da ihm diese Kategorisierung zur Verfügung gestellt wird, um seine Social Business Objects zu filtern. Somit entspricht dieser Schritt auch dem SoNBO-Konzept und damit dem Verständnis des Social Business Objects und Social Concepts (vgl. Abschnitt 4.1).

5.5.2.2 Ontologie (= Social Network of Concepts und Subconcepts)

Das sinnvolle Verknüpfen dieser *Social Concepts* bzw. *Social Subconcepts* untereinander ergibt die Ontologie bzw. das Social Network of (Sub-)Concepts (vgl. Abb. 5.5 und Abb. 5.6), welche als Referenzmodell für ein SoNBO-Einführungsprojekt dienen soll. *Im Key Case Vössing* wurden bereits beispielhaft solche Verknüpfungen durch den IT-Leiter zur Verfügung gestellt (vgl. Abb.-A 3 auf S. 216). Im SoNBO-Konzept wird allerdings zwischen zwei Ebenen unterschieden (*Social Network of Concepts* und *Social Network of Subconcepts*). Die Trennung der Ontologie in zwei Ebenen wurde aus der Literatur übernommen (vgl. *Business Ontology* in Abb. 3.13 auf S. 45). Sie wurde in den beiden Konfigurationsprojekten angewandt und bewährte sich, weshalb sie auch für das SoNBO-Framework vorgeschlagen wird. Außerdem wurden Verknüpfungen aus der *IDEON Ontologie* (vgl. insbesondere *Unternehmensorganisationssicht* in Abb. 3.10 auf S. 42 und *Ressourcen-/Produktsicht* Abb. 3.11 auf S. 43) für die Entwicklung der als Referenzmodell dienenden Ontologie in diesem Abschnitt übernommen. Schließlich hatten die Ontologien aus den beiden *Konfigurations-Projekten* Einfluss auf den gewählten Aufbau des Referenzmodells (vgl. CRM-Informationen: Abb.-A 4 und Abb.-A 5; ERP-Informationen: Abb.-A 6 und Abb.-A 7). Dies wurde mit dem Tool SoNBO-Visualiser (Meier, 2019; Meier et al., 2020) dargestellt, welches in Abschnitt 5.5.4 vorgestellt wurde.

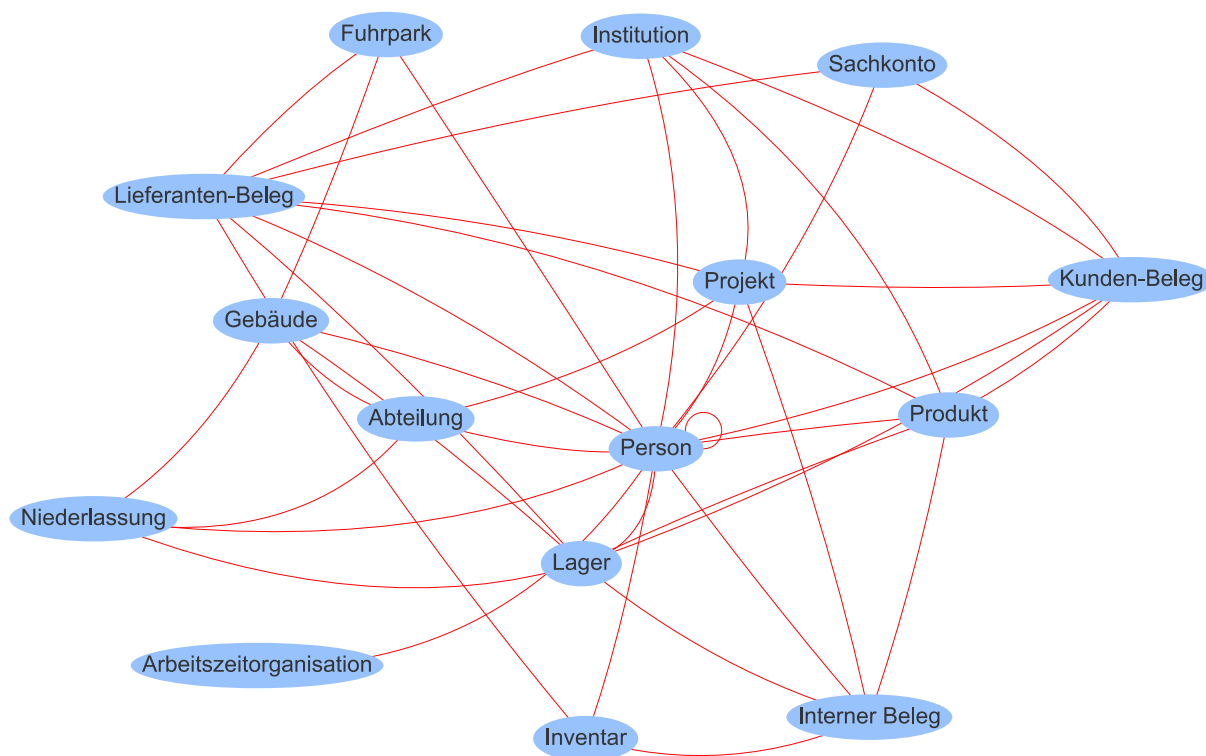


Abb. 5.5: Referenzmodell: Social Network of Concepts

Das Social Network of Concepts entsteht, indem die Social Concepts in Beziehung zueinander gesetzt werden (vgl. Abb. 5.5). Dazu wird paarweise geprüft, ob diese miteinander in Beziehung stehen. Für diese Prüfung wird das Wissen des Domain-Experten und die Analyse der Datenbanken herangezogen. Beispielsweise steht eine *Person* in Beziehung zu einem *Projekt*, da Personen in diesem Projekt arbeiten. Für diese Beziehung gibt es beispielsweise in der Projekteingabemaske das Feld *Verantwortliche*, durch welches die Kante erstellt werden kann. Diese Ebene ist für einen Menschen noch lesbar und kann von einer Maschine verarbeitet werden.

Das *Social Network of Subconcepts* entsteht ebenfalls, indem die *Social Subconcepts* miteinander verknüpft werden, die zueinander in Beziehung stehen (vgl. Abb. 5.6). Dabei orientiert man sich an den Beziehungen zu den zugehörigen Social Concepts. Allerdings bedeutet dies nicht, dass auch auf dieser Ebene der Social Subconcepts zwingend eine Beziehung besteht. Beispielsweise existiert eine Relation zwischen den Social Concepts *Person* und *Projekt*. Betrachtet man die Social Subconcepts *Mitarbeiter*, *Kundenkontakt* und *Lieferantenkontakt* sind allerdings in einigen Projekten lediglich der *Mitarbeiter* und der *Kundenkontakt* beteiligt. Das Social Network of Subconcepts ist durch die Zunahme der Anzahl der Knoten und somit der Kanten für den Menschen schwieriger lesbar. Dies kann allerdings von einer Maschine verarbeitet werden, weshalb diese Art der Visualisierung ebenfalls sinnvoll ist. Für die weitere Spezifizierung der Ontologie werden für jedes Social Subconcept die Attribute in den Quellsystemen identifiziert (Mapping), die für SoNBO relevant sind. Wie diese Attribute identifiziert werden und

somit wie das semantische und technische Mapping durchgeführt wird, wird in der nächsten Phase in Abschnitt 5.6.2 erklärt.

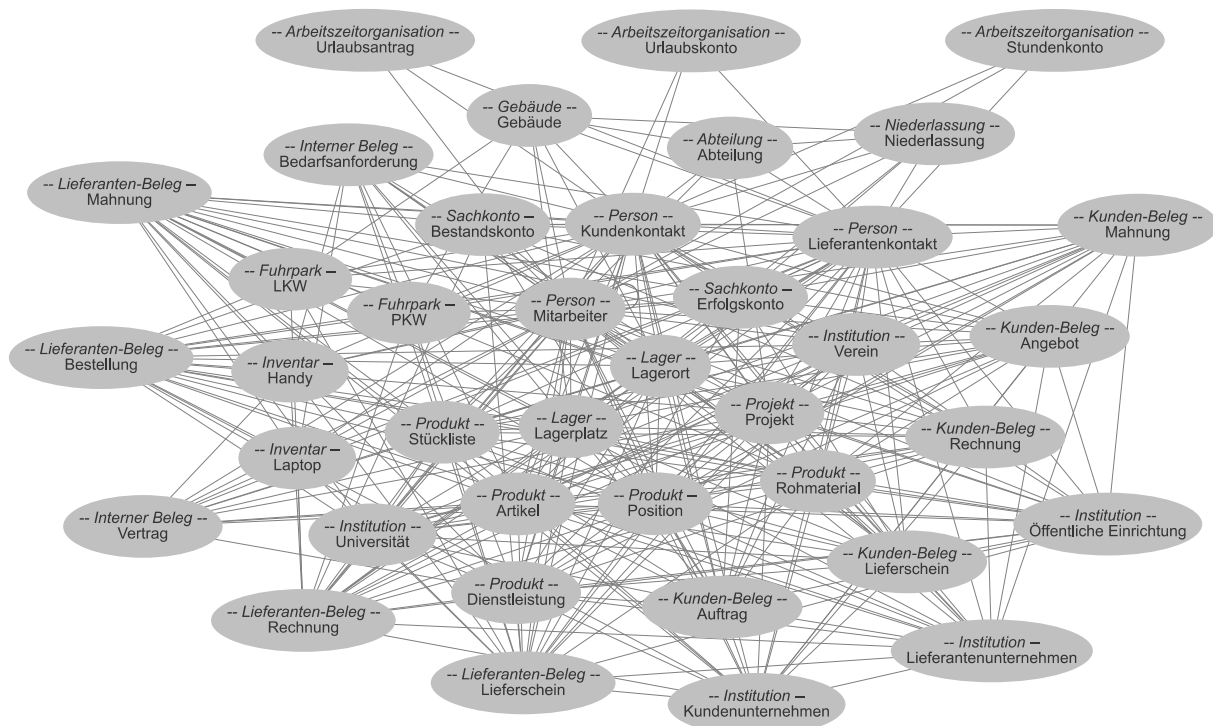


Abb. 5.6: Referenzmodell: Social Network of Subconcepts

Beide Ebenen (Social Network of Concepts und Subconcepts) sind also von Bedeutung bei der Einführung eines SoNBO-Explorers in einem Unternehmen. Da das Netzwerk mit den Mitarbeitern gemeinsam identifiziert werden soll, ist die Ebene der Social Concepts relevant, da diese vom Menschen lesbar ist. Allerdings ist für die Konfiguration im SoNBO-Explorer eine feingranularere Ebene wichtig, weshalb auch das Social Network of Subconcepts erstellt werden muss, welches dann vor allem von einer Maschine verarbeitet wird.

5.5.3 Methode: Analyse Business-Sicht und technische Sicht

In diesem Abschnitt werden Methoden vorgeschlagen, die den SoNBO-Experten bei der Erhebung des Wissens bei den Mitarbeitern (*Domain-Experten*) zur Erstellung der Ontologie *Social Network of Concepts/Subconcepts* unterstützt. Die Herausforderung im Unternehmen besteht für den SoNBO-Experten darin, ausgehend von diesen *Referenzmodellen* mit den ausgewählten Mitarbeitern die für das Unternehmen passende Ontologie zu entwickeln. Um die *Social (Sub-)Concepts* und schließlich das *Social Network of (Sub-)Concepts* zu erstellen, wurden zwei Methoden erarbeitet: Zum einen werden die Social (Sub)Concepts und deren Verknüpfungen von den Mitarbeitern durch *Interviews*, *Fragebogen* und *Workshops* erhoben. Zum anderen werden diese Informationen anhand einer *Analyse der Systemlandschaft* identifiziert, indem die Datenbankstrukturen betrachtet werden. Diese Methoden

dienen zur Erhebung des Wissens bei den Domain-Experten, um darauf aufbauend die oben erwähnte spezifische Unternehmensontologie zu erstellen. Die beiden Methoden wurden basierend auf den Erkenntnissen in den *Konfigurations-Projekten* erarbeitet. Auch wenn in den Konfigurations-Projekten diese konkreten Methoden nicht notwendig waren, weil sich das Wissen nicht auf einen *SoNBO-Experten* und einem *Domain-Experten* verteilte, sondern in den beteiligten Personen zusammen vorhanden war bzw. erst entwickelt wurde, ließ sich im Anschluss daran die Notwendigkeit ableiten, mit einer Methode das Wissen für die Ontologie bei einem Domain-Experten zu erheben. Dazu scheinen die beiden folgenden Methoden geeignet, welche im Rahmen der Evaluation auf Eignung geprüft wurden (vgl. 6.4). Es ist nicht ausgeschlossen, dass andere Methoden ebenfalls zielführend wären.

5.5.3.1 Technische Sicht: Analyse Systemlandschaft

Auf der Basis des in Phase 1 ermittelten Status quo (vgl. Abschnitt 5.4.2) bezüglich der verwendeten betrieblichen Anwendungssysteme, wird in dieser Phase der Fokus auf die Nutzung der technischen Gegebenheiten für die Strukturierung der Informationen als Social Concepts gelegt. Diese Notwendigkeit zeigte die Erfahrung aus den Konfigurations-Projekten. Bei SoNBO steht zwar die Strukturierung der Informationen als Social Concepts aus Mitarbeitersicht im Vordergrund, sodass ein intuitiver Zugriff auf die Informationen durch die Graphstruktur ermöglicht wird. Allerdings sollte sich auch an den technischen Gegebenheiten orientiert werden, sodass der Aufwand zur Konfiguration so gering wie möglich gehalten wird. Beispielsweise können die Informationen in den Systemen bereits durch Datenbanktabellen oder Zuordnung zu Kategorien strukturiert werden, sodass sich daraus Social Concepts ergeben. Für den Überblick über die zu integrierenden Informationssysteme wird eine Liste mit den Systemen erstellt und die darin befindlichen Strukturierungen in einem offenen Interview erfragt. Diese offene Methode eignet sich, da auf diese Weise auch Informationen ermittelt werden können, die vorher vom SoNBO-Experten nicht antizipiert wurden. Außerdem erfolgt die Erhebung durch den SoNBO-Experten selbst, indem dieser die Systeme analysiert. Neben der Kategorisierung der Informationen in Social (Sub)Concepts sind für die Anbindung der Systeme noch weitere Aspekte relevant: Zum einen muss bei den Systemen geprüft werden, ob eine API vorhanden ist, die für den Zugriff genutzt werden kann. Falls diese nicht existiert, kann ein JSON-String zum Teil die Funktion einer API bereitstellen, indem dieser die Daten in einem einheitlichen Format zur Verfügung stellt und somit den Austausch von Daten ermöglicht. Allerdings kann dann ein Problem bei der Übergabe der Berechtigungen entstehen, wenn der SoNBO-Explorer eingesetzt wird. Die zweite Herausforderung besteht in der Überprüfung eines bestehenden Single-Sign-ons (SSO) und inwiefern dieser durch den SoNBO-Explorer genutzt werden kann. Es sollte unbedingt vermieden werden, dass sich ein Endbenutzer (Mitarbeiter) an den Subsystemen für jede Sitzung erneut anmelden muss. Dies sind zwei technische Herausforderungen bei der Integration, die sich im *Key Case* und in den *Konfigurations-Projekten* zeigten. Weiterhin sollte in dieser Phase auch bereits erhoben werden, wie die Anbindung der Systeme erfolgen kann und wie die Queries zu formulieren sind. Für die Erstellung der Queries sind die Backend-

feldnamen der Attribute relevant, die die Ontologie spezifizieren. In dieser Phase sollte auch geprüft werden, inwiefern diese dokumentiert sind.

5.5.3.2 Business-Sicht: Workshops/Fragebogen

Für die Entwicklung des Social Network of Business Objects werden im Rahmen des Frameworks zwei Methoden vorgeschlagen, die nacheinander durchgeführt werden sollen. Diese Methoden wurden basierend auf den Erkenntnissen in den Konfigurations-Projekten entwickelt. Im ersten Schritt wird mit den Mitarbeitern ein World-Café (offene Methode) durchgeführt, das die Mitarbeiter mit Impulsen zu Gesprächen untereinander anleiten soll, welche der SoNBO-Experte verfolgt. Ein ähnliches World-Café wurde bereits im Rahmen eines IndustryConnect-Workshops (IC06, vgl. Anhang A.2, Abschnitt 2.2.2) mit Praxisvertretern durchgeführt, um den im Key Case identifizierten Ansatz mit Praxisvertretern zu diskutieren. Daraufhin wurde das World-Café für das Framework weiter angepasst. Im zweiten Schritt erhalten die Mitarbeiter einen Fragebogen (geschlossene Methode), den diese ausfüllen sollen. In dem ersten Teil sollen durch die offene Methode auch Wissen und Fehlvorstellungen identifiziert werden, die gegebenenfalls vorher nicht antizipiert wurden. Im zweiten Teil erhalten die Mitarbeiter die Liste mit Social Concepts (vgl. Referenzmodell in Abschnitt 5.5.2.1) und sie wählen die für sich relevanten aus und ergänzen die Liste gegebenenfalls. Wichtig ist dabei für den SoNBO-Experten, die Mitarbeiter auszuwählen, die im Unternehmen einen Überblick an Informationen haben (Domain-Experten), um sinnvolle Social Concepts auszuwählen.

5.5.3.2.1 Workshops (World-Café)

Der SoNBO-Ansatz sieht vor, dass für die Entwicklung der unternehmensspezifischen Ontologie *Social Network of Concepts* ein Referenzmodell (vgl. Abschnitt 5.5.2) mit den Domain-Experten auf das jeweilige Unternehmen angepasst wird. Um an das Wissen der Domain-Experten zu gelangen, wird die offene Methode World-Café vorgeschlagen. Es ist nicht ausgeschlossen, dass andere Methoden ebenfalls zu einem erfolgreichen Ergebnis führen, aber da sich die Methode *World-Café* im Forschungsprozess bewährte (vgl. IndustryConnect06-Workshop im Anhang A.2, vgl. Abschnitt 2.2.2), wurde diese für das SoNBO-Framework als geeignet identifiziert und somit aufgenommen und angepasst. Auch in der Evaluation beim KOSMOS-Verlag wurde diese Methode als geeignet bewertet (vgl. Abschnitt 6.4). Die Durchführung des World-Café an IndustryConnect06 hat folgenden Hintergrund: Zu Beginn des Forschungsprojektes wurde ein Workshop im Rahmen der IndustryConnect-Initiative (Schubert & Williams, 2020; Williams & Schubert, 2017) mit einer Fokusgruppe durchgeführt (vgl. Abschnitt 2.2.2), der das SoNBO-Konzept vorgestellt wurde und die ihre eigenen Social Concepts nennen sollten (vgl. Anhang A.2). Es wurde dazu ebenfalls die Methode World-Café verwendet und die Ergebnisse wurden in einem Arbeitsbericht veröffentlicht (Gebel-Sauer, 2017). Die Analyse der Ergebnisse zeigte, dass sowohl Social Concepts genannt wurden, die nach der eingeführten Definition (vgl. Abschnitt 4.1.2) korrekt sind (z. B. Mitarbeiter, Kunde, Lieferant, Produkt, Projekt, Angebot, Auftrag, Bestellung, Rechnung, Dokumente, Antrag), aber auch solche, die nicht korrekt sind (z. B. Verantwortung, Fähigkeiten,

Arbeitsstunden, Abwesenheit, ToDo), da sie beispielsweise eher als Attribut zu einem Social Concept einzuordnen sind. Es wird vorgeschlagen, dass diese Beispiele eines noch nicht geschärften Verständnisses der Terminologie zu Beginn der Durchführung des World-Cafés im SoNBO-Framework genannt werden, sodass die Mitarbeiter ihrerseits eine Schärfung der Begriffe vornehmen können. Weiterhin zeigte die Erfahrung, dass die Mitarbeiter die Ebenen zwischen Social Concept, Social Subconcept und Social Business Object vermischen. Um auch diesbezüglich eine Schärfung vorzunehmen, kann der SoNBO-Experte die Unterschiede immer wieder durch eine tabellarische Darstellung (vgl. Kapitel 4) bewusst machen. Beide beschriebene möglichen Herausforderungen sind allerdings für den weiteren Konfigurationsprozess nicht hinderlich, da der SoNBO-Experte aus diesen Rohdaten eine Schärfung der Terminologie vornehmen und diese dann wieder in Rückfragen mit den Domain-Experten diskutieren kann („Meintet Ihr das so?“).

Folgende Durchführung des World-Cafés im SoNBO-Framework wird empfohlen: Bevor die gemeinsame Arbeit mit den Domain-Experten beginnt, wird das SoNBO-Konzept in einer Präsentation erklärt. Anschließend werden mit der Methode World-Café verschiedene Fragestellungen thematisiert (vgl. Abb. 5.7). Bei einem World-Café kommen Mitarbeiter mit verschiedenen Hintergründen zu vorgegebenen Fragestellungen zusammen und diskutieren gemeinsam. Der Vorteil bei diesem World-Café für das SoNBO-Framework liegt unter anderem darin, dass die Personenzahl zwischen 1 und 16 beliebig ist, da der SoNBO-Experte im Vorfeld die Anzahl der relevanten Mitarbeiter nicht kennt. Die Teilnehmer bei einem World-Café finden sich in maximal 4 Gruppen zu maximal je 4 Personen zusammen und gehen gemeinsam von Tisch zu Tisch in 4 Runden. Gruppe 1 beginnt beispielsweise bei Tisch 1 und Gruppe 2 bei Tisch 2. An jedem Tisch befindet sich ein Stand-Moderator, der das Thema an jedem Tisch vorstellt und bei Unklarheiten gefragt werden kann. Außerdem ist der Moderator mit weiteren Impulsen ausgestattet, falls keine Diskussion entsteht. Die Gruppe bleibt an jedem Tisch eine gewisse Zeit (z. B. drei Minuten) und notiert die eigenen Beiträge auf einem Plakat. Dazu benutzt jede Person einen personalisierten Stift, so dass man die Eingaben auf Personen zurückführen kann. Wenn die Zeit abgelaufen ist, geht die Gruppe an den nächsten Tisch. Diese Gruppe findet dann schon die Notizen der vorherigen Gruppe vor und kann diese in ihre Diskussion mitaufnehmen und anreichern.

Zur Erhebung der Social Concepts in dem Unternehmen mithilfe des World-Cafés wurden vier Fragestellungen entwickelt (vgl. Abb. 5.7). Die Fragestellungen basieren (wie zu Beginn dieses Abschnitts erwähnt) auf dem im Jahr 2017 durchgeführten World-Café mit Praxisvertretern an IndustryConnect06, um den Key Case Vössing zu diskutieren (vgl. IndustryConnect06-Workshop im Anhang A.2, vgl. Abschnitt 2.2.2). Für das SoNBO-Framework wurden die Fragen weiterentwickelt (vgl. Abb. 5.7). Alle Fragestellungen dienen einerseits dazu, Unklarheiten bei den Mitarbeitern zu identifizieren und durch den Moderator zu beheben. Weiterhin dienen sie dazu, die Inhalte der Fragen zu erfassen.

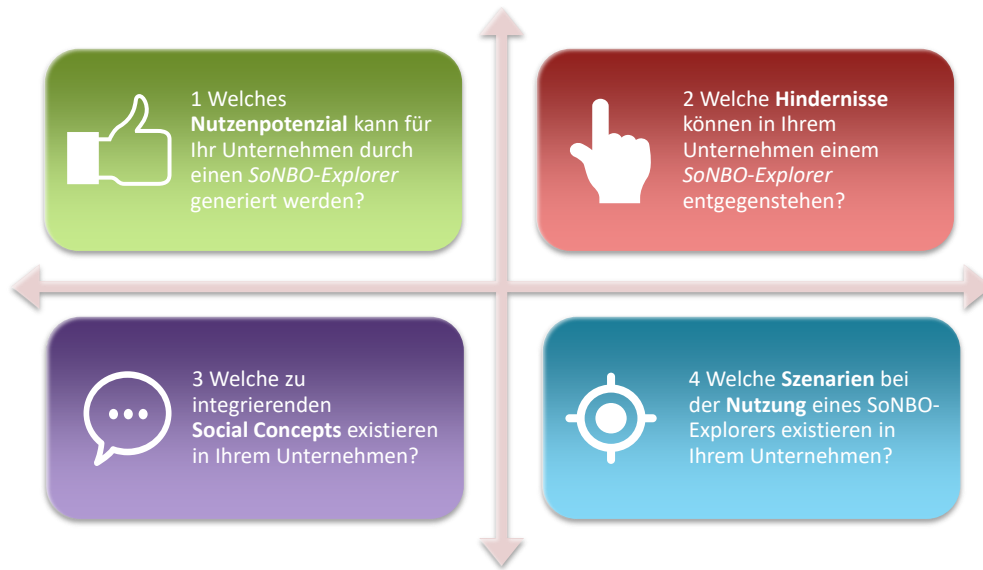


Abb. 5.7: World-Café – Fragen zur Ermittlung der Social Concepts

Zunächst sollen die Mitarbeiter gemeinsam diskutieren, welches Nutzenpotenzial durch einen SoNBO-Explorer generiert werden kann. Am nächsten Tisch soll die Frage besprochen werden, welche Hindernisse im Unternehmen einem SoNBO-Explorer entgegenstehen könnten. Durch den dritten Tisch werden konkrete Social Concepts erhoben. Am vierten Tisch werden konkrete Nutzungsszenarien erfragt. Bei diesem letzten Tisch sollen auf diese Weise noch indirekt Social Concepts ermittelt werden, die am Tisch zuvor nicht genannt wurden. Die ersten beiden Fragen dienen dazu, eventuelle unternehmenspolitische Begünstigungen oder Hürden zu identifizieren. Die Bearbeitung dieser Fragen setzt eine Einführung von SoNBO durch den SoNBO-Experten voraus, was im Vorfeld geschehen ist. Diese Rohdaten werden von dem SoNBO-Experten anschließend in ein Excel-Dokument übertragen und kategorisiert.

5.5.3.2.2 Quantitative Fragebögen

Als zweite Methode zur Entwicklung der Ontologie *Social Network of Concepts* wird im SoNBO-Framework vorgeschlagen, dass die Domain-Experten einen Fragebogen (vgl. Tab. 5.4, Tab. 5.5, Tab. 5.6) erhalten, durch welchen das unternehmensübergreifende Referenzmodell (vgl. Abschnitt 5.5.2) aus dem SoNBO-Framework bereitgestellt wird und die Domain-Experten darauf basierend das für sie relevante auswählen bzw. erweitern sollen (geschlossene Methode).

Tab. 5.4: Fragen zur Erhebung der Social Concepts

Nr.	Aufgabe	Bearbeitung in
1	Markieren Sie mit einem X die Social Subconcepts, die für <i>Unternehmen XY</i> von Bedeutung sind.	Spalte 2
2	Tragen Sie ggf. fehlende Social Concepts bzw. Social Subconcepts in den letzten Zeilen ein.	Letzte Zeilen
3	Tragen Sie ggf. die Begriffe ein, die Sie für das jeweilige Social (Sub-)Concept beim <i>Unternehmen XY</i> verwenden. <i>Beispiel: Rechnung an den Kunden → Verkaufsrechnung, Ausgangsrechnung, Kundenrechnung, ...</i>	Spalte 4
4	Tragen Sie den Namen eines konkreten Social Business Objects ein (falls Sie in Spalte 2 ein X gesetzt haben). <i>Beispiel: Ein Social Business Object zum Social Subconcept Mitarbeiter: <u>Peter Müller</u></i>	Spalte 5
5	Tragen Sie die Software als Datenquelle ein, in der sich die Social Business Objects befinden. <i>Beispiel: zu Social Subconcept Rechnung: <u>SAP</u></i>	Spalte 6
6	Tragen Sie die Attribute des jeweiligen Social Subconcepts ein, die häufig gebraucht werden. <i>Beispiel: zu Social Subconcept Rechnung: <u>Rechnungsbetrag</u></i>	Spalte 7
7	Tragen Sie die Statusveränderungen an den Social Subconcepts (in den Stammsystemen) ein, die im SoNBO-Explorer verwendet werden sollen. <i>Beispiel: zu Social Subconcept Rechnung: <u>Rechnung wurde bezahlt</u></i>	Spalte 8
8	Kreuzen Sie in der Matrix die Beziehungen zwischen den Social Subconcepts an. Falls Sie bei Aufgabe 2 Social Subconcepts nachgetragen haben, ergänzen Sie diese ebenfalls in der Matrix.	Matrix

Tab. 5.5 stellt das Raster dar, das die Mitarbeiter für das Beantworten der Fragen nutzen sollen. Dieses basiert auf der Taxonomie (vgl. Tab. 5.3) und der Ontologie (vgl. Abb. 5.5 und Abb. 5.6). Das Raster wird durch die Domain-Experten ausgefüllt, indem diese dazu strukturiert durch die Fragen aus Tab. 5.4 angeleitet werden: Zunächst wählen die Mitarbeiter die Social Subconcepts und damit auch die Social Concepts aus, welche es in ihrem Unternehmen gibt (Tab. 5.5). Anschließend wird ihnen mit der Matrix in Tab. 5.6 ermöglicht, die Beziehungen zwischen den Social Subconcepts anzugeben. Es wird also konkret das für das SoNBO-Framework entwickelte Referenzmodell den Domain-Experten zur Verfügung gestellt, sodass diese das Modell auf ihr Unternehmen anpassen können.

Tab. 5.5: Tabelle zur Erhebung der Social Concepts bei den Mitarbeitern

1	2	3	4	5	6	7	8
Social Concept		Social Sub-concept	Synonyme Bezeichnungen	Social Business Object (Beispiel)	Datenquelle (Produktname)	Attribute	Statusveränderung
Kunden-Beleg		Angebot					
		Auftrag					
		Lieferschein					
		Rechnung					
		Mahnung					
Lieferanten-Beleg		Bestellung					
		Lieferschein					
		Rechnung					
...		...					

Den Mitarbeitern wird durch das Raster ermöglicht, dass sie die Tabelle in ihrer „Unternehmenssprache“ ausfüllen können und sollen. Für den SoNBO-Explorer sollten dann allerdings die Begriffe an die hier vorgeschlagene Terminologie (vgl. Abschnitt 5.5.2.1) angepasst werden. Dies muss dann dem Mitarbeiter bewusst gemacht werden.

Tab. 5.6: Matrix für die Identifizierung der Beziehungen

	...	Stückliste	Dienstleistung	Position	Artikel	Rohmaterial	Projekt	Lagerplatz	Lagerort	Abteilung	Gebäude	Niederlassung	Universität	Verein	Öffentliche Einrichtung	Lieferantenunternehmen	Kundenunternehmen	Lieferantenkontakt	Kundenkontakt	Mitarbeiter	Vertrag	Bedarfsanforderung	Rechnung (Lieferant)	Lieferschein (Lieferant)	Bestellung	Mahnung (Kunde)	Rechnung (Kunde)	Lieferschein (Kunde)	Auftrag	Angebot	
Angebot																															
Auftrag																															
Lieferschein (Kunde)																															
Rechnung (Kunde)																															
Mahnung (Kunde)																															
Bestellung																															
Lieferschein (Lieferant)																															
Rechnung (Lieferant)																															
Bedarfsanforderung																															
Vertrag																															
Mitarbeiter																															
Kundenkontakt																															
Lieferantenkontakt																															
Kundenunternehmen																															
Lieferantenunternehmen																															
Öffentliche Einrichtung																															
Verein																															
Universität																															
Niederlassung																															
Gebäude																															
Abteilung																															
Lagerort																															
Lagerplatz																															
Projekt																															
Rohmaterial																															
Artikel																															
Position																															
Dienstleistung																															
Stückliste																															
...																															

5.5.4 Tool: SoNBO-Visualiser

In diesem Abschnitt wird der SoNBO-Visualiser vorgestellt, mit dem die entwickelte Ontologie visualisiert wird. Der SoNBO-Visualiser ist ein Tool, welches im Rahmen der SoNBO-Forschung durch eine Qualifikationsarbeit an der Universität Koblenz entwickelt (Meier, 2019) und veröffentlicht (Meier et al., 2020) wurde. Dieser visualisiert die Ontologien *Social Network of Concepts* (vgl. Abb. 5.8) und *Social Network of Subconcepts* sowie den Knowledge Graphen *Social Network of Business Objects*. Die Notwendigkeit zur Entwicklung eines solchen Tools entstand, nachdem für die Visualisierung der Ontologien mit den SoNBO-Merkmalen kein angemessenes Tool durch die Autorin der Dissertation identifiziert werden konnte.

In den SoNBO-Visualiser wird zunächst das Social Network of Concepts (vgl. Abb. 5.5) eingegeben, indem die Knoten und die Beziehungen eingerichtet werden. Auf einer weiteren Ebene werden die Social Subconcepts als Knoten eingegeben und den zugehörigen Social Concepts zugeordnet. Die Beziehungen zwischen den untergeordneten Subconcepts werden dann automatisch erzeugt auf der Basis der Beziehungen, die zwischen den übergeordneten Social Concepts bestehen (vgl. Abb. 5.6). Falls eine Beziehung dann auf der unteren Ebene doch nicht existiert, kann diese einfach wieder entfernt werden. Die beiden Ontologien sind in Abschnitt 5.5.2.2 dargestellt.

