

LEHRBUCH

Alexander Stuckenholz

Basiswissen Energieinformatik

Ein Lehr- und Arbeitsbuch für Studierende
und Anwender



Springer Vieweg

Basiswissen Energieinformatik

Alexander Stuckenholz

Basiswissen Energieinformatik

Ein Lehr- und Arbeitsbuch für Studierende
und Anwender



Springer Vieweg

Alexander Stuckenholz
Hochschule Hamm-Lippstadt
Hamm, Deutschland

ISBN 978-3-658-31808-6 ISBN 978-3-658-31809-3 (eBook)
<https://doi.org/10.1007/978-3-658-31809-3>

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

Springer Vieweg
© Der/die Herausgeber bzw. der/die Autor(en), exklusiv lizenziert durch Springer Fachmedien Wiesbaden GmbH, ein Teil von Springer Nature 2020

Das Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Jede Verwertung, die nicht ausdrücklich vom Urheberrechtsgesetz zugelassen ist, bedarf der vorherigen Zustimmung des Verlags. Das gilt insbesondere für Vervielfältigungen, Bearbeitungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

Die Wiedergabe von allgemein beschreibenden Bezeichnungen, Marken, Unternehmensnamen etc. in diesem Werk bedeutet nicht, dass diese frei durch jedermann benutzt werden dürfen. Die Berechtigung zur Benutzung unterliegt, auch ohne gesonderten Hinweis hierzu, den Regeln des Markenrechts. Die Rechte des jeweiligen Zeicheninhabers sind zu beachten.

Der Verlag, die Autoren und die Herausgeber gehen davon aus, dass die Angaben und Informationen in diesem Werk zum Zeitpunkt der Veröffentlichung vollständig und korrekt sind. Weder der Verlag, noch die Autoren oder die Herausgeber übernehmen, ausdrücklich oder implizit, Gewähr für den Inhalt des Werkes, etwaige Fehler oder Äußerungen. Der Verlag bleibt im Hinblick auf geografische Zuordnungen und Gebietsbezeichnungen in veröffentlichten Karten und Institutionsadressen neutral.

Planung: Sybille Thelen
Springer Vieweg ist ein Imprint der eingetragenen Gesellschaft Springer Fachmedien Wiesbaden GmbH und ist ein Teil von Springer Nature.
Die Anschrift der Gesellschaft ist: Abraham-Lincoln-Str. 46, 65189 Wiesbaden, Germany

Für Felix, Max und Nina

Inhaltsverzeichnis

1 Einleitung	1
Literatur	6
2 Grundlagen	9
2.1 Energieflüsse	10
2.2 Energiewende	22
2.3 Digitalisierung	27
2.4 Informationstechnik	30
2.5 Kommunikationstechnik	42
2.6 Automationstechnik und Feldbusssysteme	53
2.7 Standards der Energieinformatik	61
2.8 Zusammenfassung	67
2.9 Kontrollfragen	68
Literatur	69
3 Daten	75
3.1 Common Information Model	76
3.2 Räumliche Daten	86
3.3 Zeitreihen	95
3.4 Datenquellen	106
3.5 Zusammenfassung	113
3.6 Kontrollfragen	114
Literatur	115
4 Algorithmen	119
4.1 Prognose	121
4.2 Optimierung	132
4.3 Zusammenfassung	148
4.4 Kontrollfragen	149
Literatur	150

5 Kommunikation und Automation	153
5.1 Netzautomation	154
5.2 Gebäudeautomation	166
5.3 Smart Meter	195
5.4 Marktkommunikation	203
5.5 Zusammenfassung	212
5.6 Kontrollfragen	213
Literatur	215
6 IT-Sicherheit	221
6.1 Grundwerte	222
6.2 Rechtsrahmen und Standards	227
6.3 Kommunikationssicherheit	230
6.4 Zusammenfassung	235
6.5 Kontrollfragen	236
Literatur	237
Stichwortverzeichnis	241

Abkürzungsverzeichnis

ACSI	Abstract Communication Service Interfaces
AES	Advanced Encryption Standard
AMEV	Arbeitskreis Maschinen- und Elektrotechnik staatlicher und kommunaler Verwaltungen
AMPL	A Mathematical Programming Language
API	Application Programming Interface
APDU	Application Protocol Data Unit
ASCII	American Standard Code for Information Interchange
ASDU	Application Service Data Unit
ASN.1	Abstract Syntax Notation One
BACnet	Building Automation and Control Networks
BBMD	BACnet Broadcast Management Device
BCU	Bus Coupling Unit
BDEW	Bundesverband der Energie- und Wasserwirtschaft e. V.
BIBB	BACnet Interoperability Building Block
BIKO	Bilanzkreiskoordinator
BK	Bilanzkreis
BKV	Bilanzkreisverantwortliche
BNetza	Bundesnetzagentur
BSI	Bundesamt für Sicherheit in der Informationstechnik
CA	Certification Authority
CEN	Europäisches Komitee für Normung
CENELEC	Europäisches Komitee für elektrotechnische Normung
CDP	Controllable Double Point
CGMES	Common Grid Model Exchange Standard
CIM	Common Information Model
CLS	Controllable Local System
ComCT	Communication Client for Traders
COSEM	Companion Specification for Energy Metering

COV	Change of Value Reporting
CRC	Cyclic Redundancy Check
CSMA/CA	Carrier Sense Multiple Access/Collision Avoidance
CSMA/CD	Carrier Sense Multiple Access/Collision Detect
DACF	Day Ahead Congestion Forecast
DDC	Direct Digital Control
DHKE	Diffie-Hellman-Schlüsselaustauschverfahren
DIN	Deutsches Institut für Normung
DKE	Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE
DNP3	Distributed Network Protocol
DPT	Data Point Type
DTD	Document Type Definition
DVGW	Deutscher Verein des Gas- und Wasserfaches
exXML	Electronix Business for XML
EDA	energiewirtschaftlicher Datenaustausch
EDIFACT	United Nations Electronic Data Interchange for Administration, Commerce and Transport
EDM	Energiedatenmanagement
edna	Bundesverband Energiemarkt & Kommunikation e. V.
EEG	Erneuerbare-Energien-Gesetz
EEX	European Energy Exchange
EIB	Europäischer Installationsbus
EIC	Energy Identification Code
EMS	Energiemanagementsystem
EnWG	Energiewirtschaftsgesetz
ER	Entity-Relationship
ESB	Enterprise Service Bus
ESMP	European Style Market Profile
ESS	entso-e Scheduling System
ETRM	Energy Trading and Risk Management
ETS	Engineering-Tool-Software
ETSI	Europäisches Institut für Telekommunikationsnormen
EVU	Energieversorgungsunternehmen
FTP	File Transfer Protocol
GA	Gebäudeautomation
GAMS	General Algebraic Modeling System
GeLi Gas	Geschäftsprozess Lieferantenwechsel Gas
gMSB	grundständige Messstellenbetreiber
GOOSE	Generic Object Oriented Substation Events
GPKE	Geschäftsprozesse für die Kundenbelieferung mit Elektrizität
GPRS	General Packet Radio Service

GI	Gesellschaft für Informatik
GIS	geografische Informationssysteme
HAN	Home Area Network
HBAM	Home Building Architecture Model
HDLC	High-Level Data Link Control
HMAC	Hash-based Message Authentication Code
HMI	Human Machine Interface
HPFC	Hourly Price Forward Curve
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
ICMP	Internet Control Message Protocol
IEC	Internationale Elektrotechnische Kommission
IEEE	Institute of Electrical and Electronics Engineers
IED	Intelligent Electronic Device
IKT	Informations- und Kommunikationstechnik
IOT	Internet of Things
ITU	Internationale Fernmeldeunion
IOA	Information Object Address
IP	Internetprotokoll
IPCC	Intergovernmental Panel on Climate Change
IPsec	Internet Protocol Security
IRT	Isochronous Real Time
ISMS	Information Security Management System
ISO	Internationale Organisation für Normung
JMS	Java Message Service
jOSEF	Java-based Open-Source Smart Meter Gateway Experimentation Framework
JSON	JavaScript Object Notation
KNF	konjunktive Normalform
LAN	Local Area Network
LF	Lieferant
LLNO	Logical-Node-Zero
LMN	Local Metrological Network
LN	Logical Node
LNS	LonWorks Network Services
LON	Local Operating Network
LPHD	Logical-Node-Physical-Device
LTE	Long Term Evolution
MA	Moving Average
MAE	Mean Absolute Error
MABIS	Marktregeln für die Durchführung der Bilanzkreisabrechnung Strom
MaLo-ID	Marktlokationsidentifikationsnummer

MAPE	Mean Absolute Percentage Error
MaStR	Marktstammdatenregister
MaStRV	Verordnung der Registrierung energiewirtschaftlicher Daten
MBAP	Modbus Application Protocol
MILP	Mixed-Integer Linear Programming
MSE	Mean Squared Error
MMS	Manufacturing Messaging Specification
MP-ID	Marktpartneridentifikationsnummer
MSB	Messstellenbetreiber
MsbG	Messstellenbetriebsgesetz
NB	Netzbetreiber
OBIS	Object Identification System
ODFM	Orthogonal Frequency-Division Multiplexing
ODbL	Open Data Commons Open Database License
OMG	Object Management Group
OPC UA	Open Platform Communications United Architecture
OPSD	Open Power Systems Data Initiative
OSGeo	Open Source Geospatial Foundation
OSGP	Open Smart Grid Protocol
OSM	OpenStreetMap
OTC	Over the Counter
PCTP	Precision Time Control Protocol
RAMI 4.0	Reference Architecture Model for Industry 4.0
RBE	Report-by-Exception
RLM	registrierende Leistungsmessung
RTU	Remote Terminal Unit
NRZ	Non Return to Zero
NTP	Network Time Protocol
SCADA	Supervisory Control and Data Aquisition
SCPT	Standard Configuration Property Types
SCL	Substation Configuration Language
SGAM	Smart Grid Architecture Model
SGIS	Smart Grid Security Group
SM	Smart Meter
S/MIME	Secure/Multipurpose Internet Mail Extensions
SMG	Smart Meter Gateway
SMGA	Smart Meter Gateway Administrator
SML	Smart Message Language
SMTP	Simple Mail Transfer Protocol
SNB	Stromnetzbetreibernummer
SNVT	Standard Network Variable Types
SOA	Service Oriented Architecture

SOAP	Simple Object Access Protocol
SPS	speicherprogrammierbare Steuerung
SRID	Spational Reference ID
SQL	Structured Query Language
SV	Sampled Values
TC	Technical Committee
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TLV	Tag-Length-Value
TPDU	Transport Protocol Data Unit
OMSS	Open Metering System Specification
PDCA	Plan-Do-Check-Act
PICS	Protocol Implementation Conformance Statement
PKI	Public Key Infrastruktur
PLC	Programmable Logic Controller
UART	Universal Asynchronous Receiver Transmitter
UDP	User Datagram Protocol
ÜNB	Übertragungsnetzbetreiber
UML	Unified Modelling Language
URL	Unified Resource Locator
UTC	universelle Weltzeit
VDEW	Verband der Elektrizitätswirtschaft
VLAN	Virtual Local Area Network
VMD	Virtual Manufacturing Device
VNB	Verteilnetzbetreiber
WAN	Wide Area Network
WGS 84	World Geodetic System 1984
WIM	Wechselprozesse im Messwesen
WKT	Well-Known-Text
WSDL	Web Service Description Language
WWW	Word Wide Web
XDR	External Data Representation
XML	Extensible Markup Language
XSD	XML Schema Definition

Abbildungsverzeichnis

Abb. 2.1	Entwicklung des Endenergieverbrauchs in Deutschland zwischen 1990 und 2018 nach Sektoren. Bildquelle: Umweltbundesamt. https://www.umweltbundesamt.de/daten/energie/energieverbrauch-nach-energietaegern-sektoren	11
Abb. 2.2	Einliniendiagramm eines Kohlekraftwerks. Bildquelle: VBG Powertech e. V. https://commons.wikimedia.org/wiki/File:Energieverteilung01.png , https://creativecommons.org/licenses/by-sa/3.0/legalcode	14
Abb. 2.3	Physikalisch-technischer Energiefluss von der Erzeugung bis zum Verbrauch	15
Abb. 2.4	Stromgestehungskosten für erneuerbare Energien und konventionelle Kraftwerke in Deutschland im Jahr 2018	18
Abb. 2.5	Klassische Einspeisestruktur.....	18
Abb. 2.6	Kaufmännischer Energiefluss	19
Abb. 2.7	Anzahl der Unternehmen am Energiemarkt in Deutschland nach Tätigkeitsfeld im Jahr 2018. Datenquelle: Bundesverband der Energie- und Wasserwirtschaft e. V. (BDEW), siehe https://www.bdew.de/media/documents/Marktteilnehmer_Energie_aktuell_online_o_halbjährlich_Ki_27022020.pdf	20
Abb. 2.8	Zunehmende Spezialisierung der Unternehmen in der Energiewirtschaft. In Anlehnung an [24, S. 8]	21
Abb. 2.9	Nettoerzeugung elektrischer Energie aus regenerativen Energiequellen. Bildquelle: Fraunhofer ISE. https://www.energy-charts.de/ren_share_de.htm?source=ren-share&period=annual&year=all	24

Abb. 2.10	Trend der Emissionen einiger Treibhausgase in Deutschland zwischen 1990 und 2017. Bildquelle: Umweltbundesamt. https://www.umweltbundesamt.de/bild/trend-der-emissionen-von-kohlendioxid-methan	25
Abb. 2.11	Zukünftige Erzeugungs- und Einspeisestruktur für elektrische Energie	27
Abb. 2.12	Anwendungsklassen in der Energiewirtschaft und ihre Zuordnung zur Abwicklung des physikalisch-technischen bzw. des kaufmännischen Energieflusses	31
Abb. 2.13	Die Modellebenen der UML	34
Abb. 2.14	Als Teil eines Informationsmodells beschreibt die Klasse <i>Fotovoltaikanlage</i> das verallgemeinerte Konzept aller <i>Fotovoltaikanlage</i>	35
Abb. 2.15	Einige Beispiele für Assoziationen im UML-Klassendiagramm	36
Abb. 2.16	Vererbungshierarchie unter den Klassen <i>Erzeugungseinheit</i> und <i>Windkraftanlage</i> als UML-Klassendiagramm	36
Abb. 2.17	Ein Bildschirmfoto von Modelio mit einem Informationsmodell auf Basis eines Klassendiagramms der UML	38
Abb. 2.18	Abbildung eines Informationsmodells auf potenziell mehrere Datenmodelle	39
Abb. 2.19	Kommunikationsablauf im 7-Schicht-ISO/OSI-Modell	44
Abb. 2.20	Bus-, Ring- und Sterntopologie von Kommunikationsnetzen	46
Abb. 2.21	Aufbau eines Ethernetframes	51
Abb. 2.22	Darstellung einer Netzwerkkommunikation in Wireshark	53
Abb. 2.23	Die Ebenen der Automatisierungspyramide	54
Abb. 2.24	Netzleitwarte der Energie Steiermark	55
Abb. 2.25	Abbildung des Modbus-Standards auf die Schichten des ISO/OSI-Modells	57
Abb. 2.26	Nachrichtenformat von Modbus	58
Abb. 2.27	Modbus-Informationsmodell als Klassendiagramm der UML	59
Abb. 2.28	Normungsgremien und Organisationen im Kontext des Smart Grid	63
Abb. 2.29	Das Smart Grid Architecture Model	65
Abb. 3.1	Kommunikationsverbindungen bei der direkten Integration von Anwendungssystemen	76
Abb. 3.2	Anwendungsintegration mittels CIM	77
Abb. 3.3	Vererbungshierarchie eines Leistungsschalters im CIM-Informationsmodell	78
Abb. 3.4	Beziehung von stromleitenden Elementen zu Anschlusspunkten im CIM-Datenmodell	79
Abb. 3.5	Beispiel eines einfachen Stromnetzes aus Generator, Klemmen, Leistungsschalter und Last	79

Abb. 3.6	Objektdiagramm des Beispielnetzes auf Basis von CIM-Objekten	80
Abb. 3.7	Objektdiagramm des Beispielnetzes auf Basis von CIM-Objekten	80
Abb. 3.8	Integrationsarchitektur der openKonsequenz-Plattform	84
Abb. 3.9	Bildschirmfoto des Werkzeugs CIMSpy. Bildquelle: https://www.powerinfo.us/CIMSpy.html	84
Abb. 3.10	Klassen für die Abbildung des geografischen Orts in CIM	87
Abb. 3.11	Darstellung eines Verteilnetzes in Kombination mit geografischen Daten in CIMSpace	88
Abb. 3.12	Unterschied zwischen kartesischem und sphärischem Koordinatensystem. Bildquelle: Paul Ramsey und Mark Leslie. http://postgis.net/workshops/postgisintro/geography.html , https://creativecommons.org/licenses/by-sa/3.0/legalcode	89
Abb. 3.13	Darstellung eines einzelnen Punktes in einer Karte von pgAdmin	90
Abb. 3.14	Darstellung eines Polygons in pgAdmin	92
Abb. 3.15	Die Standorte einiger regenerativer Energieanlagen als Abfrageergebnis von PostGIS	94
Abb. 3.16	Darstellung von regenerativen Energieanlagen in einer Kartenansicht von QGis	95
Abb. 3.17	Darstellung von Windkraftanlagen in QGis. Die Größe der Kreise wird in Abhängigkeit der Nennleistung variiert	96
Abb. 3.18	Typhierarchie einer äquidistanten Zeitreihe im CIM-Basismodell	98
Abb. 3.19	Zusammenhang zwischen den Klassen MarketDocument, TimeSeries und Point im IEC 62325	99
Abb. 3.20	Der Verlauf der höchsten und niedrigsten Tagestemperaturen in Düsseldorf als Ergebnis einer Abfrage aus einem Python-Script	100
Abb. 3.21	Nutzung verschiedener Datenbanksysteme für die Verwaltung von Zeitreihendaten	101
Abb. 3.22	Anwendung von Graphite Functions auf eine Metrik	103
Abb. 3.23	Temperaturen für Düsseldorf als Liniendiagramm in Grafana	105
Abb. 3.24	Informationsmodell des Marktstammdatenregisters. Bildquelle: Dokumentation des MaStr unter https://www.marktstammdatenregister.de	107
Abb. 3.25	Eine Kartendarstellung der Daten von OpenStreetMap im Browser	110
Abb. 3.26	Das Ergebnis der Abfrage aller Elemente mit dem Power-Tag in Hamm mithilfe der Overpass API	112
Abb. 4.1	Einflussfaktoren auf die Strompreisfindung	120
Abb. 4.2	Beispiel einer Zeitreihe, die auf mehreren Zeitreihenkomponenten basiert	124
Abb. 4.3	Glättung einer Zeitreihe mithilfe des gleitenden Mittelwerts	125
Abb. 4.4	Resultat der Dekomposition einer Zeitreihe in die einzelnen Komponenten	126