Mit dem Tool kann außerdem der Datenzugriff auf die Systeme eingerichtet werden, sodass der Knowledge Graph bzw. das Social Network of Business Objects erzeugt werden können. Weiterhin können für jedes Social Subconcept auch die Attribute angegeben werden, die ebenfalls im SoNBO-Explorer verwendet werden sollen. Falls diese Informationen in den SoNBO-Visualiser eingegeben wurden, können diese Informationen auch für die Konfiguration des SoNBO-Explorers verwendet werden.

Das Tool kann also sowohl zur Visualisierung des Netzwerks als auch zur Konfiguration verwendet werden, welche dann auch für den SoNBO-Explorer genutzt werden kann. Es ist also ein Tool zur Unterstützung der Konfiguration und bildet eine Gesprächsgrundlage zwischen dem SoNBO-Experten und den Mitarbeitern, indem die Ontologie und der Knowledge Graph visualisiert werden. Da die Einbindung von Mitarbeitern ohne technisches Know-how bei SoNBO von großer Bedeutung bei der Ontologie-Erstellung ist, konnte kein existierendes Visualisierungstool verwendet werden, da in diesen die Ontologie beispielsweise als RDF-Graph angezeigt wird und somit von einem Menschen ohne technisches Know-how schwierig zu lesen ist. Das vorgestellte Tool bewährte sich auch im Praxistest (vgl. Abschnitt 6.4).

Social Network of Concepts **Social Network of Subconcepts** **Social Network of Business Objects**

Social Concept
 Debugdaten hinzufügen
 Bezeichnung
 Hinzufügen Aktualisieren Löschen

Kante
 Bezeichnung
 Von
 Bitte auswählen
 Nach
 Bitte auswählen
 Hinzufügen Aktualisieren Löschen

```
id: "cd85e33b-1387-4969-9ba4-0814d0669f17"
color: {"background": "#c0c0c0"}
description: "Lagerplatz"
markedForDeletion: 0
font: {"multi": true}
label: "<code>-- Lager --</code>\nLagerplatz"
socialConcept: "7cd3b1de-0b5e-46f9-a14e-538f251f2934"
labelConcept: "<code>-- Lager --</code>"
```

Stabilisieren Physikengine aktiviert Knoten fokussieren Gesamtes Netzwerk anzeigen

Abb. 5.8: Benutzerfrontend SoNBO-Visualiser (für Ebene: Social Network of Concepts)

5.6 Phase 3: Mapping

Das Ziel dieser Phase besteht im Erstellen des *Mappings* der Informationen in der zuvor entwickelten Ontologie auf die Daten in den Datenbanken der anzubindenden Quellsysteme. In dieser Phase ist wieder der *SoNBO-Experte* beteiligt sowie die *Domain-Experten* und der *IT-Experte*, der für den SoNBO-Explorer verantwortlich ist. Im Folgenden wird daher eine Methode und ein Tool (*SoNBO-Mapper*) für den *Informationsaustausch* zwischen diesen Akteuren vorgeschlagen, sodass das semantische und technische Mapping erstellt werden kann. Dieses wird über das Administrationsfrontend des SoNBO-Explorers eingegeben. Die Erkenntnisse stammen aus dem Key Case und den *Konfigurations-Projekten*. Die Anwendung *SoNBO-Explorer*, dessen Frontend auch die Benutzer im Unternehmen nutzen werden, wurde in Abschnitt 5.2 beschrieben.

5.6.1 Beteiligte Akteure

In dieser Phase wird vorgeschlagen, dass folgende Akteure von Bedeutung sind: Der *SoNBO-Experte* erarbeitet in dieser Phase mit den *Domain-Experten* und dem *IT-Experten* die Attribute, die für die jeweiligen Knoten notwendig sind, indem eine sinnvolle Auswahl von Attributen aus den Datenbanken der Quellsysteme getroffen wird. Die Domain-Experten konzentrieren sich dabei auf die Auswahl der Attribute, die für die Anforderungen der Benutzer relevant sein werden. Der IT-Experte fokussiert auf die technisch möglichen Attribute sowie auf die Attribute, die ggf. benötigt werden, aber im Frontend nicht ersichtlich sind. Beide Parteien werden mithilfe eines vorstrukturierten Schemas, welches als Mapping-Methode (vgl. Abschnitt 5.6.2) im Folgenden vorgestellt wird, durch den SoNBO-Experten angeleitet. Das Tool *SoNBO-Mapper* (vgl. Abschnitt 5.6.2) unterstützt dabei. Diese Erkenntnisse für die Notwendigkeit dieser Akteure in dieser Phase basieren auf der Analyse des *Key Case* und die Erfahrung aus den *Konfigurations-Projekten*.

5.6.2 Methode: Mapping und Tool: SoNBO-Mapper

In diesem Abschnitt wird eine Methode vorgeschlagen, mit welcher das *Mapping* der Informationen in der Ontologie mit den Daten in den Quellsystemen durchgeführt werden kann. Die Erfahrungen aus den *Konfigurations-Projekten* zeigte, dass zwischen dem semantischen und dem technischen Mapping unterschieden werden muss. Unter dem semantischen Mapping wird die Zuordnung der benötigten Attribute eines jeden Social Subconcepts in der Ontologie zu den Attributen in den Quellsystemen verstanden. Mit dem technischen Mapping ist die Zuordnung der Queries zu den einzelnen Attributen gemeint, um diese in den Quellsystemen abzurufen. Bei dem *Mapping* unterstützt ein auf Excel basierendes Tool (*SoNBO-Mapper*). Die Eingabe des Mappings in den SoNBO-Explorer wird Konfiguration genannt.

Im Folgenden wird das semantische und das technische Mapping erklärt, indem sowohl die Methode als auch das Tool dafür zusammen beschrieben werden.

5.6.2.1 Semantisches Mapping

Für das semantische Mapping werden mit den Domain-Experten für jedes Social Subconcept die notwendigen Attribute aus den Quellsystemen und die Verwendung in der Ontologie festgelegt (vgl. Tab. 4.5 auf S. 106). Das semantische Mapping wird in Excel durchgeführt und das Excel-Dokument wird *SoNBO-Mapper* genannt. Es wird nicht ausgeschlossen, dass es weitere Technologien gibt, die ebenfalls geeignet sind, aber Excel bewährte sich, da das Mapping für den *Domain-Experten* verständlich durchgeführt werden kann. Für die Eingabe des Mappings in den SoNBO-Explorer erfolgt dann eine Änderung der Struktur durch Verweise, sodass dies nicht mehr für den Domain-Experten, aber für den SoNBO-Explorer verarbeitbar ist. Für das Mapping in Zusammenarbeit mit dem Domain-Experten wird für jeden Knoten in einem Excel-Blatt ein Tab angelegt, der den Aufbau wie in Abb. 5.9 und Abb. 5.10 hat. Im oberen Bereich links befinden sich die Informationen über das *Social Subconcept* (Zelle A2) und

das zugehörige *Social Concept* (Zelle A1). In A3 bis A5 werden die Informationen zu dem Quellsystem notiert, in dem sich die Informationen zu dem Social Subconcept befinden. In A7 wird die Anzahl der Kanten eingetragen, zu dem der Knoten *Social Subconcept* eine Beziehung hat. Da die Kanten lediglich in eine Richtung definiert werden müssen, diese dann aber für beide Richtungen gelten, werden für jedes Social Subconcept maximal so viele Beziehungen definiert wie es Kanten besitzt. Dies wird in A8 festgehalten und dient zur Kontrolle. Die Legende befindet sich unterhalb der Tabelle (A/B45 – A/B53) und in A10 wird darauf verwiesen. Außerdem wird das Social Network of Subconcepts abgebildet und der in diesem Tab definierte Knoten umrandet (in dieser Abbildung nicht zu sehen). Dies dient zur Orientierung für den SoNBO-Experten und die eingebundenen Mitarbeiter. Mit E1 wird auf ein schematisches Mockup des SoNBO-Explorers verwiesen, da einige zu definierende Attribute im Benutzerfrontend visualisiert werden. Dies dient ebenfalls zur Orientierung für die beteiligten Akteure. Ab Zeile 28 wird die Spezifizierung des betreffenden Social Subconcepts und somit das semantische Mapping vorgenommen. In den Zellen B29 bis O29 wird die Verwendung des Attributes im SoNBO-Explorer angezeigt und falls dies eine Auswirkung auf das Benutzerfrontend hat, befindet sich in der Zeile darüber die Zahl des jeweiligen Profilbereichs (1-4). In der Zeile 30 wird der zu erwartende Eingabewert (bezogen auf die jeweilige Spalte) angezeigt. In der Regel wird ein Kreuz oder kein Kreuz erwartet. Die darunterliegende Zeile (31) zeigt die Anzahl der Eingabewerte an. Entweder muss genau ein Wert ([1]) eingegeben werden oder beliebig viele ([n]). Ab B32 werden die Attributnamen gelistet, die aus dem Quellsystem für den SoNBO-Explorer relevant sind. Dabei handelt es sich um die Attributnamen, die im Frontend des Quellsystems ersichtlich sind, da verschiedene Akteure beteiligt sind und somit auch für den Mitarbeiter deutlich wird, worum es sich handelt. Für den Datenzugriff werden in der nächsten Spalte die Backendfeldnamen der Quellsysteme eingetragen. Dies erfolgt in der Regel am Ende des semantischen Mappings. In der nächsten Spalte wird ein Beispielwert eingetragen, sodass bei den Akteuren eine Vorstellung entsteht, um welchen Attributwert es sich handelt. Ab der Spalte E wird nun mit Kreuzen die Verwendung des Attributes für den SoNBO-Explorer festgelegt. Ein Attribut muss als ID verwendet werden, sodass ein Social Business Objects eindeutig angesprochen werden kann. Bei einer Person kann dies beispielsweise durch die E-Mail-Adresse erfolgen. Außerdem ist die Zuordnung eines Social Business Objects zu dem Social Subconcept relevant. Dies kann durch den Eintrag in einer Datenbanktabelle (z. B. Lieferantenkontakte) sein oder indem die Lieferanten in dem jeweiligen Formular durch Attribute weiter klassifiziert werden (z. B. Hersteller, Dienstleister). Falls die Zuordnung zu dem Social Subconcept durch ein Attribut geschieht, wird unter F29 ein Kreuz gesetzt. In den nächsten Spalten G29 bis K29 wird die Visualisierung der Attribute im Benutzerfrontend festgelegt (Verwendung als Data Properties, siehe Ontologie-Abschnitt 3.2.1.2.2). Ein Attributwert wird beispielsweise für den Namen des Social Business Objects ausgewählt, der auch im Frontend angezeigt wird (bei einer Person z. B. Vor- und Nachname). Außerdem werden weitere Attribute angezeigt, die das Social Business Object weiter beschreiben (bei einer Person z. B. die E-Mailadresse, Büronummer, Position, ...). In der nächsten Spalte I kann ein neuer Feldname für das Attribut eingetragen werden, falls nicht der Name aus dem Quellsystem übernommen werden soll. In der nächsten Spalte wird ein Bild ausgewählt. Bei Personen handelt es sich idealerweise um ein Foto. In Spalte K werden die Attribute ausgewählt, die

im Ergebnisbereich als Kurzinfo zu dem Social Business Object angezeigt werden sollen. Da der Ergebnisbereich bestehend aus Social Business Objects auch nach Attributen gefiltert und sortiert werden soll, wird in den nächsten beiden Spalten festgelegt, nach welchen Attributen die Social Business Objects gefiltert werden sollen. Bei Personen ist es beispielsweise sinnvoll, nach Nachnamen zu sortieren. Eine Filterung ist bei Personen beispielsweise nach Abteilung sinnvoll. In *Spalte N* werden die Attribute ausgewählt, mit denen eine Beziehung bzw. Kante zu einem weiteren Social Business Object definiert werden kann (Verwendung als Object Property, siehe Ontologie-Abschnitt 3.2.1.2.2). Die Anzahl der Kreuze muss mit dem Wert in *Zelle B8* übereinstimmen. Bei einer Person ist beispielsweise das Gebäude als Attribut vermerkt, sodass mit diesem Attribut für den SoNBO-Explorer die Kante definiert werden kann. In der letzten *Spalte O* werden die Attribute festgelegt, bei denen eine Statusänderung relevant ist. Dies wird im Profil im rechten Bereich als *Recent Events* angezeigt und hat Auswirkungen auf den *Activity Stream* des Kollaborationssystems (vgl. Abschnitte 3.2.1.7 und 4.2), in das der SoNBO-Explorer integriert ist. Falls ein Mitarbeiter beispielsweise das Büro wechselt, wird in diesem Feld ein neuer Eintrag vorgenommen. Durch diese Änderung wird beispielsweise folgende Notification ausgelöst: Mitarbeiter *Peter Müller* wechselt von Raum 101 in Raum 201. Diese Notification wird in dem Profil von *Peter Müller* auf der rechten Seite angezeigt. Außerdem erhält der Mitarbeiter *Peter Müller* selbst und die mit ihm verknüpften Personen diese Notification in dem Activity Stream des Kollaborationssystems angezeigt.

Der Vorteil bei der Nutzung eines Excel-Dokumentes besteht (wie bereits angesprochen) darin, dass die Verwendung der Attribute auf diese Weise mit den beteiligten Akteuren strukturiert erarbeitet werden kann. Für die Konfiguration des SoNBO-Explorers muss dies dann in eine andere Form umformatiert werden, was im SoNBO-Mapper in einem weiteren Tab durch entsprechende Verweise geschehen kann. Somit werden nachträgliche Änderungen in der Konfigurationstabelle direkt in die Struktur des SoNBO-Explorers übertragen, sodass diese dann nur noch importiert werden muss. Für die nachträgliche Konfiguration eines weiteren Knotens kann auch das Administrationsfrontend des SoNBO-Explorers genutzt werden.

5 SoNBO-Framework

	A	B	C	D	E	F	G	H	I	J	K	
1	Social Concept (SC)	Person person										
2	Social Subconcept (SSC)	Mitarbeiter employee										
3	Quellsystem											
4	Name DB											
5	Query											
6												
7	Anzahl Kanten zu anderen Social Subconcepts											
8	Anzahl der hier definierten Kanten											
9												
10	Legende	unterhalb der Mapping-Tabelle										
11												
12												
13												
14												
15												
16												
17												
18												
19												
20												
21												
22												
23												
24												
25												
26												
27												

Abb. 5.9: Konfigurationstabelle für das semantische Mapping (I)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
26															
27															
28	Auswirkung auf Profilbereich	-	-	-	-	2	1	1	1	1	4	5	3	-	6
29	Überschrift	Feldname (Frontend)	Feldname (Backend)	Beispielwert	ID	Zuordnung zu SSC	Name des Social Business Objects	Attribute	Feldname für das Attribut (im SONBO-Explorer)	Bild	Attribute als Kurzinfo	filterbare Attribute	sortierbare Attribute	Beziehung	Statusänderung
30	Eingabewert	-	-	-	[" "; "x"]	[" "; "x"]	[" "; "x"]	[" "; "x"]	[String]	[String; "x"]	[" "; "x"]	[" "; "x"]	[" "; "x"]	[" "; "x"]	[" "; "x"]
31	Anzahl der Eingabewerte	-	-	-	[1]	[1]	[n]	[n]	[n]	[1]	[n]	[n]	[n]	[n]	[n]
32	Attribut 1	Vorname													
33	Attribut 2	Nachname													
34	Attribut 3	E-Mail Adresse													
35	Attribut 4	Telefonnummer													
36	Attribut 5	Handynummer													
37	Attribut 6														
38	Attribut 7														
39	Attribut 8														
40	Attribut 9														
41	Attribut 10														
42	Hinweis														
43															
44															
45	Legende														
46		Eingefügt aus dem Quellsystem													
47		Festlegung der Verwendung des Attributes als ID													
48		Festlegung der Verwendung des Attributes als Zuordnung zum SSC (= Social Subconcept)													
49		Festlegung der Verwendung des Attributes zur Visualisierung im Frontend													
50		Festlegung der Verwendung des Attributes zur Filterung/Sortierung													
51		Festlegung der Verwendung des Attributes zur Erstellung der Beziehung/Kante zwischen zwei Social Socialsubconcepts													
52		Festlegung der Verwendung der Änderung des Attributes zur Statusmitteilung													
53	[1]	genau ein Kreuz													
54	[n]	beliebig viele Kreuze													
55															

Abb. 5.10: Konfigurationstabelle für das semantische Mapping (II)

5.6.2.2 Technisches Mapping

Mit dem technischen Mapping wird der Datenzugriff auf die für SoNBO benötigten Attribute eingerichtet und durch die Logik des SoNBO-Explorers werden diese für den in der Tabelle angegebenen Zweck verwendet. Zu der Vorbereitung des technischen Mappings gehört in dem Excel-Tab (vgl. Abb. 5.9 und Abb. 5.10) der linke obere Bereich, indem die Query und die Datenbank eingetragen werden. Anschließend werden die Queries für die Attribute in den Quellsystemen formuliert. Dies wurde konkret für den *Key Case KOSMOS* im Rahmen der Evaluation (vgl. Abschnitt 6.5) durchgeführt, da dies abhängig vom anzubindenden Quellsystem ist.

5.7 Grenzen des Frameworks

Der Erfolg bei der Einführung und der Konfiguration eines SoNBO-Explorers ist nicht nur von der Durchführung der beschriebenen Schritte abhängig, sondern auch von der Kompetenz des durchführenden SoNBO-Experten und der Auswahl der Domain-Experten. Die Modelle, Methoden und Tools unterstützen lediglich Personen, die das notwendige Know-how besitzen.

Wie bereits angesprochen ist es eine offene Frage, ob der Datenzugriff ohne das Vorliegen einer Lizenz für das Quellsystem lizenzrechtliche Probleme birgt. Außerdem kann es sein, dass je nach Art der Anbindung bei einem Datenzugriff über eine API die Berechtigungen nicht mitübergeben werden und somit Benutzer einen Informationszugriff erhalten, für den sie nicht berechtigt sind. Weiterhin kann der SSO je nach Implementierung zu sicherheitstechnischen Problemen (z. B. bei der Art der Abspeicherung der Passwörter) führen. Darüber hinaus ist der SoNBO-Explorer noch keine professionelle Standardsoftware, die in jedem beliebigen Unternehmen eingesetzt werden kann. Das Ziel der Dissertation ist allerdings nicht, diese genannten Herausforderungen zu lösen, sondern lediglich die konzeptionelle Grundlage für SoNBO zu legen und ein Framework zu schaffen, das bei der Einführung und somit bei der Konfiguration eines SoNBO-Explorers unterstützt. Die hier identifizierten Probleme können und müssen bei einem realen Einführungsprojekt von den Verantwortlichen diskutiert und gelöst werden.

5.8 Zusammenfassung

In diesem Kapitel wurden die Schritte des Frameworks beschrieben, die für die Einführung eines SoNBO-Explorers relevant sind. Außerdem wurden die in den einzelnen Prozessschritten enthaltenen Modelle, Methoden und Tools erklärt, die für das Ziel von Bedeutung sind. Im Folgenden wird die Evaluation dieses Frameworks im Unternehmen KOSMOS-Verlag beschrieben und die erzielten Projektergebnisse und Erkenntnisse für das SoNBO-Framework werden erläutert.

6 Evaluation des SoNBO-Frameworks

Im vorangegangenen Kapitel wurde das entwickelte SoNBO-Framework beschrieben, dessen Basis die Analyse des *Key Case Vössing* (vgl. Abschnitt 3.1), die *Literaturanalyse* von existierenden Ontologie-basierten Integrationsansätzen (vgl. Abschnitt 3.2), die *Konzeptualisierung* von SoNBO (vgl. Kapitel 4) und die Erfahrung aus zwei *Konfigurations-Projekten* (Gebel-Sauer & Schubert, 2019b; Götz & Gebel-Sauer, 2018) sind. Das Ziel des SoNBO-Frameworks ist die Unterstützung des SoNBO-Experten bei der Einführung und somit Konfiguration eines SoNBO-Explorers in einem beliebigen Unternehmen. Die primäre Zielgruppe ist damit der SoNBO-Experte, der, unterstützt durch die Modelle, Methoden und Tools des Frameworks, in Zusammenarbeit mit den Mitarbeitern des Unternehmens die Einführung durchführt. Das Framework wurde wie im Design Science Research gefordert (vgl. Abschnitt 2.1) in der Praxis getestet und damit in einer ersten Form evaluiert. Dies wurde durch die Arbeit mit IndustryConnect ermöglicht (vgl. Abschnitt 2.2.2). Ein Ziel dieser Forschungsinitiative ist, diesen letzten Schritt (Rücktransfer der in der Wissenschaft gewonnenen Ergebnisse in die Praxis) nach Nunamaker et al. (2015) zu gehen (Schubert & Williams, 2020). Im Folgenden werden die Ergebnisse der Evaluation und die für das Framework relevanten Erkenntnisse vorgestellt. Für die Diskussion der Erkenntnisse wird die wissenschaftliche Literatur herangezogen.

6.1 Ziel und Hintergründe der Evaluation

Das Ziel der Evaluation besteht in der Überprüfung der Modelle, Methoden und Tools im SoNBO-Framework auf Nützlichkeit für die erfolgreiche Einführung und Konfiguration eines SoNBO-Explorers in einem beliebigen Unternehmen. Für das Erreichen des Ziels wurde der SoNBO-Explorer im Unternehmen *Franckh-Kosmos Verlags-GmbH & Co. KG* (im Folgenden: KOSMOS Verlag) eingeführt, indem dieser unter Zuhilfenahme des in dieser Dissertation entwickelten Frameworks konfiguriert wurde. Damit wurden die im Framework enthaltenen Methoden getestet und während der Anwendung bei Bedarf angepasst bzw. verfeinert.

Die Zusammenarbeit mit dem KOSMOS Verlag kam nach einem Vortrag über SoNBO an einem Roundtable¹⁶ (Südwest) der IBM in Ehningen Ende Februar 2019 zu Stande. Die Daten dazu sind im Anhang aufgeführt (vgl. A.2). Das Interesse an der Teilnahme an dem Forschungsprojekt entstand vor dem folgenden Hintergrund: Der KOSMOS Verlag setzt IBM¹⁷-Produkte ein, hatte Probleme bei der Adoption ihres IBM Connections und das Unternehmen stand vor der Herausforderung, die Informationen der

¹⁶ Roundtable Südwest – Zielgruppe: IBM Connections-Anwender im Südwesten von DACH (= Deutschland, Österreich, Schweiz); Zweck: Austausch über den Einsatz von IBM Connections

¹⁷ Die Produkte IBM Connections und IBM Notes/Domino wurden Ende 2018 von IBM an HCL verkauft. Mitte 2019 wurde der Verkauf rechtskräftig und es erfolgte eine Änderung des Namens in HCL Connections und HCL Notes/Domino. Hier wird IBM genannt, da zu diesem Zeitpunkt das Produkt noch der IBM gehörte.

historisch gewachsenen IT-Landschaft zu integrieren. Der KOSMOS Verlag ist ein geeigneter Evaluationspartner bezüglich der Größe (etwa 200 Mitarbeiter) und bezüglich der eingesetzten Software. Da dort die IBM Suite bestehend aus IBM Connections, IBM Notes und Domino (mittlerweile: HCL Connections, HCL Notes und Domino) im Einsatz war, stellten sich bei der Implementation des SoNBO-Explorers keine neuen technischen Herausforderungen (da in der Forschungsgruppe die gleiche Unternehmenssoftware eingesetzt wird).

Der KOSMOS Verlag stellt Spielwaren und Bücher für Kinder und Erwachsene her. Im Rahmen des Evaluationsprojektes wurde eine eXperience-Fallstudie aufgenommen, welche detaillierte Informationen über das Unternehmen beinhaltet (Gebel-Sauer & Grams, 2019). Im Folgenden wird das Unternehmen kurz vorgestellt mit den Informationen, die für die Evaluation relevant sind.

Der KOSMOS Verlag ist ein im 19. Jahrhundert gegründetes Unternehmen mit Sitz in Stuttgart. Es beschäftigt etwa 200 Mitarbeiter und setzt professionelle Unternehmenssoftware wie z. B. SAP ERP oder die HCL Produktpalette wie z. B. HCL Connections, HCL Notes und HCL Domino ein. KOSMOS arbeitet eng mit dem IT-Partner agentbase zusammen. Die Kundenseite und die Lagerhaltung organisiert der KOSMOS Verlag nicht selbst, da es im Verlagswesen üblich ist, einen Dienstleister (VVA = Vereinigte Verlagsauslieferung) dafür einzusetzen. Somit konzentriert sich KOSMOS auf die Produktentwicklung mit Lieferanten (Autoren, Illustratoren etc.).

Tab. 6.1: Projektablauf

Roundtable	Kick-off	1. Meeting	2. Meeting	3. Meeting ¹⁸	4. Meeting
22.02.2019	04.04.2019	22.08.2019	28.11.2019	11.03.2020 – 24.04.2020	24.08.2020- 01.10.2020
Ehningen (IBM Zentrale)	Koblenz (Universität Koblenz-Landau)	Stuttgart (KOSMOS Verlag)	Stuttgart (KOSMOS Verlag)	Web-Konferenz	Web-Konferenz
Vorbesprechung	Vorbesprechung, Phase 1	Einführung in das Projekt mit dem Management, Durchführung Phase 1, Phase 2	Durchführung Phase 2, Phase 3	Durchführung Phase 3	Durchführung Phase 3
DGE, BGS	DGE, BGS, PSC	DGE, BPR, BGS, PSC, SGR	DGE, BGS, PSC, KGO	DGE, BGS, PSC, SME, KGO	DGE, BGS, PSC, SME, KGO

An dem Evaluationsprojekt waren unterschiedliche Akteure beteiligt, die sich den Akteuren im SoNBO-Framework zuordnen lassen (vgl. Abb. 5.4): Der SoNBO-Experte war die Autorin der Dissertation (BGS).

¹⁸ Das 3. und 4. Meeting fanden in mehreren Calls als Web-Konferenz statt, da sich die Beteiligten nicht physisch getroffen haben. Da die Calls jeweils dasselbe Thema behandelten, wird die Summe der Calls jeweils als ein Meeting bezeichnet, da die Thematik in einem physischen Treffen an einem Termin behandelt worden wäre.

Die Domain-Experten und der IT-Experte wurden vom KOSMOS Verlag bereitgestellt und sind in diesem Fall Dirk Gerlach (Kürzel: DGE¹⁹), der beide Rollen einnimmt, und VPR, die vor allem als Domain-Experte dient. Das SoNBO-Expertenteam wird von weiteren Vertretern der Universität unterstützt (PSC, SGR, SME, KGO). Nach dem Termin am Roundtable fand am 04. April 2019 das Kick-off Meeting des Projektes statt (vgl. Tab. 6.1). An diesem Termin wurden bereits erste Informationen zu dem *Status quo* (entspricht 1. Phase des *SoNBO-Frameworks*) beim KOSMOS Verlag (insbesondere die im Einsatz befindlichen Systeme) gesammelt. Im Rahmen des 1. Meetings am 22. August 2019 in Stuttgart beim KOSMOS Verlag wurden die 1. Phase (*Status quo*) sowie erste Erhebungen zur Phase 2 (Ontologieentwicklung) durchgeführt. Außerdem fand eine Informationsveranstaltung zu SoNBO für die Geschäftsleitung statt. Im 2. Meeting am 28. November 2019 wurden die Ergebnisse aus dem vorherigen Meeting in Form der Ontologie *Social Network of Concepts* vorgestellt und gemeinsam weiter angepasst. Außerdem wurden die Informationen für die Phase 3 aufgenommen, indem die Mappingtabelle gemeinsam ausgefüllt wurde. Am 3. und 4. Meeting wurde die Phase 3 finalisiert, indem der SoNBO-Explorer mit der KOSMOS-Ontologie erfolgreich konfiguriert wurde und auf den Servern des KOSMOS Verlags installiert wurde.

6.2 Erkenntnisse für das SoNBO-Framework

Für die Evaluation wurde im KOSMOS Verlag das SoNBO-Framework mit den Modellen, Methoden und Tools verwendet, um einen SoNBO-Explorer einzuführen und zu konfigurieren. Dies verlief erfolgreich und das Ergebnis ist ein für den KOSMOS Verlag konfigurierter SoNBO-Explorer. Mit dem Framework kann also das Ziel (Einführung von SoNBO in einem beliebigen Unternehmen) erreicht werden. Die Differenzierung in die verschiedenen Akteure (SoNBO-Experte, Domain-Experte und IT-Experte) nach Sequeda & Miranker (2017) im SoNBO-Framework und die Auswahl der beteiligten Akteure in dem Evaluationsprojekt ist somit auch gelungen. Es wurde vorrangig mit DGE (Leitung IT) zusammengearbeitet, der für das SoNBO-Projekt im KOSMOS Verlag verantwortlich war und auch in Zukunft sein wird. Dieser wählte in Rücksprache mit dem SoNBO-Experten (BGS) die Domain-Experten aus. Die Identifizierung von geeigneten Mitarbeitern im Unternehmen blieb aber eine Herausforderung. Es ist schwierig, eindeutige und allgemeingültige Anforderungen an die benötigten Domain-Experten zu formulieren. Ein sinnvoller Weg ist, eine geeignete Person im Unternehmen zu identifizieren, die zahlreiche weitere Mitarbeiter und deren Kompetenzen kennt und in Verbindung mit dem SoNBO-Wissen die geeigneten auswählen kann (wie es bei der Evaluation gemacht wurde). Wenn die benötigten Akteure im Unternehmen feststehen, ist eine weitere Herausforderung, bei der Einführung in das SoNBO-Konzept ein für die Akteure angemessenes fachliches Niveau zu treffen, um ein Verständnis über SoNBO zu erreichen. Dies sollte vor Projektbeginn durchgeführt (bei der Evaluation im Rahmen des 1. Meetings, vgl.

¹⁹ Die Namen der Beteiligten wurden aus Datenschutzgründen mit Kürzeln anonymisiert (Ausnahme: Dirk Gerlach).

Tab. 6.1) und dann in der benötigten Phase 2 wiederholt werden. Bei dieser Einführung von SoNBO sollten auch Vertreter des Managements anwesend sein, sodass kritische Fragen direkt geklärt werden können und dem SoNBO-Projekt keine unternehmenspolitischen Hürden gestellt werden. Dies ist zwar für das Framework sinnvoll, aber wird nicht als separate Phase in das Framework mitaufgenommen, weil sich das Framework auf wissenschaftliche Inhalte fokussiert. Falls das Framework in der Praxis zum Einsatz kommt, sollte dies unbedingt mitaufgenommen werden. Bei der Beurteilung, ob die Zielgruppe ein Verständnis zu SoNBO entwickelte, muss der SoNBO-Experte die Rückfragen analysieren. Bei dieser Evaluation beispielsweise wurden folgende Fragen gestellt:

Tab. 6.2: Rückfragen zu SoNBO im Rahmen der Evaluation

Frage von Mitarbeitern des KOSMOS Verlages	Analyse durch SoNBO-Experten
Wieso habt Ihr die Informationen aus dem CRM-System und aus dem ERP-System in zwei verschiedenen SoNBO-Explorern getrennt integriert?	Antwort: Da in der Universität das CRM-System ein Produktivsystem ist und das ERP-System zu Lehrzwecken genutzt wird, ergeben sich keine konzeptionell sinnvollen Szenarien, diese Informationen in einen SoNBO-Explorer zu integrieren. Analyse: Das Prinzip von SoNBO wurde verstanden.
Was mache ich, wenn ein weiteres Projekt XY (als Social Business Object) hinzukommt?	Antwort: Wenn der Projekttyp als Social (Sub)Concept bereits konfiguriert ist, wird dieses Social Business Object ebenfalls direkt geladen. Analyse: Das Prinzip von SoNBO (Zusammenhang der Ebenen Ontologie und Knowledge Graph) wurde noch nicht verstanden und sollte nochmal vertieft werden.
Was mache ich, wenn ein weiterer Projekttyp (als Social Concept) hinzukommt?	Antwort: Dieser Projekttyp kann nachkonfiguriert werden. Analyse: Das Prinzip von SoNBO wurde schon ansatzweise verstanden und ausgehend von dieser Frage können die Vorteile des Einsatzes einer Ontologie weiter besprochen werden.
Wie ist das mit sensiblen Daten? Darf die jeder sehen oder werden Berechtigungen mitübergeben?	Antwort: Die Berechtigungen der Benutzer für die Informationen in den Quellsystemen bleiben vorhanden. Analyse: Der Zugriff auf die Informationen in den Quellsystemen wurde verstanden.

Der Ablauf der drei Phasen in dem Framework erwies sich als sinnvoll. Bei der Evaluation konnten zwar die Grenzen der drei Phasen nicht eindeutig beibehalten werden, weil es an einigen Stellen zweckmäßig war, zwischen den Phasen zu springen. Allerdings dienten die Phasen dann wieder zur Orientierung. Somit sind die Phasen im Framework beizubehalten und müssen nicht geändert werden.

Die Ergebnisse innerhalb der Phasen werden in den folgenden Abschnitten vorgestellt (vgl. Abschnitt 6.3, 6.4, 6.5). Prinzipiell lässt sich festhalten, dass die Modelle, Methoden und Tools sinnvoll gewählt und passend für die Zielgruppe waren. Wichtig ist festzuhalten, dass der SoNBO-Experte unbedingt offen bzw. flexibel gegenüber Besonderheiten im Unternehmen sein sollte. Wie bereits erwähnt ergaben sich beim KOSMOS Verlag Besonderheiten sowohl auf der Lieferanten- als auch auf der Kunden-

seite, sodass vermeintlich feststehende Social Subconcepts wie z.B. Auftrag oder Rechnung keine selbstverständlichen Knoten einer Ontologie für den KOSMOS Verlag sind. Dies wird genauer in Abschnitt 6.4 ausgeführt. Außerdem muss der SoNBO-Experte auch flexibel auf die existierenden Dokumentationen über die IT oder die geschäftsrelevanten Informationen eingehen können. Der Umgang damit und die Einbindung davon in den Konfigurationsprozess kann nicht pauschal im Framework festgehalten werden.

Neben der Evaluation wurde auch indirekt das SoNBO-Konzept (vgl. Kapitel 4) evaluiert, indem die Definitionen zu Social Concept, Social Subconcept etc. vorgestellt und von den beteiligten Akteuren auch korrekt verwendet wurden. Es konnte also ein Verständnis von den Mitarbeitern beim KOSMOS Verlag entwickelt werden, was auch auf die Klarheit der Konzeptualisierung von SoNBO zurückzuführen ist.

Im Folgenden werden die Ergebnisse der Evaluation und die Erkenntnisse, die für die jeweilige Phase abgeleitet werden können, vorgestellt.

6.3 Phase 1: Status quo

Das Ziel in der Phase besteht in der Entwicklung des Verständnisses des SoNBO-Experten zu den geschäftsrelevanten Informationen im Unternehmen.

6.3.1 Ergebnisse

Der Status quo im Unternehmen KOSMOS Verlag wurde zum einen durch eine eXperience-Fallstudie formuliert (Gebel-Sauer & Grams, 2019). Außerdem wurden Abbildungen auf der Basis des geführten teilstandardisierten Leitfaden-Interviews erstellt.

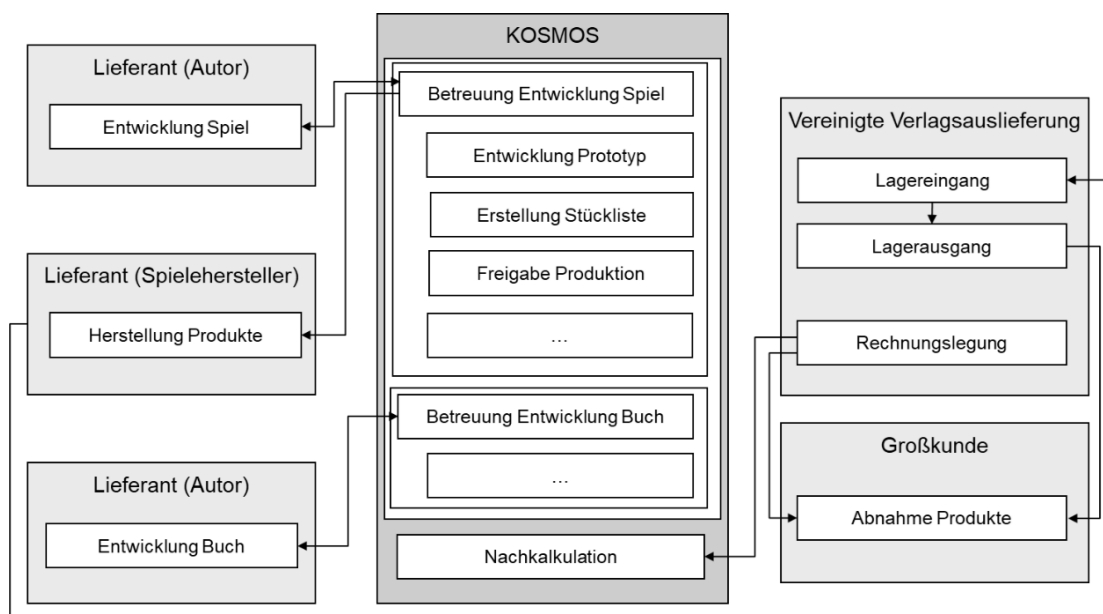


Abb. 6.1: Ergebnis Business Analyse

Für den Status quo der Business-Sicht (vgl. Abb. 6.1) wurden auf der Basis des Interviews die geschäftsrelevanten Informationen erfasst. Die beiden Produktbereiche für die Erwirtschaftung des Umsatzes sind Spiel und Buch. Im Bereich *Spiel* stellt KOSMOS Gesellschaftsspiele (wie z. B. Siedler von Catan) und Experimentierkästen her. Das Genre Experimentierkästen wurde von dem Verlag erfunden und KOSMOS ist in diesem Bereich Marktführer. Im Bereich *Buch* werden Kinder- und Jugendbücher sowie Ratgeber herausgegeben. Ein bekanntes Jugendbuch sind *Die drei ???*. Eine Besonderheit bei Unternehmen dieser Branche stellt der Vertrieb sowie die Lagerhaltung dar, da diese beiden Bereiche an ein anderes Unternehmen (VVA = Vereinigte Verlagsauslieferung) ausgelagert sind. Der KOSMOS Verlag entwickelt diese Produkte, gibt diese in Auftrag (Bücher werden in Druck gegeben, Spiele werden produziert) bei (in der Regel) asiatischen Herstellern und erhält lediglich eine kleine Auswahl zur Ansicht. Die restlichen Produkte werden in ein Lager in der Mitte Deutschlands geliefert, aus welchem sie dann auch an die Kunden (Spielwarenhandel/Buchläden) ausgeliefert werden. Die Kunden erhalten vom KOSMOS Verlag keine Angebote, Aufträge, Rechnungen etc., da dies ebenfalls durch die VVA abgewickelt wird. Die VVA erzeugt also auf der Basis der Preisliste vom KOSMOS Verlag selbst die Angebote und darauf aufbauend dann die Aufträge, Rechnungen und Mahnungen. Der KOSMOS Verlag erhält lediglich den jeweiligen Betrag monatlich überwiesen und jeden Tag eine Auswertung. In dieser Auswertung befinden sich auch Informationen über die Einzelaufträge, da die Autoren, die für den KOSMOS Verlag arbeiten, auf dieser Basis vergütet werden. Den Einkauf und somit die Lieferantenseite koordiniert der KOSMOS Verlag selbst. Der Verlag verschickt selbst Bestellungen und bucht die eingehenden Lieferscheine und Eingangsrechnungen. Die Lieferanten lassen sich in zwei Gruppen unterteilen: Bei vorrangig asiatischen *Herstellern* werden beispielsweise die fertig entwickelten Experimentierkästen bestellt. Diese bestellten Produkte (z. B. Experimentierkasten oder Material für die Entwicklung der Experimentierkästen) werden direkt vergütet. Die zweite Gruppe sind die *Autoren*. Bei diesen werden die Nutzungsrechte eingekauft, die dann absatzorientiert vergütet werden, weshalb der KOSMOS Verlag – wie oben erwähnt – die Einzelaufträge von der VVA erhält. Sie werden also erst vergütet, wenn das Produkt auch verkauft wurde. Von den Nutzungsrechten zu unterscheiden sind die Lizenzen, die eingekauft (z. B. für die Entwicklung eines Spieles zu Walking Dead) und verkauft (z. B. für die Entwicklung einer Handy-App zu Andor) werden.

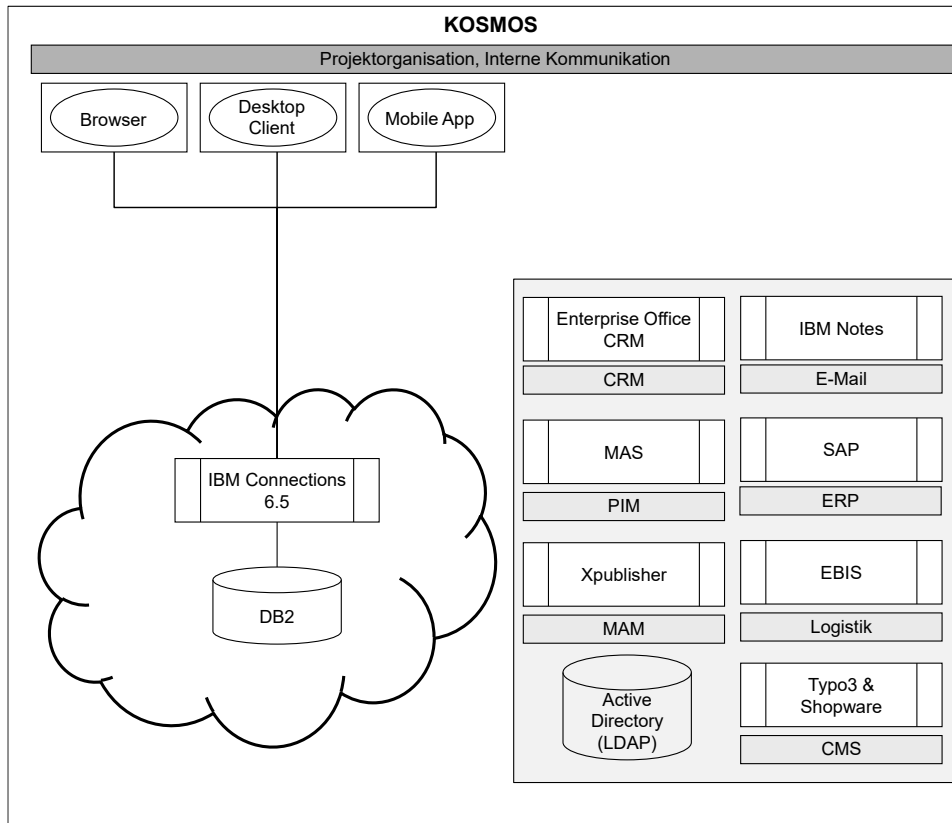


Abb. 6.2: Ergebnis Analyse IT-Landschaft

Für den Status quo bezüglich der technischen Sicht wurden die Ergebnisse ebenfalls aus dem Interview entnommen, das im Rahmen des 1. Meetings durchgeführt wurde (vgl. Tab. 6.1). Außerdem basieren diese auf der durch KOSMOS erstellten IT-Dokumentation. Auf dieser Basis wurde eine eigene Darstellung abgeleitet (vgl. Abb. 6.2). Die geschäftsrelevanten Informationen zu Produkten und Projekten befinden sich in HCL Domino/Notes, was auch für E-Mail genutzt wird. Auch die Informationen zu Lieferanten und Mitarbeitern liegen in HCL Domino/Notes. Die Belege werden im SAP-System verwaltet und werden durch die VVA täglich aktualisiert. Diese Systeme laufen auf den Servern im KOSMOS Verlag (on premises). Für die Kollaboration wird HCL Connections als Cloud-Lösung verwendet (off premises).

6.3.2 Erkenntnisse

Mit dem teilstandardisierten Leitfaden-Interview als offene Methode konnten die für den Status quo relevanten Informationen erhoben und die Eignung für das Framework konnte damit positiv getestet werden. Im Falle des KOSMOS Verlags konnte auf eine existierende IT-Dokumentation zurückgegriffen werden. Auf der Basis dieses Materials konnten mit der eXperience-Methode systematische Abbildungen entwickelt werden, die einen Überblick über den Status quo und die Besonderheiten beim KOSMOS Verlag wiedergeben. Mit den gewählten Akteuren (Domain-Experten: DGE, VPR) konnte die-

se Dokumentation zielführend erarbeitet werden. In dieser Phase kommt auf den SoNBO-Experten die Herausforderung zu, flexibel auf die existierenden Informationen (durch Interviews/Dokumentationen/...) zu reagieren. In dieser Phase fand eine Analyse der Ausgangssituation im Sinne der *Soft Systems Methodology* nach Checkland & Poulter (2010) statt, sodass auf dieser Basis Maßnahmen ergriffen werden können, um diese Ausgangssituation zu verbessern.

6.4 Phase 2: Ontologieentwicklung

Das Ziel in dieser Phase ist die Entwicklung der Ontologie *Social Network of Concepts*, um diese im SoNBO-Explorer zu konfigurieren. Dabei wurden folgende Dinge evaluiert: Zum einen wird das entwickelte Referenzmodell getestet und zum anderen die Methoden, um das Social Network of Concepts (basierend auf dem Referenzmodell) zu identifizieren. Die Ergebnisse werden mit dem Tool SoNBO-Visualiser vorgestellt.

6.4.1 Ergebnisse

Mit der Methode World-Café wurden mit den Domain-Experten DGE und VPR die Social Concepts des KOSMOS Verlages im ersten Meeting Ende August 2019 erhoben. Außerdem wurde die im Framework entwickelte Taxonomie den Domain-Experten in Form eines Fragebogens zur Verfügung gestellt, sodass sie die auf den KOSMOS Verlag zutreffenden Social Concepts auswählen konnten. Weiterhin war in dem Fragebogen vorgesehen, in einer Matrix eine Kante auszuwählen. Dies wurde bei der Evaluation allerdings nicht ausgeführt, da diese Aufgabenstellung an dieser Stelle nicht zielführend war.

Anhand dieser Datenbasis wurden die Social Concepts, Social Subconcepts und die Ontologie vom SoNBO-Experten abgeleitet, die beim nächsten Meeting Ende November 2019 nochmal gemeinsam überarbeitet wurden. Danach standen die Zieltaxonomie und die Zielontologie fest, wovon dann die Starttaxonomie und die Startontologie abgeleitet wurden (vgl. Abb. 6.3 bis Abb. 6.7). Die Ziel- und Startontologie wurden mit dem SoNBO-Visualiser dargestellt. Die Kanten waren an dieser Stelle noch nicht als final zu betrachten, da diese erst mit dem semantischen Mapping, also mit der Prüfung, inwiefern eine Kante durch die Attribute in den Quellsystemen definiert werden kann, final festgelegt werden konnte. Für die Entscheidung, welche Social Concepts in die Startontologie mitaufgenommen wurden, war auch die Business Analyse aus der vorherigen Phase ausschlaggebend. Dadurch fiel die Entscheidung des SoNBO-Experten auf die Social Concepts bzw. Social Subconcepts, die auch vom KOSMOS Verlag selbst gepflegt werden (also *Produkte*, *Projekte* und *Lieferanten*). Außerdem wurde das SSC *Mitarbeiter* für die Startontologie ausgewählt, weil der Benutzer (also der Mitarbeiter) auf seinem eigenen Knoten in das Social Network einsteigt und dieser somit immer ein wichtiger Bestandteil in einer SoNBO-Ontologie darstellt. Für diese Auswahl der Knoten in der Startontologie waren darüber hinaus die technischen Gegebenheiten relevant. Die ausgewählten SC und SSC befanden sich in HCL Notes. Da der SoNBO-Explorer ebenfalls eine Notes-Anwendung war und in den Konfigurationsprojekten bereits ein Notes-basiertes Quellsystem (GEDYS IntraWare 8) angeschlossen wurde, existierten bei der technischen Integration dieser Notes-basierten Systeme im KOSMOS Verlag für das SoN-

BO-Expertenteam weniger Hürden als bei einer Integration von externen Systemen wie z. B. die Belege aus dem SAP-System. Da der Fokus bei dem Evaluationsprojekt auf dem Framework lag und vorrangig keine technischen Aspekte hatte, wurden diese Notes-Anwendungen als Quellsysteme gewählt.

Social Concept	Social Subconcept	Social Concept	Social Subconcept
Kunden-Beleg	Angebot	Produkt	Ratgeber
	Auftrag		Kinder- und Jugendbuch
	Lieferschein		Brettspiel
	Rechnung		Experimentieren
	Mahnung		Basteln
Lieferanten-Beleg	Bedarfsanforderung		Detektiv
	Bestellung		Zaubern
	Lieferschein		Adventskalender
	Rechnung		Nutzungsrecht (Einkauf)
Person	Mitarbeiter		Vertrag
	Kundenkontakt	Lizenz (Verkauf)	
	Lieferantenkontakt	Kunde	
	Vertriebskontakt	Lieferant	
Projekt	Ratgeber	Institution	Partner
	Kinder- und Jugendbuch		Unternehmensstandort
	Brettspiel		Abteilung
	Experimentieren		Laptop
	Basteln	Inventar	Handy
	Detektiv		Stundenkonto
	Zaubern		Urlaubskonto
	Adventskalender		Urlaubsantrag
		Arbeitszeit-org.	

Legende:

grüner Hintergrund Social Concepts/Subconcepts für die Startontologie

weißer Hintergrund Ergänzung der grünen Social Concepts/Subconcepts zur Zielontologie

Abb. 6.3: KOSMOS-Start- und Zieltaxonomie (als Social Concepts und Social Subconcepts)

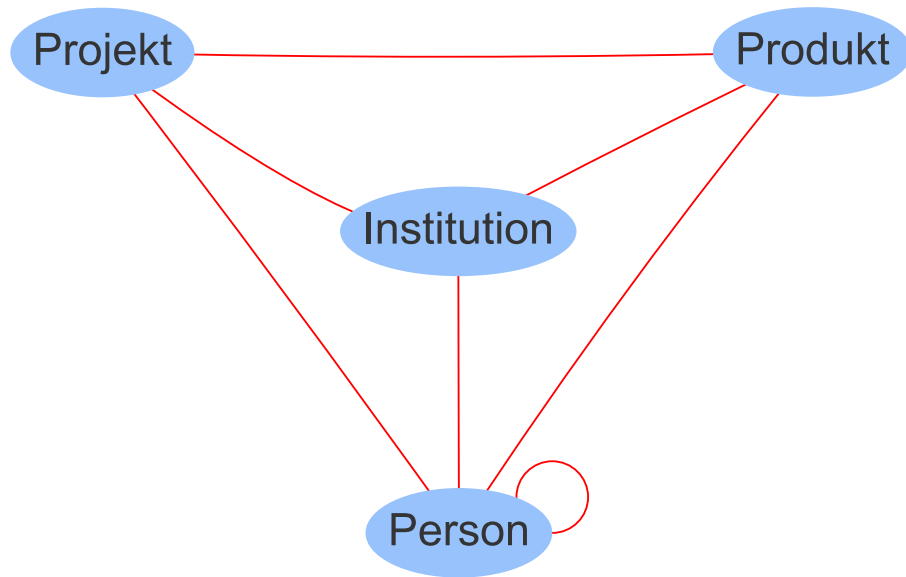


Abb. 6.4: KOSMOS-Startontologie (Social Network of Concepts)

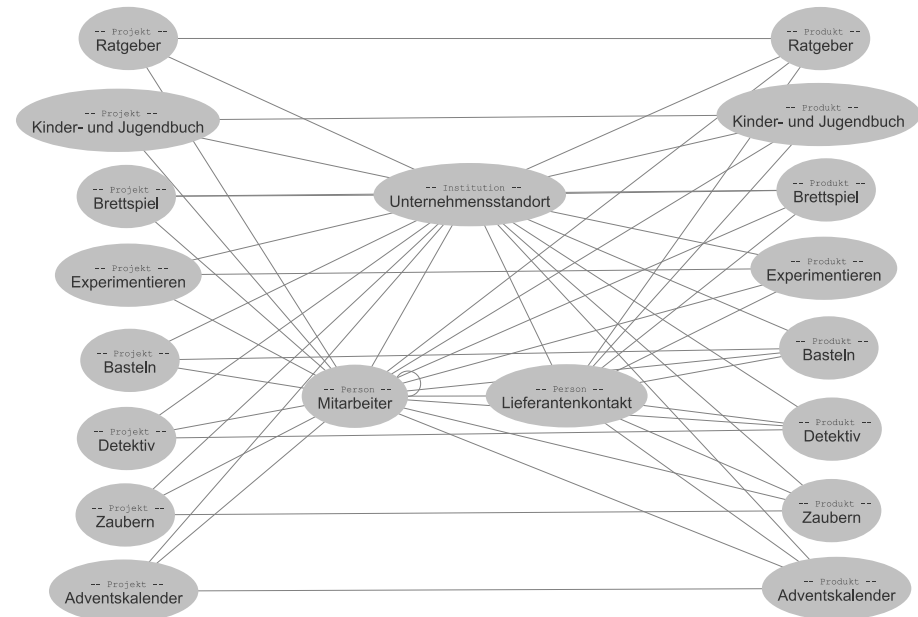


Abb. 6.5: KOSMOS-Startontologie (Social Network of Subconcepts)

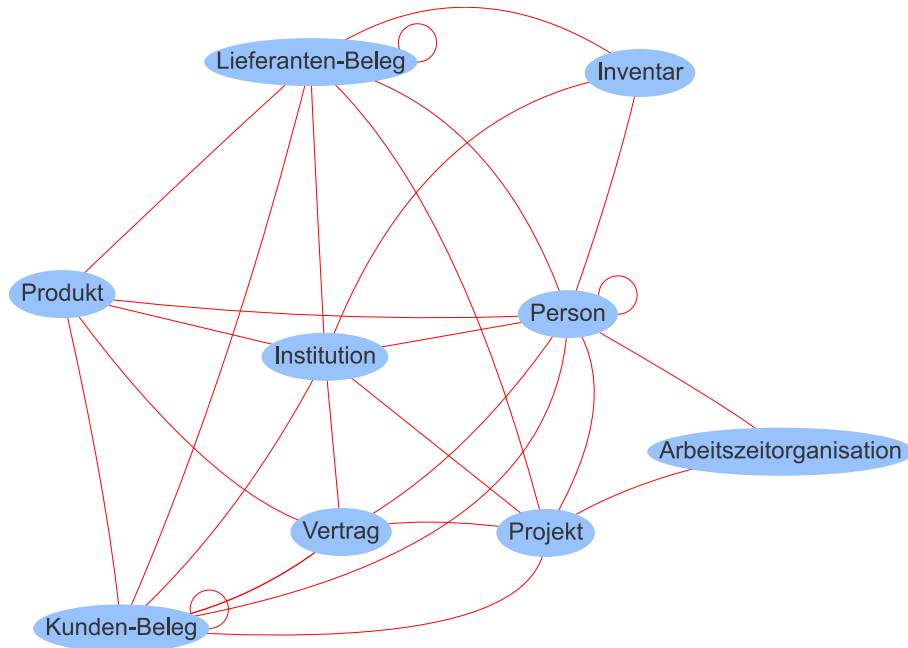


Abb. 6.6: KOSMOS-Zielontologie (Social Network of Concepts)

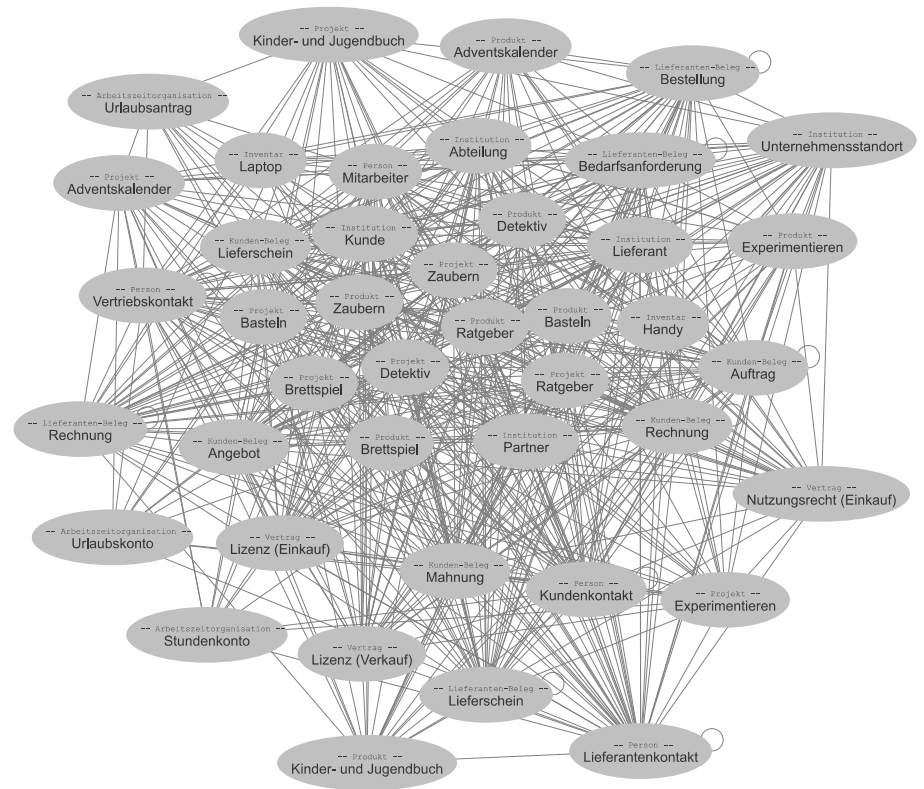


Abb. 6.7: KOSMOS-Zielontologie (Social Network of Subconcepts)

Neben der Visualisierung der Ontologie als Gesprächsgrundlage zwischen dem SoNBO-Experten und den Domain-Experten wurden Mockups über das zu erwartende Benutzerfrontend erstellt.

The screenshot displays the SoNBO-Explorer user interface for KOSMOS Verlag. At the top left, there is a 'Pfad' section with a 'Zurück zu mir' button. The main header area shows the profile of Dirk Gerlach, a 'Mitarbeiter' (Employee) at KOSMOS Verlag, with contact information and address. Below this is a 'Social Concept' graph showing relationships between various entities like 'Person', 'Institution', 'Produkt', and 'Projekt'. A sidebar on the left contains a 'Filtern' (Filter) section with options to filter by 'Enthält' (Contains) or 'Enthält nicht' (Does not contain), and a list of attribute values like 'Kinder', 'Familie', 'Erwachsene', and 'Kenner'. The main content area is a grid of business objects, each represented by a card with an icon, title, and details. The objects include 'EXIT - Mysteriöse Museum', 'VPR', 'Siedler von Catan', 'Andor', 'Fleischfressende Pflanze', 'Feen-Garten', 'KOSMOS Verlag', 'Heike Landmann', 'Laura Bender', 'Maria Busch', 'Ingo Busch', and 'Jutta Seidel'. A right sidebar shows a 'Business Object' search bar and a list of recent activity logs with timestamps and descriptions.

Abb. 6.8: Benutzerfrontend SoNBO-Explorer für den KOSMOS Verlag als Mockup

Diese Mockups waren das zweite Ergebnis dieser Phase. Ein Mockup zu dem Mitarbeiter Dirk Gerlach ist in Abb. 6.8 abgebildet.

6.4.2 Erkenntnisse

Das Vorgehen bei der Ontologieentwicklung, bei welcher eine Ontologie als Referenz nach dem Vorbild von Uschold & Gruninger (1996) verwendet wurde (vgl. Abschnitt 3.2.1.2.3), die auf den KOSMOS Verlag angepasst wurde, erwies sich als zielführend. Weiterhin wurde evaluiert, dass das wissenschaftlich fundierte Referenzmodell in Form der vorgeschlagenen Ontologie *Social Network of Concepts/Subconcepts* (vgl. Abb. 5.5 und Abb. 5.6) aus dem SoNBO-Framework auch für zukünftige Projekte als Ausgangsbasis verwendet werden kann. Für weitere Projekte ist allerdings auch denkbar, eine

putative Ontologie mithilfe eines Bootstraps aus dem Quelldatenbankschema zu erstellen (vgl. Abschnitt 3.2.1.2.3) (Sequeda, 2017). Diese kann das Referenzmodell um einen weiteren Vorschlag unterstützen, welcher dann Auswirkung auf die Ontologieentwicklung hat. Bei der Anwendung einer Ontologie als Referenzmodell muss dem SoNBO-Experten bewusst sein, dass es sich lediglich um einen Vorschlag handelt und es nicht eine Ontologie für jedes Unternehmen geben kann. Dies wurde auch beim KOSMOS Verlag deutlich durch die Besonderheit auf der Kundenseite. Da die Auslieferung von Produkten durch die VVA abgewickelt wird und der KOSMOS Verlag beispielsweise nicht selbst Angebote und Aufträge erstellt und die Lagerverwaltung organisiert, haben diese Social Concepts keine Priorität und werden somit zunächst nicht berücksichtigt. Allerdings kann das im Framework entwickelte Referenzmodell beibehalten werden mit dem Wissen, es tatsächlich als Vorschlag zu nutzen. Durch die Erfahrung des Evaluationsprojektes kann die Ontologie zukünftig auch in die drei Ebenen nach Guarino (1997) in Top-Level-, Domänen- und Anwendungsontologie aufgeteilt werden (vgl. Abschnitt 3.2.1.2.2). Eine Top-Level-Ontologie würde dann aus Konzepten bestehen, die für jedes Unternehmen gelten (z. B. Mitarbeiter). Eine Domänenontologie wäre eine Ontologie für die Verlagsbranche und die Anwendungsontologie wäre die Ontologie für das konkrete Unternehmen KOSMOS Verlag. Mit mehr Erfahrung in SoNBO-Projekten können mithilfe dieser Einteilung auf konkrete Branchen zielgruppenorientiertere Ontologien als Referenzmodelle entwickelt werden. Außerdem stellte es sich als sinnvoll heraus, die Erkenntnisse aus der erstellten Ontologie in ein Mockup für das Unternehmen zu übertragen. Dies war vorher nicht vorgesehen.

Auch die Methode durch die Einbindung der beiden Sichten (technische und Business Sicht) stellte sich als zielführend heraus, da damit die Ontologie bezogen auf den KOSMOS Verlag entwickelt werden konnte. Allerdings konnten einige Verbesserungsvorschläge identifiziert werden. Bei dem World-Café kann der Begriff *Nutzungsszenarien* auch in *Welche Fragen* umbenannt werden (vgl. Abb. 5.7 auf S. 144). Dies ist für den Mitarbeiter ggf. einfacher zu verstehen und orientiert ihn direkt. Diese Frage ist aber unbedingt beizubehalten, da darüber der SoNBO-Experte auch indirekt Social Concepts ableiten kann. Außerdem ist die Antwort auf diese Frage für die Erstellung der zielgruppengerechten Mockups hilfreich. Die Anwesenheit des SoNBO-Experten beim World-Café ist auch von großer Bedeutung, da dieser bei den Gesprächen über die Social Concepts teilhaben kann und somit für die notwendigen Fokussierungen der Mitarbeiter auf Social Concepts sorgen kann. Dies hilft ihm oder ihr bei der Ableitung der Startontologie aus der Zielontologie. Der Fragebogen wurde an die Domain-Experten im Anschluss an das World-Café am Ende des 1. Meetings im August 2019 ausgegeben. Für diesen Zeitpunkt war der Fragebogen zu lang und die Matrix (Erstellung der Kanten zwischen den Social Concepts) zu komplex. Da die Social Concepts sowieso überarbeitet wurden, war die Frage nach den Kanten zu diesem Zeitpunkt auch zu früh. Diese wurden dann nicht mehr über den Fragebogen erfragt, da der SoNBO-Experte durch die Erstellung der KOSMOS-Ontologie diese selbst als Vorschlag erstellte, welche dann diskutiert wurden. Die finalen Kanten werden dann im semantischen Mapping in Phase 3 erstellt. Es wird also empfohlen, die Frage nach den Kanten (Matrix) in Zukunft wegzulassen. Ein weiterer Nachteil war, dass durch das World-Café einige Social Concepts redundant besprochen wurden. Idealerweise hätte dieser Fragebogen mit den Ergebnissen aus dem World-Café angepasst werden müssen,

sodass die Mitarbeiter direkt auf das bereits erarbeitete Wissen aus dem World-Café hätten aufbauen können und wären auf diese Weise nicht mit redundanten Informationen überfrachtet worden. Nichtsdestotrotz konnten aus dem Fragebogen wertvolle Informationen für die Ontologie gezogen werden und bei dieser Erkenntnis handelt es sich lediglich um eine Kleinigkeit, die in einem künftigen Projekt falls möglich geändert werden sollte (wenn es z. B. für den SoNBO-Experten ein Zeitfenster zwischen dem World-Café und dem Fragebogen gibt). Außerdem sollte ein flexibler Einsatz des Fragebogens und des World-Cafés durch den SoNBO-Experten ermöglicht werden. Es könnte sich auch als sinnvoll erachten, zuerst den Fragebogen einzusetzen und im Anschluss das World-Café zu platzieren. Dies muss in Abhängigkeit der Mitarbeiter und der entstandenen Dynamik entschieden werden. Wichtig ist aber, beides einzusetzen, da mit dem World-Café die Social Concepts erfasst werden, die in dem Referenzmodell nicht antizipiert wurden, und mit dem Fragebogen die Mitarbeiter auch die Social Concepts auswählen können, die diesen nicht bewusst sind. Der SoNBO-Visualiser diente als Tool zur Visualisierung der KOSMOS-Ontologie und war hilfreich als Gesprächsgrundlage. In zukünftigen Projekten sollte dieses wieder eingesetzt werden. Zur Unterstützung der Konfiguration wurde das Tool nicht verwendet, da die Anwendbarkeit des SoNBO-Mappers dies überflüssig machte.

6.5 Phase 3: Mapping

Das Ziel in dieser Phase ist die Durchführung des semantischen und des technischen Mappings von den Informationen in der Ontologie auf die Informationen in den Quellsystemen. Dies erfolgt mithilfe des SoNBO-Mappers, welches auch Kern der Evaluation in dieser Phase ist. In diesem Excel-Dokument wird mit dem IT-Experten zunächst das semantische Mapping durchgeführt, indem für jedes Social Subconcept die notwendigen Attribute aus den Quellsystemen herausgesucht werden. Außerdem erfolgt das technische Mapping, indem die Datenbankabfragen auf diese Attribute formuliert werden. Die Erhebung dieser Informationen erfolgt im SoNBO-Mapper in einer anderen Struktur als die Eingabe dieser Informationen in den SoNBO-Explorer. Daher erfolgt außerdem in diesem Excel-Dokument eine Übertragung dieser Informationen in die Struktur der Konfiguration im SoNBO-Explorer. Dies kann als weiteres Mapping betrachtet werden.