Abb. 4.5	Exponentielles Glätten erster Ordnung.....	128
Abb. 4.6	Exponentielles Glätten zweiter Ordnung.....	130
Abb. 4.7	Exponentielles Glätten dritter Ordnung	131
Abb. 4.8	Das Prognoseblatt von Excel, mit dessen Hilfe Prognosen auf Basis exponentieller Glättung dritter Ordnung durchgeführt werden können	132
Abb. 4.9	Grafische Darstellung des zulässigen Bereichs für die Optimierung des Kraftwerkseinsatzes.....	136
Abb. 4.10	Ergebnis eines Optimierungslaufs mit Pyomo und dem Solver glpk ..	139
Abb. 4.11	Das Koepchenwerk in Herdecke.....	140
Abb. 4.12	Die beispielhafte HPFC als Eingangsdaten für die Bewirtschaftung eines Pumpspeicherwerk	141
Abb. 4.13	Das Ergebnis der Optimierung der Bewirtschaftung des Pumpspeichers	144
Abb. 5.1	Die Höchstspannungsschaltanlage der Firma Amprion GmbH in Hattingen	154
Abb. 5.2	Kommunikationsarchitektur in und mit Schaltanlagen	156
Abb. 5.3	Hierarchischer Aufbau der Standardfamilie IEC 60870-5	158
Abb. 5.4	Das ASDU-Format von IEC 60870-5	159
Abb. 5.5	Elementares Klassenmodell des Abstract Communication Service Interface des IEC 61850	163
Abb. 5.6	Interner, logischer Aufbau eines Leistungsschalters nach IEC 61850 und der Zusammenhang mit der Sekundärtechnik. In Anlehnung an [2]	164
Abb. 5.7	Abbildung des Datenmodells und der Zugriffsdienste auf Kommunikationsprotokolle in IEC 61850	165
Abb. 5.8	Gebäuderelevanter Endenergieverbrauch für Raumwärme, Raumkühlung, Warmwasser und Beleuchtung. Bildquelle: Umweltbundesamt. https://www.umweltbundesamt.de/indikator-energieverbrauch-fuer-gebaeude	167
Abb. 5.9	Kommunikationsstandards auf den Ebenen der Gebäudeautomation. In Anlehnung an [23]	169
Abb. 5.10	Versuchsaufbau mit KNX und BACnet im Gebäudeautomationslabor an der Hochschule Hamm-Lippstadt	170
Abb. 5.11	KNX und die sieben Schichten des ISO/OSI-Modells	170
Abb. 5.12	Aufbau eines Datentelegramms in KNX	171
Abb. 5.13	Ausschnitt aus dem KNX Interworking Model in Form eines Klassendiagramms der UML	173
Abb. 5.14	Bildschirmfoto der ETS-Software. Copyright durch KNX Association	174
Abb. 5.15	LonWorks und das ISO/OSI-Modell	176

Abb. 5.16	BacNet-Versuchsstand zur Simulation einer Heizungsanlage im Gebäudeautomationslabor der Hochschule Hamm-Lippstadt	180
Abb. 5.17	Die vier Schichten des BACnet-Kommunikationsstapels	181
Abb. 5.18	Ausschnitt aus dem BACnet-Informationsmodell als Klassendiagramm der UML	183
Abb. 5.19	Ausschnitt aus dem UML-Klassendiagramm des OPC-UA-Informationsmodells	189
Abb. 5.20	Beispielhafte Adressräume von zwei OPC-UA-Servern mit Knoten und Beziehungen	189
Abb. 5.21	Beispiel für ein Beziehungsgeflecht zwischen Objekten, Objekttypen, Variablen und Variablentypen in OPC UA. In Anlehnung an [42, S. 10]	190
Abb. 5.22	Kommunikationsvorgaben des BSI zum LMN eines SMG. Bildquelle: [21, S. 51]	198
Abb. 5.23	Aufbau eines M-Bus-Telegramms	198
Abb. 5.24	Aufbau eines Physical Device nach COSEM	199
Abb. 5.25	Kommunikationsvorgaben für das WAN eines SMG. Bildquelle: [21, S. 31]	201
Abb. 5.26	Ablauf des Anwendungsfalls Lieferbeginn aus dem GPKE. Bildquelle: Bundesnetzagentur (BNetzA): Anlage zum Beschluss BK6-06-009, S. 33	204
Abb. 5.27	Informationsmodell eines ESS-Fahrplans als Klassendiagramm der UML	210
Abb. 5.28	Bildschirmfoto von ComCT	211
Abb. 6.1	Die vier Phasen des Deming-Kreislaufs	229
Abb. 6.2	Die Struktur des IEC 62351 und die Zuordnung der Abschnitte zu verwandten Standards	231



Einleitung

1

Die digitale Datenverarbeitung ist aus dem Lebensalltag der Unternehmen heute nicht mehr wegzudenken. Kaum ein Unternehmen kommt mehr ohne entsprechende Informations- und Kommunikationstechnik (IKT) aus. Auch die Energiewirtschaft mit ihrer Aufgabe der sicheren, kostengünstigen und ökologisch nachhaltigen Versorgung mit Energie nutzt an vielen Stellen moderne IKT.

In allen Bereichen der Energiewirtschaft finden sich Anwendungsfälle für IKT. Im Kontext der physikalisch-technischen Energieflüsse werden die Aufgaben der Mess-, Steuer- und Regelungstechnik umgesetzt. Dabei werden Zustandsdaten von Netzen und Erzeugungsanlagen in sog. SCADA-Systemen erfasst, verarbeitet und visualisiert. Kraftwerke und Netze werden in Echtzeit gesteuert und geregelt, der Einsatz von Betriebsmitteln wird mithilfe von Optimierungssystemen geplant. Mit der kaufmännischen Abwicklung der Energieversorgung finden sich weitere Ansatzpunkte für eine moderne IKT. Energieprodukte werden auf elektronischen Märkten gehandelt, dazu werden massenhaft Marktdaten verarbeitet, Preise prognostiziert und das Risiko von Handelsportfolios wird bewertet.

Der Einsatz von Methoden der Datenverarbeitung und der digitalen Kommunikation in der Energiewirtschaft mündet in der Disziplin der **Energieinformatik**. Die Energieinformatik beschäftigt sich dabei unter anderem mit der effizienten Nutzung von IKT, um ein sicheres, kostengünstiges und ökologisch nachhaltiges Energiesystem zu schaffen (siehe [3, S. 3 f.]). Die Wikipedia definiert die Energieinformatik dabei als eine Wissenschaft, bei der die Methoden der Informatik und angrenzender Gebiete wie der Elektrotechnik, Kybernetik, der Automatisierung und der Wirtschaftswissenschaft genutzt werden, um Fragen komplexer Energiesysteme zu erforschen und zu lösen (siehe [6]). Die Energieinformatik ist dabei eine angewandte Informatik, sie wendet Techniken und Methoden auf eine bestimmte Problemdomäne, nämlich die Problemdomäne der Energiewirtschaft,

an. Sie spielt sowohl im technisch-physikalischen sowie im kaufmännischen Bereich der Energiewirtschaft eine bedeutende Rolle (siehe Abschn. 2.1).

Neben den klassischen Anwendungsfällen entstehen aktuell ganz neue Bereiche für die Energieinformatik. Die Energiewende und die fortschreitende Digitalisierung sorgen hier für ganz neue Herausforderungen. In der Energiewende sind viele Konzepte nicht ohne die Energieinformatik umzusetzen. Durch intelligente Steuerung der Lastflüsse soll z. B. der Energiebedarf bereits auf regionaler Ebene gedeckt werden (*Microgrids*). Die intelligente Lastverschiebung kann den Netzausbau reduzieren (*Demand Side Management*). Auf Basis von Echtzeitdaten und Prognosen soll das sich selbst regulierende Netz entstehen (*Smart Grid*) und unterschiedliche Erzeugungseinheiten werden mit der Hilfe intelligenter Steuerungen zu virtuellen Kraftwerken zusammengefasst. Die Digitalisierung stellt dazu ganz neue Mittel bereit. Virtualisierung und cloudbasierte Systeme erzeugen auch im Kontext der Energiewirtschaft neue Flexibilität. Die Methoden von Big Data sorgen für bessere Prognosen und Entscheidungen. Technologien rund um das Internet der Dinge unterstützen die Realisierung von Konzepten, wie z. B. Smart- oder Microgrids. Das Konzept der *Blockchain* kann für Transparenz und Kostenreduktion bei der Abwicklung des Energiehandels sorgen.

Die Energieinformatik hat sich längst als eigenständiger Zweig in Wissenschaft und Lehre etabliert. In der Gesellschaft für Informatik (GI) existiert z. B. eine entsprechende Fachgruppe. Auch eine Reihe von Forschungseinrichtungen und Lehrstühlen setzen sich in Deutschland (und im Rest der Welt) intensiv mit Fragen der Energieinformatik auseinander. In Deutschland ist hier sicher das Forschungs- und Entwicklungsinstitut OFFIS in Oldenburg zu nennen. Auch werden an mehreren Hochschulen entsprechende Bachelorstudiengänge angeboten, z. B. an der Hochschule Ruhr-West oder der Fachhochschule Aachen. An der FH-Oberösterreich existiert auch ein entsprechender Masterstudiengang. Darüber hinaus findet sich die Energieinformatik oft als Vertiefungs- oder Wahlbereich in Informatik- oder Ingenieursstudiengängen, so z. B. an der Carl von Ossietzky Universität in Oldenburg bzw. an der Hochschule Hamm-Lippstadt (siehe [7] und [8]).

Wissenschaftliche Forschung manifestiert sich bekanntlich in entsprechenden Publikationen. Im Kontext der Energieinformatik existieren hierzu diverse Fachkonferenzen und -zeitschriften, die sich mit dem Thema befassen. Im deutschsprachigen Raum ist hier sicher die Konferenz Energieinformatik zu nennen. Sie findet jährlich statt und wird vom OFFIS-Institut aus Oldenburg organisiert. Darüber hinaus existieren viele weitere internationale Konferenzen, z. B.:

- ACM SIGCOMM E-Energy Conference,
- International Conference on Smart Energy Grid Engineering (SEGE),
- International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm),
- International Conference on Informatics for Environmental Protection (EnviroInfo).

Eine Liste von relevanten Konferenzen findet sich auch auf den Webseiten der Interessengemeinschaft Strommarkttreffen (siehe [5]). Neben den Konferenzen sind Fachzeitschriften ein weiterer Kanal für Publikationen. Mit dem Springer Open Journal on Energy Informatics existiert ein entsprechendes Journal, welches sich ausschließlich mit der Energieinformatik befasst.

Auch die Politik sieht in der Energieinformatik ein wichtiges Themenfeld. Das 7. Energieforschungsprogramm der Bundesregierung nimmt explizit Stellung zur Digitalisierung der Energiewende (siehe [4, S. 63 f.]). Dabei werden unter anderem die folgenden Themen als wichtig erachtet:

- Internet of Things, Big Data und künstliches Lernen,
- Simulationsmethoden und Mensch-Maschine-Interaktion,
- Datenhandling, Datenschutz, Datennutzungsrechte und Standardisierung,
- IKT-Sicherheit.

Wie wichtig der deutschen Politik der Einsatz moderner informations- und kommunikationsbasierter Energiesysteme ist, zeigt sich auch im Förderprogramm **E-Energy**. In den Jahren 2008 bis 2013 wurde E-Energy als ressortübergreifendes Leuchtturmprojekt der Bundesregierung umgesetzt (siehe [2]). Ziel von E-Energy war es, die digitale Vernetzung und Optimierung des Energieversorgungssystems durch den Einsatz moderner IKT zu untersuchen. Da der Großteil der Einspeisung regenerativer Energie in den Verteilnetzen stattfindet, lag der Hauptfokus des Projekts auf diesem Bereich. Es sollte untersucht werden, wie mithilfe von Informations- und Kommunikationstechnik der Netzbetrieb optimiert bzw. der Netzausbau reduziert werden kann.

Im Rahmen von E-Energy wurden sechs Modellregionen mit entsprechenden Projekten gefördert. In den Modellregionen waren dabei unterschiedliche Netz- und Einspeisestrukturen vorhanden, sodass verschiedenartige Konzepte erprobt werden konnten (siehe [1, S. 37]). In mehreren Projekten entstanden dabei IKT-basierte Strommarktplätze, auf denen regionale Energieprodukte automatisiert gehandelt werden konnten. Auf ihnen wurden sowohl Haushalts- als auch gewerbliche Kunden mit verschiebbaren Lasten und Energiedienstleiter zusammengeführt. Ein wichtiger Bestandteil der Projekte war meist auch die Erprobung von flexiblen Geschäfts- und Anreizmodellen, um Lastspitzen zu reduzieren. Auf der Seite von Haushaltskunden wurden dafür verschiedene Konzepte entwickelt, z. B. verschiedene Smart Meter bzw. Assistenzsysteme wie der Energiebutler sowie der Kommunikationsstandard EEBus.

Die Ergebnisse der E-Energy-Projekte wurden in 13 zentralen Erkenntnissen zusammengefasst (siehe [2, S. 4 f.]). Insgesamt wurde gezeigt, dass durch den geschickten Einsatz von IKT im Energiesystem der Netzausbau reduziert, Energie eingespart und regenerative Erzeuger besser integriert werden können. Es ergeben sich zudem neue Marktchancen und Geschäftsmodelle. Aus den Projekten flossen viele Ergebnisse in

aktuelle Standards und Produkte ein. Im Projekt eTelligence wurde z. B. der Standard IEC 61850 erweitert und eine entsprechende Open-Source-Implementierung realisiert (siehe Abschn. 5.1). Die IT-Security-Lösungen des E-DeMa-Projekts flossen in das Schutzprofil für den Smart Meter Gateway ein (siehe Abschn. 5.3). In mehreren Projekten wurde mit EEBus ein neuer Kommunikationsstandard entwickelt und erprobt, der gut für das Lastmanagement in Haushalten genutzt werden kann. Zudem entstand der sog. Energy Name Service, der für die standardisierte Identifikation von Elektrofahrzeugen beim Ladevorgang genutzt werden kann. Viele dieser Standards und Produkte werden im weiteren Verlauf detailliert dargestellt.

Das vorliegende Werk ist als Lehrbuch und als erste Einführung in die vielfältigen Themenbereiche der Energieinformatik gedacht. Die Materialien sind dabei im Rahmen einer gleichnamigen Lehrveranstaltung im Bachelorstudiengang Energietechnik und Ressourcenoptimierung an der Hochschule Hamm-Lippstadt entstanden. Bestimmte Fragestellungen, Methoden und Technologien ziehen sich dabei wie ein roter Faden durch alle Bereiche der Energieinformatik. Entlang dieser Fragestellungen ist dieses Lehrbuch organisiert.

Daten werden als das Gold des 21. Jahrhunderts bezeichnet. Ohne sinnvolle Verknüpfung von Daten entsteht keine Information und es können keine belastbaren Entscheidungen getroffen werden. Auch in der Energiewirtschaft ist dies nicht anders. Typischerweise müssen Daten in entsprechenden Informationsmodellen strukturiert werden, damit diese automatisiert verarbeitet werden können. Zudem werden Datenmodelle bzw. -formate benötigt, um Daten zu speichern und über Netze auszutauschen. Dieses Buch wird sich daher intensiv mit den typischen Daten, Informationsmodellen, Werkzeugen und Datenquellen für die Energiewirtschaft auseinandersetzen. In Abschn. 3.1 wird mit dem CIM ein allgemeingültiges Informationsmodell vorgestellt, auf dessen Basis viele Elemente des physikalisch-technischen und des kaufmännischen Energieflusses abgebildet werden können. Räumliche Angaben und Zeitreihen spielen in der Energiewirtschaft eine besondere Rolle. Entsprechend werden diese Arten von Daten in den Abschn. 3.2 und 3.3 besonders behandelt. Abschn. 3.4 geht auf einige frei zugängliche Datenquellen ein, die z. B. für die Modellbildung, Prognose und Simulation in der Energiewirtschaft genutzt werden können.

Daten müssen verarbeitet, verknüpft und ausgewertet werden, damit intelligente Entscheidungen getroffen werden können. Entsprechend spielen Algorithmen und mathematische Modelle in der Energiewirtschaft eine wichtige Rolle. Mithilfe solcher Modelle wird es z. B. möglich, den Zusammenhang zwischen dem Strompreis, der Außentemperatur und der Windeinspeisung auf Basis eines Gleichungssystems zu beschreiben, um eine Prognose des Strompreises zu erhalten. Andere Modelle dienen der Simulation von Stromnetzen oder Erzeugungskapazitäten, um zukünftige Entwicklungen des Energiesystems besser bewerten zu können. Auch Optimierungsverfahren spielen eine wichtige Rolle. Ein Modell eines Kraftwerksparks oder eines Versorgungsnetzes kann mithilfe von Optimierungsverfahren so parametrisiert werden, dass die Anlagen kostenoptimal dimensioniert und betrieben werden können. Um anhand der Modelle entsprechende Lösungen zu berechnen,

wird der Methodenapparat der Mathematik und der Informatik genutzt. Dieses Lehrbuch wird einige Grundsätze zur Modellierung entsprechender Systeme vorstellen und auf die Verfahren zur Lösung solcher Systeme eingehen (siehe Kap. 4).

Eine sichere, kostengünstige und ökologisch nachhaltige Energieversorgung wird durch eine Vielzahl von technischen Anlagen und Geschäftsprozessen mit einer großen Menge unterschiedlicher Akteure getragen. An vielen Stellen des Energiesystems müssen Daten nicht nur erhoben, sondern auch sicher und zuverlässig über weite Strecken übertragen werden können. Anwendungsfälle für Kommunikationssysteme finden sich z. B. in der Automation der Energieanlagen, bei denen über ganz Deutschland verteilte Assets in Echtzeit überwacht, gesteuert und geregelt werden müssen. Mit entsprechenden Standards werden wir uns in Abschn. 5.1 befassen. Um den Energieverbrauch zu senken bzw. den Komfort und die Sicherheit zu erhöhen, werden Automatisierungssysteme auch innerhalb von Gebäuden verstärkt eingesetzt. Entsprechend existiert auch dort ein hoher Bedarf, Daten in Informationsmodellen abilden zu können und sicher über entsprechende Kommunikationssysteme zu übertragen. Einige der dort eingesetzten Verfahren und Standards werden daher in Abschn. 5.2 vorgestellt. Um in Zukunft Energieerzeugung und Verbrauch in Echtzeit aufeinander abstimmen zu können, sollen Verbrauchsdaten zunehmend mithilfe intelligenter Messsysteme erhoben und zeitnah übertragen werden können. Wie dies in Deutschland mithilfe von Smart Metern technisch umgesetzt werden soll, wird in Abschn. 5.3 besprochen. An vielen Geschäftsprozessen der Energiewirtschaft sind aufgrund der Spezialisierung der Unternehmen immer mehr unterschiedliche Akteure beteiligt. Wenn ein Endkunde z. B. seinen Versorger wechselt, müssen mehrere Markttrollen in einem geordneten Prozess Informationen untereinander austauschen. Dabei bedarf es klarerer Vorgaben, bei denen geregelt ist, wer welche Daten an wen und in welchem Format zu übertragen hat. Im Abschn. 5.4 werden entsprechende Verfahren, Protokolle und Systeme für Deutschland dargestellt.

An allen Stellen des Energiesystems kommen IKT-Systeme zum Einsatz, sei es für die Steuerung von Energieanlagen oder für die Abwicklung des Energiehandels. Für eine verlässliche Energieversorgung spielt dabei die Informationssicherheit eine überlebensnotwendige Rolle. Grundwerte, wie Verfügbarkeit, Vertraulichkeit und Integrität, müssen für die relevanten IKT-Systeme jederzeit sichergestellt werden. Was dies für die Energiewirtschaft bedeutet, welche Gefahren diese Werte bedrohen, mit welchen Maßnahmen diesen Gefahren begegnet werden kann und welcher Rechtsrahmen hier gültig ist, wird daher im Kap. 6 behandelt.

In diesem Lehrbuch werden ausschließlich solche Themen behandelt, die spezifisch für die Energieinformatik sind oder dort besondere Anwendungsfälle finden. Auf allgemeine Themen der betrieblichen Datenverarbeitung, die in jedem Unternehmen (auch der Energiewirtschaft) anfallen (z. B. Human Resources, Controlling oder Finance), wird daher nicht weiter eingegangen.

Dieses Lehrbuch richtet sich mit den zuvor beschriebenen Inhalten vor allem an Studierende und AnwenderInnen, die sich fachlich mit Fragen der Energieversorgung bzw. der Informationstechnik auseinandersetzen und einen Einblick in die Informati-

onsverarbeitung im Bereich der Energieversorgung erhalten wollen. Zum einen können InformatikerInnen von den Lehrinhalten profitieren, die bislang keinen Kontakt mit der Energiewirtschaft hatten. Sie erhalten einen Einblick in die Struktur der Wertschöpfungskette der deutschen Energiewirtschaft und bekommen einen ersten Überblick zu den relevanten informationstechnischen Verfahren und Standards, die in dieser Branche genutzt werden. Auf der anderen Seite richtet sich dieses Buch auch an EnergiewirtschaftlerInnen bzw. EnergietechnikerInnen. Sie erhalten einen Einblick in die relevanten Verfahren, Standards und Technologien der Informatik, die in ihrem Bereich eine Rolle spielen.