6.5.1 Ergebnisse

Das semantische Mapping für die Informationen aus der Startontologie auf die Informationen in den Quellsystemen wurde im zweiten Meeting im November 2019 mit dem IT-Experten erhoben. Dazu wurden die Attribute in den Quellsystemen identifiziert, die die Social Subconcepts weiter spezifizierten (vgl. Tab. 4.5 auf S. 106). Außerdem wurden erste Vorbereitungen für das technische Mapping durchgeführt, indem die konkreten Quellsysteme für die jeweiligen Social Subconcepts identifiziert wurden. Die Queries wurden im Nachgang zu diesem Meeting durch den SoNBO-Experten formuliert (= technisches Mapping), da es sich bei den Quellsystemen um Notes-Anwendungen handelte und daher die Queries aus vorhergehenden Projekten bekannt waren. Dieses Mapping wurde in einer Nachbereitung von dem SoNBO-Experten aufbereitet und an den IT-Experten zurückgespielt, sodass dieser

die Backendfeldnamen eintragen und daraufhin noch offene Fragen geklärt werden konnten. Anschließend wurde die Struktur der Attribute für den SoNBO-Explorer abgeleitet und die Konfigurationstabellen wurden für den Import in den SoNBO-Explorer erstellt. Dieses Vorgehen wird beispielhaft an dem Mapping für das Social Subconcept *Mitarbeiter* erläutert: In Abb. 6.9 ist ein Ausschnitt dieses Mappings für das Social Subconcept *Mitarbeiter* abgebildet. In der *Spalte B* wurden die Attribute aus den Quellsystemen aufgelistet und ab *Spalte E* wurde die Verwendung dieses Attributes für den SoNBO-Explorer festgelegt. Beispielsweise wurde in *Spalte N* festgelegt, welches der Attribute für die Erstellung der Beziehung verwendet wurde. In diesem Beispiel wurde das Feld *Manager* verwendet, um die Kante zwischen zwei Mitarbeitern zu erzeugen. Die Formulierung der Query, um diese Kanten zu erzeugen, wurden in einem weiteren Tab des SoNBO-Mappers (vgl. Abb. 6.10) formuliert. Dazu wurden paarweise die Social Subconcepts gelistet, zwischen denen eine Kante existieren sollte (*Spalte B und C*). Die Formulierung der Query erfolgte in *Spalte J*. Für die Kante *Mitarbeiter – Mitarbeiter* erfolgte dies in *Zeile 2 und 3*. Dies war schon Teil des technischen Mappings. Abschließend wurde dieses erhobene Mapping in dem SoNBO-Mapper auf die Struktur der Konfigurationstabelle des SoNBO-Explorer abgebildet. Dies ist beispielhaft in Abb. 6.11 zu sehen. In Abb. 6.12 ist ein Screenshot des Administrationsfrontends zu sehen, in dem sich die Konfiguration aus dem SoNBO-Mapper nach dem Import befindet. Dieses Vorgehen erfolgte analog für die anderen Social Subconcepts. Abb. 6.13 zeigt einen Screenshot des Benutzerfrontends (Beispiel: SSC Mitarbeiter *Dirk Gerlach*) des für KOSMOS konfigurierten SoNBO-Explorers. In diesem Screenshot ist im Ergebnisbereich eine Mitarbeiterin zu sehen, die durch einen dunkelgrauen Balken aus Gründen des Datenschutzes anonymisiert wurde. Dieser Knoten erscheint dort, weil sie die Vorgesetzte von Dirk Gerlach ist und somit eine Kante zwischen diesen beiden Knoten (Dirk Gerlach und *Mitarbeiterin*) existiert. Diese kam durch die formulierte Query in Abb. 6.10 zu Stande. Abb. 6.14 zeigt einen weiteren Screenshot (SSC Projekt *Glibber-Organe*).

6 Evaluation des SoNBO-Frameworks

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
25																		
26																		
27																		
28	Auswirkung auf Profilbereich	-	-	-	-	2	1	1	1	1	4	5	3	-	6			
29	Überschrift	Feldname (Frontend)	Feldname (Backend)	Beispielwert	ID	Zuordnung zu SSC	Name des Social Subconcepts	Attribute	SoNBO-Attribut (String)	Bild (String: "x")	Attribute als Kurzinfo (": "x")	filterbare Attribute (": "x")	sortierbare Attribute (": "x")	Beziehung (": "x")	Statusänderung (": "x")	JSON Key	JSON ID	Hinweis für JSON
30	Eingabewert	-	-	-	[* ; "x"]	[* ; "x"]	[* ; "x"]	[* ; "x"]	[* ; "x"]	[* ; "x"]	[* ; "x"]	[* ; "x"]	[* ; "x"]	[* ; "x"]	[* ; "x"]	-	-	
31	Anzahl der Eingabewerte	-	-	-	[1]	[1]	[n]	[n]	[n]	[1]	[n]	[n]	[n]	[n]	[n]	-	-	
32	Attribut 1	Vorname	Firstname	Boris			x				x					Vorname		
33	Attribut 2	Nachname	Lastname	Saltzmann			x				x		x			Nachname		
34	Attribut 3	E-Mail Adresse	InternetAddress	boris.saltzmann@kosmos.de	x			x	Symbol							E-Mail Adresse	x	
35	Attribut 4	Telefonnummer	OfficePhoneNumber	49 711 2191 295				x	Telefon		x					Telefon		
36	Attribut 5	Handynummer	CellPhoneNumber	-				x	Mobil							Mobil		
37	Attribut 6	Manager	Manager	LandmannKosmos								x		x (Mitarbeiter)		Manager		"@Name"
38	Attribut 7	Titel	JobTitle	Entwickler				x	-		x				x	Titel		
39	Attribut 8	Abteilung	Department	Entwicklung				x	Abteilung							Abteilung		
40	Attribut 9	Firma	CompanyName	Frank-Kosmos Verlage-GmbH & Co. KG				x	-			x		x (Unternehmensstandort)		Firma		
41	Attribut 10	Standort	OfficeCity	Stuttgart				x	-		x		x		x	Standort		
42	Attribut 11	-	Form	Person		x										Form		
43	Attribut 12	-	FullName	"CN=Boris Saltzmann@kosmos.de,OU=Entwickler,OU=Stolpmann"												-		
44	Attribut 13	-	-	Boris Saltzmann												Name		
45	Attribut 14	Foto-URL	PhotoURL	http://photo.do?keu=aab8b87c-54f8-475e-a77f-a3d658876857												Photo		
46	Hinweis	Bild	-	-						URL aus Connections								
47																		
48																		
49	Legende	Eingefügt aus dem Quellsystem Festlegung der Verwendung des Attributes als ID Festlegung der Verwendung des Attributes als Zuordnung zum SSC (= Social Subconcept) Festlegung der Verwendung des Attributes zur Visualisierung im Frontend Festlegung der Verwendung des Attributes zur Filterung/Sortierung Festlegung der Verwendung des Attributes zur Erstellung der Beziehung/Kante zwischen zwei Social Socialsubconcepts Festlegung der Verwendung der Änderung des Attributes zur Statusmitteilung [1] genau ein Kreuz [n] beliebig viele Kreuze																

Abb. 6.9: Ausschnitt aus dem SoNBO-Mapper für KOSMOS: Konfiguration zu SSC Mitarbeiter (SC: Person)

	A	B	C	D	E	F	G	H	I	J	K	L	M
	Kantennr.	[1] Source: Social Subconcept	[2] Target: Social Subconcept	Feldname Frontend [2]	Feldname Backend [2]	Attribut (SoNBO-Attribut)	Quellsystem [2]	Datenbank [2]	View [2]	Query	QueryCommand	Kommentar	Query
1	1	Mitarbeiter	Mitarbeiter	Manager	Manager	Name	Domino Directory	names.nsf	SoNBOEmployee	FIELD Manager CONTAINS "Name"	FTSearch		PersonEm_PersonEm_1
2		Mitarbeiter	Mitarbeiter	-	FullName	Manager	Domino Directory	names.nsf	SoNBOEmployee	FIELD FullName CONTAINS "Manager"	FTSearch		PersonEm_PersonEm_2
3	2	Mitarbeiter	Lieferantenkontakt	Zuständige	AddressOwner	Name	EO Einkauf	pavone/eo_einkauf.nsf	SoNBOSupplier	FIELD AddressOwner CONTAINS "Name"	FTSearch	hier: Lieferanten = Autoren	PersonEm_PersonSu
4		Lieferantenkontakt	Mitarbeiter	-	FullName	Zuständige	Domino Directory	names.nsf	SoNBOEmployee	FIELD FullName CONTAINS "Zuständige"	FTSearch		PersonSu_PersonEm
5	3	Unternehmensstandort	Mitarbeiter	Firma	CompanyName	Kosmos Verlags-GmbH	Domino Directory	names.nsf	SoNBOEmployee	FIELD CompanyName CONTAINS "Francckh-Kosmos Verlags-GmbH & Co. KG"	FTSearch		Institution_PersonEm
6		Mitarbeiter	Unternehmensstandort	-	CompanyName	Kosmos Verlags-GmbH	selbst geschrieben von Simon	Institution.nsf	SoNBOLocation	FIELD CompanyName CONTAINS "Francckh-Kosmos Verlags-GmbH & Co. KG"	FTSearch		PersonEm_Institution
7	4	Mitarbeiter	Ratgeber(Proj.)	Project leader	MainProjectLeader	Name	Genius	geniusgpEn_portfolio.nsf	SoNBOProject	FIELD MainProjectLeader CONTAINS "Name"	FTSearch		PersonEm_Project
8		Ratgeber(Proj.)	Mitarbeiter	-	FullName	Projektleiter	Domino Directory	names.nsf	SoNBOEmployee	FIELD FullName CONTAINS "Projektleiter"	FTSearch		Project_PersonEm
9	5	Mitarbeiter	Kinder- und Jugendbuch(Proj.)	Project leader	MainProjectLeader	Name	Genius	geniusgpEn_portfolio.nsf	SoNBOProject	FIELD MainProjectLeader CONTAINS "Name"	FTSearch		PersonEm_Project
10		Kinder- und Jugendbuch(Proj.)	Mitarbeiter	-	FullName	Projektleiter	Domino Directory	names.nsf	SoNBOEmployee	FIELD FullName CONTAINS "Projektleiter"	FTSearch		Project_PersonEm
11	6	Mitarbeiter	Brettspiel(Proj.)	Project leader	MainProjectLeader	Name	Genius	geniusgpEn_portfolio.nsf	SoNBOProject	FIELD MainProjectLeader CONTAINS "Name"	FTSearch		PersonEm_Project
12		Brettspiel(Proj.)	Mitarbeiter	-	FullName	Projektleiter	Domino Directory	names.nsf	SoNBOEmployee	FIELD FullName CONTAINS "Projektleiter"	FTSearch		Project_PersonEm
13										FIELD			

Abb. 6.10: Ausschnitt aus dem SoNBO-Mapper für KOSMOS: Konfiguration der Kanten zwischen den SSC

6 Evaluation des SoNBO-Frameworks

A	B	C	D	E	F	G	H	I	J	K	L	M
Attribut: A. aus "1.2 Social Subconcepts" (SoNBO)	selbst benennen	leer lassen (wird automatisch erzeugt)	leer lassen (wird automatisch erzeugt)	Datasource: DS aus "2.1 Datasources"	Query: Q. aus "2.2 Queries"	Attribut: Key aus JSON	aus Auswahl wählen	selbst benennen	selbst benennen	selbst benennen	leer lassen (wird automatisch erzeugt)	leer lassen (wird automatisch erzeugt)
attributeNodeType	attributeName	attributeNameDE	attributeNameEN	attributeDataSource	attributeQuery	attributeFieldName	attributeDatatyp	attributeDisplayName	attributeFilterable	attributePreview	attributeIcon	attributeDoc
Mitarbeiter	Name			PersonEmployee	getPersonEmployee	Name	String		0 false	false		
Mitarbeiter	Vorname			PersonEmployee	getPersonEmployee	Vorname	String		0 false	false		
Mitarbeiter	Nachname			PersonEmployee	getPersonEmployee	Nachname	String		0 false	false		
Mitarbeiter	E-Mail Adresse			PersonEmployee	getPersonEmployee	E-Mail Adresse	String		1 false	false		
Mitarbeiter	Telefon			PersonEmployee	getPersonEmployee	Telefon	String		3 false	true		
Mitarbeiter	Mobil			PersonEmployee	getPersonEmployee	Mobil	String		3 false	false		
Mitarbeiter	Manager			PersonEmployee	getPersonEmployee	Manager	String		2 true	false		
Mitarbeiter	Titel			PersonEmployee	getPersonEmployee	Titel	String		2 false	true		
Mitarbeiter	Abteilung			PersonEmployee	getPersonEmployee	Abteilung	String		2 true	true		
Mitarbeiter	Firma			PersonEmployee	getPersonEmployee	Firma	String		4 true	false		
Mitarbeiter	Standort			PersonEmployee	getPersonEmployee	Standort	String		4 false	true		
Mitarbeiter	Photo			PersonEmployee	getPersonEmployee	Photo	String		0 false	true		
Lieferantenkontakt	Name			PersonSupplier	getPersonSupplier	Name	String		0 false	false		
Lieferantenkontakt	Lieferantennummer			PersonSupplier	getPersonSupplier	SAP Lieferantennummer	String		1 false	false		
Lieferantenkontakt	Vorname			PersonSupplier	getPersonSupplier	Vorname	String		0 false	false		
Lieferantenkontakt	Nachname			PersonSupplier	getPersonSupplier	Nachname	String		0 false	false		
Lieferantenkontakt	E-Mail Adresse			PersonSupplier	getPersonSupplier	E-Mail Adresse	String		1 false	false		
Lieferantenkontakt	Telefon			PersonSupplier	getPersonSupplier	Telefonnummer	String		3 false	true		
Lieferantenkontakt	Mobil			PersonSupplier	getPersonSupplier	Handynummer	String		3 false	false		
Lieferantenkontakt	Homepage			PersonSupplier	getPersonSupplier	Homepage	String		2 false	false		
Lieferantenkontakt	Klasse			PersonSupplier	getPersonSupplier	Klasse	String		4 true	true		
Lieferantenkontakt	Zuständige			PersonSupplier	getPersonSupplier	Zuständige	String		4 true	false		
Lieferantenkontakt	Firma			PersonSupplier	getPersonSupplier	Firma	String		2 false	true		
Lieferantenkontakt	Land			PersonSupplier	getPersonSupplier	Land	String		2 false	false		
Lieferantenkontakt	Ort			PersonSupplier	getPersonSupplier	Ort	String		2 false	false		
Lieferantenkontakt	Titel			PersonSupplier	getPersonSupplier	Titel	String		1 false	false		
Lieferantenkontakt	ID			PersonSupplier	getPersonSupplier	ID	String		0 false	false		
Ratgeber (Proj.)	Material number			Project	getProject	Material number	String		1 false	true		
Ratgeber (Proj.)	Project code			Project	getProject	Project code	String		1 false	true		
Ratgeber (Proj.)	Market			Project	getProject	Market	String		0 false	false		
Ratgeber (Proj.)	Segment			Project	getProject	Segment	String		1 true	false		
Ratgeber (Proj.)	Project description			Project	getProject	Project description	String		0 false	false		
Ratgeber (Proj.)	Projektleiter			Project	getProject	Project leader	String		2 true	false		
Ratgeber (Proj.)	Projektstart			Project	getProject	Project start	String		4 false	false		
Ratgeber (Proj.)	Projekttende			Project	getProject	Project end	String		4 false	true		
Ratgeber (Proj.)	Fortschritt			Project	getProject	Progress	String		2 false	false		
Ratgeber (Proj.)	Type			Project	getProject	Type	String		1 false	false		
Ratgeber (Proj.)	Location			Project	getProject	Location	String		3 false	false		
Ratgeber (Proj.)	Zielpreis			Project	getProject	Target price	String		4 false	false		
Ratgeber (Proj.)	Project team			Project	getProject	Project team	String		0 false	false		
Kinder- und Jugendbuch (Proj.)	Material number			Project	getProject	Material number	String		0 false	false		
Kinder- und Jugendbuch (Proj.)	Project code			Project	getProject	Project code	String		0 false	false		
Kinder- und Jugendbuch (Proj.)	Market			Project	getProject	Market	String		0 false	false		
Kinder- und Jugendbuch (Proj.)	Segment			Project	getProject	Segment	String		1 true	false		
Kinder- und Jugendbuch (Proj.)	Project description			Project	getProject	Project description	String		0 false	true		
Kinder- und Jugendbuch (Proj.)	Projektleiter			Project	getProject	Project leader	String		2 true	true		
Kinder- und Jugendbuch (Proj.)	Projektstart			Project	getProject	Project start	String		2 true	false		
Kinder- und Jugendbuch (Proj.)	Projekttende			Project	getProject	Project end	String		2 true	true		

Abb. 6.11: Ausschnitt aus dem SoNBO-Mapper für KOSMOS: Übertragung der Konfiguration auf die Struktur im SoNBO-Explorer

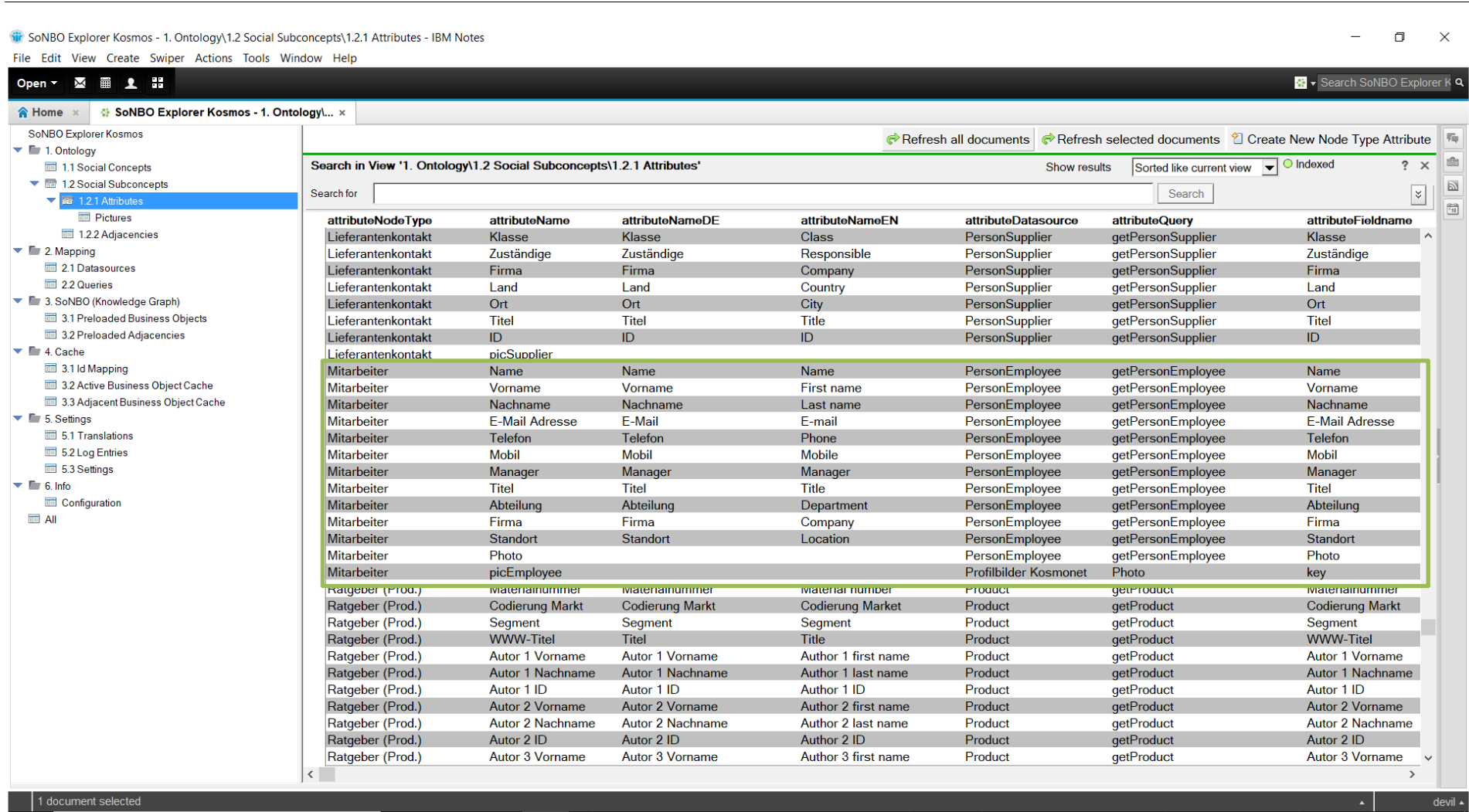


Abb. 6.12: Screenshot des Administrationsfrontends vom SoNBO-Explorer für KOSMOS

Pfad

Dirk Gerlach

↑

Zurück zu mir

Filtern

Filter hinzufügen:

Enthält

Enthält nicht

Attributschlüssel

Abteilung ▼

Attributwerte

Buchhaltung

IT

Hinzufügen

Aktive Filter:

Social Business Object

Dirk Gerlach

Mitarbeiter

E-Mail: [redacted]

Manager: [redacted]

Titel: Leiter IT & CDO

Abteilung: IT

Telefon: +49 [redacted]

Mobil: +49 15 [redacted]

Firma: Franckh-Kosmos Verlags-GmbH & Co. KG

Standort: Stuttgart

Social Concept

3 all 1 Institution 2 Person

Social Subconcept

3 all 2 Mitarbeiter 1 Unternehmensstandort

Sortieren Titel ▼ Aufsteigend Absteigend

Social Business Objects

Franckh-Kosmos Verlags-GmbH & Co. KG

Unternehmensstandort

KOSMOS Stadt: Stuttgart

E-Mail: info@kosmos.de

[redacted]

Mitarbeiter

Telefon: +49 711 [redacted]

Abteilung: IT

Standort: Stuttgart

[redacted]

Mitarbeiter

Telefon: +49 711 [redacted]

Abteilung: Buchhaltung

Standort: Stuttgart

Dunkelgrauer Balken: Anonymisiert aus Gründen des Datenschutzes

Abb. 6.13: Benutzerfrontend SoNBO-Explorer für den KOSMOS Verlag als Screenshot (Knoten: *Mitarbeiter* Dirk Gerlach)

Pfad
Glibber-Organ
Zurück zu mir

Social Business Object
Glibber-Organ
Experimentieren (Proj.)

Projektcode: 2019-009
Segment: Science
Format: Cases
Type: Science - Novi

Location: DACH
Unternehmensstandort: Franckh-Kosmos Verlags GmbH & Co. KG - Stuttgart

Projektleiter: [redacted]
Projektgesundheit: 0
Fortschritt: 0
Projektphase: Initiation

Projektstart: 10/30/2019
Projektende: 05/21/2020 12:00:00 AM

Social Concept
2 all 1 Institution 1 Person

Social Subconcept
2 all 1 Mitarbeiter 1 Unternehmensstandort

Sortieren Titel Aufsteigend Absteigend

Social Business Objects

Franckh-Kosmos Verlags-GmbH & Co. KG
Unternehmensstandort
KOSMOS Stadt: Stuttgart
E-Mail: info@kosmos.de

Mitarbeiter
Abteilung: Entwicklung
Standort: Stuttgart

Filtern
Filter hinzufügen:
 Enthält
 Enthält nicht

Attributschlüssel
Abteilung

Attributwerte
Entwicklung

Hinzufügen

Aktive Filter:

Dunkelgrauer Balken: Anonymisiert aus Gründen des Datenschutzes

Abb. 6.14: Benutzerfrontend SoNBO-Explorer für den KOSMOS Verlag als Screenshot (Knoten: *Projekt Glibber-Organ*)

Der Import der Konfigurationstabellen in den SoNBO-Explorer (also von Abb. 6.11 nach Abb. 6.12) wurde im dritten Meeting durchgeführt.

Tab. 6.3: Ablauf Phase 3 (3. und 4. Meeting)

	Startkonfiguration	Startontologie	Installation bei KOSMOS
Zeitraum	11.03.2020 – 19.03.2020	20.03.2020 – 24.03.2020	24.08.2020 – 01.10.2020
Server	Devil (FG BAS) (Domino 9.01)	Devil (FG BAS) (Domino 9.01)	Ikarus (KOSMOS) (Domino 9.01)
Schritte	<ul style="list-style-type: none"> - Erstellung JSON-Strings in Quellsystemen - Import der Startkonfiguration (Office 2003) - Testen der Startkonfiguration 	<ul style="list-style-type: none"> - Erstellung der restlichen JSON-Strings in Quellsystemen - Import der Startontologie (Office 2003) - Testen der Startontologie 	<ul style="list-style-type: none"> - Erstellung JSON-Strings in Quellsystemen auf Ikarus - Anpassungen der Spalte „datasourceDatabase“ in der View „2.1 Datasources“ im SoNBO-Explorer - Kopieren der Institution.nsf auf Ikarus - Kopieren des SoNBO-Explorers mit der konfigurierten Startontologie auf Ikarus
Verwendeter Teil der Ontologie	Person (Mitarbeiter); Produkt (Experimentieren)	Startontologie	Startontologie
Zeitaufwand	Ca. 100 h	Ca. 30 h	Ca. 20 h

Dieses Meeting fand nicht physisch, sondern virtuell in mehreren Web-Calls statt (vgl. Tab. 6.3). Dazu wurde die Konfiguration des SoNBO-Explorers mit der KOSMOS-Ontologie auf dem Server (Devil) der Universität durchgeführt, sodass dieser konfigurierte SoNBO-Explorer dann lediglich auf den Server im KOSMOS Verlag kopiert werden musste. Damit die Konfiguration getestet werden konnte, wurden vom KOSMOS Verlag die anzubindenden Datenbanken mit einigen Beispieldatensätzen zur Verfügung gestellt. Parallel dazu traf der IT-Experte DGE vom KOSMOS Verlag Vorbereitungen auf dem Server im KOSMOS Verlag: Bis auf eine der anzubindenden Datenbanken (Projektdatenbank) lagen alle benötigten Quellsysteme auf einem Domino 9.01 Server (Ikarus), auf welchem auch der SoNBO-Explorer installiert werden sollte. Da es einfacher war, wenn alle anzubindenden Quellsysteme auf einem Server liegen, wurde von der Projektdatenbank, die auf einem Domino 10.01 Server (Diomedes) lag, eine Replik auf dem Ikarus erstellt. Außerdem wurden die von dem SoNBO-Explorer benötigten Libraries auf dem Server Ikarus installiert. Das Vorgehen für die Konfiguration der Startontologie wurde auf dem Devil in der FG BAS der Universität folgendermaßen strukturiert: Zunächst wurde eine (1) Startkonfiguration aus der Startontologie (vgl. Abb. 6.4 und Abb. 6.5) gewählt, welche aus den Social Subconcepts *Mitarbeiter* (Social Concept: Person) und *Experimentieren* (Social Concept: Produkt) bestand. Die Wahl fiel auf diese beiden Knoten, da der *Mitarbeiter* einer der zentralsten Social Subconcepts durch den Social-Network-Ansatz ist. Das Produkt *Experimentieren* wurde gewählt, weil dazu Beispieldatensätze vorhanden waren und dies eines der zentralsten Produkte im KOSMOS Verlag

war. Außerdem war die Produktdatenbank schon länger bei KOSMOS im Einsatz und dadurch waren darin mehr Datensätze vorhanden als in der Projektdatenbank, die erst vor kurzem eingeführt wurde. Da die Quellsysteme als Notes-Anwendungen keine API besaßen, wurden in den Quellsystemen JSON-Strings für das Transformieren der benötigten Attribute in ein einheitliches Datenformat eingerichtet, sodass ein Datenaustausch ermöglicht wurde. Anschließend wurde die Startkonfiguration zu den beiden Social Subconcepts importiert und getestet. Der Import der Startkonfiguration und das Testen nahm einen hohen Zeitaufwand von etwa 96 Stunden durch drei Personen (SME, BGE und KGO) in Anspruch. Hilfreich bei der Konfiguration waren die Informationen, die in dem SoNBO-Mapper (= Excel-Dokument) enthalten waren, da das Administrationsfrontend des SoNBO-Explorers für die Konfiguration nicht intuitiv war. Dies war bedingt dadurch, dass der SoNBO-Explorer als Prototyp in einem Studentenprojekt entstand. Die konkreten Herausforderungen und die daraus abgeleiteten Erkenntnisse (vgl. Tab. 6.4) werden im nächsten Abschnitt beschrieben. Mit diesen Erkenntnissen wurden die Informationen im Excel-Dokument nochmal überarbeitet, sodass die Startontologie daraufhin final war. Der anschließende Import der Startontologie war dann durch diese Erkenntnisse aus den Herausforderungen bei der Startkonfiguration wesentlich schneller möglich und somit konnte die Startontologie erfolgreich im SoNBO-Explorer konfiguriert werden. Dazu wurden wieder die JSON-Strings in den weiteren Datenbanken formuliert, die Startontologie importiert und wieder getestet. Die weitere Konfiguration bis hin zur Eingabe der Zielontologie war nicht Ziel des Forschungsprojektes. Diese kann durch den KOSMOS Verlag selbst in den SoNBO-Mapper eingegeben und importiert werden oder weitere Knoten können über das Administrationsfrontend sukzessive hinzugefügt werden. Abschließend wurde der auf dem Devil in der Universität konfigurierte SoNBO-Explorer auf den Ikarus des KOSMOS Verlags kopiert. Nachdem konkrete Servereinstellungen angepasst wurden, ergab das Testen, dass die Konfiguration korrekt vorgenommen wurde und der SoNBO-Explorer auch für die Live-Daten (und nicht nur für die Testdaten) die Daten aus den Quellsystemen im Sinne der Konfiguration integrierte. Der nächste Schritt ist die Integration in KosmoNet (HCL Connections beim KOSMOS Verlag), welcher unabhängig vom Dissertationsprojekt durchgeführt wird, da der Fokus bei der Evaluation die Konfiguration war.

6.5.2 Erkenntnisse

Mithilfe des SoNBO-Mappers konnten alle für die Konfiguration notwendigen Informationen durch das semantische und technische Mapping erhoben werden, sodass der Einsatz davon in zukünftigen SoNBO-Einführungsprojekten beibehalten werden kann (vgl. erste Zeile in Tab. 6.4). Durch das Excel-Dokument, welches das erhobene Mapping auch in die Struktur des SoNBO-Explorers übersetzt, ist auch der SoNBO-Mapper als geeignet für die Einführung von SoNBO in einem Unternehmen evaluiert worden, da im Code nichts angepasst werden musste. Allerdings ergaben sich in dem Konfigurationsprozess einige Herausforderungen, die im Folgenden beschrieben werden. Außerdem wird die daraus abgeleitete Erkenntnis für zukünftige Projekte dargelegt (vgl. Tab. 6.4).

Tab. 6.4: Herausforderungen bei der Startkonfiguration und daraus abgeleitete Erkenntnisse

	Beschreibung des Sachverhaltes	Erkenntnis
Aufbau im SoNBO-Mapper (Excel-Dokument)	Hilfreicher Aufbau des semantischen Mappings im Excel-Dokument für die Konfiguration	Beibehalten der Struktur des Mappings im Excel-Dokument als Unterstützung für die Konfiguration
Attribute	Unklarheit beim Zugriff auf die Attribute (Backendfeld, JSON-Key, SoNBO-Attribut) Beispiel: Attribut für ID (= <i>Form</i> bei Mitarbeiter) war nicht im JSON-String und somit nicht als SoNBO-Attribut gelistet	Klarheit schaffen im Excel-Dokument und in dem Administrationsfrontend des SoNBO-Explorers
Formatierung	Fehlende Formatierung im JSON-String	Korrekte Erstellung der JSON-Strings (Verwendung von @Text, @Name, @If)
	Falsche Formatierung in den Excel-Spalten (insbes. bei Zahlen)	Anpassung der Formatierung im Excel-Dokument bzw. Anpassung im SoNBO-Explorer (Umwandlung in String)
Semantisches Mapping	Falsche Informationen beim Erheben des Mappings Beispiel: Bei Produkt/Projekt wurde Location als Kante angegeben. Aber: Location = Science (und nicht KOSMOS)	Konkretes Beispiel von Beginn an kreieren, das beim Testen unterstützt Engeres Arbeiten mit anzubindenden Datenbanken von Beginn an
Datenqualität	Mangelnde Datenqualität I: Unvollständige Testdatensätze durch KOSMOS bereitgestellt (→ kein Testen möglich bei 12 SSC und 2 Kanten)	Konkretes Beispiel von Beginn an kreieren, das beim Testen unterstützt. Engeres Arbeiten mit anzubindenden Datenbanken von Beginn an
	Mangelnde Datenqualität II: Unvollständige Datensätze in den Quellsystemen bei KOSMOS (→ problematisch bei relevanten Attributen wie ID, Zuordnung SSC (Produkt, Projekt) und Kante zwischen Projekt, Produkt, Lieferantenkontakt)	(Empfehlungen für das Unternehmen zur Datenpflege formulieren.)
Kanten	Formulierung der Queries für die Kanten (nur in eine Richtung)	Queries müssen in Zukunft in beide Richtungen formuliert werden

Eines der zentralen Probleme bestand darin, dass der SoNBO-Explorer auf verschiedene Arten von Attributen zugreift. Dies soll folgendes Beispiel verdeutlichen (vgl. Tab. 6.5): Beispielsweise hat das Experimentierprojekt *Kristall-Seife* das Attribut *Project leader* als Frontendfeldname, während der Backendfeldname *MainProjectLeader* ist. Als Value steht für dieses Projekt *Boris Saltzmann/Kosmos* in der Projektdatenbank, die als Quellsystem dient. Im JSON-String wird als Key *Project leader* gewählt und als SoNBO-Attribut *Projektleiter*. Diese drei verschiedenen Attribute, die vom SoNBO-Explorer verwendet werden, haben folgenden Hintergrund: Der (1) *Backendfeldname* ist unabhängig vom SoNBO-Explorer im Quellsystem vorhanden und kann nicht verändert werden. Der (2) *JSON-Key* zu dem Backendfeldnamen ist frei wählbar bei der Erstellung des JSON. Das (3) *SoNBO-Attribut* zu dem Backendfeldname

ist ebenfalls frei wählbar und wird im SoNBO-Explorer so gewählt werden, dass die Bezeichnung für den User sinnvoll ist, da dies im Frontend des SoNBO-Explorers angezeigt wird. Der Value ist dabei immer gleich. In Tab. 6.5 sind die verschiedenen Attribute beispielhaft gegenübergestellt. Prinzipiell macht es Sinn, für den JSON-Key denselben Begriff wie für das SoNBO-Attribut zu verwenden, was auch für zukünftige Projekt gemacht werden sollte.

Tab. 6.5: Zugriff auf verschiedene Arten von Attributen

Social (Sub)Concept <i>Social Business Object</i>	Frontend-feldname	Backend-feldname	Value im Backend = Value im JSON	Key im JSON	SoNBO-Attribut
<i>Projekt (Exp.) Kristall-Seife</i>	<i>Project leader</i>	<i>MainProjectLeader</i>	<i>"Boris Saltzmann/ Kosmos"</i>	<i>Project leader</i>	<i>Projektleiter</i>
<i>Person (Mita.) Boris Saltzmann</i>	<i>Manager</i>	<i>Manager</i>	<i>„Heike Landmann/ Kosmos“</i>	<i>Manager (Hinzufügen: „@Name“)</i>	<i>Manager</i>

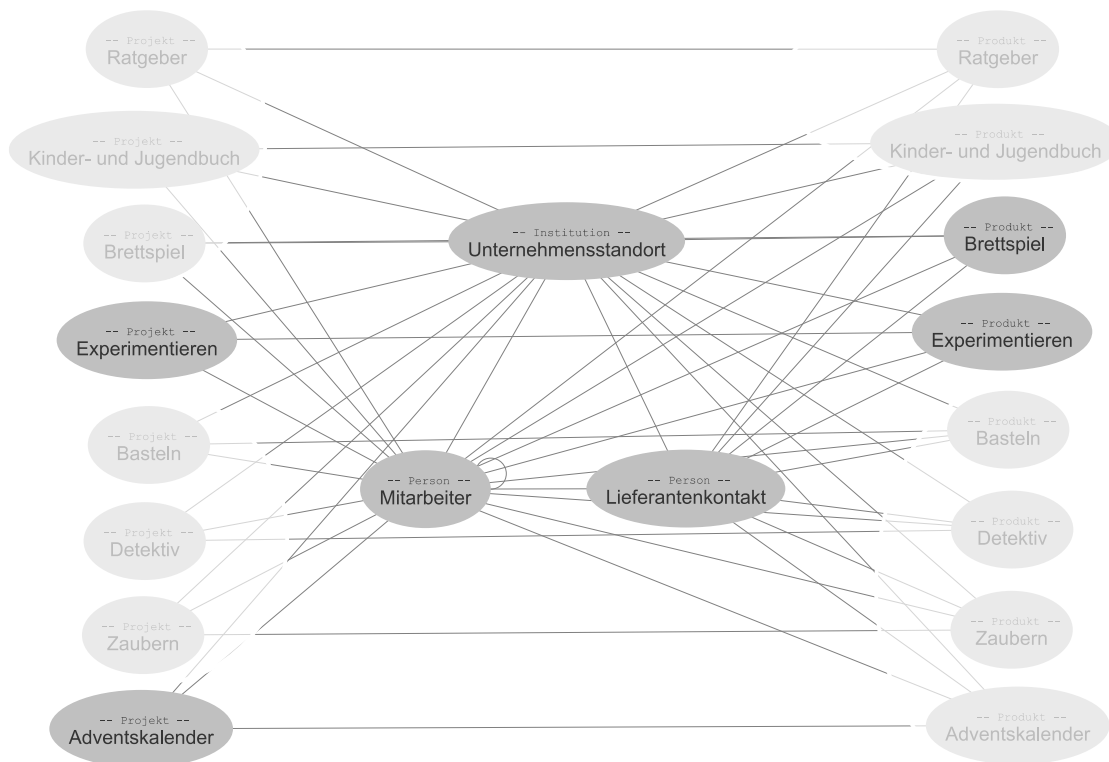
Die Problematik beim Laden der Daten aus den Quellsystemen durch den SoNBO-Explorer besteht nun darin, dass für die verschiedenen Zwecke zur Spezifizierung der Ontologie (vgl. Tab. 6.6) jeweils auf ein anderes Attribut in der Query zugegriffen wird. Dies liegt vermutlich darin begründet, dass bei der Entwicklung des Prototyps nicht auf eine Einheitlichkeit geachtet wurde. Während auf den Frontendfeldnamen nie zugegriffen wird, wird für die Erzeugung der Kante der *Backendfeldname* benötigt und für die Zuordnung des jeweiligen *Social Business Objects* zum *Social Subconcept* der *Key* im *JSON-String*. Für die restlichen Zugriffe (ID, Titel des SBO, Visualisierung der Attribute im Frontend und auch für den anderen Teil der Query für die Kante) werden die *SoNBO-Attribute* benötigt. Da dies bei der vorbereiteten Konfiguration und den formulierten Queries nicht bekannt war, entstand Tab. 6.5 erst nach langem Testen der Startkonfiguration. Beispielsweise erfolgt die Zuordnung zum SSC *Mitarbeiter* über die Form, mit der diese angelegt werden, welche aber nicht in den JSON-Key mitaufgenommen wurde. Daher wurde zu Beginn der Knoten nicht geladen. Mit dieser Erkenntnis wurden die Queries für die Startontologie angepasst und entsprechende Hinweise im SoNBO-Mapper und im SoNBO-Explorer mitaufgenommen. Längerfristig wird es sinnvoll sein, eine Code-Anpassung im SoNBO-Explorer vorzunehmen, sodass lediglich ein Attribut (z. B. SoNBO-Attribut) für die Queries verwendet werden muss. Auf diese Weise ist für die Formulierung der Queries nicht mehr diese Tabelle notwendig.

Tab. 6.6: Verwendung der Attribute für den jeweiligen Zweck zur Spezifizierung der Ontologie

Zweck	Frontend-feldname	Backend-feldname	Value im Backend = Value im JSON	Key im JSON	SoNBO-Attribut
ID	-				x
Zuordnung zum SSC	-		x	x	
Titel SBO					x
Verwendung im SoNBO-Explorer Frontend	-				x
Kante	-	x	x		x

Ein weiteres Problem bestand darin, dass die Values im JSON-String nicht korrekt formatiert wurden. Während der JSON-String zunächst die Attribute in Keys übersetzt, ermöglicht dieser auch, Formatierungen vorzunehmen. Beispielsweise ist der Value in der ersten Zeile in Tab. 6.5 „*Boris Saltzmann/Kosmos*“, allerdings soll nur der Name „*Boris Saltzmann*“ weitergegeben werden. Diese Kürzung kann mit der Funktion „@Name“ erzielt werden. Ein Formatierungsproblem bestand auch z. B. bei Zahlen. Dazu wurden die Spalten in Excel in numerische Werte (Zahl) umformatiert, wobei im SoNBO-Explorer in der entsprechenden Spalte auch die Information hinterlegt wurde, dass es sich bei dem Value um eine Zahl handelt. Eine weitere Hürde bei der Konfiguration bestand in der Erhebung des semantischen Mappings (vgl. fünfte Zeile in Tab. 6.5): Für Projekte und Produkte wurde beispielsweise das Attribut *Location* für die Herstellung der Kante zum Unternehmensstandort angegeben. Dieses konnte dafür aber nicht verwendet werden, da der Value *Science* und nicht *KOSMOS Verlag* ist. Im Übrigen existieren in den Daten keine Kanten zu dem Unternehmensstandort, weshalb diese (als einzige) manuell hinzugefügt wurden. Ein weiteres Problem bestand in der Formulierung der Kanten. Diese sind im SoNBO-Konzept immer bidirektional und müssen auch in beide Richtungen in der Query formuliert werden. In der Vorbereitung wurden diese nur in eine Richtung formuliert. Ein weiteres Problem bestand in der Datenqualität vom KOSMOS Verlag, sodass das Testen und somit die Ursachensuche in der Konfiguration erschwert wurde (vgl. letzte Zeile in Tab. 6.5). Weiterhin war eine Herausforderung, dass die dem SoNBO-Team zur Verfügung gestellten Datensätze unvollständig bzw. zum Teil noch gar nicht vorhanden waren. Ersteres war insofern hinderlich, als dass es beim Testen zunächst als Problem identifiziert werden musste und das Anfragen weiterer Datensätze Zeit in Anspruch nahm. Letzteres ist ein Datenpflegeproblem, was im SoNBO-Projekt nicht gelöst werden kann. Allerdings kann der SoNBO-Explorer bei der Datenpflege unterstützen, da dieser unvollständige Daten im Benutzerfrontend sichtbar macht. Als Erkenntnis lässt sich für diese Probleme ableiten, dass von Beginn an die Qualität der Testdatensätze zunächst überprüft werden muss, indem beispielsweise für jedes Social Subconcept ein jeweils konkretes Social Business Object ausgewählt wird, zwischen denen auch

Beziehungen bestehen. Für das Testen bedeutete dies, dass durch die unzureichenden Datensätze zwölf Social Subconcepts und zwei Kanten (Lieferantenkontakt – *Produkte*; *Projekte* und *Produkte*²⁰) (vgl. Abb. 6.15) nicht getestet werden konnten. Allerdings ergab sich nach der Installation des SoNBO-Explorers beim KOSMOS Verlag und somit der Anbindung von den Live-Systemen, dass die Knoten und Kanten korrekt konfiguriert waren.



ausgegraut: kein Testen möglich auf dem Devil (Testumgebung)

Abb. 6.15: Erfolgreich konfigurierte und getestete Ontologie

Abschließend lässt sich festhalten, dass die Konfiguration zwar erfolgreich war, der Beginn (Startkonfiguration) aber durch die Kombination der beiden Hauptprobleme (vgl. Tab. 6.4), nämlich erstens das unvollständige Verständnis über die Verwendung der Attribute in den Queries und zweitens die mangelnden Datensätze zum Testen, sehr mühsam war. Diese Herausforderungen wurden aber im Laufe der Konfiguration deutlich, sodass diese Probleme zunächst gelöst werden mussten. Der SoNBO-Mapper als Excel-Dokument ist weiterhin ein geeignetes Instrument und durch die Anpassungen auch zukünftig anwendbar.

²⁰ Hinweis: Lieferantenkontakt ist ein Social Subconcept und Produkt/Projekt ein Social Concept – Daher sind diese kursiv, da diese stellvertretend für die Social Subconcepts zu sehen sind.

Der SoNBO-Mapper ist als pragmatisches Tool zu verstehen, für das es Technologien gibt, die geeigneter sind. Wenn beispielsweise bei der Beschreibung der Ontologie RDF verwendet werden würde, könnte auch eine Mappingsprache aus diesem Bereich verwendet werden (vgl. Abschnitt 3.2.1.1.3). Falls Klarheit über den Input (Daten und Datenbankschema) existiert und davon kaum abgewichen werden soll, kann *Direct Mapping* für das Erzeugen der RDF Version dieser Daten verwendet werden. Bei R2RML (RDB to RDF Mapping Language) erfolgt ein Customising des Mappings und die RDF Daten entstehen nach einem individuellen Design (Rodríguez-Muro & Rezk, 2015; Villazon-Terrazas, Garcia-Santa, Ren, Faraotti, et al., 2017). Für SoNBO wäre eher R2RML denkbar, da die Ontologie nicht nur ausgehend von dem Datenbankschema, sondern vor allem auch unter der Berücksichtigung der Mitarbeiter entstehen soll. Dies kann dann individuell eingerichtet werden.

6.6 Zusammenfassung

Zusammenfassend lässt sich festhalten, dass sowohl die *indirekte* Evaluation des SoNBO-Konzeptes als auch die *direkte* Evaluation (als die konkrete Anwendung) des SoNBO-Frameworks erfolgreich waren. Dies wurde von DGE sowohl explizit bestätigt, indem er das Vorgehen mit konkreten Aussagen positiv bewertete, als auch implizit bestätigt, da er schnell ein Verständnis zu SoNBO entwickelte sowie die korrekten Begriffe verwendete und den richtigen Input in den jeweiligen Phasen gab. Das SoNBO-Framework konnte durch die Erkenntnisse aus der Evaluation noch weiter spezifiziert und verfeinert werden (z. B. Anpassungen im SoNBO-Mapper). Für den Einsatz des Frameworks in weiteren Projekten ist hier nochmal zu erwähnen, dass die Phasen nicht statisch zu sehen sind, sondern zwischen den Phasen auch gesprungen werden kann. Außerdem ist die Auswahl der Akteure von großer Bedeutung. Der SoNBO-Experte benötigt umfangreiches Wissen über das Konzept und den SoNBO-Explorer und die Mitarbeiter müssen so ausgewählt werden, dass sie mit ihrem Wissen auch unterstützen können. Außerdem muss der SoNBO-Experte auch flexibel auf möglicherweise noch nicht im Framework antizipierte Sachverhalte in Unternehmen (wie z. B. Anpassung der Ontologie auf die Besonderheit der Kundenseite beim KOSMOS Verlag) eingehen können.

Eine Grenze der Evaluation des Frameworks ist der Technologiebezug. Während das Framework technologieagnostisch formuliert ist, erfolgte die Evaluation notwendigerweise in einer *konkreten* Technologie. Dadurch fokussiert sich die Evaluation auf eine Einführung von SoNBO mit HCL-Technologie, und es ist möglich, dass die Verwendung einer anderen Technologie zu leicht anderen oder zusätzlichen Erkenntnissen geführt hätte.

7 Ausblick

In diesem Abschnitt wird der Beitrag der Dissertation für die Praxis und die Wissenschaft aufgezeigt. Außerdem werden die Grenzen der Dissertation und der weitere Forschungsbedarf identifiziert.

7.1 Beitrag für die Wissenschaft: Erweiterung des Wissens im Bereich Ontologie-basierte Informationsintegration

Diese Dissertation leistet einen Beitrag für die Wissenschaft, da die Erkenntnisse in einem Zusammenspiel aus Wissenschaft und Praxis im Sinne der Wirtschaftsinformatik entstanden und die konstruierten Artefakte konkrete und relevante Probleme lösen (Baskerville et al., 2011; Frank, 2006; Junglas et al., 2011; Nett & Stevens, 2009; Orlikowski & Baroudi, 1991; Österle et al., 2011) (vgl. Kapitel 2): Die Forschungsfragen wurden durch die Praxis inspiriert (Key Case Vössing, vgl. 3.1) und die Forschungsergebnisse basierten auf empirischen Daten (Key Case Vössing; Ergebnis Literaturanalyse), welche mit rigorosen Forschungsmethoden (Fallstudie, strukturierte Literaturanalyse) erhoben und ausgewertet wurden. Die Ergebnisse wurden in die Praxis zurücktransferiert und auf einem Detailniveau evaluiert (Key Case KOSMOS). Dies wurde durch die Forschungsinitiative IndustryConnect forciert und ermöglicht (Schubert & Williams, 2020). Auf diese Weise wurde der Design-Science-Research-Zyklus vollständig durchlaufen (Kuechler & Vaishnavi, 2008). Durch den Rücktransfer der Forschungsergebnisse in die Praxis wurde die letzte Meile („last research mile“ nach Nunamaker et al. (2015, p. 10)) gegangen, was nach ebd. den größten Einfluss auf Wissenschaft und Gesellschaft hat („may ultimately have the greatest impact on science and society“). Die dadurch entstandenen nützlichen Artefakte (belegt durch die Evaluation) leisten einen *inhaltlichen* und *methodischen* Beitrag für die Wissenschaft. Der *inhaltliche* Beitrag ist die Bereitstellung des neuen Ontologie-basierten Informationsintegrationsansatzes SoNBO (vgl. Kapitel 4). Dieser neue Ansatz entstand, indem eine Software aus einem Unternehmen als innovativ im Rahmen der Forschungsinitiative IndustryConnect identifiziert wurde, diese Merkmale mithilfe von Ontologie-basierten Informationsintegrationskonzepten aus der Wissenschaft Informatik (Ontologien, OBDA, EKG) beschrieben wurden und anschließend darauf aufbauend ein Einführungskonzept entwickelt wurde, welches in der Praxis evaluiert wurde. Somit fand ein Transfer von Theorien aus der Disziplin Informatik in die Wirtschaftsinformatik statt, wodurch das SoNBO-Konzept entstand, was als geeignet evaluiert wurde und somit ein nützliches Artefakt darstellt. Im SoNBO-Konzept spielt wie bei den Ontologie-basierten Konzepten aus der Informatik der Knowledge Graph eine tragende Rolle, welcher aus einer Ontologie und einer Faktenbasis besteht. Allerdings hebt sich SoNBO durch folgende Eigenschaften von den Konzepten aus der Informatik ab: Die Ontologie (Social Network of (Sub-)Concepts) ist *unternehmensspezifisch*, da die Anforderungen der Mitarbeiter eines konkreten Unternehmens bei der Entwicklung der Ontologie zusätzlich zu den technischen Gegebenheiten (Aufbau der Quellsysteme) berücksichtigt werden. Bei SoNBO entsteht die Ontologie nicht nur bottom-up aus den Daten heraus, sondern auch (und vor allem) top-down durch den Einbezug des Wissens der Mitarbeiter. Der daraus resultierende Knowledge Graph ist bei SoNBO für jeden Benutzer

einzigartig, da ein Benutzer nur den Teil des Knowledge Graphen für den Informationszugriff nutzen kann, zu dem dieser berechtigt ist. Weiterhin besitzt der Knowledge Graph, der bei SoNBO *Social Network of Business Objects* genannt wird, *soziale Prinzipien* aus Social Software: Der Knowledge Graph ist ein *Enterprise Social Network* bestehend aus Knoten und Kanten wie dieser aus Social Software bekannt ist, mit einem Unterschied. Während ein Knoten im *Enterprise Social Network* der *Social Software* immer ein Mensch ist, ist der Knoten bei SoNBO ein Mensch *oder* ein Ding (z. B. Rechnung). Die Netzwerkstrukturen bestehend aus Menschen in Social Software wurden durch SoNBO also auf Dinge übertragen. Die Kante zwischen zwei Dingen oder einem Ding und einem Menschen kommt zu Stande, indem existierende Attribute aus den Quellsystemen (z. B. Sachbearbeiter einer Rechnung) verwendet werden. Somit kommt die Kante bei SoNBO durch den Administrator zu Stande, während sie bei Social Software durch den Benutzer entsteht (durch die soziale Funktion *follow*). Jeder Knoten wird als *Social Profile* in der jeweiligen Software im Benutzerfrontend dargestellt. Somit existieren Social Profiles bei SoNBO nicht nur für Menschen, sondern auch für Dinge. Der Informationszugriff erfolgt durch die Navigation von Knoten zu Knoten oder: von *Social Profile* zu *Social Profile* als Query by Navigation (vgl. Abschnitt 3.2.1.4), indem dem Benutzer die Informationen aus der Sicht eines Knotens angezeigt werden. Ein weiteres *soziales Prinzip*, das aus der Social Software auf SoNBO übertragen wurde, ist die Steigerung der Awareness durch Notifications. Diese wird bei dem Benutzer gesteigert, indem die Knoten Notifications generieren, die Aktualisierungen des Knotens beinhalten, welche durch das System erzeugt werden. Während die Notifications bei der Social Software in den Activity Stream gepostet werden, werden sie bei SoNBO auch in das Profil des betreffenden Knotens als Recent Events (im Benutzerfrontend) gepostet. Die Social Software spielt bei SoNBO nicht nur bezüglich des Übertragens der Eigenschaften dieses Softwaretyps auf SoNBO eine Rolle. Die Software, die SoNBO umsetzt und somit die Informationen aus dem Softwaretyp ERP-System integriert, wird SoNBO-Explorer genannt und wird wiederum in den Softwaretyp ECS (Enterprise Collaboration Software) integriert. ECS umfassen die Eigenschaften von Social Software. Auf diese Weise werden diese beiden in Unternehmen oftmals getrennt betrachteten Softwaretypen *ERP-Systeme* und *Kollaborationssysteme* und die darin enthaltenen Informationen miteinander verknüpft. Im Zuge dessen wird ein weiteres soziales Prinzip auf SoNBO übertragen, indem die eben beschriebenen Notifications, die Änderungen eines Objektes beinhalten (aus den ERP-Systemen), nicht nur in das Benutzerfrontend des SoNBO-Explorers, sondern auch in den *Activity Stream* des ECS gepostet werden. Somit wird die Awareness der Benutzer des ECS gegenüber den Informationen eines ERP-Systems gesteigert, was keine traditionelle Funktionalität eines ERP-Systems ist. Der Benutzer kann also aus dem ECS auf die Informationen der ERP-Systeme zugreifen. Um es mit den Worten der Praxis zu sagen: Das Zugreifen auf die Informationen der transaktionalen Systeme durch das Kollaborationssystem wird durch SoNBO möglich.

Neben diesem inhaltlichen leistet die Dissertation auch einen *methodischen Beitrag*. Aufbauend auf dem SoNBO-Konzept wird in dieser Dissertation ein Framework bereitgestellt, das bei einem SoNBO-Einführungsprojekt in einem Unternehmen unterstützt und ein nützliches Artefakt für die Praxis darstellt. Für die Entwicklung dieses Frameworks wurde der Aufbau von Einführungsmethoden aus der Wirtschaftsinformatik (ARIS, IRESS) herangezogen. Zwei zentrale Schritte sind in dem Framework die

Ontologie-Erstellung und das Mapping der Informationen in der Ontologie auf die Informationen in den Quellsystemen. Diese beiden Schritte existieren auch in der wissenschaftlichen Literatur (z. B. Pay-as-you-go-Methodology in Abschnitt 3.2.2.3.3) und insbesondere die Ontologie-Erstellung ist nach Sequeda & Miranker (2017) ein offener Forschungsbereich. Die beiden Schritte wurden auf SoNBO durch die eben beschriebenen Besonderheiten des SoNBO-Konzeptes angepasst. Für den besonderen Aufbau der Ontologie ist eine spezielle Unternehmensontologie notwendig, die dann unter Einbezug der Business-Sicht und technischen Sicht auf das Unternehmen angepasst wird. Außerdem wird eine pragmatische Mapping-Methode bereitgestellt, die den Mitarbeiter einbeziehen kann. Das SoNBO-Framework leistet somit einen methodischen Beitrag für das Ontologie-basierte Konzept SoNBO, da die Ontologie-Erstellung und das Mapping nachweislich kein triviales Vorgehen, sondern eine Herausforderung ist (Sequeda & Miranker, 2017). Weiterhin leistet das Ergebnis einen methodischen Beitrag für die übrigen Ontologie-basierten Konzepte aus dem Semantic Web, da das hier entwickelte Framework als praxistauglich evaluiert wurde und somit Einfluss auf die Informatik-Konzepte haben kann. Innerhalb des Frameworks soll die für das Framework entwickelte Unternehmensontologie hervorgehoben werden, die als Referenzmodell dient. Auch wenn bereits zahlreiche Unternehmensontologien existieren (vgl. Abschnitt 3.2.1.2.4), kann die hier entwickelte Ontologie einen Beitrag zu den existierenden leisten.

Abschließend lässt sich festhalten, dass die Dissertation die verschiedenen Gebiete *Praxis*, *Informatik* und *Wirtschaftsinformatik* tangiert und somit *interdisziplinär* einen Beitrag leistet, indem sie diese drei Bereiche zusammenbringt.

7.2 Beitrag für die Praxis: Anwendung des SoNBO-Frameworks in der Praxis

Die Ergebnisse der Dissertation stellen einen Beitrag für die Praxis dar, welcher durch die Evaluation der Forschungsergebnisse in der Praxis als letzter Schritt des Design-Science-Research-Zyklus (Kuechler & Vaishnavi, 2008) demonstriert wurde (vgl. Abschnitt 2). Eine solche Zusammenarbeit von Wissenschaft und Praxis sowie das Bereitstellen von nützlichen Artefakten für die Praxis wird in der Literatur gefordert (Kuechler & Vaishnavi, 2008; Österle et al., 2011; Schubert & Williams, 2020). Außerdem wird in der Literatur bemängelt, dass der Einsatz von Ontologie-basierten Informationsintegrationskonzepten aus der Informatik (z. B. EKG unter der Verwendung von LED) in der Praxis noch zu wenig Anwendung finden (Galkin, Auer, & Scerri, 2016). Durch die Evaluation wurde gezeigt, dass in dieser Dissertation praxistaugliche Artefakte entwickelt wurden und im Evaluationsunternehmen auch eingesetzt werden.

Als Praxisbeitrag stellt diese Dissertation zum einen die Beschreibung eines Konzepts für eine Ontologie-basierte Informationsintegration zur Verfügung, die den Mitarbeitern einen intuitiven Informationszugriff auf die Informationen der transaktionalen Systeme unter der Verwendung von sozialen Prinzipien ermöglicht. Durch die wissenschaftliche und technologieagnostische Beschreibung der Inhalte der Software bei Vössing konnte diese wissenschaftliche Beschreibung als SoNBO-Konzept in dem Unternehmen KOSMOS eingesetzt und somit als praxistauglich evaluiert werden. Es ist davon

auszugehen, dass dies auch in weiteren Unternehmen auf diese Weise evaluiert wird. Dadurch ist der Grundstein für eine Übertragung von SoNBO auf weitere Unternehmen gelegt. Bei der Evaluation war einerseits die Art der Beschreibung des Konzeptes Untersuchungsgegenstand. Zum anderen stellt die Dissertation ein Framework zur Verfügung, das Unternehmen bei der strukturierten Einführung einer Ontologie-basierten Integrationssoftware (hier: SoNBO-Explorer) unterstützt. Dieses wurde ebenfalls als praxistauglich evaluiert. Das Framework stellt Modelle, Methoden und Tools bereit, um SoNBO auf ein beliebiges Unternehmen zu übertragen, indem ein SoNBO-Explorer eingeführt wird. Dabei spielt der Einbezug der Mitarbeiter eine tragende Rolle. Das Framework wurde evaluiert, indem es in einem konkreten Unternehmen (KOSMOS Verlag) angewandt und somit ein SoNBO-Explorer konfiguriert und eingeführt wurde. Auf diese Weise wurde die Nützlichkeit des Artefaktes demonstriert. Dadurch wurde der DSR-Zyklus vollständig durchlaufen. Es ist auch hier davon auszugehen, dass das SoNBO-Framework auf ein beliebiges weiteres Unternehmen angewandt werden und auf diese Weise SoNBO übertragen werden kann. Die gewonnenen Erkenntnisse aus der Evaluation sind bereits in die Verbesserung des Frameworks eingeflossen.

7.3 Grenzen der Dissertation

Prinzipiell sind die Ergebnisse der Dissertation als erste, praxistaugliche und wissenschaftlich fundierte Vorschläge für die Beschreibung des neuen Ontologie-basierten Integrationsansatzes SoNBO aufzufassen. In die Entwicklung der Artefakte floss der aktuelle Erkenntnisstand aus Praxis und Wissenschaft ein. Durch neue Erkenntnisse in den beiden Bereichen können weitere Erkenntnisse gewonnen werden, welche das SoNBO-Konzept und das SoNBO-Framework verfeinern.

Konkret ist bei dem SoNBO-Konzept eine Grenze die Berücksichtigung der Quellsysteme, aus denen bei SoNBO die Informationen integriert werden. Der Fokus liegt in dieser Dissertation auf betriebliche Anwendungssysteme, wobei die Beispiele aus ERP-Systemen stammen. Es wurde allerdings bei der Beschreibung des Konzeptes versucht, es möglichst offen und technologieagnostisch zu halten, sodass die Quellsysteme auch konzeptionell um weitere Quellsystemtypen (z. B. Produktivsysteme) erweitert werden könnten. Bei dem SoNBO-Framework ist eine Grenze, dass der Erfolg des Framework-Einsatzes insbesondere von den beteiligten Akteuren abhängig ist und somit die hier entwickelten Modelle, Methoden und Tools lediglich in dem Einführungsprozess unterstützen können. Die Konfiguration ist ein zeitaufwendiger Prozess, da durch die Fokussierung auf die individuellen Anforderungen eines spezifischen Unternehmens und damit verbunden durch den starken Einbezug des Menschen wenig Automatisierung stattfinden kann. Die Bedeutung des Menschen in der Entwicklung einer Ontologie und dem Mapping ist auch bei OBDA von großer Bedeutung (vgl. Abschnitt 3.2.2.3), wobei der Einfluss bei SoNBO noch größer ist. Es ist aber nicht auszuschließen, dass dies durch z. B. künstliche Intelligenz verbessert werden kann.

Außerdem lässt sich festhalten, dass durch die Evaluation die Anwendbarkeit des Konzeptes und des Frameworks in einer ersten Form demonstriert wurde. Es ist allerdings davon auszugehen, dass mehr Zyklen und somit Erfahrung aus weiteren Einführungsprojekten notwendig sein werden, um weitere

Verbesserungsvorschläge für die Verfeinerung des SoNBO-Frameworks zu erhalten und somit die Tauglichkeit der Modelle, Methoden und Tools im SoNBO-Framework für bessere, fehlerfreiere und schnellere Einführungsprojekte auszubauen. Auf diese Weise wurde auch das ARIS-Toolset (vgl. Abschnitt 3.2.1.6.2) mit Erfahrungen aus dem praktischen Einsatz angereichert und somit weiterentwickelt (Scheer, 1998).

Die Grenzen liegen weiterhin darin, dass im Rahmen der Dissertation nicht alle Themen angesprochen werden konnten, die die Integration tangieren. Nach Schumann (1992, p. 10) wird die Integration der Informationsverarbeitung in (1) *Integrationsreichweite* (innerbetrieblich oder zwischenbetrieblich), (2) *Integrationsgegenstand* (Datenintegration, Funktionsintegration oder Programmintegration) und (3) *Integrationsrichtung* (horizontal oder vertikal) unterteilt. Bezüglich der (1) *Integrationsreichweite* wurde in dieser Dissertation auf die innerbetriebliche Integration fokussiert. Es ist aber nicht auszuschließen, dass auch eine zwischenbetriebliche Integration durch SoNBO umgesetzt werden kann. Dies ist beispielsweise ein Forschungsbereich bei *Linked Enterprise Data* (vgl. Abschnitt 3.2.2.4). Bezüglich des (2) *Integrationsgegenstandes* kann die Arbeit im Bereich *Datenintegration* platziert werden, da es bei SoNBO vor allem um die Integration von Daten aus Quellsystemen und weniger um das Integrieren von Funktionen oder um das Integrieren einzelner Programme geht. Die Datenintegration, für die der SoNBO-Mapper eine pragmatische Lösung bereitstellt und in Zukunft noch weiter verfeinert werden sollte, ist an sich auch ein eigenständiges Forschungsgebiet mit Unterthemen wie automatische Datenweitergabe oder unternehmensübergreifende Datenbanken (Schumann, 1992). Für die Herausforderung der Integration in Unternehmen existieren zahlreiche Tools von Softwareherstellern, um den Aufwand bei einer Informationsintegration zu reduzieren (Bernstein & Haas, 2008). Die Architekturen zur Informationsintegration lassen sich in Data Warehouse Loading, Virtual Data Integration, Message Mapping, Object-to-Relational Mappers, Document Management und Portal Management klassifizieren. Dazu werden Technologien wie z. B. XML, Schema Standards, Data Cleansing, Schema Mapping, Schema Matching, Information Extraction und Dynamic Web Technologies verwendet (Bernstein & Haas, 2008). Die Architekturen besitzen Eigenschaften, die auch bei SoNBO vorliegen, weshalb auch die Erfahrung und konkret die Tools aus diesen Ansätzen auf SoNBO übertragen werden können. Dies wurde in der Dissertation nicht behandelt, da es bei dem Übertragen dieser Eigenschaften eher um das Verfeinern des SoNBO-Konzeptes und des SoNBO-Frameworks gehen würde und diese bei der Entwicklung nicht in einem angemessenen Maße unterstützt hätte. Bezüglich des dritten Bereichs nach Schumann (1992) (3) *Integrationsrichtung* lässt sich die Dissertation bei der vertikalen Integration einordnen, da das Unterstützen der Prozesse im Rahmen einer horizontalen Integration (im Sinne einer Prozessorientierung) bei SoNBO nicht im Vordergrund steht (Schumann, 1992).