Für beide Zielgruppen werden im nächsten Kapitel einige Grundlagen erarbeitet. Für InformatikerInnen werden zunächst die Wertschöpfungskette der Energiewirtschaft und der dort vorhandene physikalisch-technische und der kaufmännische Energiefluss dargestellt. Durch die Energiewende und die zunehmende Digitalisierung steht die Energiewirtschaft dabei vor großen Herausforderungen. Auf beide Themen werden wir daher einen kurzen Blick werfen. Für EnergiewirtschaftlerInnen existiert zudem eine Einführung in die Grundbegriffe der Informations- und Kommunikationstechnik sowie der Automatisierungstechnik unter Einsatz von Feldbusssystemen.

Ein Lehrbuch über die Energieinformatik kann das Thema kaum erschöpfend behandeln. Es sei daher an dieser Stelle darauf hingewiesen, dass das vorliegende Werk keinen Anspruch auf Vollständigkeit erhebt. Zu vielen Themen, die es nicht in das Buch geschafft haben, ließe sich noch Weiteres ausführen, z. B. zu der Simulation von Energiesystemen, dem Demand Side Management oder Smart Cities. Das Ziel dieses Lehrbuchs ist aber keine erschöpfende Behandlung aller nur denkbaren Themen der Energieinformatik, sondern eine erste Einführung für Studierende und Anwender.

Literatur

1. *E-Energy Abschlussbericht – Ergebnisse und Erkenntnisse aus der Evaluation der sechs Leuchtturmprojekte*. Baum Consult GmbH, Mai 2014. URL: <https://www.digitale-technologien.de/DT/Redaktion/DE/Downloads/abgesamt-begleitforschung.pdf> (besucht am 09.06.2020).
2. *Smart Energy made in Germany – Erkenntnisse zum Aufbau und zur Nutzung intelligenter Energiesysteme im Rahmen der Energiewende*. Bundesministerium für Wirtschaft und Energie (BMWi), Mai 2014. URL: https://www.ppc-ag.de/wp-content/uploads/2017/8/BMWi-E-Energy_Abschlussbrosch_Mai_2014.pdf (besucht am 10.06.2020).
3. Fabian Rigoll. „Nutzerorientiertes Energiedatenmanagement“. Dissertation. Karlsruhe: KIT-Bibliothek, 2017. URL: <http://digibib.ubka.uni-karlsruhe.de/volltexte/1000068109>.
4. Bundesministerium für Wirtschaft und Energie (BMWi), Hrsg. *7. Energieforschungsprogramm der Bundesregierung*. Sep. 2018. URL: <https://www.bmwi.de/Redaktion/DE/Publikationen/Energie/7-energieforschungsprogrammder-bundesregierung.pdf> (besucht am 05.09.2010).
5. *Konferenzkalender*. 2019. URL: <https://www.strommarkttreffen.org/energy-conferences>.
6. *Energieinformatik*. 2020. URL: <https://de.wikipedia.org/wiki/Energieinformatik> (besucht am 21.11.2019).

7. Studiengang Energieinformatik an der Carl von Ossietzky Universität, Oldenburg. URL: <https://uol.de/ei> (besucht am 14.07.2020).
8. Studiengang Energietechnik und Ressourcenoptimierung an der Hochschule Hamm-Lippstadt. URL: <https://www.hshl.de/studieren/studiengaenge/bachelorstudiengaenge/energietechnik-und-ressourcenoptimierung/> (besucht am 14.07.2020).



Grundlagen

2

In diesem Kapitel werden zunächst einige Grundlagen vermittelt, die für das weitere Verständnis der nachfolgenden Kapitel von besonderer Bedeutung sind.

Die ersten beiden Abschnitte wenden sich dabei an InformatikerInnen, die bislang noch keine tieferen Kenntnisse über die Funktionsweise der Energiewirtschaft erlangt haben. Abschn. 2.1 stellt zunächst die Wertschöpfungskette der deutschen Energiewirtschaft dar. Es wird beschrieben, welche Aufgaben im Rahmen des technisch-physikalischen und des kaufmännischen Energieflusses entstehen und welche Akteure dabei miteinander interagieren. Dabei konzentrieren wir uns in diesem Lehrbuch auf die Versorgung mit leitungsgebundener Energie. Der Hauptfokus wird dabei auf die elektrische Energie gelegt, da diese besondere Ansprüche für Steuerung und Regelung stellt. Durch die sog. Energiewende entstehen dabei aktuell ganz neue Herausforderungen, denen insbesondere auch durch Methoden und Verfahren der Energieinformatik begegnet werden soll. Diese Herausforderungen und die entsprechenden Konsequenzen werden in Abschn. 2.2 dargestellt.

Die daran anschließenden Abschnitte wenden sich an EnergiewirtschaftlerInnen. Abschn. 2.3 zeigt einige Veränderungen auf, die im Rahmen der allgegenwärtigen Digitalisierung und der weiten Verbreitung verschiedener Technologien für die Energiewirtschaft entstehen. Die Abschn. 2.4, 2.5 und 2.6 bauen dann das notwendige Rüstzeug auf, damit im weiteren Verlauf des Buches die Informationsmodelle und Kommunikationsprotokolle dargestellt werden können, die in der Energiewirtschaft eine besondere Bedeutung besitzen.

Abgerundet wird das Kapitel durch die Darstellung des SGAM, eines Vorschlags der IEC für eine allgemeine Systemarchitektur des Smart Grid. Diese Architektur beschreibt, wie die Komponenten eines Energiesystems miteinander interagieren müssen, um das intelligente und herstellerunabhängige Stromnetz von morgen realisieren zu können. Das

SGAM bezieht sich dabei auf eine ganze Reihe von weiteren Standards, die im Laufe dieses Lehrbuchs näher besprochen werden.

2.1 Energieflüsse

Unser modernes Leben benötigt ständig **Energie**. Energie besitzt dabei die Fähigkeit, mechanische Arbeit zu verrichten, Wärme abzugeben oder Licht auszustrahlen. Streng genommen kann Energie nicht verbraucht oder erzeugt, nur umgewandelt werden. Dennoch hat sich der Begriff *Energieverbrauch* weitestgehend etabliert.

Die international anerkannte SI-Einheit für Energie ist Joule. Ein Joule entspricht dabei genau einer Wattsekunde (Ws), also einer Sekunde lang die Leistung von einem Watt. Entsprechend gilt:

$$\begin{aligned}
 1kWh &= 1kW \times 1h \\
 &= 1000W \times 1h \\
 &= 1000Wh = 1000W \cdot 3600s \\
 &= 3,6 \times 10^6 J = 3,6MJ
 \end{aligned} \tag{2.1}$$

Zum Vergleich:

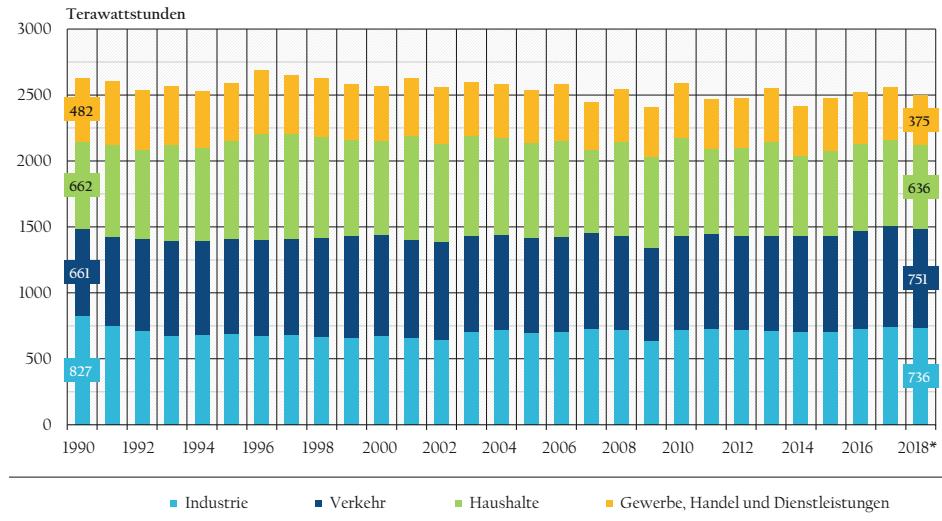
- 1 Watt entspricht ungefähr der Leistung des menschlichen Herzens,
- 20 Watt entsprechen ungefähr der Leistung des menschlichen Gehirns (siehe [35]).

Im Jahr 2016 wurden in Deutschland ca. 2542 Terrawattstunden (TWh) an Endenergie verbraucht. 2542 TWh sind 9151,2 Petajoule (PJ), wobei ein Petajoule für $1 \cdot 10^{15}$ Joule steht. Als Endenergie wird diejenige Energiemenge bezeichnet, die beim Endnutzer ankommt. Etwaige Umwandlungs- oder Übertragungsverluste sind darin nicht enthalten.

Traditionell wird der Energieverbrauch in mehrere **Energiesektoren** aufgeteilt. Man unterscheidet dabei die Bereiche Industrie, Haushalte, Gewerbe/Handel/Dienstleistungen und Verkehr. Abb. 2.1 zeigt die Entwicklung des Endenergieverbrauchs in Deutschland in den Jahren 1990 bis 2018. Insgesamt ist der Endenergieverbrauch in dieser Zeit in Deutschland leicht gesunken. Nur im Verkehrssektor ist der Endenergieverbrauch in dieser Zeit deutlich von 661 TWh im Jahr 1990 auf bis zu 751 TWh im Jahr 2018 gestiegen.

Für den Endenergieverbrauch muss die Energie in einer bestimmten Form bereitgestellt werden. Diese Endenergieträger sind meist Erdölprodukte wie Benzin oder Diesel, Erdgas oder elektrische Energie. Diese Energieformen müssen allerdings durch Umwandlung zunächst aus Primärenergieträgern gewonnen werden. Primärenergieträger sind:

Entwicklung des Endenergieverbrauchs nach Sektoren



* vorläufige Angaben

Quelle: Umweltbundesamt auf Basis AG Energiebilanzen, Auswertungstabellen zur Energiebilanz der Bundesrepublik Deutschland 1990 bis 2018, Stand 10/2019

Abb. 2.1 Entwicklung des Endenergieverbrauchs in Deutschland zwischen 1990 und 2018 nach Sektoren. Bildquelle: Umweltbundesamt. <https://www.umweltbundesamt.de/daten/energie/energie-verbrauch-nach-energietaegern-sektoren>

- fossile Energieträger wie Steinkohle, Braunkohle, Erdgas und Erdöl,
- Kernenergie,
- regenerative Energieträger wie Sonne, Wind, Biomasse, Wasserkraft oder Geothermie.

Elektrische Energie ist ein Endenergieträger, der sich als besonders vielseitig erwiesen hat. Er kann leicht übertragen und wieder in andere Energieformen, z. B. Wärme, Licht und Bewegung, umgewandelt werden. Um elektrische Energie nutzen zu können, muss diese aber erst aus den Primärenergieträgern gewonnen, man sagt auch *erzeugt* werden.

Elektrische Energie wird im Allgemeinen anhand von Ladungsbewegungen in Leitern übertragen. Diese Ladungsbewegung wird durch die elektrische Stromstärke (Einheit Ampere) und durch die elektrische Spannung (Einheit Volt) charakterisiert. Die Stromstärke beschreibt dabei die Größe eines Ladungsflusses pro Zeit zwischen zwei Punkten. Die Spannung beschreibt dabei quasi den Druck oder die Kraft auf diese Ladungen. Ohne Spannung also kein Strom. Im elektrischen Stromkreis ist die Leistung P gleich dem Produkt der Stromstärke mit der Spannung: $P = U \cdot I$. Diese Leistung kann über die Zeit hinweg genutzt werden, um Arbeit zu verrichten, z. B. im Elektromotor (Bewegung) oder in einer Heizspule (Wärme). Bei der Übertragung der Energie entstehen Leistungsverluste,

die proportional zum Quadrat der Stromstärke sind. Man bevorzugt für den Energietransport daher hohe Spannungen gegenüber hohen Strömen.

Bewegen sich die Ladungen in einem Leiter gleichförmig in eine Richtung, wird dies als **Gleichstrom** bezeichnet. Wechseln die Ladungen aber wiederholt ihre Richtung, spricht man von **Wechselstrom**. Bei Wechselstrom pulsieren Strom und Spannung mit einer gleichförmigen Frequenz. In Europa liegt diese Frequenz (außer bei Bahnstrom) bei 50 Hz.

Die Energieübertragung kann auch mit Wechselstrom realisiert werden, was einige Vorteile aufweist. Wechselstrom ist zunächst durch die Drehung in Generatoren leicht zu erzeugen, was eine Folge der Kreisbewegung ist. Besonders hilfreich sind bei Wechselstrom aber sog. Transformatoren. Diese sind in der Lage, Spannung und Strom bei gleichbleibender Leistung ineinander zu überführen, sodass für den Energietransport über lange Strecken hohe Spannungen genutzt werden können.

Die Nutzung von Wechselstrom weist aber auch Nachteile auf. Spulen oder Kondensatoren nehmen beim Verbraucher Ladung auf und geben diese dann zeitverzögert wieder an das Netz ab. Dies geschieht durch den Auf- und Abbau von elektrischen oder magnetischen Feldern. Diese Ladung muss zwar transportiert werden, erzeugt aber sonst keine Wirkung beim Verbraucher. Die Leistung, die dabei verloren geht, wird als **Blindleistung** bezeichnet. Blindleistung belastet das Netz, leistet aber sonst keinen Beitrag, d. h., Blindleistung sollte beim Energietransport möglichst vermieden werden, was mitunter technisch aufwendig ist. Bei einem einphasigen Wechselstrom pulsiert zudem die übertragene Energiemenge mit doppelter Netzfrequenz. Da eine möglichst gleichförmige Energiemenge übertragen werden soll, wird heute meist **Dreiphasendrehstrom** eingesetzt.

Erdgas bindet Energie chemisch in Form von Kohlenwasserstoffen. Diese kann durch Verbrennung genutzt werden. Der Hauptbestandteil von natürlichem Erdgas ist Methan. Der Energiegehalt (Heizwert) hängt von der genauen Zusammensetzung des Gasgemisches ab und kann zwischen 10 kWh/kg (sog. L-Gas) und 14 kWh/kg (sog. H-Gas) schwanken. Erdgas wird anhand von Rohrleitungen unter hohem Druck transportiert. Aufgrund von Wandreibung und Turbulenzen fällt der Druck in den Rohrleitungen mit zunehmender Länge ab. Der Druckverlust steigt zudem quadratisch mit der Strömungsgeschwindigkeit des Gases an. Um den Druckverlust zu kompensieren, muss der Druck über die Länge der Rohrleitungen immer wieder erhöht werden. Dies geschieht in **Verdichterstationen**. Bei Transport über größere Strecken wird ein erheblicher Teil der Energie für den Pumpenbetrieb aufgewendet.

Elektrischer Strom und Erdgas haben einige gemeinsame Eigenschaften:

- **Leitungsgebunden**

Strom und Gas werden über Netze übertragen und verteilt. Diese Netze stellen ein natürliches Monopol dar. Der diskriminierungsfreie Zugang wird entsprechend durch die Bundesnetzagentur (BNetzA) überwacht. Im elektrischen Übertragungsnetz

in Deutschland existieren aktuell vier Regelzonen, bei Erdgas existieren zwei sog. Marktgebiete.

- **Mischprodukt**

Sowohl bei Strom als auch bei Gas existiert keine direkte Zuordnung zwischen Entnahme und Einspeisung (Mischprodukt). Bei elektrischer Energie spricht man daher auch vom Strommix aus unterschiedlichen Quellen.

- **Fungibel**

Elektrizität ist ein homogenes Produkt mit gleichbleibenden Eigenschaften (Funktionalität). Sie kann daher leicht börslich gehandelt werden. Bei Gas existieren zwar unterschiedliche Brennwerte (z. B. H-/L-Gas), aber auch Gas wird börslich gehandelt.

Erdgas kann mit überschaubarem Aufwand gespeichert werden. Gas ist kompressabel und kann daher in entsprechenden Speichern (z. B. Kavernen- oder Porenspeichern) vorgehalten werden. Dies wird auch im großen Stil so umgesetzt. Im Sommer wird z. B. das Erdgas in die Gasspeicher eingespeist, im Winter dann entnommen. Im Gasnetz ist es auch wegen der Speicherfähigkeit des Rohrleitungssystems weniger kritisch, wenn Entnahme und Einspeisung sich kurzfristig nicht genau überdecken. Für elektrische Energie ist dies jedoch anders. Es existiert aktuell keine wirklich effiziente und preiswerte Speichertechnologie von nennenswertem Ausmaß oder vergleichbarem Energiegehalt wie bei Gasspeichern. Pumpspeicher sind zwar technisch erprobt, aber neue Anlagen sind in Deutschland kaum noch umzusetzen (NIMBY-Effekt). Auch chemische Speicher, wie z. B. Batterien, sind bislang nicht in dem Umfang verfügbar- bzw. bezahlbar. Dies ist ein Hauptproblem der elektrischen Energieversorgung. Mithilfe von Speichern wären auch die Netzregelung und ein weiterer Ausbau von regenerativen Energien leichter umzusetzen.

Die Konsequenz ist signifikant. Zu jedem Zeitpunkt muss in das Stromnetz genau so viel elektrische Leistung eingespeist werden, wie gerade entnommen wird. Ansonsten bricht die Versorgung mit elektrischer Energie zusammen. Diese Deckung von Einspeisung und Entnahme bezeichnet man als **Lastdeckung**. Die **Netzfrequenz** ist dabei der wichtigste Indikator, ob eine Unter- oder Überdeckung vorhanden ist. Entspricht im Netz die Last (Energieentnahme) der Einspeisung, beträgt die Netzfrequenz exakt 50 Hz. Ist zu viel Leistung im Netz, steigt die Netzfrequenz, ist zu wenig Leistung im Netz, sinkt die Netzfrequenz.

Leitungsgebundene Energieformen wie elektrischer Strom und Erdgas benötigen Netze, damit die Energie zum Endkunden transportiert werden kann. Elektrische Energienetze basieren auf einer komplexen Struktur aus Leitern, wie z. B. Freileitungen und Erdkabeln. Der strukturelle Aufbau solcher Netze wird auch als Netztopologie bezeichnet. Die Topologie wird meist in Form von **Einliniendiagrammen** dargestellt, welche eine Form eines Schaltplans darstellen, die alle relevanten Elemente des Netzes, z. B. Leitungen, Generatoren oder Transformatoren, beinhaltet. Dreiphasige Netze werden dabei meist vereinfacht dargestellt, indem die drei Leiter zu nur einem zusammengefasst werden. Abb. 2.2 zeigt als Beispiel das Einliniendiagramm eines Kohlekraftwerks.