7.4 Weiterer Forschungsbedarf

Der Fokus der Dissertation liegt auf der Integration von Informationen aus betrieblichen Anwendungssystemen mit dem Fokus auf ERP-Systemen. Das Konzept ist allerdings so offengehalten, dass auch Social Concepts aus weiteren Systemtypen miteingebunden werden können. Beispielsweise kann aus

dem betrieblichen Anwendungssystem *Enterprise Collaboration Systems* ein Social Document (z. B. Blog) als Social Concept dienen. Ein Anwendungsfall wären Blogbeiträge aus einer Community zu einem Projekt XY, zu welchem Informationen auch in einem Projektmanagement-System liegen. Durch SoNBO können diese Informationen zusammengeführt werden. Für die Struktur der Social Documents gibt es bereits eine Ontologie, sodass bei der Integration dieses Quellsystems auch darauf zurückgegriffen werden kann (Williams, Mosen, & Schubert, 2020). Ein weiteres Beispiel für die Erweiterung der Quellsysteme ist das Einbeziehen von Maschinen. In diesem Zusammenhang ist eine *Maschine* ein Social Concept, die beispielsweise über SoNBO durch die *Recent Events* darüber informieren und warnen kann, dass eine gewisse Temperatur überstiegen wurde. Diese Information kann durch einen Sensor an der Maschine ausgelöst werden, wodurch IoT (Internet of Things) eine Rolle spielt (Sheth, 2016). Durch die Informationen dieser beiden Quellsystemtypen kann die Unternehmensontologie (vgl. Abb. 5.5 auf S. 139 und Abb. 5.6 auf S. 140), die als Referenzmodell dienen soll, erweitert werden.

Außerdem kann SoNBO bezüglich der Ontologieentwicklung und dem Mapping weiterentwickelt werden. Während der Einbezug des Menschen nach wie vor eine große Rolle spielt, kann dennoch z. B. das Mapping weiter automatisiert werden, indem die Ontologie für die Informationen in den Quellsystemen mithilfe von Schema Integration (Batini & Lenzerini, 1984; Delfmann, Herwig, & Łukasz, 2009; Havel, Steinhorst, Dietrich, & Delfmann, 2014; Rahm & Bernstein, 2001) teilweise automatisiert erfolgen kann. Schema Integration hat seine Ursprünge in den frühen 1980er Jahren. Da bereits zu dieser Zeit Schemata unabhängig voneinander in verschiedenen Domänen entwickelt wurden und somit eine unterschiedliche Struktur und Terminologie aufwiesen, entstand die Notwendigkeit, diese Schemata zu integrieren. Ein erster Schritt bei der Schema Integration besteht darin, diese Interschema-Beziehung zu identifizieren und übereinstimmende Elemente unter einem zusammenhängenden, integrierten Schema zu vereinheitlichen. Bei dieser Integration werden Programme oder Abfragen erstellt, die die Übersetzung von Daten aus den ursprünglichen Schemata in die integrierte Darstellung ermöglichen. Eine Variation der Schema Integration besteht darin, ein unabhängig entwickeltes Schema in ein gegebenes konzeptionelles Schema zu integrieren. Dies erfordert wieder die Abstimmung der Struktur und der Terminologie der beiden Schemata und wird *Schema Matching* genannt (Rahm & Bernstein, 2001). Dieses unabhängig entwickelte Schema soll dann als einheitliches Schema als gemeinsame Sprache in verschiedenen Kontexten verwendet werden. Dieses Prinzip erinnert an SoNBO, bei dem eine Ontologie (als Schema für den Informationszugriff) auf die Informationen in den Quellsystemen gemappt werden soll. Auch wenn Schema Matching typischerweise manuell durchgeführt wird, gibt es auch Techniken, die dies unterstützen (Rahm & Bernstein, 2001).

Weiterhin ist eine vollständige Umsetzung des SoNBO-Konzeptes mit Semantic-Web-Konzepten denkbar, indem voll auf diesen Technologie-Stack zurückgegriffen wird. Beispielsweise kann eine Graph-Datenbank für die Speicherung der Ontologie eingesetzt werden, wie dies von Social Media und Google bekannt ist (vgl. Abschnitt 3.2.1.5). Außerdem kann für die Beschreibung der Ontologie auch eine Ontologie-Beschreibungssprache (z. B. RDF) eingesetzt werden, wodurch dann auch eine Abfragesprache (z. B. SPARQL) verwendet werden kann (vgl. Abschnitt 3.2.1.1). Auf diese Weise werden die

Standards aus der Informatik verwendet. Wichtig ist dabei dann nur, dass auch immer eine für den Menschen lesbare Ontologie aus dem RDF-Graph erzeugt (oder in eine andere Darstellung wie Labeled Property Graph transformiert) werden kann (Tomaszuk, 2016), sodass der Mensch bei der Entwicklung weiterhin mit einbezogen werden kann. Ein Vorteil bei der Verwendung von RDF wäre, dass Reasoning (vgl. Abschnitt 3.2.1.2) ermöglicht wird. Außerdem könnte die Mappingsprache R2RML (RDB to RDF Mapping Language) eingesetzt werden (vgl. Abschnitt 3.2.1.1), bei der ein Customising des Mappings nach einem individuellen Design erfolgt, sodass auf diese Weise die RDF Daten entstehen (Rodríguez-Muro & Rezk, 2015; Villazon-Terrazas, Garcia-Santa, Ren, Faraotti, et al., 2017). Unabhängig vom Einsatz weiterer Technologien besteht auch Verbesserungsbedarf in der technischen Umsetzung des SoNBO-Explorers. Für eine Anpassung des Codes, um die Konfiguration zu erleichtern, können die Erkenntnisse aus der Evaluation miteinbezogen werden. Beispielsweise könnte das Administrationsfrontend so überarbeitet werden, dass das Tool *SoNBO-Mapper* zunehmend überflüssig wird. Für die Durchführung des Mappings wurde ein pragmatischer Ansatz in Form des Tools *SoNBO-Mapper* gewählt, was durch ein erweitertes Administrationsfrontend zunehmend abgelöst werden könnte. Außerdem wurden bei der Evaluation der Activity Stream und die Recent Events nicht berücksichtigt, da der für das SoNBO-Framework eingesetzt SoNBO-Explorer diese Funktionalitäten nicht besaß. Dies sollte in weiteren Projekten ebenfalls einbezogen werden bezüglich der technischen Umsetzung und der konzeptionellen Einbindung in die Konfiguration. Bei dieser Erweiterung würde dann insbesondere das Mapping tangiert werden, da die Attribute erhoben werden müssen, deren Änderungen in den Quellsystemen die Notifications auslösen sollen.

Ein weiterer Forschungsbedarf ist die Weiterentwicklung einer Unternehmensontologie im SoNBO-Framework als Referenzmodell für weitere SoNBO-Projekte, die zukünftig in die drei Ebenen *Top-Level-Ontologie* (= branchenübergreifend), *Domänenontologie* (= branchenbezogen) und *Anwendungsontologie* (= unternehmensbezogen) unterteilt werden kann (vgl. Abschnitt 3.2.1.2.2). Eine weitere Möglichkeit, eine Ontologie als Ausgangsbasis zu erstellen, sind *putative* Ontologien, die mithilfe eines Bootstraps aus dem Quelldatenbankschema abgeleitet werden (Sequeda, 2017). Diese können die Referenzmodelle im SoNBO-Framework unterstützen, sodass auf diese Weise das Einbeziehen der technischen Gegebenheiten verstärkt unterstützt wird. Es ist außerdem davon auszugehen, dass die Konfiguration des SoNBO-Explorers nicht nur durch die Methoden- und Modellorientierung sondern auch durch eine zunehmende Erfahrung (z. B. in konkreten Branchen) effizienter durchgeführt werden kann, wie dies auch bei Einführungsprojekten mithilfe des ARIS-Toolsets der Fall ist (vgl. Abschnitt 3.2.1.6.2).

8 Zusammenfassung

Das Ziel der Dissertation bestand in der Weiterentwicklung des Wissens im Bereich der Ontologie-basierten Informationsintegration durch die Erweiterung dieses Integrationsansatzes um das Konzept Social Network of Business Objects (SoNBO) und durch die Entwicklung von wissenschaftlich fundierten und praxisrelevanten Modellen, Methoden und Tools (= Framework), die eine Einführung von SoNBO in ein beliebiges Unternehmen ermöglichen (vgl. Forschungsziele in Tab. 1.1 auf S. 10). Dieses Ziel wurde erreicht, indem zunächst der Status quo in der Praxis (*Teilziel 1a*) zu einem Ontologie-basierten Integrationsansatz im Rahmen des Key Case Vössing durch die Methode Tiefenfallstudie erhoben wurde. In diesem Key Case war eine Individualsoftware durch das IT-Team des Unternehmens Vössing entwickelt worden, welche die Informationen aus den Quellsystemen Ontologie-basiert integriert. Diese Entwicklung erfolgte in diesem Unternehmen intuitiv ohne die Verwendung von Semantic-Web-Technologien und weist Merkmale auf, die noch nicht in der wissenschaftlichen Literatur existieren. Die beiden Hauptmerkmale sind die (1) Verwendung einer *unternehmensspezifischen Ontologie*, die sowohl die Anforderungen des Unternehmens²¹ als auch die Struktur der Quellsysteme berücksichtigt (also nicht nur bottom-up, sondern auch top-down). Weiterhin wurde in dieser Applikation (2) die Struktur eines *nutzerspezifischen Knowledge Graphen* als Social Network of Business Objects realisiert, der für jeden Nutzer spezifisch ist und *soziale Prinzipien* aus *Social Software* aufweist. Außerdem wurde der Status quo zu *Ontologie-basierten Integrationsansätzen* (*Teilziel 1b*) in der wissenschaftlichen Literatur identifiziert, indem eine strukturierte Literaturanalyse durchgeführt wurde. Dabei wurden zwei zentrale Konzepte identifiziert (EKG und OBDA), die diese Ansätze mit Merkmalen wie z. B. *Materialisierung* oder *Virtualisierung* des *Knowledge Graphen* beschreiben. Weiterhin wurde im Laufe des SoNBO-Forschungsprojektes ein SoNBO-Explorer auf der Basis des Key Case Vössing im Rahmen von Qualifikationsarbeiten entwickelt und eine Konfiguration dafür durchgeführt (Gebel-Sauer & Schubert, 2019b; Götz & Gebel-Sauer, 2018). Auf der Basis dieser Erkenntnisse wurde eine Konzeptualisierung von SoNBO als Ontologie-basierter Integrationsansatz (*Teilziel 2a*) vorgenommen, indem der in dem Unternehmen identifizierte Ansatz mit den Konzepten aus dem Semantic Web beschrieben wurde und die darüber hinaus existierenden Merkmale (soziale Prinzipien, unternehmensspezifische Ontologie, nutzerspezifischer Knowledge Graph) hinzugefügt wurden. Dies ist die Grundlage für das SoNBO-Framework (*Teilziel 2b*), das eine strukturierte Einführung von SoNBO mit (Referenz-)Modellen, Methoden und Tools in einem beliebigen Unternehmen ermöglicht. Dieses SoNBO-Framework wurde anschließend auch in einem Unternehmen (KOSMOS Verlag) evaluiert (*Teilziel 3*). Die Ergebnisse aus der Evaluation sind zum einen, dass das Framework ein valides und funktionierendes Instrument für die Einführung von SoNBO darstellt. Zum anderen entstanden Erkenntnisse für eine

²¹ Das z. B. bei OBDA verwendete Beispiel (vgl. Abschnitt 3.2.2.3.3) macht deutlich, dass bei der Ontologieentwicklung zwar die Anforderungen des Unternehmens miteinbezogen werden sollen, allerdings fehlt bei den Use Cases fundiertes betriebswirtschaftliches Wissen. Hier liegt der Fokus eher auf der Demonstration der technischen Umsetzung.

Verbesserung des Frameworks, die aufgelistet wurden, wodurch das Framework verfeinert und angepasst wurde. Abschließend wurden die Grenzen der Ergebnisse der Dissertation und der weitere Forschungsbedarf (*Teilziel 4*) ermittelt.

Diese Dissertation liefert einen Erkenntnisgewinn sowohl für die Wissenschaft als auch für die Praxis. Für die Wissenschaft wurde ein neuer Ontologie-basierter Integrationsansatz entwickelt, der sich durch den Aufbau der Ontologie und des Knowledge Graphen von den Ansätzen aus dem Semantic Web unterscheidet. Die Ontologie ist unternehmensspezifisch und bindet bei der Entwicklung neben den technischen Gegebenheiten auch die Perspektive der Mitarbeiter des Unternehmens mit ein. Der aus der Ontologie und der Faktenbasis entstehende Knowledge Graph ist userspezifisch, da dieser für jeden Benutzer einzigartig ist. Der Knowledge Graph ist wie ein Enterprise Social Network (bekannt aus Social Software) aufgebaut, da er auch aus Knoten und Kanten besteht. Er besitzt somit soziale Prinzipien (Social Network, Social Profile, Notifications zu Änderungen eines Knotens für Awareness). Außerdem wurde für die Wissenschaft ein Framework entwickelt, das bei der Einführung eines SoNBO-Explorers in einem Unternehmen unterstützt. Dieses wurde auch in der Praxis evaluiert, was ein entscheidender Schritt im DSR darstellt (Kuechler & Vaishnavi, 2008) und wodurch die Nützlichkeit des Artefaktes demonstriert wurde. Durch die wissenschaftliche Beschreibung von SoNBO fand ein Transfer von Theorien aus der Informatik (Semantic Web) in die Wirtschaftsinformatik statt. Für die Praxis entstand ein anwendbares Artefakt, das für Softwareeinführungsprojekte in Unternehmen eingesetzt werden kann.

Eine Grenze der Dissertation liegt in der Datenbasis, die für die Entwicklung des SoNBO-Konzeptes und des SoNBO-Frameworks verwendet wurde. Die Datenbasis stammt aus der Wissenschaft und aus der Praxis und es ist nicht auszuschließen, dass der Einbezug eines weiteren wissenschaftlichen Fachgebietes oder eines weiteren Evaluationsprojektes neue Erkenntnisse für die Verfeinerung der Ergebnisse der Dissertation bereitstellen. In der *Wissenschaft* ist das Feld der Informationsintegration facettenreich und es existieren neben Ontologie-basierten Integrationskonzepten noch zahlreiche weitere Architekturen oder Tools, die bei der Integration und dem Mapping der Daten unterstützen (Bernstein & Haas, 2008) und somit Einfluss auf das SoNBO-Konzept und SoNBO-Framework haben können (vgl. Abschnitt 7.3). Weiterhin wird mit SoNBO lediglich ein Teil der Dimensionen der Integration nach Schumann (1992) angesprochen (z. B. Fokus auf innerbetriebliche und nicht zwischenbetriebliche Integration). Außerdem stammen die Erkenntnisse für die Verfeinerung des Frameworks aus *einem* Evaluationsprojekt in der *Praxis* und es ist davon auszugehen, dass das Framework mit den Modellen, Methoden und Tools durch weitere Einführungsprojekte verbessert wird. Der weitere Forschungsbedarf besteht darin, den SoNBO-Explorer mit weiteren Technologien umzusetzen (z.B. Semantic-Web-Technologien), um auch bei der Konfiguration von den Vorteilen dieser Technologien zu profitieren. Das bei der Konfiguration unterstützende Tool *SoNBO-Mapper* stellte beispielsweise einen pragmatischen Lösungsansatz dar und es ist nicht ausgeschlossen, dass dieses durch weitere Technologien weiterentwickelt werden kann. Schema Integration bietet abseits von Semantic-Web-Technologien beispielsweise ein technisches Hilfsmittel, die unternehmensspezifische Ontologie auf die Daten in den

Quellsystemen zu mappen, was in zukünftigen Projekten angewendet werden kann. Konzeptionell kann das SoNBO-Konzept um weitere Quellsystemtypen ergänzt werden, weshalb das Konzept diesbezüglich offen formuliert ist. Beispielsweise kann die Ontologie *Social Network of Concepts* um weitere Social Concepts aus einem Enterprise Collaboration System ergänzt werden (z. B. Wiki, Blog, Forum). Eine weitere Möglichkeit ist, Produktivsysteme anzubinden, so dass ein Social Concept z. B. auch eine Maschine sein kann.

Abschließend sei festgehalten, dass die Ontologie-basierten Informationsintegrationsansätze und insbesondere die Ontologieentwicklung prinzipiell ein offener Forschungsbereich sind und in der wissenschaftlichen Literatur eine stärkere Übertragung auf bzw. Anwendung in Unternehmen gefordert wird. SoNBO wurde speziell für den Einsatz im Unternehmen entwickelt und liefert damit einen wertvollen Beitrag für diese Forderung. Es bleibt spannend, zu sehen, wie sich dieser Bereich in den nächsten Jahren oder Jahrzehnten entwickeln wird.

Literaturverzeichnis

- Alexopoulos, P., Zhao, Y., Pan, J. Z., & Zhu, M. (2017). Construction of Enterprise Knowledge Graphs (II)*. In J. Z. Pan, G. Vetere, J. M. Gomez-Perez, & H. Wu (Eds.), *Exploiting Linked Data and Knowledge Graphs in Large Organisations* (pp. 117–146). Cham: Springer.
- Allemang, D. (2010). Semantic Web and the Linked Data Enterprise. In D. Wood (Ed.), *Linking Enterprise Data* (pp. 3–23). Boston: Springer.
- Alter, S. (2017). Nothing is more practical than a good conceptual artifact... which may be a theory, framework, model, metaphor, paradigm or perhaps some other abstraction. *Information Systems Journal*, 27(5), 671–693.
- Araujo, T. H. D., Avena, B. T., Braghetto, K. R., & Wassermann, R. (2017). OntoMongo - Ontology-Based Data Access for NoSQL. In *IX Seminar on Ontology Research in Brazil* (pp. 55–66).
- Auer, S. (2016). Interview with Frank van Harmelen on “Linked Data and Business Information Systems.” *Business & Information Systems Engineering*, 58(5), 371–373.
- Auer, S., Petzsch, R., & Unbehauen, J. (2014). Datenintegration im Unternehmen mit Linked Enterprise Data. In T. Pellegrini, H. Sack, & S. Auer (Eds.), *Linked Enterprise Data* (pp. 85–101). Berlin, Heidelberg: Springer.
- Baader, F., Horrocks, I., & Sattler, U. (2004). Description Logics. In S. Staab & R. Studer (Eds.), *Handbook on Ontologies* (2nd ed., pp. 21–44). Berlin, Heidelberg: Springer.
- Bächle, M., & Kolb, A. (2015). *Einführung in die Wirtschaftsinformatik*. München: Oldenbourg Verlag.
- Bagosi, T., Calvanese, D., Hardi, J., Komla-Ebri, S., Lanti, D., Rezk, M., ... Xiao, G. (2014). The Ontop Framework for Ontology Based Data Access. In D. Zhao, J. Du, H. Wang, P. Wang, D. Ji, & J. Z. Pan (Eds.), *The Semantic Web and Web Science. 8th Chinese Conference, CSWS 2014* (pp. 67–77). Berlin, Heidelberg: Springer.
- Barrasa, J. (2017). RDF Triple Stores vs. Labeled Property Graphs: What’s the Difference? Retrieved from <https://neo4j.com/blog/rdf-triple-store-vs-labeled-property-graph-difference/>
- Baskerville, R., Lyytinen, K., Sambamurthy, V., & Straub, D. (2011). A response to the design-oriented information systems research memorandum. *European Journal of Information Systems*, 20, 11–15.
- Batini, C., & Lenzerini, M. (1984). A Methodology for Data Schema Integration in the Entity Relationship Model. *IEEE Transactions on Software Engineering*, SE-10(6), 650–664.
- Berners-Lee, T. (2006). Linked Data. Retrieved from <https://www.w3.org/DesignIssues/LinkedData.html>
- Berners-Lee, T. (2008). Conceptual Graphs and the Semantic Web. Retrieved October 12, 2017, from <https://www.w3.org/DesignIssues/CG.html>
- Bernstein, P. A., & Haas, L. M. (2008). Information integration in the enterprise. *Communications of the ACM*, 51(9), 72–79.
- Bertolazzi, P., Krusich, C., & Missikoff, M. (2001). An Approach to the Definition of a Core Enterprise Ontology: CEO. In *Proceedings of the International Workshop on Open Enterprise Solutions: Systems, Experiences and Organizations (OES-SEO2001)*.
- Bizer, C. (2009). The Emerging Web of Linked Data. *IEEE Intelligent Systems*, 24(5), 87–92.
- Blaikie, N. (2010). *Designing Social Research. The Logic of Anticipation* (2nd ed.). Cambridge: Polity Press.

- Bleiholder, J., & Schmid, J. (2008). Datenintegration und Deduplizierung. In K. Hildebrand, M. Gebauer, H. Hinrichs, & M. Mielke (Eds.), *Daten- und Informationsqualität. Auf dem Weg zur Information Excellence* (1st ed., pp. 123–142). Wiesbaden: Vieweg+Teubner.
- Blumauer, A. (2014a). From Taxonomies over Ontologies to Knowledge Graphs. Retrieved October 20, 2017, from <https://semantic-web.com/2014/07/15/from-taxonomies-over-ontologies-to-knowledge-graphs/>
- Blumauer, A. (2014b). Linked Data in Unternehmen. Methodische Grundlagen und Einsatzszenarien. In T. Pellegrini, H. Sack, & S. Auer (Eds.), *Linked Enterprise Data* (pp. 3–20). Berlin, Heidelberg: Springer.
- Blumauer, A. (2014c). SKOS as a Key Element in Enterprise Linked Data Strategies. In *Proceedings of the Industry Track at the International Semantic Web Conference 2014*.
- Brandes, U. (2010). Graphentheorie. In C. Stegbauer & R. Häußling (Eds.), *Handbuch Netzwerkforschung* (pp. 345–353). Wiesbaden: VS Verlag für Sozialwissenschaften.
- Bretzke, W. R. (1980). *Der Problembefug von Entscheidungsmodellen*. Tübingen: Mohr Siebeck.
- Brickley, D., & Miller, L. (2000). The FOAF Project.
- Calvanese, D., Cogrel, B., Komla-Ebri, S., Kontchakov, R., Lanti, D., Rezk, M., ... Xiao, G. (2017). Ontop: Answering SPARQL queries over relational databases. *Semantic Web*, 8(3), 471–478.
- Calvanese, D., Cogrel, B., Komla-Ebri, S., Lanti, D., Rezk, M., & Xiao, G. (2015). How to Stay Ontop of Your Data: Databases, Ontologies and More. In *The Semantic Web: ESWC (European Semantic Web Conference) 2015 Satellite Events* (pp. 20–25). Springer.
- Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Poggi, A., Rodríguez-Muro, M., ... Savo, D. F. (2011). The MASTRO system for ontology-based data access. *Semantic Web*, 2(1), 43–53.
- Calvanese, D., de Giacomo, G., Lembo, D., Lenzerini, M., Poggi, A., & Rosati, R. (2007). Ontology-based database access. In *Proceedings of the 15th Italian Symposium Advanced Database Systems (SEBD 2007)* (pp. 324–331).
- Casanave, C. (1995). Business-Object Architectures and Standards. In J. Sutherland, D. Patel, C. Casanave, G. Hollowell, & J. Miller (Eds.), *Business Object Design and Implementation* (pp. 7–28). Berlin, Heidelberg, New York: Springer.
- Chaves, L. G., & de Carvalho, R. B. (2016). Analysis of Application Integration and Portals. In Á. Rocha, A. M. Correia, H. Adeli, L. P. Reis, & M. M. Teixeira (Eds.), *New Advances in Information Systems and Technologies* (Vol. 1, pp. 351–358). Springer.
- Checkland, P. (2000). Soft systems methodology: a thirty year retrospective. *Systems Research and Behavioral Science*, 17, S11–S58.
- Checkland, P., & Poulter, J. (2010). Soft Systems Methodology. In M. Reynolds & S. Holwell (Eds.), *Systems Approaches to Managing Change: A Practical Guide* (pp. 191–242). London: Springer.
- Chein, M., & Mugnier, M.-L. (2009). *Graph-based Knowledge Representation. Computational Foundations of Conceptual Graphs*. London: Springer.
- Chmielewicz, K. (1994). *Forschungskonzeptionen der Wirtschaftswissenschaft* (3rd ed.). Stuttgart: Schäffer-Poeschel.
- Creswell, J. W. (2009). *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches* (3rd ed.). Los Angeles, London, New Delhi, Singapore: SAGE.
- Creswell, J. W. (2012). *Educational Research: Planning, Conducting, and Evaluating Quantitative and Qualitative Research* (4th ed.). Boston: Pearson.
- Creswell, J. W., & Clark, V. L. P. (2011). *Designing and Conducting Mixed Methods Research* (2nd ed.).

- Los Angeles, London, New Delhi, Singapore, Washington DC: SAGE.
- Crotty, M. (1998). *The Foundations of Social Research. Meaning and Perspective in the Research Process*. London, Thousand Oaks, New Delhi: SAGE.
- Davenport, T. H. (1998). Putting the Enterprise into the Enterprise System. *Harvard Business Review*, 76(4), 121–131.
- De Nicola, A., Misceo, F., & Missikoff, M. (2005). A Core Business Ontology for eProcurement: a First Proposal.
- De Nicola, A., & Missikoff, M. (2016). A Lightweight Methodology for Rapid Ontology Engineering. *Communications of the ACM*, 59(3), 79–86.
- De Nicola, A., Missikoff, M., & Navigli, R. (2009). A software engineering approach to ontology building. *Information Systems*, 34(2), 258–275.
- Decker, S., Erdmann, M., Fensel, D., & Studer, R. (1999). Ontobroker: Ontology Based Access to Distributed and Semi-Structured Information. In R. Meersman, Z. Tari, & S. Stevens (Eds.), *Database Semantics: Semantic Issues in Multimedia Systems* (pp. 351–369). Boston, MA: Springer.
- Delfmann, P., Herwig, S., & Łukasz, L. (2009). Unified Enterprise Knowledge Representation with Conceptual Models - Capturing Corporate Language in Naming Conventions. In *ICIS 2009 Proceedings - Thirtieth International Conference on Information Systems*.
- Denaux, R., Ren, Y., Villazon-Terrazas, B., Alexopoulos, P., Faraotti, A., & Wu, H. (2017). Knowledge Architecture for Organisations. In J. Z. Pan, G. Vetere, J. M. Gomez-Perez, & H. Wu (Eds.), *Exploiting Linked Data and Knowledge Graphs in Large Organisations* (pp. 57–84). Cham: Springer.
- Dengel, A. (2012). *Semantische Technologien: Grundlagen. Konzepte. Anwendungen*. Berlin, Heidelberg: Springer Spektrum.
- Dietz, J. L. G. (2006). *Enterprise Ontology: Theory and Methodology*. Berlin, New York: Springer.
- Dietzsch, A., & Goetz, T. (2005). Nutzen-orientiertes Management einer Service-orientierten Unternehmensarchitektur. In *7. Internationale Tagung Wirtschaftsinformatik (WI2005)* (pp. 1519–1538).
- Edlich, S., Friedland, A., Hampe, J., & Brauer, B. (2010). *NoSQL: Einstieg in die Welt nichtrelationaler Web 2.0 Datenbanken*. München: Hanser.
- Ehrlinger, L., & Wöß, W. (2016). Towards a Definition of Knowledge Graphs. In *SEMANTICS 2016: Posters and Demos Track*.
- Eicker, S., & Nietsch, M. (1999). Standards zum objektorientierten Paradigma. *Wirtschaftsinformatik*, 41(4), 358–370.
- Eisenhardt, K. M. (1989). Building Theories from Case Study Research. *The Academy of Management Review*, 14(4), 532–550.
- Färber, M., Bartscherer, F., Menne, C., & Rettinger, A. (2018). Linked Data Quality of DBpedia, Freebase, OpenCyc, Wikidata, and YAGO. *Semantic Web*, 9(1), 77–129.
- Feilmayr, C., & Wöß, W. (2016). An analysis of ontologies and their success factors for application to business. *Data and Knowledge Engineering*, 101, 1–23.
- Fernández-López, M., Gómez-Pérez, A., & Juristo, N. (1997). METHONTOLOGY: From Ontological Art Towards Ontological Engineering. In *Proceedings of the Ontological Engineering AAAI-97 Spring Symposium Series*.
- Fink, A. (2014). *Conducting Research Literature Reviews: From the Internet to Paper* (4th ed.). Los

Angeles, London, New Delhi, Singapoore, Washongton DC: SAGE.

- Flyvbjerg, B. (2006). Five Misunderstandings About Case-Study Research. *Qualitative Inquiry*, 12(2), 219–245.
- Fox, M. S. (1992). The TOVE Project - Towards a Common-Sense Model of the Enterprise. In *Industrial and Engineering Applications of Artificial Intelligence and Expert Systems. 5th International Conference (IEA/AIE 1992)* (pp. 25–34). Berlin, Heidelberg: Springer.
- Frank, U. (1999). Component Ware: Software-technische Konzepte und Perspektiven für die Gestaltung betrieblicher Informationssysteme. *Information Management & Consulting*, 14(2), 11–18.
- Frank, U. (2006). *Towards a Pluralistic Conception of Research Methods in Information Systems Research*. Essen.
- Frischmuth, P., Auer, S., Tramp, S., Unbehauen, J., Holzweißig, K., & Marquardt, C.-M. (2013). Towards Linked Data based Enterprise Information Integration. In *WaSABi'13 Proceedings of the 2013th International Conference on Semantic Web Enterprise Adoption and Best Practice* (pp. 26–34).
- Frischmuth, P., Klímek, J., Auer, S., Tramp, S., Unbehauen, J., Holzweißig, K., & Marquardt, C.-M. (2012). Linked Data in Enterprise Information Integration. *Semantic Web*.
- Furrer, F. J. (2014). Eine kurze Geschichte der Ontologie: Von der Philosophie zur modernen Informatik. *Informatik-Spektrum*, 37(4), 308–317.
- Gagnon, M. (2007). Ontology-Based Integration of Data Sources. In *10th International Conference on Information Fusion (FUSION 2007)*. IEEE.
- Galkin, M., Auer, S., Kim, H., & Scerri, S. (2016). Integration Strategies for Enterprise Knowledge Graphs. In *IEEE Tenth International Conference on Semantic Computing (ICSC)* (pp. 242–245). IEEE.
- Galkin, M., Auer, S., & Scerri, S. (2016). Enterprise Knowledge Graphs: A Backbone of Linked Enterprise Data. In *IEEE/WIC/ACM International Conference on Web Intelligence* (pp. 497–502). IEEE.
- Galkin, M., Auer, S., Vidal, M.-E., & Scerri, S. (2017). Enterprise Knowledge Graphs: A Semantic Approach for Knowledge Management in the Next Generation of Enterprise Information Systems. In *19th International Conference on Enterprise Information Systems* (pp. 88–98). SCITEPRESS.
- Gebel-Sauer, B. (2017). *Anwender-Umfrage: Informationsmanagement und Informationsintegration mit dem SoNBO-Ansatz in Unternehmen. CEIR-Forschungsbericht* (Vol. 02). Universität Koblenz-Landau.
- Gebel-Sauer, B., & Grams, S. (2019). *Fallstudie Kosmos: KosmoNet - Ablösung des Intranets durch IBM Connections. Unveröffentlichter Arbeitsbericht CEIR*. Universität Koblenz-Landau.
- Gebel-Sauer, B., & Schubert, P. (2016). *Fallstudie Vössing: Social und Digital Workplace mit IBM Connections. Unveröffentlichter Arbeitsbericht CEIR*. Universität Koblenz-Landau.
- Gebel-Sauer, B., & Schubert, P. (2019a). Entwicklung einer Definition für Social Business Objects (SBO) zur Modellierung von Unternehmensinformationen. In *International Conference on Wirtschaftsinformatik (WI)* (pp. 199–213).
- Gebel-Sauer, B., & Schubert, P. (2019b). ERP objects become “social”: a process model for the identification of social business objects for an enterprise knowledge graph. In *Pre-ICIS SIGDSA Symposium on Inspiring mindset for Innovation with Business Analytics and Data Science* (pp. 1–12). München.
- Gericke, A., & Winter, R. (2009). Entwicklung eines Bezugsrahmens für Konstruktionsforschung und Artefaktkonstruktion in der gestaltungsorientierten Wirtschaftsinformatik. In J. Becker, H.

- Krcmar, & B. Niehaves (Eds.), *Wissenschaftstheorie und gestaltungsorientierte Wirtschaftsinformatik* (pp. 195–210). Dordrecht, Heidelberg, London, New York: Springer.
- Gewehr, B., Gebel-Sauer, B., & Schubert, P. (2017). Social Network of Business Objects (SoNBO): An Innovative Concept for Information Integration in Enterprise Systems. *Procedia Computer Science*, 121, 904–912.
- Giese, M., Soylu, A., Vega-Gorgojo, G., Waaler, A., Haase, P., Jimenez-Ruiz, E., ... Rosati, R. (2015). Optique: Zooming in on Big Data. *Computer*, 48(3), 60–67.
- Glitsch, J. H., & Schubert, P. (2017). IRESS: Identification of Requirements for Enterprise Social Software. In *Procedia Computer Science* (pp. 866–873). Barcelona: Elsevier B.V.
- Gomez-Perez, J. M., Pan, J. Z., Vetere, G., & Wu, H. (2017). Enterprise Knowledge Graph: An Introduction. In J. Z. Pan, G. Vetere, J. M. Gomez-Perez, & H. Wu (Eds.), *Exploiting Linked Data and Knowledge Graphs in Large Organisations* (pp. 1–15). Cham: Springer.
- Gorjatshev, K. (2019). *Bachelor-Arbeit: Entwicklung einer Unternehmensontologie am Beispiel von Microsoft Dynamics NAV und der Fallstudie Müllerland*. Universität Koblenz-Landau.
- Götz, F. (2018). *Development of a prototype for integrating multiple software systems with a user interface for graph-based navigation*. MSc-Arbeit. Universität Koblenz-Landau.
- Götz, F., & Gebel-Sauer, B. (2018). Vom CRM-System ins Social Network of Business Objects (SoNBO): Entwicklung eines Prototyps für eine innovative Informationsintegration. In *Multikonferenz Wirtschaftsinformatik (MKWI) 2018* (pp. 1995–2006).
- Gradmann, S. (2013). Semantic Web und Linked Open Data. In R. Kuhlen, W. Semar, & D. Strauch (Eds.), *Grundlagen der praktischen Information und Dokumentation. Handbuch zur Einführung in die Informationswissenschaft und -praxis* (6th ed., pp. 218–227). Berlin, Boston: De Gruyter Saur.
- Grimm, S., Hitzler, P., & Abecker, A. (2007). Knowledge Representation and Ontologies. In R. Studer, S. Grimm, & A. Abecker (Eds.), *Semantic Web Services. Concepts, Technologies and Applications* (pp. 51–106). Berlin, Heidelberg, New York: Springer.
- Gruber, T. R. (1993). A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*, 5(2), 199–220.
- Gruninger, M., & Fox, M. S. (1995). Methodology for the Design and Evaluation of Ontologies. In *Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing*.
- Guarino, N. (1997). Semantic Matching: Formal Ontological Distinctions for Information Organization, Extraction, and Integration. In M. T. Pazienza (Ed.), *Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology* (pp. 139–170). Berlin, Heidelberg: Springer.
- Guarino, N., Oberle, D., & Staab, S. (2009). What Is an Ontology? In S. Staab & R. Studer (Eds.), *Handbooks on Ontologies* (2nd ed., pp. 1–20). Dordrecht, Heidelberg, London, New York: Springer.
- Gutwin, C., & Greenberg, S. (1996). Workspace awareness for groupware. In *Conference on Human Factors in Computing Systems - Proceedings* (pp. 208–209).
- Haas, J., & Malang, T. (2010). Beziehungen und Kanten. In C. Stegbauer & R. Häußling (Eds.), *Handbuch Netzwerkforschung* (pp. 89–98). Wiesbaden: VS Verlag für Sozialwissenschaften.
- Haase, P. (2017). Metaphactory. Retrieved December 4, 2017, from <https://metaphacts.com/product>
- Haase, P., Herzig, D. M., Kozlov, A., Nikolov, A., & Trame, J. (2019). metaphactory: A platform for knowledge graph management. *Semantic Web*, 10(6), 1109–1125.
- Hart, C. (1998). *Doing a Literature Review: Releasing the Social Science Research Imagination*. London, Thousand Oaks, New Delhi: SAGE.

- Hart, C. (2001). *Doing a Literature Search: A Comprehensive Guide for the Social Sciences* (1st ed.). London: SAGE.
- Harth, A. (2009). VisiNav: Visual Web Data Search and Navigation. In *20th International Conference on Database and Expert Systems Applications* (pp. 214–228). Berlin, Heidelberg: Springer.
- Havel, J.-M., Steinhorst, M., Dietrich, H.-A., & Delfmann, P. (2014). Supporting terminological standardization in conceptual models - A plugin for a meta-modelling tool. In *ECIS 2014 Proceedings - Twenty Second European Conference on Information Systems*.
- He, W., & Xu, L. Da. (2014). Integration of Distributed Enterprise Applications: A Survey. *IEEE Transactions on Industrial Informatics*, 10(1), 35–42.
- Heath, T., & Bizer, C. (2011). *Linked Data: Evolving the Web into a Global Data Space*. Morgan & Claypool.
- Hepp, M. (2008). GoodRelations: An Ontology for Describing Products and Services Offers on the Web. In *Knowledge Engineering: Practice and Patterns. 16th International Conference, EKAW 2008* (pp. 329–346). Berlin, Heidelberg: Springer.
- Herden, O., & Redenz, K. (2012). Open Source Graphdatenbanken - Konzepte und Klassifikation. In *Informatik 2012, 42. Jahrestagung der Gesellschaft für Informatik e.V. (GI)* (pp. 334–346).
- Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information Systems Research. *MIS Quarterly*, 28(1), 75–105.
- Hildebrand, K. (1990). Was sind Software Tools? In K. Hildebrand (Ed.), *Software Tools: Automatisierung im Software Engineering* (Vol. 42, pp. 11–38). Berlin, Heidelberg: Springer.
- Hislop, D. (2005). *Knowledge Management in Organizations. A Critical Introduction*. Oxford, New York: Oxford University Press.
- Hoede, C. (1995). On the ontology of knowledge graphs. In *Conceptual Structures: Applications, Implementation and Theory. Third International Conference on Conceptual Structures, ICCS '95* (pp. 308–322). Berlin, Heidelberg: Springer.
- Holten, R. (2003). Integration von Informationssystemen. *Wirtschaftsinformatik*, 45(1), 41–52.
- Hu, B., & Svensson, G. (2010). A Case Study of Linked Enterprise Data. In P. F. Patel-Schneider, Y. Pan, P. Hitzler, P. Mika, L. Zhang, J. Z. Pan, ... B. Glimm (Eds.), *9th International Semantic Web Conference* (pp. 129–144). Berlin, Heidelberg: Springer.
- Huff, A. S. (2009). Literature Review. In *Designing Research for Publication* (pp. 147–179). London: SAGE.
- Humm, B., Ege, B., & Reibold, A. (2015). Corporate Semantic Web. In B. Ege, B. Humm, & A. Reibold (Eds.), *Corporate Semantic Web: Wie semantische Anwendungen in Unternehmen Nutzen stiften* (pp. 1–9). Berlin, Heidelberg: Springer Vieweg.
- Huynh, D. F., & Karger, D. R. (2009). Parallax and Companion: Set-based Browsing for the Data Web. In *International World Wide Web Conference*.
- Jalali, S., & Wohlin, C. (2012). Systematic Literature Studies: Database Searches vs. Backward Snowballing. In *IEEE International Symposium on Empirical Software Engineering and Measurement* (pp. 29–38). New York: IEEE.
- Jetschni, J., & Meister, V. G. (2017). Schema Engineering for Enterprise Knowledge Graphs: A Reflecting Survey and Case Study. In *2017 Eighth International Conference on Intelligent Computing and Information Systems (ICICIS)* (pp. 271–277). IEEE.
- Jung, R. (2006). *Architekturen zur Datenintegration. Gestaltungsempfehlungen auf der Basis fachkonzeptueller Anforderungen*. Wiesbaden: Deutscher Universitäts-Verlag.

- Junglas, I., Niehaves, B., Spiekermann, S., Stahl, B. C., Weitzel, T., Winter, R., & Baskerville, R. (2011). The inflation of academic intellectual capital: the case for design science research in Europe. *European Journal of Information Systems*, 20(1), 1–6.
- Kharlamov, E., Hovland, D., Jimenez-Ruiz, E., Lanti, D., Lie, H., Pinkel, C., ... Horrocks, I. (2015). Ontology Based Access to Exploration Data at Statoil. In M. Arenas, O. Corcho, E. Simperl, M. Strohmaier, M. D'Aquin, K. Srinivas, ... S. Staab (Eds.), *14th International Semantic Web Conference* (pp. 93–112). Cham: Springer.
- Kharlamov, E., Jimenez-Ruiz, E., Pinkel, C., Rezk, M., Skjæveland, M. G., Soyulu, A., ... Waaler, A. (2015). OPTIQUE: Ontology-Based Data Access Platform. In *Proceedings of the International Semantic Web Conference 2015 Posters & Demonstrations Track*.
- Kharlamov, E., Jimenez-Ruiz, E., Zheleznyakov, D., Bilidas, D., Giese, M., Haase, P., ... Waaler, A. (2013). Optique: Towards OBDA Systems for Industry. In *The Semantic Web: ESWC 2013 Satellite Events* (pp. 125–140). Berlin, Heidelberg: Springer.
- Kim, H., Lee, J., & Yang, J. (2015). Towards an Enterprise Entity Hub: Integration of General and Enterprise Knowledge. In G. Qi, K. Kozaki, J. Z. Pan, & S. Yu (Eds.), *Semantic Technology. 5th Joint International Conference* (pp. 301–310). Cham: Springer.
- Kitchenham, B. A., & Charters, S. M. (2007). *Guidelines for performing Systematic Literature Reviews in Software Engineering: Version 2.3* (EBSE Technical Report). Keele University and University of Durham.
- Kobilarov, G., & Dickinson, I. (2008). Humboldt: Exploring Linked Data. In *Proceedings of the WWW2008 Workshop on Linked Data on the Web*.
- Kogalovsky, M. R. (2012). Ontology-Based Data Access Systems. *Programming and Computer Software*, 38(4), 167–182.
- Kontchakov, R., Lutz, C., Toman, D., Wolter, F., & Zakharyashev, M. (2011). The Combined Approach to Ontology-Based Data Access. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence* (pp. 2656–2661). AAAI Press.
- Kontchakov, R., Rodríguez-Muro, M., & Zakharyashev, M. (2013). Ontology-Based Data Access with Databases: A Short Course. In S. Rudolph, G. Gottlob, I. Horrocks, & F. van Harmelen (Eds.), *Reasoning Web. Semantic Technologies for Intelligent Data Access. 9th International Summer School 2013* (pp. 194–229). Berlin, Heidelberg: Springer.
- Krcmar, H. (2015). *Einführung in das Informationsmanagement* (2nd ed.). Berlin, Heidelberg: Springer.
- Krötzsch, M., & Thost, V. (2016). Ontologies for Knowledge Graphs: Breaking the Rules. In P. Groth, E. Simperl, A. Gray, M. Sabou, M. Krötzsch, F. Lecue, ... Y. Gil (Eds.), *15th International Semantic Web Conference* (pp. 376–392). Cham: Springer.
- Krötzsch, M., & Weikum, G. (2016). Editorial. *Journal of Web Semantics*, 37–38, 53–54.
- Kuechler, W., & Vaishnavi, V. K. (2008). On theory development in design science research: anatomy of a research project. *European Journal of Information Systems*, 17(5), 489–504.
- Kvale, S., & Brinkmann, S. (2009). *Interviews: Learning the Craft of Qualitative Research Interviewing* (2nd ed.). Thousand Oaks, London, New Delhi, Singapore: SAGE.
- Leser, U., & Naumann, F. (2007). *Informationsintegration: Architekturen und Methoden zur Integration verteilter und heterogener Datenquellen*. Heidelberg: dpunkt.verlag.
- Liebhart, D. (2007). *SOA goes real. Serviceorientierte Architekturen erfolgreich planen und einführen*. München, Wien: Hanser.
- Lima, J. F., Amaral, C. M. G., & Molinaro, L. F. R. (2010). Ontology: An Analysis of the Literature. In *ENTERprise Information Systems. International Conference, CENTERIS 2010* (Vol. 110, pp. 426–

- 435). Berlin, Heidelberg: Springer.
- Linthicum, D. S. (2000). *Enterprise Application Integration* (1st ed.). Boston: Addison-Wesley.
- Linthicum, D. S. (2003). *Next Generation Application Integration: From Simple Information to Web Services* (1st ed.). Boston: Addison-Wesley.
- Linthicum, D. S. (2009). *Cloud Computing and SOA Convergence in Your Enterprise: A Step-by-Step Guide* (1st ed.). Upper Saddle River, NJ: Addison-Wesley.
- Madni, A. M., Lin, W., & Madni, C. C. (2001). IDEON: An Extensible Ontology for Designing, Integrating, and Managing Collaborative Distributed Enterprises. *Systems Engineering*, 4(1), 35–48.
- Maedche, A., Staab, S., & Studer, R. (2001). Ontologien. *Wirtschaftsinformatik*, 43(4), 393–395.
- Maynard, M. (1994). Methods, Practice and Epistemology: The Debate about Feminism and Research. In M. Maynard & J. Purvis (Eds.), *Researching women's lives from a feminist perspective* (pp. 10–26). London: Taylor & Francis.
- McCusker, J. P., McGuinness, D. L., Erickson, J. S., & Chastain, K. (2016). What is a Knowledge Graph? Retrieved October 12, 2017, from https://www.authorea.com/users/6341/articles/107281-what-is-a-knowledge-graph/_show_article
- McGuinness, D. L., & van Harmelen, F. (2004). OWL Web Ontology Language Overview. Retrieved September 16, 2017, from <https://www.w3.org/TR/owl-features/>
- Meier, S. (2019). *Entwicklung eines Prototyps zur Visualisierung eines unternehmensspezifischen SoNBO (Knowledge Graphen) am Beispiel von CRM-Informationen*. BSc-Arbeit. Universität Koblenz-Landau.
- Meier, S., Gebel-Sauer, B., & Schubert, P. (2020). Knowledge Graph for the Visualisation of CRM Objects in a Social Network of Business Objects (SoNBO): Development and Evaluation of the SoNBO Visualiser. In *Procedia Computer Science*.
- Miao, Q., Meng, Y., & Zhang, B. (2015). Chinese Enterprise Knowledge Graph Construction based on Linked Data. In *Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015)* (pp. 153–154). IEEE.
- Miles, M. B., Huberman, A. M., & Saldaña, J. (2014). *Qualitative Data Analysis: A Methods Sourcebook* (3rd ed.). Los Angeles, London, New Delhi, Singapore, Washington DC: SAGE.
- Nagypál, G. (2007). Ontology Development. In R. Studer, S. Grimm, & A. Abecker (Eds.), *Semantic Web Services. Concepts, Technologies and Applications* (pp. 107–134). Berlin, Heidelberg, New York: Springer.
- Nett, B., & Stevens, G. (2009). Business Ethnography - Aktionsforschung als Beitrag zu einer reflexiven Textgestaltung. In J. Becker, H. Krcmar, & B. Niehaves (Eds.), *Wissenschaftstheorie und gestaltungsorientierte Wirtschaftsinformatik* (pp. 43–68). Dordrecht, Heidelberg, London, New York: Springer.
- Newcomer, E., & Lomow, G. (2004). *Understanding SOA with Web Services* (1st ed.). Upper Saddle River: Addison Wesley Professional.
- Nogales, A., Sicilia, M.-A., García-Barriocanal, E., & Sánchez-Alonso, S. (2013). Exploring the potential for mapping Schema.org microdata and the Web of Linked Data. In *Metadata and Semantics Research. 7th Research Conference, MTSR 2013* (pp. 266–276). Cham: Springer.
- Noy, N. F. (2004). Ontology Mapping. In S. Staab & R. Studer (Eds.), *Handbook on Ontologies* (2nd ed., pp. 573–590). Berlin, Heidelberg: Springer.
- Noy, N. F., & McGuinness, D. L. (2001). Ontology Development 101: A Guide to Creating Your First Ontology.

- Nunamaker, J. F., Briggs, R. O., Derrick, D. C., & Schwabe, G. (2015). The Last Research Mile: Achieving Both Rigor and Relevance in Information Systems Research. *Journal of Management Information Systems*, 32(3), 10–47.
- Nurdiati, S., & Hoede, C. (2008). 25 years development of knowledge graph theory: the results and the challenge. *Memorandum*, 2(1876).
- Nüttgens, M., & Zimmermann, V. (1998). Geschäftsprozeßmodellierung mit der objektorientierten Ereignisgesteuerten Prozeßkette (oEPK). In M. Maicher & H.-J. Scheruhn (Eds.), *Informationsmodellierung. Referenzmodelle und Werkzeuge* (pp. 23–35). Wiesbaden: Springer.
- Okoli, C., & Schabram, K. (2010). A Guide to Conducting a Systematic Literature Review of Information Systems Research. *Sprouts: Working Papers on Information*, 10(26).
- Orlikowski, W. J., & Baroudi, J. J. (1991). Studying Information Technology in Organizations: Research Approaches and Assumptions. *Information Systems Research*, 2(1), 1–28.
- Österle, H., Becker, J., Frank, U., Hess, T., Karagiannis, D., Krcmar, H., ... Sinz, E. J. (2011). Memorandum on design-oriented information systems research. *European Journal of Information Systems*, 20(1), 7–10.
- Paulheim, H. (2017). Knowledge Graph Refinement: A Survey of Approaches and Evaluation Methods. *Semantic Web*, 8(3), 489–508.
- Peppers, K., Rothenberger, M., Tuunanen, T., & Vaezi, R. (2012). Design Science Research Evaluation. In K. Peppers, M. Rothenberger, & B. Kuechler (Eds.), *Design Science Research in Information Systems. Advances in Theory and Practice. 7th International Conference, DESRIST 2012* (pp. 398–410). Berlin, Heidelberg: Springer.
- Peppers, K., Tuunanen, T., & Niehaves, B. (2018). Design science research genres: introduction to the special issue on exemplars and criteria for applicable design science research. *European Journal of Information Systems*, 27(2), 129–139.
- Peirce, C. S. (1885a). On the Algebra of Logic: A Contribution to the Philosophy of Notation. *American Journal of Mathematics*, 7(2), 180–196.
- Peirce, C. S. (1885b). On the Algebra of Logic [Continued]. *American Journal of Mathematics*, 7(3), 197–202.
- Pelletier, C., Goossenaerts, J., & Szirbik, N. (2001). A centralized translation interface based on the PSIM ontology. In *Proceedings of the 8th International Conference on Emerging Technologies and Factory Automation (ETFA 2001)* (pp. 653–656). IEEE.
- Poggi, A., Lembo, D., Calvanese, D., de Giacomo, G., Lenzerini, M., & Rosati, R. (2008). Linking Data to Ontologies. In S. Spaccapietra (Ed.), *Journal on Data Semantics X* (pp. 133–173). Berlin, Heidelberg, New York: Springer.
- Psyllidis, A. (2015). Ontology-Based Data Integration from Heterogeneous Urban Systems: A Knowledge Representation Framework for Smart Cities. In *The 14th International Conference on Computers in Urban Planning and Urban Management (CUPUM 2015)*.
- Pujara, J., Miao, H., Getoor, L., & Cohen, W. (2013). Knowledge Graph Identification. In H. Alani, L. Kagal, A. Fokoue, P. Groth, C. Biemann, J. X. Parreira, ... K. Janowicz (Eds.), *12th International Semantic Web Conference* (pp. 542–557). Heidelberg, New York, Dordrecht, London: Springer.
- Quilitz, B., & Leser, U. (2008). Querying Distributed RDF Data Sources with SPARQL. In S. Bechhofer, M. Hauswirth, J. Hoffmann, & M. Koubarakis (Eds.), *The Semantic Web: Research and Applications. 5th European Semantic Web Conference, ESWC 2008* (pp. 524–538). Berlin, Heidelberg: Springer.
- Rahm, E., & Bernstein, P. A. (2001). A survey of approaches to automatic schema matching. *VLDB Journal*, 10(4), 334–350.

- Randolph, J. J. (2009). A Guide to Writing the Dissertation Literature Review. *Practical Assessment Research and Evaluation*, 14(13), 1–13.
- Riedle, M. (2018). *Entwicklung einer Applikation auf Basis eines Knowledge Graphen für Objekte in Navigation und Realisierung eines Integrationsinterface mit dem SoNBO-Ansatz*. MSc-Arbeit. Universität Koblenz-Landau.
- Rodríguez-Muro, M., & Calvanese, D. (2011). Dependencies: Making Ontology Based Data Access work in practice. In *5th Alberto Mendelzon International Workshop on Foundations of Data Management (AMW2011)*.
- Rodríguez-Muro, M., Kontchakov, R., & Zakharyashev, M. (2013). Ontology-Based Data Access: Ontop of Databases. In H. Alani, L. Kagal, A. Fokoue, P. Groth, C. Biemann, J. X. Parreira, ... K. Janowicz (Eds.), *12th International Semantic Web Conference* (pp. 558–573). Berlin, Heidelberg: Springer.
- Rodríguez-Muro, M., & Rezk, M. (2015). Efficient SPARQL-to-SQL with R2RML mappings. *Journal of Web Semantics*, 33, 141–169.
- Rosemann, M., & Schütte, R. (1998). Multiperspektivische Referenzmodellierung. In J. Becker, M. Rosemann, & R. Schütte (Eds.), *Referenzmodellierung: State-of-the-Art und Entwicklungsperspektiven* (pp. 22–44). Berlin, Heidelberg: Springer.
- Ruan, T., Xue, L., Wang, H., Hu, F., Zhao, L., & Ding, J. (2016). Building and Exploring an Enterprise Knowledge Graph for Investment Analysis. In P. Groth, E. Simperl, A. Gray, M. Sabou, M. Krötzsch, F. Lecue, ... Y. Gil (Eds.), *15th International Semantic Web Conference* (pp. 418–436). Cham: Springer.
- Sabou, M. (2016). An Introduction to Semantic Web Technologies. In *Semantic Web Technologies for Intelligent Engineering Applications* (pp. 53–81). Cham: Springer.
- Sabou, M., Ekaputra, F. J., Ionescu, T., Musil, J., Schall, D., Haller, K., ... Biffi, S. (2018). Exploring Enterprise Knowledge Graphs: A Use Case in Software Engineering. In A. Gangemi, R. Navigli, M.-E. Vidal, P. Hitzler, R. Troncy, L. Hollink, ... M. Alam (Eds.), *The Semantic Web. 15th International Conference, ESWC 2018* (pp. 560–575). Cham: Springer.
- Sack, H. (2014). Linked Data Technologien - Ein Überblick. In T. Pellegrini, H. Sack, & S. Auer (Eds.), *Linked Enterprise Data* (pp. 21–62). Berlin, Heidelberg: Springer.
- Scheer, A.-W. (1996). Architekturen für das Information Engineering. In H. Heilmann, L. J. Heinrich, & F. Roithmayr (Eds.), *Information Engineering: Wirtschaftsinformatik im Schnittpunkt von Wirtschafts-, Sozial- und Ingenieurwissenschaften* (pp. 235–260). Berlin, Boston: De Gruyter.
- Scheer, A.-W. (1998). *ARIS — Modellierungsmethoden, Metamodelle, Anwendungen* (3rd ed.). Berlin, Heidelberg: Springer.
- Scheer, A.-W. (2002). *ARIS — Vom Geschäftsprozess zum Anwendungssystem* (4th ed.). Berlin, Heidelberg: Springer.
- Scheer, A.-W., Nüttgens, M., & Zimmermann, V. (1997). *Objektorientierte Ergebnisgesteuerte Prozeßkette (oEPK) - Methode und Anwendung* -. Saarbrücken.
- Schelp, J., & Winter, R. (2002). Enterprise Portals und Enterprise Application Integration. *HMD Praxis Der Wirtschaftsinformatik*, 225, 6–20.
- Schmidt-Volkmar, P. (2008). *Betriebswirtschaftliche Analyse auf operationalen Daten*. Gabler Edition Wissenschaft. Wiesbaden: Gabler.
- Schmidt, W., Fleischmann, A., & Gilbert, O. T. (2009). Subjektorientiertes Geschäftsprozessmanagement. *HMD Praxis Der Wirtschaftsinformatik*, 46(2), 52–62.
- Schubert, P. (2003). E-Business-Integration. In P. Schubert, R. Wölfle, & W. Dettling (Eds.), *E-Business-Integration: Fallstudien zur Optimierung elektronischer Geschäftsprozesse* (pp. 1–22). München,

Wien: Hanser.

- Schubert, P. (2018). Joint Work and Information Sharing in the Modern Digital Workplace: How the Introduction of “Social” Features Shaped Enterprise Collaboration Systems. In K. Riemer, S. Schellhammer, & M. Meinert (Eds.), *Collaboration in the Digital Age: How Technology Enables Individuals, Teams and Businesses* (pp. 45–59). Berlin, Heidelberg: Springer.
- Schubert, P., & Glitsch, J. H. (2015). Adding Structure to Enterprise Collaboration Systems: Identification of Use Cases and Collaboration Scenarios. In *Procedia Computer Science* (Vol. 64, pp. 161–169).
- Schubert, P., & Glitsch, J. H. (2016). Use Cases and Collaboration Scenarios: How employees use socially-enabled Enterprise Collaboration Systems (ECS). *International Journal of Information Systems and Project Management*, 4(2), 41–62.
- Schubert, P., & Williams, S. (2020). IndustryConnect: WI-Forschung mit und für die Praxis. *HMD - Praxis Der Wirtschaftsinformatik*, 1–16.
- Schubert, P., & Winkelmann, A. (2020). *Betriebswirtschaftliche Anwendungssysteme (Fokus: Enterprise Resource Planning)*.
- Schumann, M. (1992). *Betriebliche Nutzeffekte und Strategiebeiträge der großintegrierten Informationsverarbeitung*. Berlin, Heidelberg, New York, London, Paris, Tokyo, Hong Kong, Barcelona, Budapest: Springer-Verlag.
- Schwade, F., & Schubert, P. (2017). Social Collaboration Analytics for Enterprise Collaboration Systems: Providing Business Intelligence on Collaboration Activities. In *Proceedings of the 50th Hawaii International Conference on System Sciences (HICSS)* (pp. 401–410).
- Schwarze, J. (2000). *Einführung in die Wirtschaftsinformatik* (5th ed.). Herne, Berlin: NWB.
- Seale, C. (1999). Quality in Qualitative Research. *Qualitative Inquiry*, 5(4), 465–478.
- Semantic Arts. (2019). gist.
- Sequeda, J. F. (2017). Ontology Based Data Access: Where do the Ontologies and Mappings come from? In J. L. Reutter & D. Srivastava (Eds.), *Proceedings of the 11th Alberto Mendelzon International Workshop on Foundations of Data Management and the Web*.
- Sequeda, J. F., & Miranker, D. P. (2017). A Pay-As-You-Go Methodology for Ontology-Based Data Access. *IEEE Internet Computing*, 21(2), 92–96.
- Seubert, M. (1998). Business-Objekte und objektorientiertes Prozeßdesign. In J. Becker, M. Rosemann, & R. Schütte (Eds.), *Referenzmodellierung: State-of-the-Art und Entwicklungsperspektiven* (pp. 107–128). Berlin, Heidelberg: Springer.
- Sharma, S., & Osei-Bryson, K.-M. (2008). Organization-Ontology Based Framework for Implementing the Business Understanding Phase of Data Mining Projects. In *Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS 2008)*. New York: IEEE.
- Sheth, A. (2016). Internet of Things to Smart IoT Through Semantic, Cognitive, and Perceptual Computing. *IEEE Intelligent Systems*, 31(2), 108–112.
- Sims, O. (1995). The OMG Business Object Facility and the OMG Business Object. In J. Sutherland, D. Patel, C. Casanave, G. Hollowell, & J. Miller (Eds.), *Business Object Design and Implementation* (pp. 37–46). Berlin, Heidelberg, New York: Springer.
- Singhal, A. (2012). Introducing the Knowledge Graph: things, not strings. Retrieved October 20, 2017, from <https://googleblog.blogspot.de/2012/05/introducing-knowledge-graph-things-not.html>
- Smith, B., & Welty, C. (2001). Ontology: Towards a New Synthesis. In *Proceedings of the international conference on Formal Ontology in Information Systems (FOIS '01)* (pp. iii–ix). New York: ACM.

- Song, D., Schilder, F., Hertz, S., Saltini, G., Smiley, C., Nivarthi, P., ... Bennett, D. (2017). Building and Querying an Enterprise Knowledge Graph. *IEEE Transactions on Services Computing*.
- Sowa, J. F. (1976). Conceptual Graphs for a Data Base Interface. *IBM Journal of Research and Development*, 20(4), 336–357.
- Sowa, J. F. (1984). *Conceptual structures: Information processing in mind and machine*. Longman Higher Education.
- Sowa, J. F. (2008). Conceptual Graphs. In F. van Harmelen, V. Lifschitz, & B. Porter (Eds.), *Handbook of Knowledge Representation* (1st ed., pp. 213–237). Amsterdam, Kidlington: Elsevier.
- Soylu, A., & Giese, M. (2016). Qualifying Ontology-based Visual Query Formulation. In *Flexible Query Answering Systems 2015. Proceedings of the 11th International Conference FQAS 2015* (pp. 243–255). Cham: Springer.
- Soylu, A., Giese, M., Jimenez-Ruiz, E., Kharlamov, E., Zheleznyakov, D., & Horrocks, I. (2017). Ontology-based end-user visual query formulation: Why, what, who, how, and which? *Universal Access in the Information Society*, 16(2), 435–467.
- Soylu, A., Giese, M., Jimenez-Ruiz, E., Vega-Gorgojo, G., & Horrocks, I. (2016). Experiencing OptiqueVQS: a multi-paradigm and ontology-based visual query system for end users. *Universal Access in the Information Society*, 15(1), 129–152.
- Soylu, A., Kharlamov, E., Zheleznyakov, D., Jimenez-Ruiz, E., Giese, M., & Horrocks, I. (2015). Ontology-Based Visual Query Formulation: An Industry Experience. In *Advances in Visual Computing. 11th International Symposium, ISVC 2015* (pp. 842–854). Cham: Springer.
- Soylu, A., Kharlamov, E., Zheleznyakov, D., Jimenez-Ruiz, E., Giese, M., Skjæveland, M. G., ... Horrocks, I. (2018). OptiqueVQS: A visual query system over ontologies for industry. *Semantic Web*, 9(5), 627–660.
- Spahn, M., Kleb, J., Grimm, S., & Scheidl, S. (2008). Supporting Business Intelligence by Providing Ontology-Based End-User Information Self-Service. In *Proceedings of the First International Workshop on Ontology-supported Business Intelligence (OBI 2008)*. New York: ACM.
- Staab, S. (2002). Wissensmanagement mit Ontologien und Metadaten. *Informatik-Spektrum*, 25(3), 194–209.
- Staab, S., Angele, J., Decker, S., Erdmann, M., Hotho, A., Maedche, A., ... Sure, Y. (2000). Semantic community Web portals. *Computer Networks*, 33(1–6), 473–491.
- Staab, S., & Schnurr, H.-P. (2002). Knowledge and Business Processes: Approaching an Integration. In R. Dieng-Kuntz & N. Matta (Eds.), *Knowledge Management and Organizational Memories* (pp. 75–88). Boston, MA: Springer.
- Staab, S., Studer, R., Schnurr, H.-P., & Sure, Y. (2001). Knowledge processes and ontologies. *IEEE Intelligent Systems*, 16(1), 26–34.
- Stachowiak, H. (1973). *Allgemeine Modelltheorie*. Wien, New York: Springer-Verlag.
- Stokman, F. N., & de Vries, P. H. (1988). Structuring Knowledge in a Graph. In G. C. van der Veer & G. Mulder (Eds.), *Human-Computer Interaction* (pp. 186–206). Berlin, Heidelberg: Springer.
- Stuckenschmidt, H. (2011). *Ontologien: Konzepte, Technologien und Anwendungen*. Berlin, Heidelberg: Springer.
- Studer, R., Benjamins, V. R., & Fensel, D. (1998). Knowledge engineering: Principles and methods. *Data and Knowledge Engineering*, 25(1–2), 161–197.
- Suárez-Figueroa, M. C., Gómez-Pérez, A., Motta, E., & Gangemi, A. (2012). Introduction: Ontology Engineering in a Networked World. In M. C. Suárez-Figueroa, A. Gómez-Pérez, E. Motta, & A.