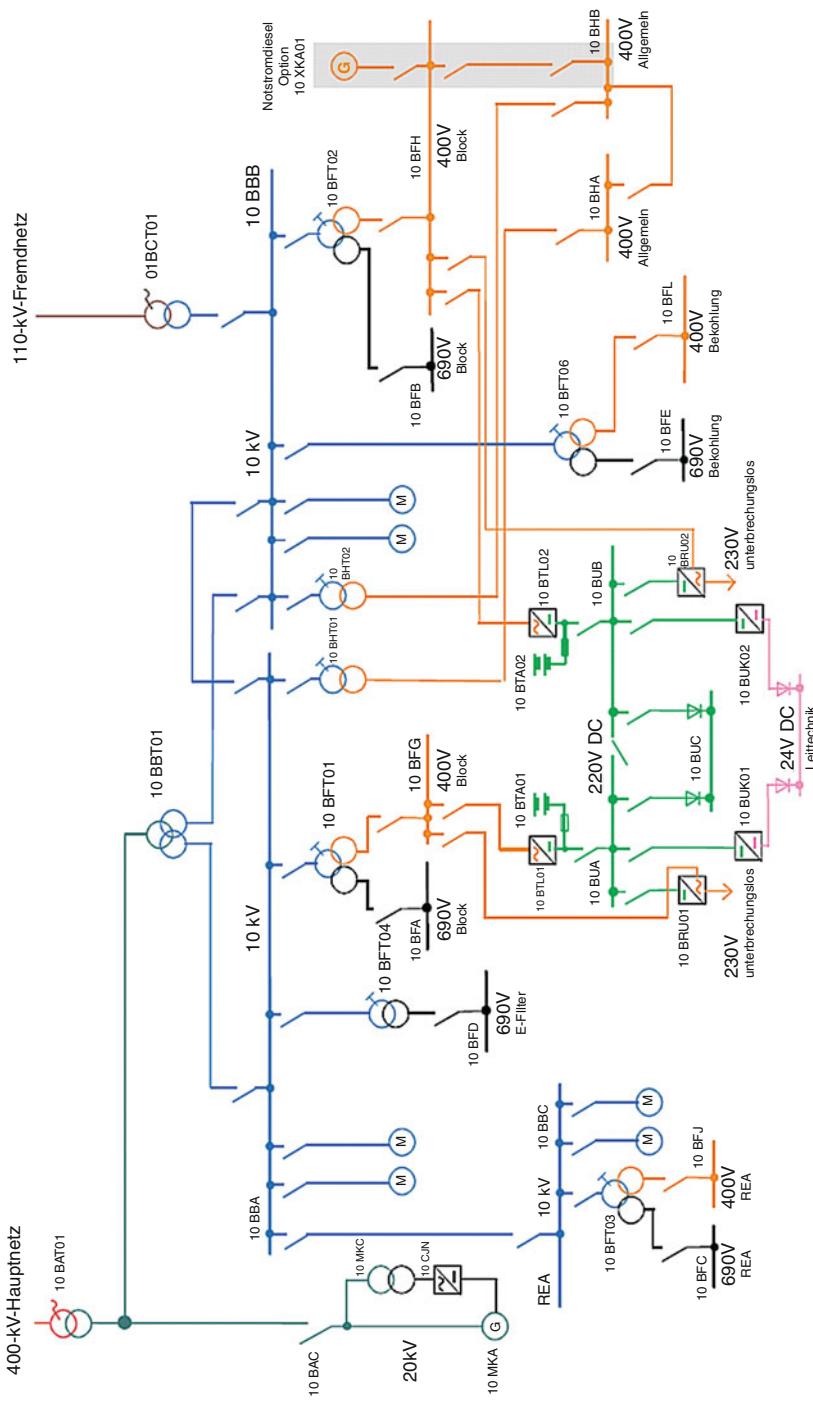


Abb. 2.2 Einliniendiagramm eines Kohlekraftwerks. Bildquelle: VBG Powertech e. V. <https://commons.wikimedia.org/wiki/File:Energieverteilung01.png>, <https://creativecommons.org/licenses/by-sa/3.0/legalcode>

Die Übertragung der Energie über die Netze basiert auf physikalisch-technischen Gesetzen, wie den Kirchhoff'schen Regeln, den Maxwell'schen Gleichungen oder der Thermodynamik. Die Netze verbinden dabei die Energiequellen (z. B. Kraftwerke) mit den Endkunden. Erzeugung, Übertragung, Verteilung und Verbrauch der Energie basiert dabei auf einem geordneten Prozess, an dem mehrere voneinander unabhängige Akteure beteiligt sind. Man spricht auch vom **physikalisch-technischen Energiefluss**. Abb. 2.3 zeigt den physikalisch-technischen Energiefluss, der bei der Erzeugung beginnt und beim Endkunden mit Messung und Abrechnung endet.

Die Entnahme der Energie durch den **Endkunden** (z. B. Haushaltkunden) erfolgt meist spontan und unangemeldet. Die Nachfrage variiert dabei stark nach einer Vielzahl von Faktoren, z. B. Tages- und Wochenrhythmus, Temperatur, Helligkeit oder Preis. Es ist daher mitunter schwierig bis unmöglich, den Verbrauch korrekt zu prognostizieren. Auch für die Dimensionierung der Netze ist der Unterschied zwischen sog. Basis- und Spitzenlast eine Herausforderung. Es existieren allerdings gewisse Grundmuster, die sich nach Tageszeit, Wochentag usw. richten. Diese finden sich auch in Form von sog. Standardlastprofilen wieder.

Um Leitungsverluste zu minimieren, existieren sowohl bei Erdgas als auch bei elektrischem Strom Hierarchien innerhalb der Netze. Bei elektrischen Netzen unterscheidet man mindestens zwischen den überregionalen Übertragungs- und den regionalen Verteilnetzen. Bei Erdgas wird ebenfalls zwischen einem überregionalen Ferntransportnetz (z. B. 84 bar Druck) und dem regionalen Verteilnetz (bis ca. 16 bar Druck) unterschieden.

Das elektrische **Übertragungsnetz** dient zur überregionalen Übertragung von Energie mit sehr hohen Spannungen (220–380 kV). Es ist hochgradig vermascht, d. h., beim Ausfall einer Teilstrecke kann die Energie meist über andere Wege transportiert werden. Die Energieflüsse ändern dabei in Abhängigkeit der Lasten mitunter ihre Richtung. Dies macht das Management der Übertragungsnetze kompliziert. Konventionelle Großkraftwerke sind i. d. R. an das Übertragungsnetz angeschlossen. Es macht den kleinsten Teil des gesamten Netzes aus, also ca. 2–3 % der Leitungslängen (siehe [25]). Das deutsche Übertragungsnetz ist in das europäische Verbundsystem integriert. Auf europäischer Ebene sind die Übertragungsnetze miteinander verbunden. Die Netzfrequenz ist synchronisiert und Energie kann zwischen den Ländern ausgetauscht werden. Dabei können Probleme in einem Land leicht Auswirkungen auf ganz Europa haben. Im Jahr 2018 hat ein Problem in der Kontrollregion Serbien, Mazedonien und Montenegro dazu geführt, dass die Netzfrequenz in ganz Europa über einen längeren Zeitraum unter 50 Hz lag. Die Konsequenz war auch in Deutschland spürbar. Viele Radiowecker und Küchenuhren

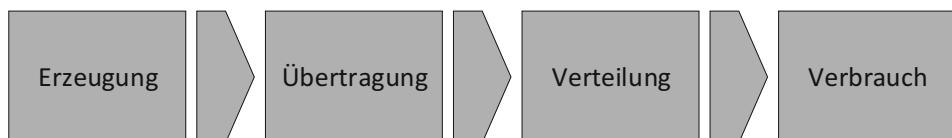


Abb. 2.3 Physikalisch-technischer Energiefluss von der Erzeugung bis zum Verbrauch

gingen eine Zeit lang nach (siehe [52]). Im Jahr 2006 kam es zu einem größeren Stromausfall in Europa und Teilen von Deutschland, als eine planmäßige Abschaltung zweier Hochspannungsleitungen bei der Ausschiffung eines Kreuzfahrtschiffes aus der Meyer Werft in Papenburg zu einer unkontrollierten Kettenreaktion führte (siehe [11]).

Das Übertragungsnetz teilt sich in Deutschland in vier **Regelzonen** auf. Zwischen diesen Regelzonen existiert ein bidirektonaler Energiefluss. Jede Regelzone wird durch ein eigenes Unternehmen geführt. Dies sind die **Übertragungsnetzbetreiber (ÜNB)** Tennet, Amprion, Transnet BW und 50Hertz. Die Übertragungsnetzbetreiber haben beim Betrieb der Netze insbesondere die Aufgabe der **Frequenzhaltung**. Die europäischen Übertragungsnetzbetreiber arbeiten dafür eng zusammen und sind dafür auch im Verband entso-e organisiert. Bei einer Abweichung der Netzfrequenz ergreifen die Übertragungsnetzbetreiber entsprechende Maßnahmen, um die Netzfrequenz zu stabilisieren. Bei einer zu geringen Netzfrequenz muss mehr Leistung, bei einer zu hohen Netzfrequenz weniger Leistung in das Netz eingespeist werden. Dafür wird dann sog. Regelleistung eingesetzt. Positive Regelleistung bedeutet, dass zusätzliche Leistung ins Netz eingespeist wird. Negative Regelleistung bedeutet, dass weniger Leistung ins Netz eingespeist wird. Regelleistung wird durch Kraftwerke oder Industrieanlagen erbracht. Da Netzbetreiber keine eigenen Anlagen betreiben dürfen (siehe Energiewirtschaftsgesetz (EnWG), Abschn. 3), wird diese in einem Auktionsverfahren ausgeschrieben.

Um die Energieflüsse in den Regelzonen bereits im Vorfeld überblicken zu können, existiert das Konzept der **Bilanzkreise**. Dieses Konzept existiert sowohl bei elektrischer Energie als auch bei Erdgas in leicht abweichender Form (siehe §20 EnWG). Bilanzkreise kann man sich als virtuelle Energiemengenkonten vorstellen. Sie dienen der Bilanzierung von Energiemengen und der Abrechnung von Ausgleichsenergie. Jede Entnahm- oder Einspeisestelle muss dabei genau einem Bilanzkreis zugeordnet sein. Jeder Erzeuger, Energiehändler oder Versorger verwaltet mindestens einen Bilanzkreis, er ist dann *bilanzkreisverantwortlich*. In der konkreten Ausgestaltung wird dazu ein Bilanzkreisvertrag mit dem ÜNB geschlossen, der auf einem Standardbilanzkreisvertrag basiert (siehe [57]). Jede Energielieferung wird anhand von Energieflüssen zwischen den Bilanzkreisen abgebildet. Kauft z. B. ein Stadtwerk Energie bei einem Erzeuger, wird Energie von einem Bilanzkreis in einen anderen verschoben. Verkauft dieses Stadtwerk wiederum die Energie an Endkunden, existieren wiederum mehrere Energieflüsse in die Bilanzkreise, in denen die Endkunden verortet sind. Alle Bilanzkreise müssen am Tag der Lieferung ausgeglichen sein, d. h., alle Entnahmen müssen durch entsprechende Einspeisungen gedeckt sein. Die ÜNB überwachen diesen Energiesaldo und stellen den Bilanzkreisverantwortlichen im Abweichungsfall Ausgleichsenergie in Rechnung.

Den Übertragungsnetzen sind die **Verteilnetze** untergeordnet. Sie machen den größten Teil der Leitungslängen aus. Die Verteilnetze sind ursprünglich darauf ausgelegt, die Energie unidirektional aus den Übertragungsnetzen zu beziehen. Der Energiefluss auf der Ebene der Verteilnetze ist (bisher) gerichtet, was das Management der Verteilnetze (noch) überschaubar komplex macht. Verteilnetze sind nicht untereinander verbunden, ein Energieaustausch über größere Strecken findet nur anhand des Übertragungsnetzes statt.

Verteilnetze sind meist radial als Baumstruktur oder als Ring angeordnet. Die Verteilnetze werden durch die sog. Verteilnetzbetreiber (VNB) betrieben. Dies sind meist regionale Gesellschaften, wie z. B. Stadtwerke. Die Verteilnetzbetreiber haben beim Betrieb der Netze insbesondere die **Spannungshaltung** als Aufgabe. Dafür werden z. B. Transformatoren mit schaltbarem Übersetzungsverhältnis oder steuerbare Blindleistungskompensationsanlagen genutzt. Auf der Ebene der Verteilnetze speisen fast alle Solaranlagen und knapp 90 Prozent der Windkraftanlagen ihre Energie ein (siehe [19]). Dies stellt eine zunehmende Herausforderung dar.

Entsprechend der Last muss die richtige Menge elektrischer Leistung zeitgleich in das Netz eingespeist werden. Elektrische Energie muss dabei zunächst aus einer anderen Energieform umgewandelt werden. Hierfür stehen unterschiedliche Technologien zur Verfügung. Im Jahr 2017 haben in Deutschland folgende Technologien zur Nettostromerzeugung beigetragen (siehe [44, S. 11]):

- thermische Erzeugung: Braunkohle = 24,3 %, Steinkohle = 14,3 %, Erdgas = 8,9 %
- nuklear = 13,1 %
- erneuerbare Energien: Wasser, Biogas, PV, Wind = 38,2 %

Über die Einsatzreihenfolge dieser Erzeugungseinheiten bestimmen sowohl Politik als auch der Markt. Auf Basis des Erneuerbare-Energien-Gesetz (EEG) haben erneuerbare Erzeuger zunächst Vorrang, dann erst werden sog. konventionelle Erzeugungseinheiten eingesetzt. Bei der konventionellen Erzeugung gilt das Prinzip der kostenminimalen Lastdeckung. Um eine bestimmte Last zu decken, werden nach den Produktionskosten sortiert zunächst die günstigsten Kraftwerke eingesetzt, danach die nächstteureren usw. Die Einsatzreihenfolge der Kraftwerke wird dabei auch als **Merit Order** bezeichnet. Bei dieser Staffelung sind lediglich die sog. Grenzkosten, also die variablen Kosten der Erzeugungseinheiten entscheidend. Im Gegensatz dazu werden die Vollkosten der Stromerzeugung mithilfe einer bestimmten Technologie als **Gestehungskosten** bezeichnet. Die Gestehungskosten reichen aktuell von ca. 22 €ct pro kWh für Gaskraftwerke bis zu unter 4 €ct für Solarmodule. Abb. 2.4 zeigt die Stromgestehungskosten für regenerative Energiequellen und konventionelle Kraftwerke an Standorten in Deutschland im Jahr 2018 (siehe [49]).

Klassischerweise speisen Großkraftwerke, welche die elektrische Energie auf Basis fossiler Brennstoffe oder nuklear erzeugen, direkt in das Übertragungsnetz ein. Der Großteil der Verbraucher entnimmt die Energie hingegen aus den Verteilnetzen, welche die Energie wiederum aus dem Übertragungsnetz beziehen. Abb. 2.5 zeigt die klassische Einspeise- und Entnahmestruktur in Deutschland.

Wie wir gesehen haben, hat die Versorgung mit leitungsgebundener Energie viele physikalisch-technische Nebenbedingungen. Darüber hinaus sind die Unternehmen, die an der Versorgung von Endkunden mit Energie beteiligt sind, reguläre Wirtschaftsunternehmen. Sie betreiben Wertschöpfung, d. h., sie haben ein Interesse, Gewinne zu erwirtschaften. In dieser Perspektive spielt z. B. der Handel mit Energieprodukten eine

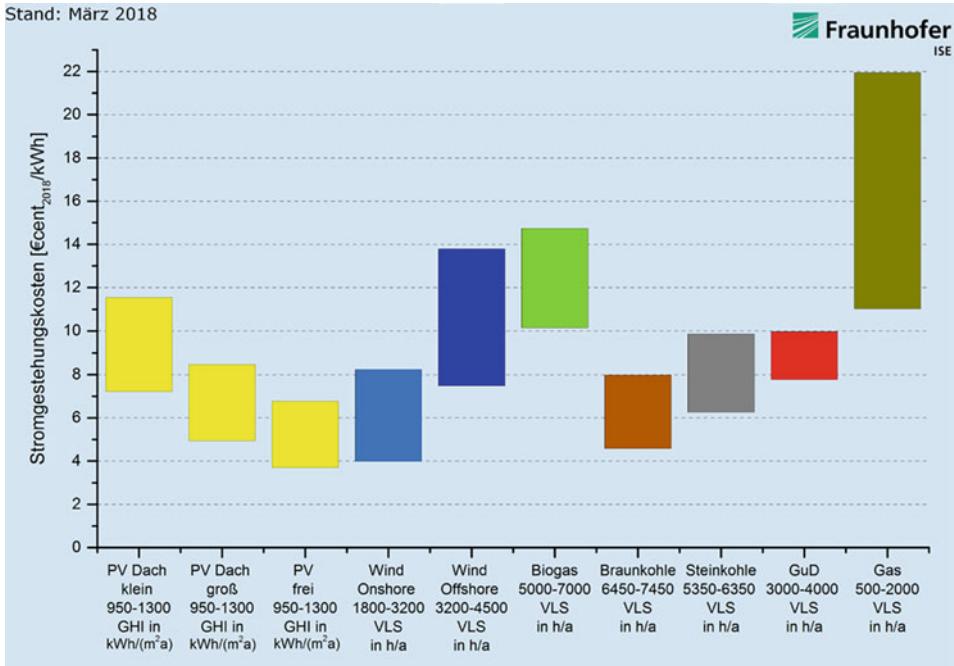


Abb. 2.4 Stromgestehungskosten für erneuerbare Energien und konventionelle Kraftwerke in Deutschland im Jahr 2018

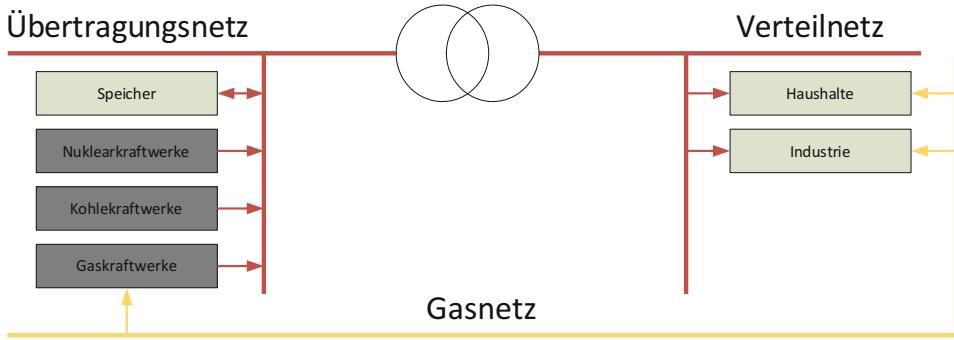


Abb. 2.5 Klassische Einspeisestruktur

große Rolle. Es entstehen Märkte, an denen sich Versorger mit Energie eindecken können. Erzeuger können diese Märkte nutzen, um ihre Marktrisiken mithilfe von Termingeschäften zu reduzieren. Im Kontext des Handels mit Energieprodukten spricht man auch von dem **kaufmännischen Energiefluss** (siehe [53, S. 50] bzw. [16, S. 6]). Abb. 2.6 zeigt den kaufmännischen Energiefluss. Er beginnt ebenfalls mit der Erzeugung und endet mit der Messung und Abrechnung beim Endkunden.

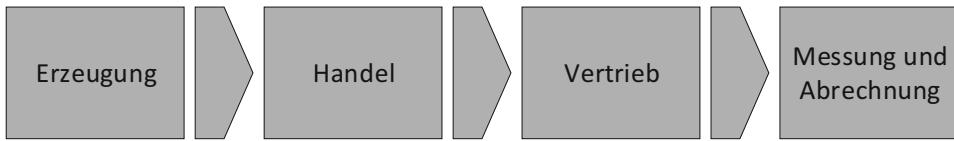


Abb. 2.6 Kaufmännischer Energiefluss

Kern und Träger des kaufmännischen Energieflusses ist die **Energiewirtschaft**. Ziele und Aufgaben der Energiewirtschaft sind im EnWG festgehalten. Als Ziele werden eine sichere, preisgünstige, verbraucherfreundliche, effiziente und umweltverträgliche leitungsgebundene Versorgung der Allgemeinheit mit Strom und Gas vorgegeben, die zunehmend auf erneuerbaren Energien beruht. Zudem wird ein wirksamer und unverfälschter Wettbewerb angestrebt. Die Ziele haben dabei durchaus bewusst eine Reihenfolge: Die sichere Versorgung wird als Erstes genannt, danach erst folgenden die weiteren Ziele, wie der Preis usw. Träger der Energiewirtschaft sind die Energieversorgungsunternehmen (EVU). Sie erzeugen, handeln oder übertragen elektrische Energie bzw. beliefern Endkunden mit Energieprodukten, wie elektrischer Energie, Erdgas oder Fernwärme. Ein Großteil der konventionellen Erzeugungskapazität liegt dabei zwar noch immer in der Hand einiger großer Unternehmen, im Energievertrieb und bei den Verteilnetzen finden sich aber viele Hundert sonstiger Unternehmen, i. d. R. Stadtwerke, regionale Erzeuger und Versorger. Abb. 2.7 zeigt die Anzahl und die Verteilung von Unternehmen auf die einzelnen Marktbereiche in der deutschen Energiewirtschaft mit dem Stand Januar 2020.

Im Jahr 1998 wurde durch Neuregelung des Energiewirtschaftsrechts die Energiewirtschaft **liberalisiert**. Bis dahin besaßen die Energieversorgungsunternehmen in ihren Marktgebieten ein Monopol und waren vollständig vertikal integriert. Ein solcher Energieversorger erbrachte in einem bestimmten Marktgebiet sämtliche Dienstleistungen wie Erzeugung, Verteilung und Vertrieb vollständig aus einer Hand bzw. bediente sich einer Vertriebspartnerschaft mit Stadtwerken. Die Liberalisierung des Energiewirtschaftsgesetzes sorgte zunächst für ein **Unbundling**, also die Entkopplung von Erzeugung, Übertragung und Verteilung von elektrischer Energie. Ehemals integrierte Unternehmen mussten entsprechend aufgespalten werden. Der Netzbetrieb wird seitdem als natürliches Monopol reguliert und durch die Bundesnetzagentur (BNetzA) überwacht. Es gilt zudem ein allgemeiner Netzanschluss- und Netzzugangsanspruch. Elektrischer Strom (später Erdgas) wurde zum Handelsgut an Großhandelsmärkten (z. B. Börse). Endkunden können infolge ihrer Energielieferanten frei wählen. Spätestens seit der Liberalisierung stehen die Unternehmen der Energiewirtschaft im Wettbewerb miteinander. Sie müssen effizient wirtschaften und ihre Wertschöpfung optimieren, um Gewinn zu erzielen. Durch gesetzliche Vorgaben haben sich mit der Zeit immer spezifischere Unternehmensarten entwickelt. Diese Entwicklung ist noch nicht abgeschlossen und mit Energiewende und Digitalisierung werden sicher neue Unternehmen entstehen. Abb. 2.8 zeigt diese zeitliche Entwicklung der Spezialisierung von Unternehmen in der Energiewirtschaft.

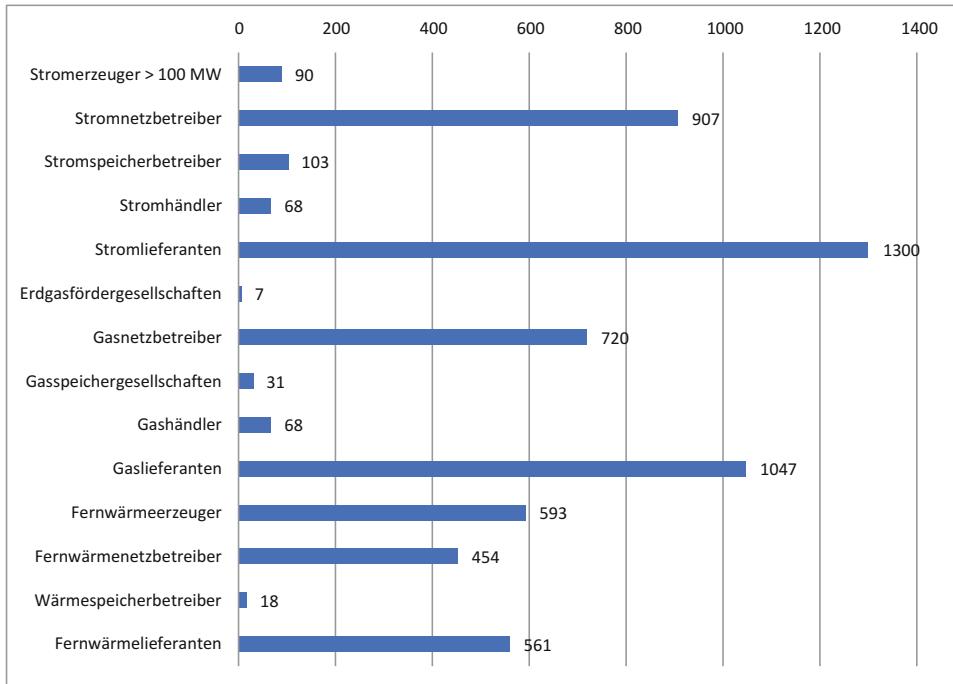


Abb. 2.7 Anzahl der Unternehmen am Energiemarkt in Deutschland nach Tätigkeitsfeld im Jahr 2018. Datenquelle: Bundesverband der Energie- und Wasserwirtschaft e. V. (BDEW), siehe https://www.bdew.de/media/documents/Marktteilnehmer_Energie_aktuell_online_o_halbjährlich_Ki_27022020.pdf

Der kaufmännische Energiefluss lässt die Rolle des **Energievertriebs** entstehen. Der Vertrieb beliefert einen Endkunden an einer Messstelle mit einem Energieprodukt (Strom oder Gas). Der Vertrieb ist das Kerngeschäft vieler EVU, wie z. B. den Stadtwerken. Dazu bedarf es vielfältiger Aktivitäten (siehe auch [53, S. 301 f.]). Im Vertrieb muss z. B. die zu liefernde Energie beschafft werden. Dies mündet im **Portfoliomangement**, welches wiederum aus einer Menge von Einzelaktivitäten besteht, z. B. die Analyse des Absatz- und Beschaffungspotfolios, die Ermittlung des Mengen-, Leistungs- und Flexibilitätsbedarfs, die Wahl einer Beschaffungsstrategie und der Handel mit Energieprodukten an Großhandelsmärkten (siehe [16]).

Zum Energievertrieb gehört auch die Rechnungsstellung, die entsprechende Messwerte benötigt, die beim Endkunden erfasst werden müssen. Technisch werden dazu entsprechende Messeinrichtungen genutzt. Bei Haushaltskunden kommen dabei meist noch sog. Ferraris-Zähler zum Einsatz, bei denen die Messwerte manuell, einmal jährlich erfasst und übermittelt werden müssen. Industriekunden bzw. Besitzer von regenerativen Erzeugungsanlagen (z. B. Fotovoltaikanlagen) verfügen meist über elektronische Zähler. Der Einsatz von Smart-Metern, bei denen die Messwerte digital erhoben und an den Messstellenbetreiber übertragen werden, ist in Deutschland noch nicht sehr weitverbreitet.

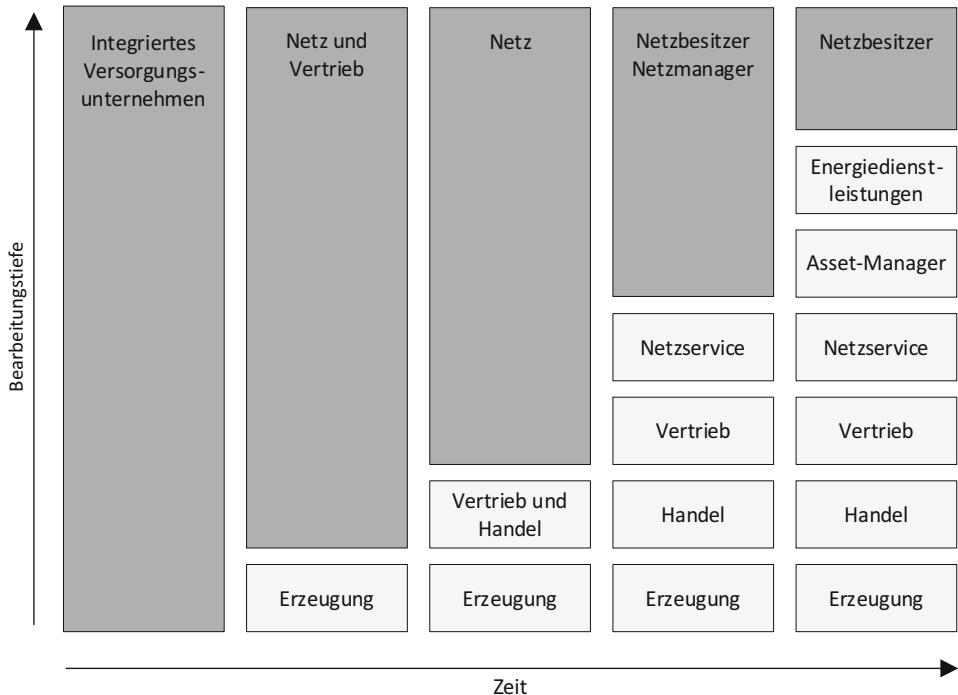


Abb. 2.8 Zunehmende Spezialisierung der Unternehmen in der Energiewirtschaft. In Anlehnung an [24, S. 8]

Eine weitere wichtige Rolle im kaufmännischen Energiefluss ist die des **Energiehändlers**. Diese Rolle muss von vielen Akteuren im Versorgungsmarkt eingenommen werden, z. B. den Kraftwerksbetreibern oder den Energieversorgern. Energiehändler kaufen oder verkaufen Energieprodukte (*Commodities*) auf einem **Großhandelsmarkt**. Energieprodukte können dabei standardisierte Stromlieferkontrakte wie *Base-* oder *Peakload* sein, Brennstoffe wie Steinkohle, Erdgas oder Erdöl als auch CO₂-Zertifikate und exotischere Produkte wie Wetterderivate. Für den Handel mit Energieprodukten stehen unterschiedliche Märkte zur Verfügung. Standardisierte Produkte, wie *Base-* oder *Peakload* können anonym an Energiebörsen wie der European Energy Exchange (EEX) gehandelt werden. Auch der direkte Handel zwischen Unternehmen ist nicht ungewöhnlich, was auch als Over the Counter (OTC) bezeichnet wird. Für Regelleistung stehen eigene Plattformen zur Verfügung, auf denen der Handel meist auf Basis von Auktionen abgewickelt wird.

Der physikalisch-technische und der kaufmännische Energiefluss sind nicht getrennt voneinander zu betrachten. Markt und Netz haben einen engen Bezug zueinander. Ein gemeinsamer Markt mit gleichen Preisen setzt voraus, dass Energie größtenteils ungehindert transportiert werden kann (siehe [38]). Dass aber Windenergie aktuell aus Norddeutschland nicht ungehindert nach Süddeutschland transportiert werden kann, hatte

zur Konsequenz, dass das ursprünglich gemeinsame Marktgebiet für Deutschland und Österreich im Jahr 2018 in zwei Marktgebiete aufgespalten werden musste.

Das Ziel der Energiewirtschaft ist es zwar, so viel Markt wie möglich zuzulassen, die Systemstabilität aber dabei nicht zu gefährden. Werden im Netzbereich kritische Systemgrenzen erreicht, z. B. bei Spannung, Netzfrequenz oder Betriebsmittelauslastung, ist es den Netzbetreibern daher erlaubt, Eingriffe in den Stromhandel vorzunehmen. Ein Beispiel einer solchen Maßnahme ist der **Redispatch**, also die Anpassung der Leistungseinspeisung von Erzeugern.

Der BDEW hat den Zusammenhang zwischen Netz und Markt mithilfe eines Ampelkonzepts dargestellt (siehe z. B. [41, S. 61 ff.]). Im roten Bereich finden sich Anwendungsfälle, welche die Systemstabilität sicherstellen. Der gelbe Bereich stellt den Übergangsbereich vom Netz- zum Marktbetrieb dar, im grünen Bereich finden sich Anwendungsfälle für die marktorientierte Bewirtschaftung des Energiesystems.

2.2 Energiewende

Im letzten Abschnitt wurden einige Grundbegriffe zur leitungsgebundenen Energieversorgung dargestellt. Dabei spielen die Aspekte des physikalisch-technischen und des kaufmännischen Energieflusses eine herausragende Rolle. Die Energiewirtschaft ist allerdings derzeit stark in Bewegung. So zeigte das Beratungshaus Price Waterhouse Coopers bereits im Jahr 2016, dass die Energiewirtschaft aktuell durch zwei Megatrends beeinflusst wird. Die Energiewende und die Digitalisierung (siehe [33]). Beide Themenbereiche haben starke Auswirkungen auf den physikalisch-technischen als auch auf den kaufmännischen Energiefluss. Der folgende Abschnitt wird einige Hintergründer zur Energiewende darstellen.

Die Wissenschaft geht heute weitgehend von der Existenz des anthropogenen Treibhauseffekts aus. Der Treibhauseffekt wird dabei durch die Wirkung von bestimmten Treibhausgasen, z. B. CO₂, in der Erdatmosphäre auf die Temperatur am Erdboden erklärt. Durch das Verbrennen von Erdöl, Kohle und Gas wird allerdings zusätzliches CO₂ in die Atmosphäre eingebracht. In Konsequenz sorgt dies für eine Erwärmung der Erdatmosphäre, was auch als **Klimawandel** bezeichnet wird.

Der Weltklimarat (Intergovernmental Panel on Climate Change (IPCC)) hat in bislang fünf Sachstandsberichten zum Klimawandel Stellung bezogen. Eine letzte Stellungnahme wurde im Jahr 2013 finalisiert (siehe [22]), der 6. Bericht ist seit 2017 in Arbeit. Die zusammengetragenen Ergebnisse sprechen eine klare Sprache. Der Temperaturanstieg seit der vorindustriellen Zeit bis zum Jahr 2017 beträgt etwa 1 °C. Die Erderwärmung wird mit einer sehr hohen Wahrscheinlichkeit (>95 %) durch den menschgemachten Ausstoß von Treibhausgasen verursacht. Die Erderwärmung nimmt weiter zu, wenn die Emission von Treibhausgasen nicht (deutlich) reduziert wird, und eine weitere Erderwärmung hat schwerwiegende und unumkehrbare Konsequenzen.

Es gilt daher, die Erderwärmung zu bremsen bzw. zu stoppen. Der IPCC schlägt eine Grenze von max. 1,5 °C vor (siehe [48]). Um dieses Ziel zu erreichen, muss die Emission von Treibhausgasen deutlich reduziert oder gar gestoppt werden. Maßnahmen zur Begrenzung der Auswirkungen eines menschengemachten Klimawandels werden daher seit einiger Zeit auf verschiedenen nationalen und internationalen Ebenen ergriffen. Die Europäische Union hat beispielsweise das im Jahr 2005 geschlossene Kyotoprotokoll ratifiziert, in welchem verbindliche Zielwerte für den Ausstoß von Treibhausgasen festgelegt wurden. Im Rahmen dessen wurde durch die EU im Jahr 2007 das sog. Klima- und Energiepaket 2020 beschlossen. Dieses beinhaltet die sog. 20-20-20-Ziele:

- Senkung der Treibhausgasemissionen um 20 % (gegenüber 1990),
- 20 % der Energie aus erneuerbaren Energien,
- Verbesserung der Energieeffizienz um 20 %.

Deutschland hat sich darüber hinaus dazu verpflichtet, seine Treibhausgasemissionen bis zum Jahr 2020 um 40 %, bis 2030 um 55 %, bis 2040 um 70 % und bis 2050 um 80–95 % zu senken. Diese Ziele werden nach aktuellem Stand aber wohl nicht erreicht werden.

Eine Senkung des Ausstoßes von Treibhausgasen kann aber dann nur gelingen, wenn auch die Versorgung mit elektrischer Energie und Wärme auf regenerative Quellen umgestellt wird. Anstelle fossiler Energieträger müssen zunehmend regenerative Energiequellen wie Wasser-, Windkraft, Solar, Biogas oder Geothermie genutzt werden. Diese stoßen keine oder vergleichsweise wenig Klimagase aus und dienen somit dem übergeordneten Ziel, zu einer ökologisch nachhaltigen Energieversorgung zu kommen. Die Transformation der Energiesysteme in Richtung einer regenerativen Energieversorgung wird auch als **Energiewende** bezeichnet. Die Energiewende in Deutschland wurde nicht zuletzt durch das EEG eingeleitet. Es garantiert u. a. den erneuerbaren Energien eine bevorzugte Einspeisung in das Stromnetz (Anschluss- und Abnahmepflicht) und regelt eine feste Einspeisevergütung für die Erzeuger. Durch diese Förderung hat in den letzten Jahren ein starker Zubau erneuerbarer Energien stattgefunden. Abb. 2.9 zeigt den Anstieg des Anteils regenerativer Erzeuger an der Nettoerzeugung in Deutschland in den Jahren 2002 bis zum Mai 2020.

Der Anteil der erneuerbaren Energien an der Nettostromerzeugung der öffentlichen Stromversorgung ist vom Jahr 2002 mit 8,6 % auf über 46 % im Jahr 2019 gestiegen. Der CO₂-Ausstoß des deutschen Strommixes ist dadurch von 765 g/kWh im Jahr 1990 auf 480 g/kWh im Jahr 2018 gesunken. Insgesamt kann ein Rückgang der Treibhausgasemissionen in Deutschland verzeichnet werden. Abb. 2.10 zeigt den Rückgang der Treibhausgasemissionen mit dem Bezugsjahr 1990 bis zum Jahr 2017.

Die Vermarktung der regenerativ erzeugten elektrischen Energie übernehmen seit 2010 die vier Übertragungsnetzbetreiber. Der Differenzbetrag zwischen den erzielten Erlösen und den Kosten für die Einspeisevergütung wird in Form der EEG-Umlage auf

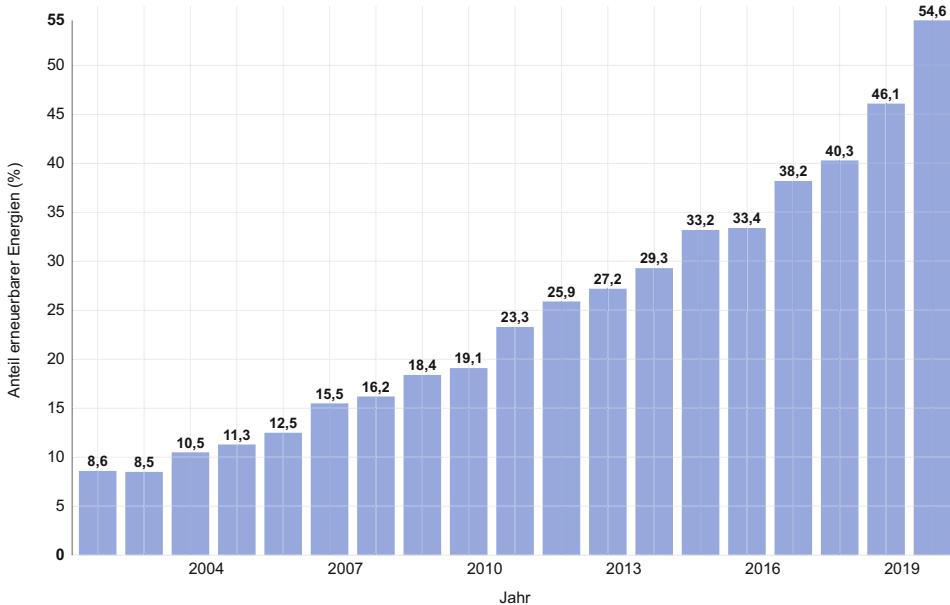


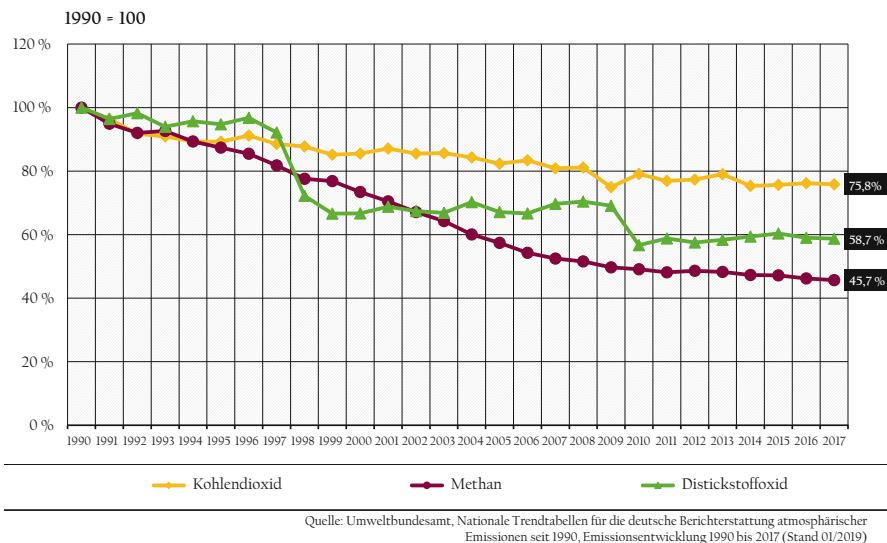
Abb. 2.9 Nettoerzeugung elektrischer Energie aus regenerativen Energiequellen. Bildquelle: Fraunhofer ISE. https://www.energy-charts.de/ren_share_de.htm?source=ren-share&period=annual&year=all

die (meisten) Endkunden umgelegt. Die EEG-Umlage macht aktuell ca. ein Viertel des Endkundenstrompreises aus (siehe [63]).