- Gangemi (Eds.), *Ontology Engineering in a Networked World* (pp. 1–6). Berlin, Heidelberg: Springer.
- Sutherland, J., Casanave, C., Miller, J., Patel, D., & Hollowell, G. (Eds.). (1995). *Business Object Design and Implementation. OOPSLA '95 Workshop Proceedings*. London: Springer.
- ter Hofstede, A. H. M., Proper, H. A., & van der Weide, T. P. (1996). Query Formulation as an Information Retrieval Problem. *The Computer Journal*, 39(4), 255–274.
- Thomas, G. (2011). A Typology for the Case Study in Social Science Following a Review of Definition, Discourse, and Structure. *Qualitative Inquiry*, 17(6), 511–521.
- Thomas, O. (2005). *Das Modellverständnis in der Wirtschaftsinformatik: Historie, Literaturanalyse und Begriffsexplikation. Veröffentlichungen des Instituts für Wirtschaftsinformatik*. Saarbrücken.
- Tittmann, P. (2019). *Graphentheorie: Eine anwendungsorientierte Einführung* (3rd ed.). München: Carl Hanser Verlag.
- Tomaszuk, D. (2016). RDF Data in Property Graph Model. In E. Garoufallou, I. S. Coll, A. Stellato, & J. Greenberg (Eds.), *Metadata and Semantics Research. 10th International Conference, MTSR 2016* (pp. 104–115). Cham: Springer.
- Torraco, R. J. (2005). Writing Integrative Literature Reviews: Guidelines and Examples. *Human Resource Development Review*, 4(3), 356–367.
- Tunkelang, D. (2009). *Faceted Search*. Morgan & Claypool.
- Uschold, M. (1996). Building Ontologies: Towards a Unified Methodology. In *16th Annual Conference of the British Computer Society Specialist Group on Expert Systems*.
- Uschold, M. (2018). *Demystifying OWL for the Enterprise*. Morgan & Claypool.
- Uschold, M., & Gruninger, M. (1996). Ontologies: Principles, Methods and Applications. *The Knowledge Engineering Review*, 11(2), 93–136.
- Uschold, M., King, M., Moralee, S., & Zorgios, Y. (1998). The Enterprise Ontology. *The Knowledge Engineering Review*, 13(1), 31–89.
- Vaishnavi, V. K., & Kuechler, W. (2004). Design Science Research in Information Systems. Retrieved January 16, 2017, from <http://www.desist.org/design-research-in-information-systems/>
- Vaishnavi, V. K., & Kuechler, W. (2015). *Design Science Research Methods and Patterns. Design Science Research Methods and Patterns* (2nd ed.). Boca Raton: CRC Press, Taylor & Francis Group.
- Villazon-Terrazas, B., Garcia-Santa, N., Ren, Y., Faraotti, A., Wu, H., Zhao, Y., ... Pan, J. Z. (2017). Knowledge Graph Foundations. In J. Z. Pan, G. Vetere, J. M. Gomez-Perez, & H. Wu (Eds.), *Exploiting Linked Data and Knowledge Graphs in Large Organisations* (pp. 17–55). Cham: Springer.
- Villazon-Terrazas, B., Garcia-Santa, N., Ren, Y., Srinivas, K., Rodríguez-Muro, M., Alexopoulos, P., & Pan, J. Z. (2017). Construction of Enterprise Knowledge Graphs (I). In J. Z. Pan, G. Vetere, J. M. Gomez-Perez, & H. Wu (Eds.), *Exploiting Linked Data and Knowledge Graphs in Large Organisations* (pp. 87–116). Cham: Springer.
- Volz, R., Handschuh, S., Staab, S., Stojanovic, L., & Stojanovic, N. (2004). Unveiling the hidden bride: deep annotation for mapping and migrating legacy data to the Semantic Web. *Journal of Web Semantics*, 1(2), 187–206.
- vom Brocke, J., Simons, A., Niehaves, B., Riemer, K., Plattfaut, R., & Cleven, A. (2009). Reconstructing the Giant: On the Importance of Rigour in Documenting the Literature Search Process. In *European Conference on Information Systems (ECIS 2009)* (pp. 2206–2217).
- vom Brocke, J., Simons, A., Riemer, K., Niehaves, B., Plattfaut, R., & Cleven, A. (2015). Standing on the

- Shoulders of Giants: Challenges and Recommendations of Literature Search in Information Systems Research. *Communications of the Association for Information Systems*, 37(9), 205–224.
- von Rosing, M., & Laurier, W. (2015). An Introduction to the Business Ontology. *International Journal of Conceptual Structures and Smart Applications*, 3(1), 20–41.
- W3C. (2012). OWL 2 Web Ontology Language.
- W3C. (2014a). RDF 1.1 Concepts and Abstract Syntax.
- W3C. (2014b). The Organization Ontology.
- Wache, H., Vögele, T., Visser, U., Stuckenschmidt, H., Schuster, G., Neumann, H., & Hübner, S. (2001). Ontology-Based Integration of Information — A Survey of Existing Approaches. In *International Joint Conference on Artificial Intelligence (Workshop: Ontologies and Information Sharing)* (pp. 108–117).
- Walsham, G. (1995). Interpretive case studies in IS research: nature and method. *European Journal of Information Systems*, 4(2), 74–81.
- Webster, J., & Watson, R. T. (2002). Analyzing the Past to Prepare for the Future: Writing a Literature Review. *MIS Quarterly*, 26(2), xiii–xxiii.
- Weller, K. (2014). Ontologien. In R. Kuhlen, W. Semar, & D. Strauch (Eds.), *Grundlagen der praktischen Information und Dokumentation. Handbuch zur Einführung in die Informationswissenschaft und -praxis* (6th ed., pp. 207–218). Berlin, Boston: De Gruyter.
- Weske, M. (1999). Business-Objekte: Konzepte, Architekturen, Standards. *Wirtschaftsinformatik*, 41(1), 4–11.
- Weske, M. (2012). *Business Process Management: Concepts, Languages, Architectures* (2nd ed.). Berlin, Heidelberg: Springer.
- Wilde, T., & Hess, T. (2006). *Methodenspektrum der Wirtschaftsinformatik: Überblick und Portfoliobildung. Arbeitsbericht 2006/2: Institut für Wirtschaftsinformatik und Neue Medien der Ludwig-Maximilians-Universität München*.
- Wilde, T., & Hess, T. (2007). Forschungsmethoden der Wirtschaftsinformatik. Eine empirische Untersuchung. *Wirtschaftsinformatik*, 49(4), 280–287.
- Williams, S. P. (2011). Das 8C-Modell für kollaborative Technologien. In P. Schubert & M. Koch (Eds.), *Wettbewerbsfaktor Business Software* (pp. 11–21). München: Hanser.
- Williams, S. P., Mosen, J., & Schubert, P. (2020). The Structure of Social Documents. In *53rd Hawaii International Conference on System Sciences* (pp. 2825–2834). Maui, HI, USA.
- Williams, S. P., & Schubert, P. (2017). Connecting Industry: Building and Sustaining a Practice-based Research Community. In *Proceedings of the 50th Hawaii International Conference on System Sciences (HICSS)* (pp. 5400–5409).
- Wohlin, C. (2014). Guidelines for Snowballing in Systematic Literature Studies and a Replication in Software Engineering. In *18th International Conference on Evaluation and Assessment in Software Engineering*. New York: ACM.
- Zaino, J. (2017, October). Onboarding to Enterprise Knowledge Graphs. *DATAVERSITY*.
- Zimmermann, F.-O. (1999). *Betriebliche Informationssysteme in virtuellen Organisationen*. Wiesbaden: Deutscher Universitäts-Verlag.

Anhang

A.1 Organisatorische Daten zur Erhebung des Key Case Vössing

Thema des Meetings	Erläuterung des beim Ingenieurbüro entwickelten, integrierten IT-Systems. Ausgangspunkt waren die Vortragsfolien von der IBM Social Business Conference mit dem Titel "The Social Network of Business Objects".
Datum	12.10.2016
Zeit	13:00 - 16:30 Uhr
Ort	Düsseldorf (Hauptsitz Ingenieurbüro Vössing)
Eingeladene Teilnehmer	BGE, Vössing DKR, Vössing PSC, Universität Koblenz BGS, Universität Koblenz FGO, Universität Koblenz
Leitung	BGE/BGS
Protokollführung	FGO
Sonstiges	Tonbandaufnahme; Transkript

A.2 Dokumentation des Forschungsprozesses

Mon/Jahr	Ereignis	Beschreibung	Beteiligte	Ort	Relevanz für Diss (Kapitel)
Blau: Erkenntnisse aus/für Praxis, Grün: Erkenntnisse aus/für Wissenschaft					
Apr 2016	Industry Connect04	Erste Präsentation des "Self-Service" durch BGE	BGE, PSC, BGS	Koblenz	3, 4
Okt 2016	Erhebung Key Case Vössing (1/2)	Detaillierte Erhebung des Key Case Vössing durch ein fast vierstündiges qualitatives Interview	BGE, PSC, BGS, FGO	Düsseldorf	3, 4
Apr 2017	Industry Connect06	Durchführung World-Café mit IC-Mitgliedern, um den Key Case Vössing mit Fachpublikum zu diskutieren. Leitfragen: (1) Welches Nutzenpotenzial könnte in deinem Unternehmen durch einen „Self-Service“ generiert werden? (2) Welche Hindernisse könnten in deinem Unternehmen einem „Self-Service“ entgegenstehen? (3) Was wären deine zu integrierenden Business Objects im Unternehmen? (4) Woher (aus welchem System) würden deine zu integrierenden Business Objects stammen?	BGE, PSC, BGS + IC-Mitglieder	Koblenz	5
Mai 2017	Einreichung des akzeptierten Papers (1/5)	"Social Network of Business Objects (SoNBO): An Innovative Concept for Information Integration in Enterprise Systems" Verschriftlichung des Key Case Vössing als wissenschaftliche Publikation	BGE, BGS, PSC	Koblenz	3, 4
Mai 2017	DNUG44	Präsentation des Key Case Vössing auf der DNUG44 und Diskussion mit Fachpublikum	PSC, BGS	Berlin	3, 4
Sep 2017	Literaturrecherche (1/3)	Initial durchgeführte Literaturrecherche	BGS	Koblenz	1, 2, 3
Okt 2017	Einreichung des akzeptierten Papers (2/5)	"Vom CRM-System ins Social Network of Business Objects (SoNBO): Entwicklung eines Prototyps für eine innovative Informationsintegration" Verschriftlichung des in der Universität entwickelten SoNBO-Explorers (durch die MA Götz und deren Betreuerin Berit Gebel-Sauer)	FGO, BGS	Koblenz	5
Feb 2018	Einreichung Exposé für die Dissertation	Verschriftlichung des Dissertationsvorhabens	BGS	Koblenz	-

Mon/Jahr	Ereignis	Beschreibung	Beteiligte	Ort	Relevanz für Diss (Kapitel)
Blau: Erkenntnisse aus/für Praxis, Grün: Erkenntnisse aus/für Wissenschaft					
Feb 2018	Erhebung Key Case Vössing (2/2)	Diskussion des in der Universität entwickelten SoNBO-Explorers mit dem IT-Team von Vössing	FGO, BGS, PSC, BGE + IT-Team Vössing	Düsseldorf	3, 4
Apr 2018	MA Götz (Konfigurations-Projekt 1/2)	Entwicklung des SoNBO-Explorers für CRM-Informationen (GEDYS IntraWare 8) als Prototyp	FGO, BGS, PSC	Koblenz	5
Jun 2018	DNUG45	Diskussion über "Business Objects" mit Fachpublikum	BGS + Fachpublikum	Darmstadt	4
Aug 2018	Einreichung des akzeptierten Papers (3/5)	"Entwicklung einer Definition für Social Business Objects (SBO) zur Modellierung von Unternehmensinformationen", Basis: empirische Erhebung des DNUG45-Fachpublikums & Literaturanalyse	BGS, PSC	Koblenz	4
Nov 2018	MA Riedle (Konfigurations-Projekt 2/2)	Entwicklung des SoNBO-Explorers für ERP-Informationen (Microsoft Dynamics NAV) als Prototyp	MRI, BGS, PSC	Koblenz	5
Feb 2019	IBM Roundtable	Präsentation des SoNBO-Konzeptes IBM Connections-Anwender, um einen Evaluationspartner zu gewinnen.	BGS, DGE, BGE	Ehningen	6
Mrz 2019	BA Meier	Entwicklung des SoNBO-Visualiser zur Visualisierung der Unternehmensontologie als Prototyp	BGS	Koblenz	5
Apr 2019	"SoNBO für den KOSMOS" (Kick-off)	Kick-off Meeting für die Evaluation	BGS, PSC, DGE	Koblenz	6
Jun 2019	DNUG46	Präsentation des SoNBO-Visualiser und Diskussion mit Fachpublikum	BGS + Fachpublikum	Essen	5
Jun 2019	Literaturrecherche (2/3)	Strukturierte Literaturrecherche	BGS	Koblenz	3, 4, 5
Aug 2019	"SoNBO für den KOSMOS" (1/4)	1. Meeting für die Evaluation	BGS, PSC, SGR, DGE, VPR	Stuttgart	6
Sep 2019	Einreichung des akzeptierten Papers (4/5)	"ERP objects become 'social': a process model for the identification of social business objects for an enterprise knowledge graph"	BGS, PSC	Koblenz	5
Nov 2019	"SoNBO für den KOSMOS" (2/4)	2. Meeting für die Evaluation	BGS, SMI, PSC, DGE	Stuttgart	6

Mon/Jahr	Ereignis	Beschreibung	Beteiligte	Ort	Relevanz für Diss (Kapitel)
Blau: Erkenntnisse aus/für Praxis, Grün: Erkenntnisse aus/für Wissenschaft					
Dez 2019	Ergänzung zu Literaturrecherche (2/3)	Austausch zu Ontologie-basierten Integrationsansätzen und Knowledge Graph mit Fachexperten Prof. Dr. Stefan Staab	BGS, PSC, SST	Koblenz	3
Jan 2020	Literaturrecherche (3/3)	Berücksichtigung aktueller Publikationen im Rahmen der strukturierten Literaturrecherche	BGS	Koblenz	3, 4, 5
Mrz 2020	"SoNBO für den KOSMOS" (3/4)	3. Meeting für die Evaluation	BGS, SMI, PSC, DGE, KGO	virtuell	6
Apr 2020	Einreichung des akzeptierten Papers (5/5)	"Knowledge Graph for the Visualisation of CRM Objects in a Social Network of Business Objects (SoNBO): Development and Evaluation of the SoNBO Visualiser"	SMI, BGS, PSC	Koblenz	5
Okt 2020	"SoNBO für den KOSMOS" (4/4)	4. Meeting für die Evaluation	BGS, SMI, PSC, DGE, KGO	virtuell	6
Okt 2020	Einreichung Dissertation	Finalisierung der Dissertation	BGS	Koblenz	-

Tab.-A 1: Social (Sub)Concept eines CRM-Systems Konfigurations-Projekt 1 (eigene Darstellung)

Social Concept	Social Subconcept
(1) Forschungsgruppe research group	Professor professor
	Wiss. Mitarbeiter res. assistant
	Nicht-wiss. Mitarbeiter non-scienc. staff
	Techn. Mitarbeiter techn. assistant
	HiWi stud. assistant
(2) Lehre teaching	Vorlesung lecture
	Übung exercise
	Seminar seminar
	Qualifikationsarbeit thesis
	Praktikum student project
(3) Forschung research	Dissertation PhD
	Projekt project
(4) Kontakt contacts	Professor professor
	Wiss. Mitarbeiter res. assistant
	Nicht-wiss. Mitarbeiter non-scienc. staff
	Student student
	Techn. Mitarbeiter techn. assistant
	Kontaktperson contact person
(5) Institution institution	Universität university
	Unternehmen company
	Öffentliche Verwaltung public administration
	Verein association

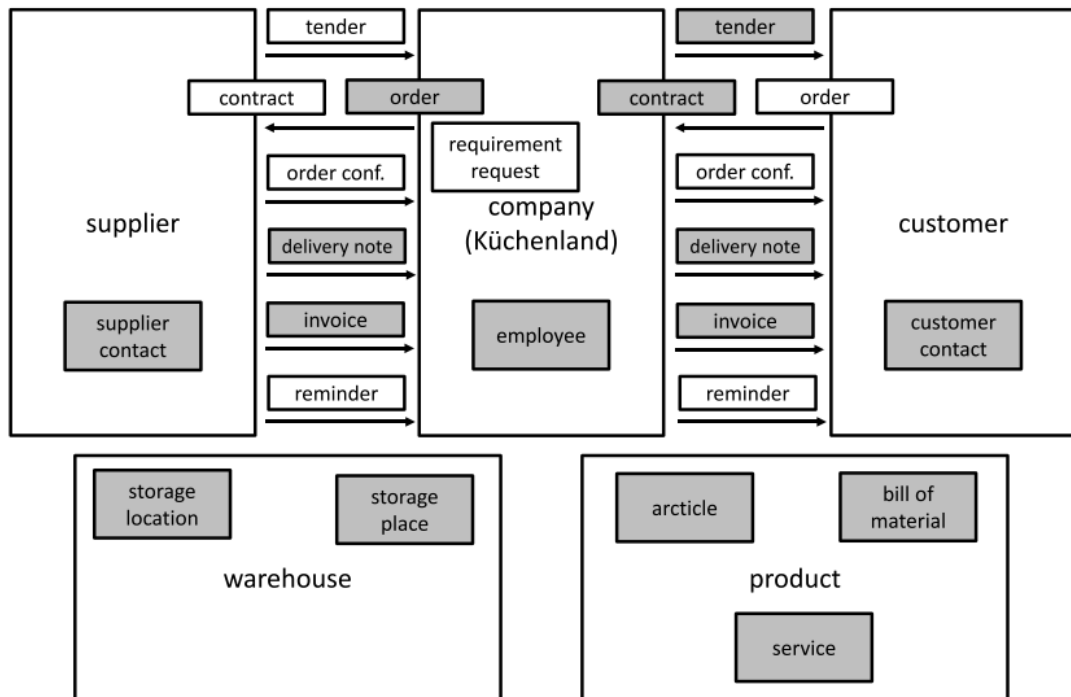


Abb.-A 2: Social (Sub)Concept eines ERP-Systems Konfigurations-Projekt 2 (Gebel-Sauer & Schubert, 2019b, p. 6)

A.4 Grundlage für die Entwicklung der Ontologie (als Referenzmodell) im SoNBO-Framework

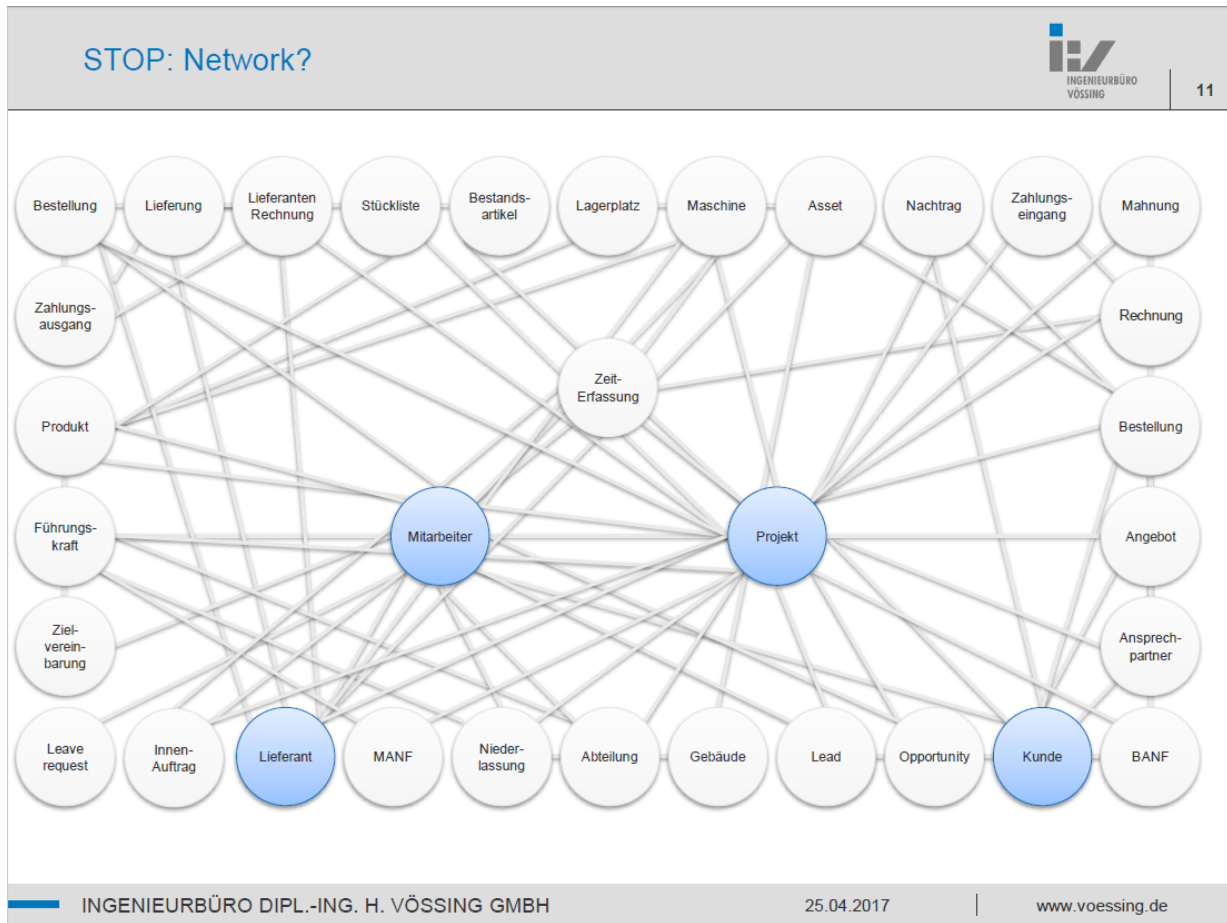


Abb.-A 3: Social Network of Concepts im Key Case Vössing

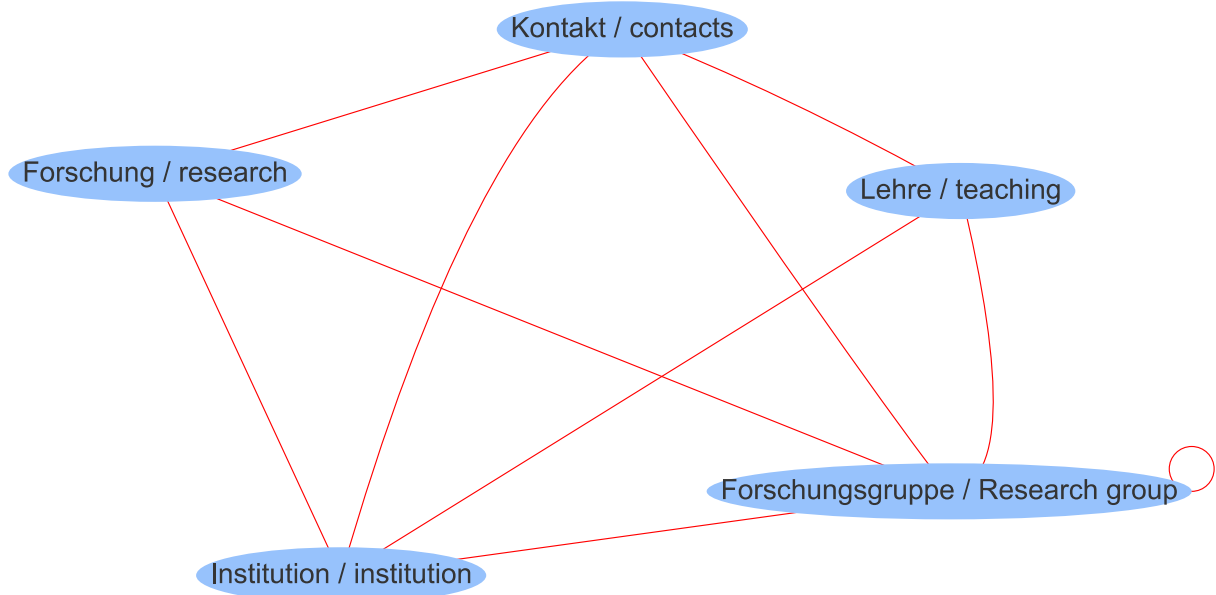


Abb.-A 4: Social Network of Concepts für CRM-Informationen nach dem Konfigurations-Projekt 1

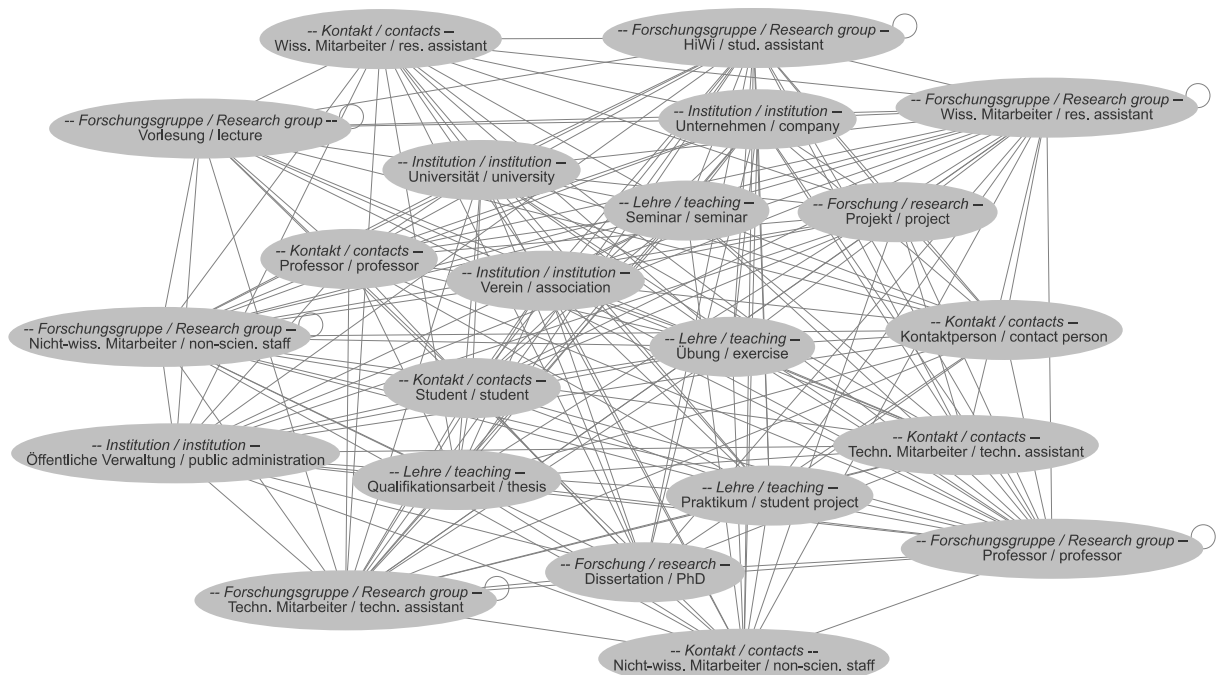


Abb.-A 5: Social Network of Subconcepts für CRM-Informationen nach dem Konfigurations-Projekt 1

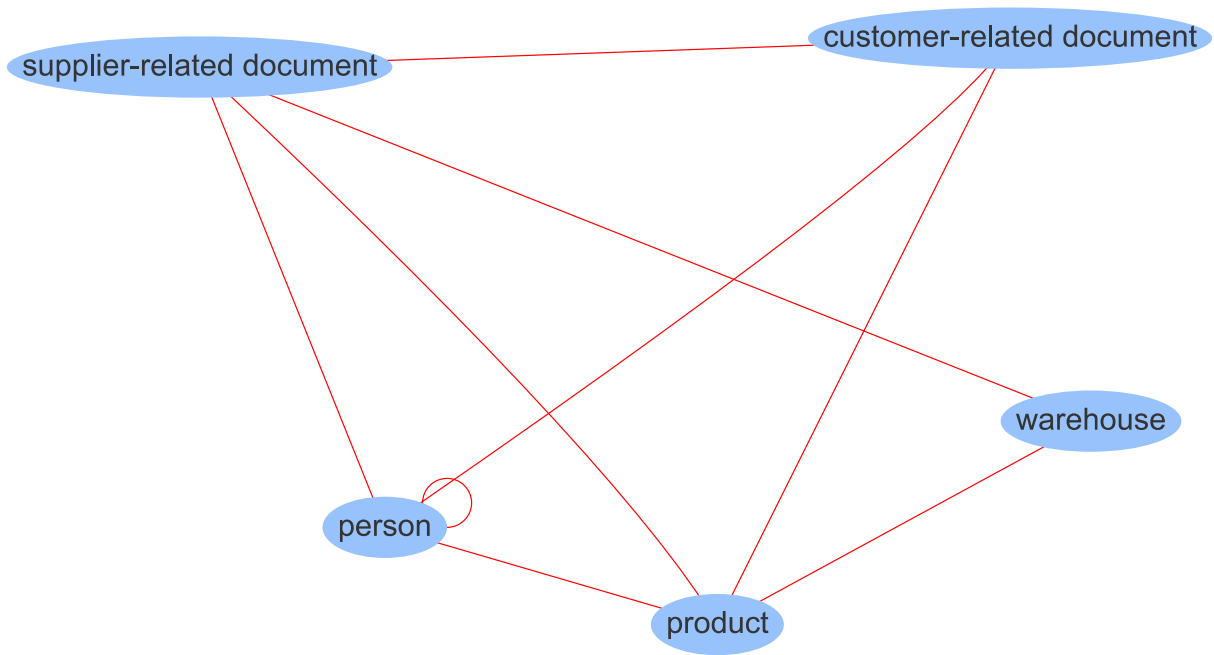


Abb.-A 6: Social Network of Concepts für ERP-Informationen nach dem Konfigurations-Projekt 2
(Gebel-Sauer & Schubert, 2019b, p. 7)

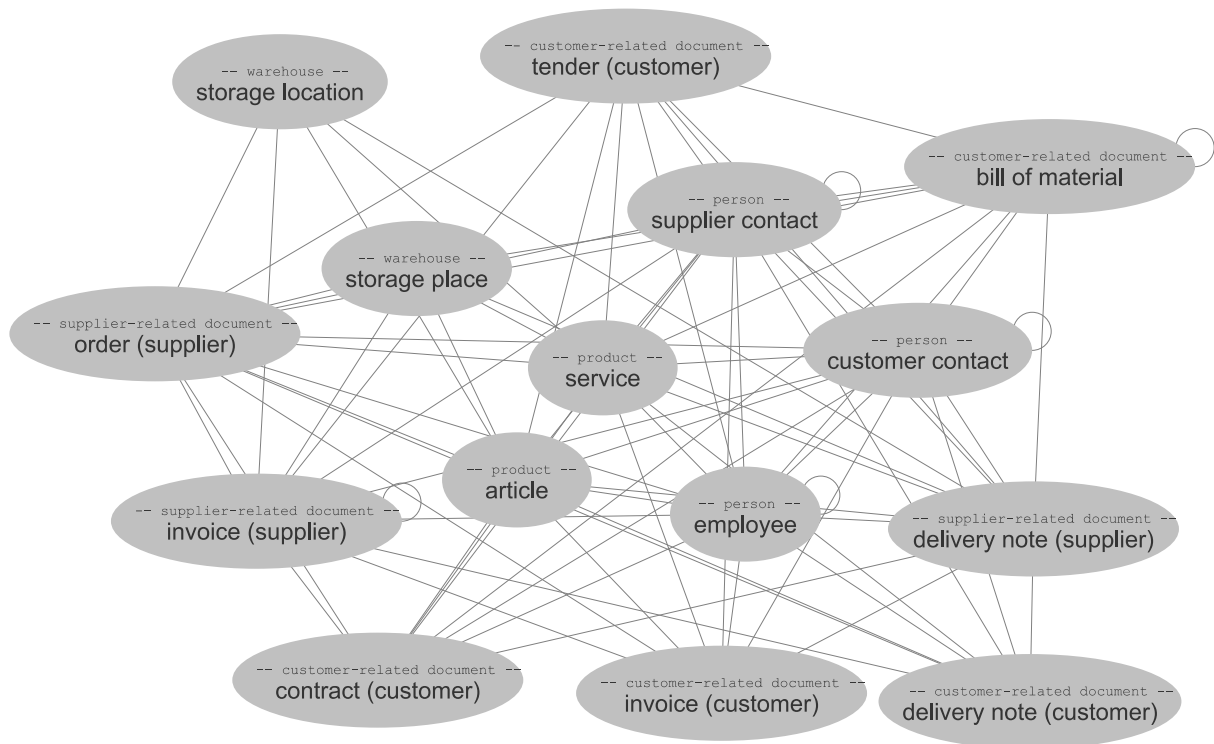


Abb.-A 7: Social Network of Subconcepts für ERP-Informationen nach dem Konfigurations-Projekt 2
(Gebel-Sauer & Schubert, 2019b, p. 8)



Curriculum Vitae

Ausbildung

- Okt 2016 – Feb 2021 **Universität Koblenz-Landau** Koblenz
Promotionsstudium Wirtschaftsinformatik
- Apr 2015 – Nov 2016 **Universität Koblenz-Landau** Koblenz
Zertifikatsstudiengang **Informatik** (Erweiterung des **Master**)
- Okt 2007 – Mär 2013 **Johannes Gutenberg-Universität** Mainz
Staatsexamen Lehramt an Gymnasien (mit **Master** vergleichbar)
Mathematik, Geographie
Examensarbeit mit 1,0
- Aug 1998 - Mär 2007 **Görres-Gymnasium** Koblenz
Abitur

Berufserfahrung

- Seit Okt 2016 **Universität Koblenz-Landau** Koblenz
Wissenschaftliche Mitarbeiterin (FB04 Informatik; FG BAS)
- Jul 2015 – Jul 2017 **Hart Consult International GmbH** Weißenthurm
Customizing ERP-CRM-System *work4all!*, Projektmanagement
- Feb 2015 – Jan 2016 **Bauunternehmung Willi Sauer GmbH & Co. KG** Koblenz
Öffentlichkeitsarbeit, Erstellung und Betreuung der Homepage
- Okt 2011 – Jan 2015 **Eichendorff-Gymnasium** Koblenz, **Otto Schott-Gymnasium** Mainz
Lehrerin (Mathematik, Erdkunde)

Koblenz, den 24.02.2021



Ort, Datum

Berit Gebel-Sauer