Durch den Ausbau der erneuerbaren Energien ist es auf dem deutschen Strommarkt in den letzten Jahren zu einer gewissen Überkapazität gekommen. Diese Überkapazität betrug im Jahr 2017 bis zu 10 GW (siehe [27]). Viele fossile Kraftwerke, insbesondere Braunkohlekraftwerke, können jedoch technisch bedingt nicht schnell an- und wieder abgefahren werden, um so flexibel auf den Energiebedarf im Netz zu reagieren. Als Resultat wurde in den letzten Jahren sehr viel Energie in Nachbarländer abgeführt. Im ersten Halbjahr 2018 hat Deutschland einen Exportüberschuss von 22 TWh erzielt (siehe [44]). Das ist mehr elektrische Energie, als alle Gaskraftwerke im selben Zeitraum produziert haben. Zu einer Transformation der Energiesysteme kann es allerdings nur dann kommen, wenn im Gleichschritt mit dem Ausbau der erneuerbaren Energien auch Erzeugung aus fossilen Quellen reduziert wird. Dies wird insbesondere durch den im Jahr 2019 beschlossenen Ausstieg aus der Kohleverstromung nun schrittweise umgesetzt.

Im Februar 2019 hat die Kommission für Wachstum, Strukturwandel und Beschäftigung (auch Kohlekommission genannt) ihren Abschlussbericht vorgelegt, der den langfristigen Ausstieg aus der Kohleverstromung empfiehlt (siehe [55]). Es wird vorgeschlagen, bis zum Jahr 2022 insgesamt 7 GW an Braunkohlekraftwerken vom Netz zu nehmen, bis zum Jahr 2030 weitere 13 GW abzuschalten und spätestens 2035 die Verstromung von Kohle vollständig zu beenden. Im Bezug zur gewollten Senkung von

Trend der Emissionen von Kohlendioxid, Methan und Distickstoffoxid



Quelle: Umweltbundesamt, Nationale Trendtabellen für die deutsche Berichterstattung atmosphärischer Emissionen seit 1990, Emissionsentwicklung 1990 bis 2017 (Stand 01/2019)

Abb. 2.10 Trend der Emissionen einiger Treibhausgase in Deutschland zwischen 1990 und 2017.
Bildquelle: Umweltbundesamt. <https://www.umweltbundesamt.de/bild/trend-der-emissionen-von-kohlendioxid-methan>

Treibhausgasen und der damit verbundenen Einhaltung der Emissionsziele sind diese Maßnahmen sicher richtig und empfehlenswert. Für Versorger allerdings geraten durch diese politischen Beschlüsse langfristige Investitionsentscheidungen in konventionelle Kraftwerke in Gefahr. Konventionelle Kraftwerke haben nicht selten eine technische Lebensdauer von 30–50 Jahren. Die Rentabilität der Anlagen ist in dieser Zeit nur dann gegeben, wenn entsprechende Betriebsstunden pro Jahr erreicht werden.

Der Zubau von erneuerbaren Energien und die gleichzeitige Abschaltung von konventionellen Kraftwerken hat drastische Konsequenzen für den Versorgungsmarkt. Die Erzeugung von elektrischer Energie aus erneuerbaren Quellen hat sowohl auf die technisch-physikalische als auch auf die kaufmännische Ebene Einfluss auf die Energieversorgungsstruktur. Wesentlicher Faktor ist die fehlende Regelfähigkeit erneuerbarer Energiequellen. Im Vergleich zu den konventionellen Kraftwerken sind die erneuerbaren Energien (abgesehen von Biogasanlagen) nicht regelbar. Die Erzeugung hängt von der Wetterlage (Wind, Sonne) ab und kann nur für kurze Zeiträume in die Zukunft halbwegs sicher prognostiziert werden. In Konsequenz wird die zentrale Frequenzhaltung der Netze immer schwieriger. Dies zeigt sich auch daran, dass die Anzahl der sog. Redispatch-Maßnahmen seit 2012 stark angestiegen ist (siehe [58, S. 10]). Im Ergebnis müssen konventionelle Kraftwerke aktuell mehr Flexibilität stellen.

Ein weiterer Unterschied zwischen konventionellen und regenerativen Erzeugern betrifft die **Einspeisestruktur**. Konventionelle Großkraftwerke speisen i. d. R. zentral in das Übertragungsnetz ein. Regenerative Erzeuger sind jedoch zu über 90 % an das Verteilnetz angebunden und über ganz Deutschland verteilt (siehe z. B. [25]). Die Technologien der Energiewende, also Windkraft, Solarenergie, Stromspeicher, Elektromobilität und Wärmepumpen, bringen eine wesentlich verteiltere Struktur mit sich. In März 2019 stehen in Deutschland den aktuell 763 konventionellen Kraftwerken mittlerweile mehr als 1,7 Millionen regenerative Erzeugungseinheiten gegenüber. Dies verursacht neue, bidirektionale Energieflüsse zwischen Übertragungs- und Verteilnetzen, wozu insbesondere die Verteilnetze ursprünglich nicht ausgelegt waren. Obwohl Dezentralität an sich keinen Wert darstellt, bietet sie aber Chancen (siehe [37]). So können z. B. regionale Energiemärkte entstehen, um den Netzausbau zu reduzieren.

In den letzten Jahren sind Technologien entstanden, die das Potenzial haben, die Energiesektoren zu einem holistischen Ansatz einer ganzheitlichen Energieversorgung zu koppeln. Im Zentrum steht dabei die elektrische Energie, da diese zunehmend regenerativ erzeugt werden kann. Wird diese Energie in Sektoren eingesetzt, in denen klassisch kaum elektrische Energie genutzt wurde, kann es in allen Sektoren zu einer Reduktion von Klimagasen kommen. Der sektorenübergreifende Einsatz von Energie wird als **Sektorenkopplung** bezeichnet und kann als ein Eckpfeiler der Energiewende bezeichnet werden. Um die Sektorenkopplung zu realisieren, existieren heute einige technische Möglichkeiten. Ein vielversprechender Ansatz ergibt sich aus sog. Power-to-X-Anlagen. Diese erlauben es, aus elektrischer Energie Wasserstoff, Methan oder auch künstliche Treibstoffe zu synthetisieren. Das Gas kann dann wieder zur Wärmeversorgung oder zur Verstromung genutzt werden. Im letzteren Fall dient das Gas als chemischer Energiespeicher. Ein Konsortium aus verschiedenen Netzbetreibern (Tennet, Gasunie und Thyssengas) plant in diesem Kontext derzeit eine 100 MW starke Pilotanlage, die im Jahr 2022 in Betrieb genommen werden soll (siehe [54]). Auch der ÜNB Amprion plant gemeinsam mit der Open Grid Europe GmbH eine entsprechende Anlage (siehe [61]).

Anhand der **Elektromobilität** kann auch der Verkehrssektor von der regenerativen Erzeugung elektrischer Energie profitieren. Die Elektromobilität ist aktuell eine der wenigen realistischen Alternativen zu Verbrennungsmotoren, die es erlaubt, regenerative Energie einzusetzen. Auch moderne Wärmepumpen sind in der Lage, den Wärme- bzw. Kältebedarf auf Basis elektrischer Energie zu bedienen. Dadurch kann in diesem Bereich die Verwendung von fossilen Brennstoffen ebenfalls (Erdgas, Erdöl) zurückgedrängt werden. Infolge erscheint eine dekarbonisierte Energieversorgung in allen Sektoren als möglich. Abb. 2.11 zeigt eine mögliche zukünftige Einspeisestruktur mit weniger zentralen Großkraftwerken, einer regenerativen Erzeugung, die an die Verteilnetze angegeschlossen ist, und einer Sektorenkopplung auf Basis von Power-to-Gas.

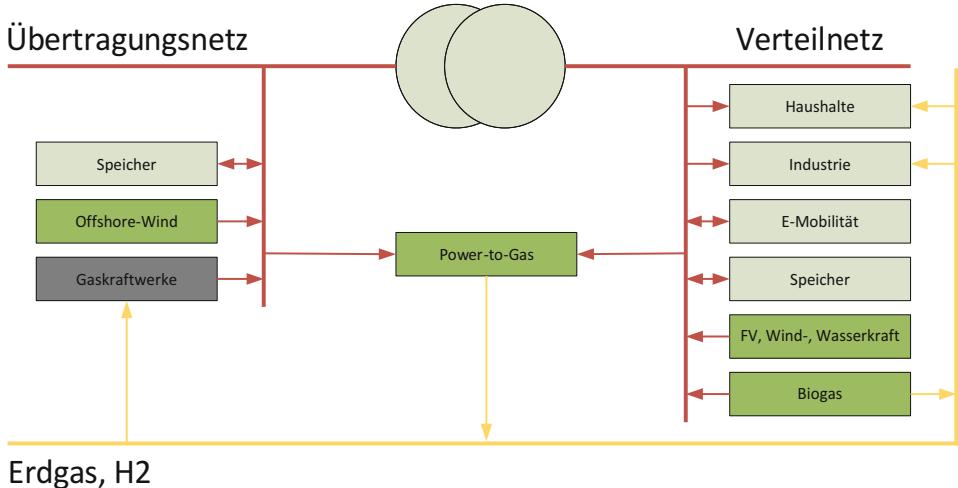


Abb. 2.11 Zukünftige Erzeugungs- und Einspeisestruktur für elektrische Energie

2.3 Digitalisierung

Die Energiewende verändert die Struktur der Energieversorgung rasant. Neben der Energiewende gibt es allerdings noch ein weiteres Tophema der Energiewirtschaft. Auch die Digitalisierung bzw. die **digitale Revolution** führt zu erheblichen Veränderungen (siehe [33]). Im 20. Jahrhundert standen vor allem Automatisierung und Optimierung im Fokus der Informationstechnologie der Energiewirtschaft. Diese Themen sind weiterhin wichtig und richtig, seit dem 21. Jahrhundert rücken aber zunehmend innovative Geschäftsmodelle, Autonomisierung, Flexibilisierung und Individualisierung in den Vordergrund (siehe [56]). Der disruptive Einsatz von digitalen Technologien wird auch als digitale Revolution bezeichnet. Big Data und künstliche Intelligenz setzen neue Maßstäbe in der Informationsverarbeitung, kleinste eingebettete Rechner dringen in alle Lebensbereiche vor, Roboter werden autonom und Produktion vernetzt und integriert. Die sogenannte Industrie 4.0 zielt auf die individualisierte Fertigung mit Losgröße 1 ab.

Die technologischen Entwicklungen der digitalen Revolution sind auch in der Energiebranche ein wichtiger Treiber. Die Existenz IT-basierter Technologien erzeugt Druck, sie auch in der Energiebranche einzusetzen. Man erhofft sich dadurch Kostensparnisse durch höhere Effizienz bzw. neue Geschäftsmodelle. Ohne die digitalen Technologien wären viele Ansätze der Energiewende nicht zu realisieren. Synonym dafür steht die Bestrebung, das sog. Smart Grid zu realisieren. Bei diesem intelligenten Netz werden Erzeuger, Speicher und Verbraucher und Netzbetriebsmittel in Übertragungs- und Ver-

teilnetzen zu jedem Zeitpunkt durch verteilte Intelligenz so aufeinander abgestimmt, dass Leistungs- oder Lastspitzen automatisch durch den Einsatz von Informations- und Kommunikationstechnik (IKT) ausgeglichen werden. Alle Komponenten müssen dazu miteinander vernetzt werden, wodurch sie in der Lage sind, in Echtzeit Zustandsdaten miteinander auszutauschen. In einem Smart Grid werden also nicht nur Energie, sondern auch Daten transportiert. Entsprechende Datenflüsse werden in Deutschland durch das Ausrollen der sog. Smart Meter aktuell umgesetzt. Diese digitalen Zähler erheben und übertragen Last- und Netzzustandsdaten an den Netzbetreiber und schaffen mit ihrer Datenbasis eine Grundvoraussetzung für das Smart Grid (siehe Abschn. 5.3). Darüber hinaus entstehen aktuell viele weitere IKT-basierte Konzepte in der Energiewirtschaft oder werden erprobt, z. B. virtuelle Kraftwerke oder der Energiehandel auf Basis der Blockchain. Mithilfe digitaler Technologien, wie Vernetzung, intelligente Steuerungen oder Virtualisierung, sind heute Ansätze umsetzbar, die noch vor wenigen Jahren völlig undenkbar waren.

Ein Beispiel für die Weiterentwicklung zukünftiger Automatisierungstechnik ist das **Internet der Dinge** (engl. Internet of Things (IOT)). Das IOT setzt voraus, dass einzelne Maschinen, Geräte oder Anlagen eine virtuelle Identität besitzen. Eine solche virtuelle Identität spiegelt Meta- und Zustandsdaten von Komponenten, die möglichst in Echtzeit erhoben, übertragen und verarbeitet werden können. Es entstehen sog. cyberphysische Systeme, bei denen die Daten dann mithilfe intelligenter Algorithmen und künstlicher Intelligenz ausgewertet werden können. Auf solchen Plattformen setzen Ansätze, wie Industrie 4.0, Smart Grid oder Smarthomes, auf.

Die Vernetzung intelligenter Geräte setzt allerdings eine Kommunikationsinfrastruktur voraus. Dabei spielen offene Standards eine bedeutende Rolle, da ansonsten Inkompatibilitäten zwischen Systemen drohen oder man dauerhaft an einzelne Anbieter technischer Lösungen gebunden ist. Mit dem Protokollstack des Internets existiert mittlerweile ein solcher Standard. Im Büroumfeld basiert ein großer Teil der digitalen Nachrichtenübertragung heute auf der Internetprotokollfamilie. Anwendungsdienste nutzen diese Infrastruktur und werden heute oft webbasiert auf Basis von Http bereitgestellt (sog. Mikroservices). Im IoT-Bereich und in der Prozessautomation existieren viele weitere Kommunikationsstandards. Es kommen Nachrichtenprotokolle und Übertragungsverfahren wie Modbus, MQTT oder ZigBee zur Anwendung. Auch in der Energiewirtschaft und der Gebäudetechnik haben sich spezifische Standards, wie z. B. IEC 61850, KNX oder LONWorks, etabliert.

Die Digitalisierung wird auch durch die Möglichkeiten des **Cloud-Computings** getrieben. Kernkonzepte des Cloud-Computings sind die Virtualisierung von IT-Komponenten, wie z. B. Rechner, Netzwerke oder Speicher, und die standardisierte Vernetzung über das Internet. IT-Infrastruktur kann so flexibel und bei Bedarf gestartet und wieder gestoppt werden, ohne sie selbst vorhalten zu müssen. Dies erhöht die Auslastung von Rechenzentren und spart dadurch unter anderem Energie ein. IT-Infrastruktur wird dadurch zum *Commodity*, welches in nahezu beliebigem Umfang abgerufen werden kann. Anbieter wie

Amazon, Microsoft oder Google bieten heute umfangreiche Plattformen an, auf deren Basis auch Lösungen für die Energiewirtschaft umgesetzt werden können.

Durch die weite Verbreitung von digitalen Systemen fallen mittlerweile riesige Datenmengen an. Diese Daten entstehen u. a. durch den Geschäftsbetrieb ganz automatisch, sie müssen nicht erst erzeugt werden. Beispiele sind Markt- und Wetterdaten, Surfverhalten, Konsumgewohnheiten, Lieblingsfilme, Fotos, Bewegungsdaten von Smart-TV, Informationen aus sozialen Netzen oder Gesundheitsinformationen. Im Ergebnis verdoppelt sich die weltweite Datenmenge im Schnitt alle zwei Jahre. Große, oft unstrukturierte Datenmengen werden heute oft als **Big Data** bezeichnet. Dank der Verfügbarkeit von Speicherplatz und Rechenkraft können diese Datenmengen heute aber verarbeitet werden. Dazu kommen u. a. statistische Verfahren bzw. Algorithmen aus dem maschinellen Lernen zum Zug, z. B. Regression, Klassifizierung oder Assoziationsanalyse. Das Ziel sind meist bessere Prognosen oder Risikoeinschätzungen. Auch in der Energiewirtschaft sind die Verfahren rund um Big Data wertvoll (siehe z. B. [34]).

Die **Blockchain** ist seit einiger Zeit in aller Munde. Sie bildet die zentrale Idee hinter den virtuellen Währungen wie Bitcoin oder Ethereum. Eine Blockchain ist dabei zunächst lediglich ein Konzept zur Speicherung von Daten ähnlich einer Datenbank. Eine Blockchain basiert dabei aber auf einigen neuen Ideen. Datenblöcke werden mithilfe kryptografischer Methoden so miteinander verkettet, dass eine nachträgliche Manipulation der Daten ausgeschlossen ist. Eine Blockchain wird dezentral, d. h. verteilt auf viele Rechner gespeichert. Da viele Kopien einer Blockchain existieren, herrscht Transparenz über den Inhalt. Es existiert ein Konsensmechanismus, der die Beteiligten dazu zwingt, neue Inhalte untereinander abzustimmen, bevor dieser in die Blockchain aufgenommen werden kann. Das Konzept der Blockchain kann in vielen Szenarien sinnvoll eingesetzt werden. Insbesondere kann eine vertrauenswürdige dritte Instanz ersetzt werden, wenn zwei Parteien eine Transaktion abschließen wollen. Auch große Unternehmen, wie die deutsche Börse AG, sehen offenbar großes Potenzial (siehe [59]). In der Energiewirtschaft kommen solche Transaktionen an vielen Stellen vor, z. B. immer dann, wenn Verträge geschlossen oder Informationen transparent und unmanipulierbar zu veröffentlichen sind. Beispiele dafür sind der Wechsel eines Kunden von einem zu einem anderen Versorger, der (börsliche) Handel mit Energieprodukten oder der Herkunftsnnachweis regenerativer Energie. Es existieren mittlerweile einige interessante Projekte, welche den Einsatz einer Blockchain im energiewirtschaftlichen Kontext zeigen. So kann eine Blockchain z. B. dazu genutzt werden, um Energieflüsse abzurechnen oder lokale Energiemarkte zu realisieren (siehe [51]) oder die Marktkommunikation zwischen Marktteilnehmern zu erleichtern (siehe [60]). Es gibt sicher noch viele weitere sinnvolle Anwendungsfälle in der Energiewirtschaft.

Nicht zuletzt aufgrund steigender Energiepreise stehen Energieversorger heute zunehmend unter Konkurrenzdruck. Mithilfe von Vergleichsplattformen wie Check24 oder Verivox ist ein Anbieterwechsel nur jeweils einen Mausklick entfernt. Für Versorger entsteht daher zunehmend die Notwendigkeit, den Service und Dienstleistungsbereich weiter auszubauen. Dabei können bestehende Geschäftsfelder um Energiedienstleistungen

erweitert werden, z. B. Fotovoltaik-Contracting oder Energieberatung. Aber auch energieferne Dienstleistungen rücken in den Fokus, z. B. Telekommunikation, Mobilität oder Smarthome. Häufig spielen digitale Prozesse dabei eine herausragende Rolle. Für viele Versorger bedeutet dies einen Wandel in der Unternehmenskultur. Der Kunde und seine ganzheitlichen Bedürfnisse abseits der reinen Energieversorgung rücken stärker in den Mittelpunkt. Dazu gehört auch die Schaffung eines digitalen Kommunikationskanals, z. B. mithilfe nützlicher Webangebote oder sozialer Netze.

2.4 Informationstechnik

In den letzten beiden Abschnitten wurden einige Grundbegriffe der Energiewirtschaft dargestellt. Die Wertschöpfungskette der Energiewirtschaft wurde sowohl anhand des physikalisch-technischen als auch des kaufmännischen Energieflusses erläutert. Darüber hinaus wurden zukünftige Herausforderungen für die Energiewirtschaft dargestellt. Die Energiewende und die Digitalisierung werden großen Einfluss auf die Struktur der Energiesysteme, die Unternehmen, die Methoden und Verfahren haben.

Bevor nun Verfahren, Methoden und Standards der Energieinformatik dargestellt werden, müssen zunächst einige Grundlagen der Informations- und Kommunikationstechnik eingeführt werden. Im folgenden Abschnitt werden daher zunächst einige Begriffe wie Daten, Information, Informations- und Datenmodell erläutert und die Informationsverarbeitung mithilfe von Datenbanken dargestellt.

Ohne IKT kann eine sichere, preisgünstige, verbraucherfreundliche, effiziente, umweltverträgliche und leitungsgebundene Energieversorgung heute nicht mehr sichergestellt werden (siehe auch EnWG). Die Entwicklung und Bereitstellung entsprechender IT-Systeme sind Hauptaufgaben der Energieinformatik. Die IKT ist dabei ein Sammelbegriff für alle Technologien, die im betrieblichen Kontext dazu dienen, Daten automatisiert zu erfassen, zu speichern, zu übertragen, zu transformieren und zu visualisieren. In der Energiewirtschaft unterstützt die IKT die Abwicklung und Planung des physikalisch-technischen und kaufmännischen Energieflusses.

Für die MitarbeiterInnen eines Energieunternehmens manifestiert sich die IKT meist in Form von spezifischen Anwendungssystemen (Software). Mithilfe solcher Anwendungssysteme können im Kontext eines betrieblichen Geschäftsprozesses administrative oder dispositivo Aufgaben bearbeitet werden, z. B. das Portfoliomanagement im Energiehandel, die Auslegung eines Übertragungsnetzes oder die Überwachung eines Kraftwerks. Der Betrieb von Anwendungssystemen setzt eine technische Infrastruktur aus Rechnern und Kommunikationsnetzen voraus. In der Energiewirtschaft werden sowohl Branchen- als auch Individuallösungen eingesetzt. Branchenlösungen werden von spezialisierten Softwareanbietern erstellt und werden in mehr als einem Energieunternehmen genutzt. Individuallösungen hingegen werden betriebsintern bzw. durch externe Anbieter spezifisch auf die Anforderungen eines einzelnen Unternehmens hin entwickelt und eingesetzt. Da die physikalischen Gesetze weltweit gleich sind, existieren besonders im Bereich

dieser Energieflüsse einige Branchenlösungen mit hoher funktionaler Reife, z. B. für die Netzplanung, Simulation oder Überwachung (siehe z. B. [64]).

Auch im Bereich des Energiehandels finden sich mit den **Energy-Trading-and-Risk-Management-(ETRM-)Systemen** diverse Standardlösungen, z. B. OpenLink Endur, Allegro oder Murex. Große Teile des kaufmännischen Energieflusses und das daraus resultierende Marktmodell basieren allerdings auf nationalen Gesetzen und Regularien. In diesem Bereich haben sich in Deutschland spätestens seit der Liberalisierung der Energiewirtschaft eine ganze Reihe von speziellen IT-Dienstleistern etabliert.

Aus den fachlichen Anforderungen der Energiewirtschaft sind einige Klassen von typischen Anwendungssystemen entstanden. In der Literatur werden viele dieser Anwendungssysteme oft auch unter dem Sammelbegriff der **Energiemanagementsysteme (EMS)** zusammengefasst. Abb. 2.12 zeigt einige dieser Klassen von Anwendungssystemen. Es wird zudem eine Zuordnung zur Abwicklung des physikalisch-technischen oder des kaufmännischen Energieflusses vorgenommen.

Daten sind heute der Rohstoff für alle Informations- und Entscheidungsprozesse in Unternehmen (siehe [10, S. 53]). Man bezeichnet Daten daher auch als *Gold des 21. Jahrhunderts*. Daten sind dabei zunächst nichts anderes als von Maschinen verarbeitbare Zeichen. Nützlich werden solche Daten erst, wenn sie zu Informationen werden. Daten werden zu Informationen, wenn sie eine Bedeutung (Semantik) erhalten. Aus der Ziffernfolge 4,7 (Daten) wird der prognostizierte Strompreis am morgigen Tag (Informationen). Informationen können wiederum miteinander verknüpft werden, um Wissen zu generieren. Die Information über den morgigen Strompreis und den Produktionskosten eines eigenen

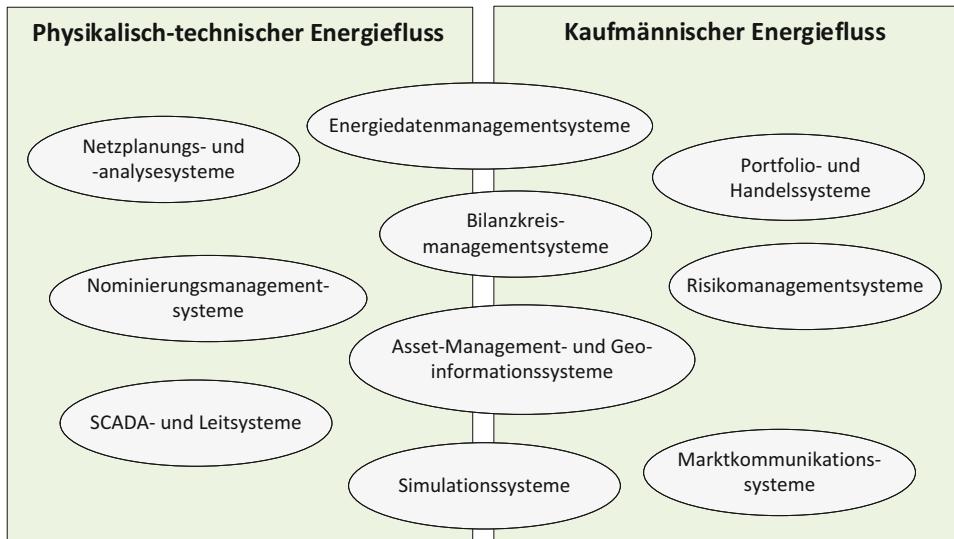


Abb. 2.12 Anwendungsklassen in der Energiewirtschaft und ihre Zuordnung zur Abwicklung des physikalisch-technischen bzw. des kaufmännischen Energieflusses

Kraftwerksblocks wird dann zum Wissen über eine Opportunität. Ohne Informationen können auch in Energieunternehmen keine gesicherten Entscheidungen getroffen werden. Ziel der Energieinformatik muss es daher sein, die betrieblichen Abläufe optimal mit Daten zu versorgen (sog. Datenlogistik). Daten müssen zudem so aufbereitet werden, dass sie zu Informationen werden und daraus Wissen generiert werden kann. Daten können nach unterschiedlichen Gesichtspunkten kategorisiert werden. Eine wichtige Unterscheidung ist die in Instanz- und Metadaten.

Instanzdaten stellen die eigentlichen Nutzdaten dar. In einer Bilddatei sind dies z. B. die Angaben über die Farbwerte (RGB) eines jeden Pixels. Im Gegensatz dazu machen **Metadaten** Aussagen über anderen Daten. Für die Bilddatei ist dies z. B. der Dateiname oder das Erstellungsdatum. Metadaten sind oft eine Voraussetzung, um Instanzdaten verarbeiten zu können. Ohne die Breite, Höhe und Auflösung zu kennen, kann der Datenstrom einer Bilddatei nicht korrekt interpretiert werden. Metadaten können unterschiedliche Aussagen über Daten machen. Beschreibende Metadaten, wie z. B. der Autor eines Buches, können für das Auffinden und Identifizieren einer Ressource genutzt werden. Strukturelle Metadaten machen Aussagen über den inneren Aufbau einer Sammlung von Daten. Administrative Metadaten, wie z. B. das Erstellungsdatum oder Zugriffsrechte, helfen bei der Verwaltung einer Ressource.

Im Kontext der Datenverarbeitung im Unternehmen werden Daten meist zwischen **Stamm- oder Sachdaten** und **Bewegungsdaten** unterschieden. Stammdaten stellen Grundinformationen über betrieblich relevante Objekte dar. Sie bilden eine wichtige Grundlage für alle Prozesse betrieblicher Informationsverarbeitung. Stammdaten werden nur selten verändert. Dies ist nur dann notwendig, wenn sich auch die Grundinformationen der beschriebenen Objekte ändern, z. B. durch Austausch oder Erneuerung. Im Kontext des Asset-Managements von Energieanlagen sind Art und Technologie der Anlage, der Hersteller, das Baujahr, der geografische Ort und diverse kaufmännische Daten sicherlich Beispiele für Stammdaten. Diese Daten werden dann in entsprechenden Stamm- oder Sachdatensystemen verwaltet.

In Abgrenzung zu den Stammdaten existieren auch sog. Bewegungsdaten. Sie entstehen meist durch die Wertschöpfung des Unternehmens, z. B. durch energiewirtschaftliche Aktivitäten, wie Vertrieb, Handel oder den Betrieb der Netze. Bewegungsdaten beziehen sich dabei grundsätzlich auf Stammdaten und können nicht ohne sie existieren. In der Energiewirtschaft äußern sich Bewegungsdaten sehr häufig in Form sog. Zeitreihen, also mehrere Datenpunkte mit einer eindeutigen Zuordnung zu einem Erfassungs- oder Gültigkeitszeitpunkt. Beispiele für solche Zeitreihen sind Marktpreise, Verbräuche oder Lastkurven. Abschn. 3.3 befasst sich intensiv mit Zeitreihen. Im Kontext des Asset-Managements sind Störhäufigkeiten und -ursachen, Zustands- und Nutzungsdaten einige Beispiele für Bewegungsdaten. Diese Daten werden meist in weiteren, spezifischen Systemen, wie Betriebstagebüchern oder Störungsinformationssystemen, erfasst und visualisiert.

Die Qualität (nicht nur) des Asset-Managements hängt entscheidend von der **Qualität** der Stamm- und Bewegungsdaten ab (siehe [42, S. 27]). Aktivitäten wie eine Netz-

planung und -simulation, Zustandsbewertung, technisches Risikomanagement und eine Instandhaltungsplanung können nur sinnvolle Ergebnisse produzieren, wenn die notwendigen Eingangsparameter in einer gewissen Mindestqualität vorliegen. Der Begriff der Datenqualität kann anhand unterschiedlicher Kriterien bewertet werden (siehe u. a. [12] und [3]).

- **Intrinsische Datenqualität**

Glaubhaftigkeit, Genauigkeit, Objektivität, Reputation

- **Kontextuelle Datenqualität**

Mehrwert, Relevanz, Zeitnähe, Vollständigkeit, Datenmenge

- **Repräsentationelle Datenqualität**

Interpretierbarkeit, Verständlichkeit, Konsistenz der Darstellung, Knappheit der Darstellung

- **Zugriffsqualität**

Verfügbarkeit, Zugriffssicherheit

Viele dieser Qualitätskriterien sind leider verhältnismäßig subjektiv. Es ist daher mitunter schwierig, eine objektive Bewertung der Datenqualität vorzunehmen (siehe [12]). Die Vollständigkeit der Daten kann meist noch recht einfach bewertet werden. Dazu muss lediglich die vorhandene Datenmenge mit der maximal möglichen Datenmenge in Bezug gesetzt werden. Die Frage kann z. B. sein, von wie vielen aller Leistungsschalter, die in Betrieb sind, auch wirklich Zustandsdaten vorhanden sind. Andere Kriterien, wie die Glaubhaftigkeit oder der Mehrwert, sind dabei etwas schwieriger zu bewerten.

Qualitativen Mängeln von Daten kann auf zweierlei Weise begegnet werden. Bei externen Daten, deren Qualität nicht aktiv beeinflusst werden kann, muss mit der minderen Qualität bewusst umgegangen werden. Entscheidungen auf Basis dieser Daten müssen bewusst die Unsicherheit mit einkalkulieren. Die Qualität eigener Daten kann hingegen aktiv beeinflusst werden. Am wirkungsvollsten und nachhaltigsten werden Datenfehler bei ihrer Entstehung bekämpft (siehe [12]). Bei der manuellen Dateneingabe oder bei der Erhebung durch Sensoren können entsprechende Schutzmechanismen eingeführt werden.

Daten können nur dann automatisiert in Anwendungssystemen verarbeitet werden, wenn ihnen ein **Informationsmodell** zugrunde liegt. Ein solches Informationsmodell legt die konzeptuelle Struktur der Daten explizit fest. Das Informationsmodell gibt den Daten eine Bedeutung, erst dadurch können die Daten sinnvoll interpretiert und verarbeitet werden. Durch die Kenntnis des Informationsmodells resultiert auch, welche Operationen sinnvollerweise auf den Daten möglich sind. Obwohl Postleitzahlen und Temperaturen beides numerische Werte sind, ergibt die Berechnung eines Durchschnittswerts auf Postleitzahlen nur wenig Sinnvolles. Ohne ein Informationsmodell kann ein Anwendungssystem mit einzelnen Ziffern oder einer Zeichenkette nichts anfangen. Ein solches Informationsmodell wird auch für die Datenübertragung benötigt. Im Rahmen von Kommunikationsprotokollen wird ein solches Modell entweder explizit oder implizit festgelegt.

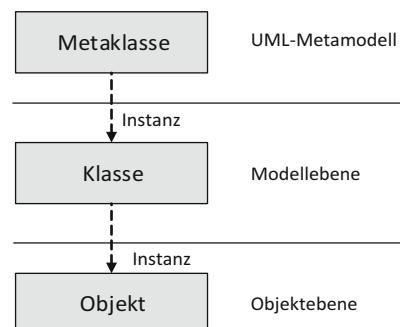
Ein Informationsmodell ist eine abstrakte Beschreibung der konzeptuellen Datenstruktur. Es beschreibt die konkreten Objekte eines abgegrenzten Geltungsbereichs (Problemdomäne) mitsamt ihren Eigenschaften und Beziehungen. Ein Informationsmodell kann mithilfe verschiedener Modellierungssprachen erfasst und grafisch visualisiert werden. Die sog. Entity-Relationship-(ER-)Modelle liegen konzeptionell z. B. sehr nah bei den relationalen Datenbanken. In den letzten Jahren hat sich die sog. Unified Modelling Language (UML) als Standardmodellierungssprache positioniert, die von der Object Management Group (OMG) definiert wird. Die **UML** basiert auf dem Paradigma der **objektorientierten Programmierung** und basiert auf einer dreischichtigen Infrastruktur (siehe z. B. [39]). Das Metamodell der UML beschreibt eine abstrakte Syntax für Elemente auf der Modellebene. Elemente der Modellebene sind Instanzen (z. B. Klassen) des Metamodells. Entsprechend sind Elemente auf der Objektebene Instanzen der Modellebene. Im Prinzip sind noch weitere Abstraktionsebenen denkbar, die mitunter notwendig sind, um Informationsmodelle in der UML abzubilden, die nicht auf Basis der UML erstellt wurden (siehe z. B. [9]). Abb. 2.13 zeigt die drei Modellebenen der UML.

In der Regel werden Informationsmodelle der UML auf der Modellebene erstellt. Ein Klassendiagramm auf dieser Ebene besteht dann aus **Klassen**, die über **Attribute** beschrieben werden. Klassen können unterschiedliche Arten von Beziehungen untereinander eingehen, die mithilfe von **Assoziationen**, **Aggregation**, **Komposition** und **Vererbung** darstellbar sind.

Klassen ...

- sind die wichtigsten Bausteine eines objektorientierten Informationsmodells.
- sind eine Verallgemeinerung von konkreten Objekten, quasi ein Konzept.
- sind Vorlagen oder Baupläne, welche die Struktur für konkrete Objekte vorgeben.
- werden für alle dinglichen oder abstrakten Elemente mit gleichen Eigenschaften gebildet.
- besitzen einen eindeutigen Namen, z. B. Generator, und werden im Klassendiagramm als Rechteck dargestellt.

Abb. 2.13 Die Modellebenen der UML



Als strukturelle Merkmale besitzen Klassen sog. Attribute. Jedes Attribut hat einen Namen und basiert auf einem geeigneten Datentyp. Ein solcher Datentyp kann ein primitiver Datentyp, wie Integer (Ganzzahl), oder ein komplexer Datentyp sein (z. B. KapazitätProLänge). Ein komplexer Datentyp wird wiederum durch eine Klasse definiert.

Beispiel 2.4.1 Es existieren viele konkrete Fotovoltaikanlage (Objekte). Die Gemeinsamkeiten dieser Fotovoltaikanlage können in der Klasse Fotovoltaikanlage zusammengefasst werden. Abb. 2.14 zeigt die Klasse Fotovoltaikanlage in ihrer üblichen Notation eines UML-Klassendiagramms.

Die Klasse besitzt u. a. die Attribute Fläche und Ort. Die Objekte der Klasse Fotovoltaikanlage tragen individuelle Werte in diesen Attributen. Die Fläche basiert auf einem primitiven Datentyp (z. B. Float). Der Ort hingegen basiert auf einem komplexen Datentyp (Klasse GeoLocation). Dieser wird wiederum durch eine geografische Höhe und Breite beschrieben.

Beziehungen zwischen Klassen nennt man Assoziationen. Eine Assoziation drückt die Fähigkeit von Objekten dieser Klassen aus, im Bezug zu anderen Objekten stehen zu können. Wo Klassen die Baupläne für Objekte darstellen, sind Assoziationen die Baupläne für Objektbeziehungen.

Beispiel 2.4.2 Ein konkreter Transformator (Objekt) kennt (Beziehung) genau eine Eingangs- und eine Ausgangsspannungsebene (Objekte).

Assoziationen werden im Klassendiagramm als Linie zwischen Klassen dargestellt. An beiden Enden einer Assoziation wird zudem über die Multiplizität angegeben, wie viele Objekte an der Beziehung teilnehmen sollen. Abb. 2.15 zeigt einige Beispiele für Assoziationen und Multiplizitäten im UML-Klassendiagramm.

Die objektorientierte Programmierung hat zudem den Begriff der Vererbung eingeführt. Die Vererbung ist eine Beziehung zwischen Klassen, die angelehnt an die Biologie in gewisser Weise zueinander ähnlich sind. Vererbung bedeutet, dass eine Basisklasse (auch Super-, Ober- oder Elternklasse) alle ihre Eigenschaften an eine abgeleitete Klasse (auch Kind- oder Subklasse) weitergibt. Die abgeleitete Klasse besitzt dadurch automatisch alle Attribute der Basisklasse, kann aber zusätzliche Attribute einführen. Mithilfe der Vererbung können komplizierte Hierarchien gebildet werden. Eine Vererbungshierarchie ähnelt einem Stammbaum. Je tiefer man in der Hierarchie absteigt, desto mehr Eigenschaften



Abb. 2.14 Als Teil eines Informationsmodells beschreibt die Klasse Fotovoltaikanlage das verallgemeinerte Konzept aller Fotovoltaikanlage

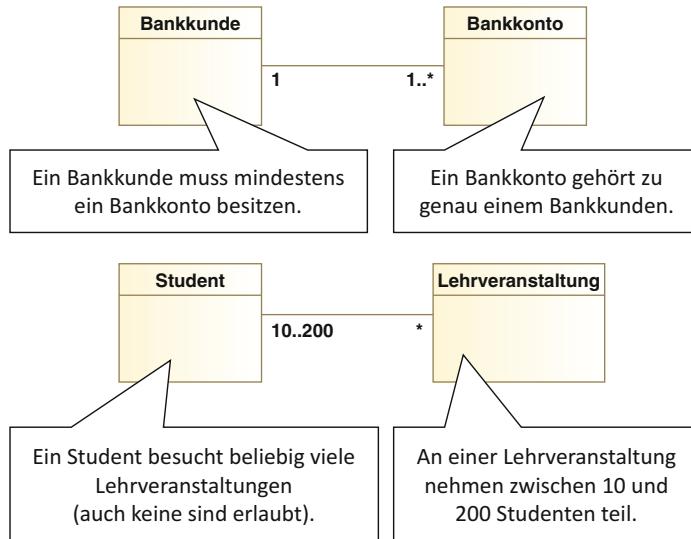


Abb. 2.15 Einige Beispiele für Assoziationen im UML-Klassendiagramm

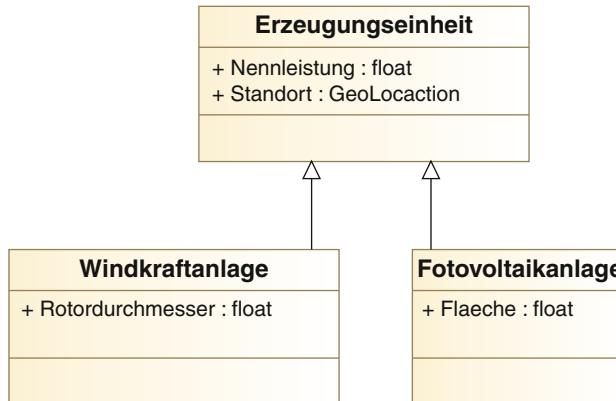


Abb. 2.16 Vererbungshierarchie unter den Klassen Erzeugungseinheit und Windkraftanlage als UML-Klassendiagramm

besitzen die Klassen, sie werden spezieller. Vererbung wird daher auch als Spezialisierung bezeichnet. Die Umkehrung hiervon ist die Generalisierung.

Beispiel 2.4.3 Eine Windkraftanlage ist eine Erzeugungseinheit. Die Klasse Windkraftanlage leitet von der Klasse Erzeugungseinheit ab, sie erbt alle Eigenschaften ihrer Basisklasse. Die Klasse Windkraftanlage fügt aber zusätzliche Eigenschaften hinzu, z. B. Bauhöhe und Flügellänge. Abb. 2.16 zeigt die Vererbungshierarchie zwischen den Klassen „Erzeugungseinheit“ und „Windkraftanlage“ als UML-Klassendiagramm.

Im Informationsmodell werden relevante Teile der Realität mithilfe von Klassen, Assoziationen und Vererbungsbeziehungen abgebildet. Das dabei entstehende Modell sollte einige Qualitätskriterien einhalten:

- **Minimalität**

Das Management von Daten ist aufwendig, es sollte daher darauf verzichtet werden, unnötige Daten vorzuhalten. Das Modell sollte nur genau Aspekte abbilden, die auch für den jeweiligen Anwendungskontext relevant sind.

- **Redundanzfrei**

Das Modell sollte so aufgebaut werden, dass Aussagen über einen Sachverhalt nur an jeweils einer Stelle getroffen werden. Änderungen können sonst zu Widersprüchen und somit zu nutzlosen Daten führen.

- **Allgemeinheit**

Im Betrieb sollten sich möglichst wenig Änderungen an einem Informationsmodell ergeben, da solche Änderungen großen Anpassungsaufwand an den Anwendungssystemen nach sich ziehen. Das Modell sollte also möglichst allgemein gehalten werden.

Um Informationsmodelle der UML zu erstellen und zu bearbeiten, existiert eine Vielzahl von elektronischen Werkzeugen. Diese unterstützen die EnergieinformatikerInnen bei der Modellierung, Visualisierung und Anpassung von Informationsmodellen. Bekannte Systeme sind z. B. das kommerzielle Enterprise Architect von Sparx Systems, welches auch bei der Pflege des Common Information Model verwendet wird (siehe Abschn. 3.1). Eine quelloffene und plattformunabhängige Alternative ist das System Modelio (siehe [62]), mit dem auch die UML-Modelle für dieses Buch erstellt wurden. Abb. 2.17 zeigt ein Bildschirmfoto von Modelio mit einem Informationsmodell in Form eines UML-Klassendiagramms.

In diesem Lehrbuch werden viele verschiedene Informationsmodelle vorgestellt, die im Kontext der Energieinformatik eine Rolle spielen und jeweils unterschiedliche Aspekte der physikalisch-technischen oder des kaufmännischen Energieflusses abbilden sollen. Manche dieser Modelle wurden direkt auf Basis der UML definiert und nutzen dann durchgängig deren Mechanismen, z. B. das sog. Common Information Model (siehe Abschn. 3.1). In vielen Fällen werden Informationsmodelle in den Spezifikationsdokumenten allerdings lediglich verbal oder mithilfe eigener Notationen beschrieben. Im Falle von älteren Kommunikationsprotokollen ergibt sich das Informationsmodell oft auch nur implizit aus den Nachrichtenformaten. In diesen Fällen ist die Darstellung der Informationsmodelle auf Basis der UML mitunter schwierig und nicht immer präzise und eindeutig umsetzbar. In diesem Lehrbuch soll dennoch der Versuch unternommen werden, möglichst viele Informationsmodelle auf Basis der UML zu präsentieren, auch dann, wenn eine solche Darstellung nicht Teil der eigentlichen Spezifikation ist. Ein Beispiel ist das Informationsmodell von Modbus (siehe Abb. 2.27). In manchen dieser Informationsmodelle können die Abstraktionsebenen nicht klar in Meta-, Modell- und Instanzebene unterschieden werden. Um diese Ebenen dann gemeinsam in einem Modell

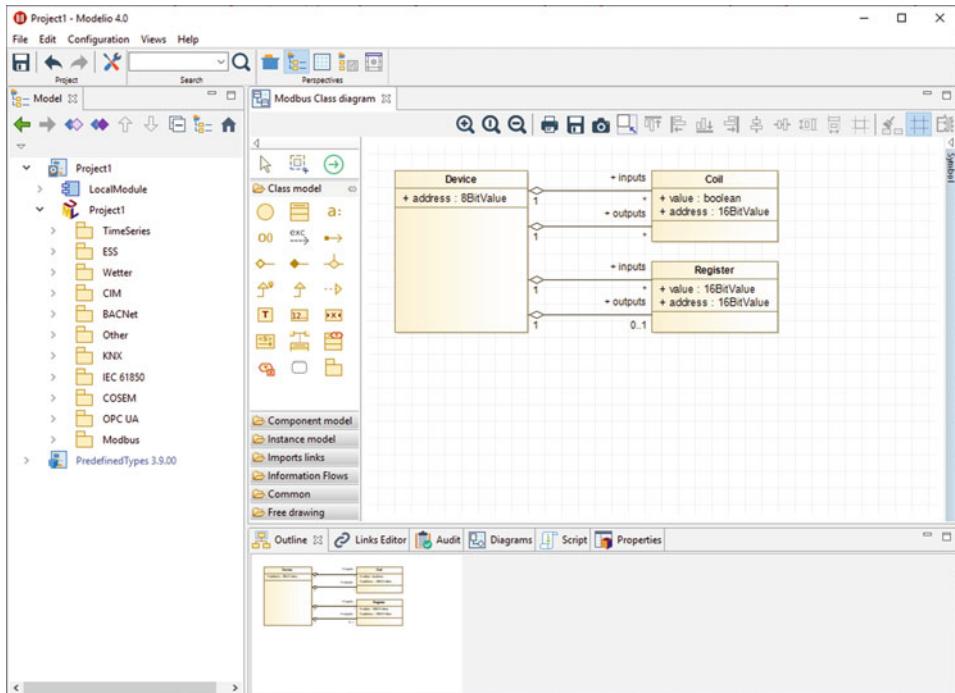
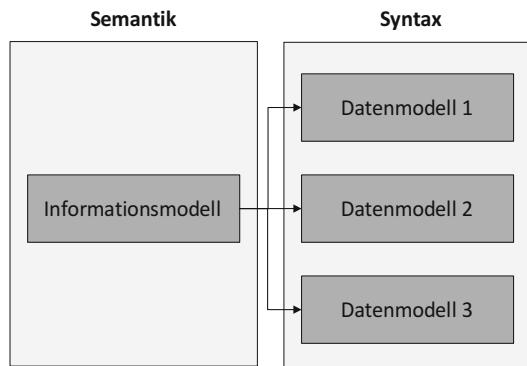


Abb. 2.17 Ein Bildschirmfoto von Modelio mit einem Informationsmodell auf Basis eines Klassendiagramms der UML

darstellen zu können, wird eine zusätzliche Beziehungsart genutzt (*Instance of*), welche die Instanziierung eines Elements darstellt (siehe z. B. das KNX-Interworking-Modell in Abb. 5.13).

Ein Informationsmodell legt die abstrakte Struktur von Daten fest. Es beschreibt die Bedeutung, die Semantik der Daten. Damit Daten aber zwischen mehreren Rechnern ausgetauscht werden können, müssen sie in einem elektronischen Dokument bzw. in einer Nachricht abgebildet werden können. Dafür wird eine Abbildung der Objekte, ihrer Attribute und Beziehungen auf Symbole und Datentypen benötigt. Diese Abbildungsregeln werden auch als Syntax bezeichnet und werden in einem **Datenmodell** festgelegt. Ein Datenmodell regelt die Darstellung der Daten eines bestimmten Informationsmodells. Das Abbilden von Daten auf ein Datenmodell wird auch als **Serialisieren** bezeichnet, das Lesen des Datenmodells als **Deserialisieren**. Im Prinzip existieren beliebig viele Möglichkeiten, Daten eines Informationsmodells auf ein Datenmodell abzubilden. Dies erklärt auch, warum für dieselben Informationen oft ganz unterschiedliche Datenmodelle in Form von Dateiformaten existieren, bei Bildern z. B. PNG, BMP oder GIF. Abb. 2.18 zeigt den Zusammenhang zwischen einem Informationsmodell und potenziell vielen Datenmodellen.

Abb. 2.18 Abbildung eines Informationsmodells auf potenziell mehrere Datenmodelle



In der Praxis haben sich einige Standards etabliert, um beliebige Informationsmodelle auf generische Datenmodelle abzubilden. Ein Beispiel ist die **Extensible Markup Language (XML)**, die auch an einigen Stellen der Energiewirtschaft genutzt wird. XML ist eine maschinenlesbare **Auszeichnungssprache (Markup Language)**, mit deren Hilfe beliebige Informationsmodelle auf ein Datenmodell abgebildet werden können.

Ein XML-Dokument ist rein textbasiert und besteht aus einer Vielzahl von **Tags**, was so viel wie Auszeichnung bzw. Etikett bedeutet. Tags treten meist paarweise als Start- und Endtag auf. `<Tag></Tag>` Ein Start-Tag wird von den Größer/Kleiner-Zeichen umschlossen: `<Tag>` Ein End-Tag wird durch den zusätzlichen Querstrich / angezeigt: `</Tag>` Jeder Start-Tag muss durch einen End-Tag geschlossen werden. Tag-Paare umschließen einfachen Text oder weitere andere Tags.

1 `<Adresse><Name>Peter</Name></Adresse>`

Es existiert zudem ein sog. Empty-Tag, der auch ohne ein End-Tag gültig ist.

1 `<EinTagDerAlleinVorkommenKann />`

Ein XML-Dokument muss wohlgeformt sein, d. h., es besitzt genau ein Wurzel-Tag und alle Start-Tags müssen durch ein End-Tag geschlossen werden. Ein Attribut ist ein Merkmal bzw. eine Eigenschaft, welche Tags näher beschreiben. Attribute können nur bei Start- oder Empty-Tags hinzugefügt werden. Ein Attribut weist einem Schlüssel einen Wert zu. Der Schlüssel ist ein eindeutiger Bezeichner. Der Wert des Attributs ist in Anführungszeichen zu setzen, z. B.:

1 `<Wire Length="10" />`

Einem Tag können beliebig viele Attribute hinzugefügt werden. Dabei existieren zunächst keine Vorgaben, welche Attribute eingeführt werden sollen. Bei der Abbildung

der Daten eines Informationsmodells werden Objekte i. d. R. auf Tags abgebildet. Bei der Abbildung von Attributen existieren verschiedene Möglichkeiten, aus denen der Entwickler im Prinzip frei wählen kann. Die beiden folgenden Abbildungen sind beide gültiges XML. Sie bilden ein Objekt der Klasse `Wire` mit der Länge 10 auf unterschiedliche Weise, aber mit demselben Informationsgehalt in XML ab.

```

1 <Wire>
2   <Length>10</Length>
3 </Wire>
4
5 <Wire Lenth="10" />
```

Mithilfe einer sog. XML Schema Definition (XSD) kann die erlaubte Struktur festgelegt und verifiziert werden. Da XML-Dokumente streng hierarchisch aufgebaut sind, können *Ist-Teil-von-Beziehungen* leicht abgebildet werden. Mithilfe von Schlüsseln können aber auch andere, allgemeinere Beziehungen ausgedrückt werden. Dieser Mechanismus ähnelt der Vorgehensweise in relationalen Datenbanken. Ein eindeutiger Schlüssel dient als Platzhalter, der an anderer Stelle als Verweis auf ein bestimmtes Objekt verwendet werden kann. Solche Schlüsselbeziehungen können ebenfalls in XSD festgelegt werden. Das folgende Beispiel bildet einige Daten einer Bibliotheksverwaltung auf ein XML-Dokument ab. Dem Buch wird der Autor über einen Schlüsselverweis zugeordnet.

```

1 <library>
2   <author id="1" firstname="Peter" lastname="Meier" />
3   <book title="SQL for dummies" author="1" />
4 </library>
```

Ein Informationsmodell gibt die Struktur von Daten vor, ein Datenmodell bildet die Daten eines Informationsmodells auf einen Datenstrom ab (siehe auch [7]). Mit einem Datenmodell können Daten zwar zwischen Rechnern ausgetauscht werden, ein Datenmodell bietet aber keine weiteren Operationen auf den Daten an. **Datenbanksysteme** hingegen sind Softwareprodukte, die Daten auf Basis eines Informationsmodells verwalten können, zudem aber vielfältige weitere Funktionen anbieten, um die Daten zu selektieren, zu aggregieren und zu transformieren. Es existieren vielfältige Arten von Datenbanksystemen, am weitesten verbreitet sind aber die sog. relationalen Datenbanksysteme, welche auf der relationalen Algebra basieren. Beispielprodukte sind MySql, PostgreSQL, Oracle DB oder der SQL-Server.

Eine relationale Datenbank verwaltet die Daten in Tabellen, in denen Datensätze abgelegt sind. Jeder Datensatz besteht aus Attributwerten. Die Anzahl der Attribute und ihr Datentyp werden durch das Tabellenschema festgelegt. Beziehungen werden auf Basis von Schlüsseln dargestellt. Ein Schlüssel dient dabei als eindeutiger Platzhalter eines Datensatzes, sodass bei einem anderen Datensatz auf diesen verwiesen werden kann.

Ein UML-Klassenmodell kann mit überschaubarem Aufwand in einer relationalen Datenbank umgesetzt werden. Klassen werden dabei auf Tabellen und die Attribute auf Tabellenspalten abgebildet. Assoziationen und Vererbung werden anhand von Schlüsselbeziehungen dargestellt.

Relationale Datenbanken haben den Vorteil, dass mit **Structured Query Language (SQL)** eine sehr mächtige Abfragesprache existiert. Zwischen den Datenbanksystemen unterschiedlicher Hersteller existieren nur im Detail Unterschiede. SQL ist eine Art Programmiersprache. Mithilfe von SQL können Daten in einer relationalen Datenbank eingefügt, verändert und abgefragt werden. Zudem dient SQL dazu, das Tabellenschema der Datenbank festzulegen. Eine der wichtigsten SQL-Anweisungen ist sicher `select`. Sie dient der Abfrage von Daten aus einer oder mehrerer Tabellen. Das Ergebnis einer solchen Anfrage ist immer eine Tabelle (Spalten und Zeilen). Eine typische Select-Anweisung hat die folgende Form:

```
1 select column1, column2, ... FROM table_name;
```

Dabei stehen `column1`, `column2` usw. für die Namen der Spalten einer Tabelle und `table_name` für den Namen eben dieser Tabelle in einer Datenbank. Die folgende Anweisung gibt für alle Zeilen und Spalten der Tabelle Daten aus:

```
1 select * from Daten;
```

Eine `Select`-Anweisung ermöglicht eine Vielzahl von Transformationen:

- Umbenennen oder Umstellen von Spalten,
- zeilenweise Berechnung von neuen Werten,
- Aggregation von Spalten, z. B. Summe oder Durchschnitt,
- Verknüpfen mehrerer Tabellen über Schlüsselbeziehungen.

Ein Datenbanksystem hilft dabei, Daten persistent zu speichern. Auch können die Daten mithilfe von SQL transformiert und aggregiert werden. Komplexe Algorithmen können mithilfe einer relationalen Datenbank aber eher nicht realisiert werden. Dazu wird eine (höhere) **Programmiersprache** benötigt, die Konzepte anbietet, um Algorithmen zu realisieren, um Daten automatisiert verarbeiten zu können. Dazu gehören meist die folgenden Ausdrucksmöglichkeiten:

- elementare Datentypen wie `int`, `float` oder `bool`,
- Ausdrücke, um z. B. arithmetische Berechnungen abilden zu können,
- Anweisungen wie `print` oder `return` und
- Kontrollstrukturen wie Verzweigungen und Schleifen.

Solche Konzepte werden im Prinzip von allen bekannten Programmiersprachen angeboten, z. B. Java, C#, C++ und viele andere. Im Bereich der Data Science bzw. des Maschine Learning wird sehr häufig die Programmiersprache Python eingesetzt. Python hat mehrere (günstige) Eigenschaften:

- interpretiert,
- dynamisches Typsystem,
- objektorientiert,
- funktionale und aspektorientierte Möglichkeiten,
- fördert einen gut lesbaren und knappen Programmierstil.

Wir werden Python in diesem Buch immer dann nutzen, wenn Algorithmen realisiert werden müssen, z. B. für die Auswertung von Zeitreihen, der Prognose, Optimierung oder Simulation. Eine Stärke von Python ist die Verfügbarkeit einer großen Menge von externen Paketen. Besonders hervorzuheben sind dabei die Pakete Pandas (Python Data Analysis Library) und Matplotlib. Pandas organisiert tabellarische Daten als Objekte vom Typ DataFrame. Für die Datenanalyse bietet dieser Datentyp eine Reihe hilfreicher Funktionen an, z. B.:

- Lesen und Schreiben aus diversen Datenquellen (Datenbanken oder CSV),
- Indexierung, Aggregation der Daten,
- Behandlung fehlender Werte u. v. m.

Das folgende Python-Script liest die Datei daten.csv in ein DataFrame ein:

```
1 import pandas as pd  
2 df = pd.read_csv('weather_data.csv', sep=',')
```

Daten eines solchen DataFrames können mithilfe von Matplotlib auch sehr einfach in Form eines Charts dargestellt werden. Der folgende Programmcode erstellt einen Chart aus einem DataFrame:

```
1 import matplotlib.pyplot as plt  
2 df.plot()  
3 plt.show()
```

2.5 Kommunikationstechnik

Im letzten Abschnitt wurden einige Grundlagen der Informationstechnik dargestellt, die für das weitere Verständnis des Buches wichtig sind. Daten werden aber nicht nur lokal

verarbeitet, sondern müssen auch zwischen Rechnern übertragen werden. Dafür existieren auch in der Energiewirtschaft unendlich viele Anwendungsfälle, z. B. unternehmensübergreifende Geschäftsprozesse, Übermitteln von Messwerten und Alarmmeldungen, Steuerung von verteilten Anlagen usw. Immer dann, wenn Rechner über Netze miteinander Informationen austauschen sollen, werden Technologien der Kommunikationstechnik benötigt. Im folgenden Abschnitt werden dazu zunächst einige Grundbegriffe eingeführt, ohne die z. B. die Automation von Energieanlagen und Gebäuden, der Datenübertragung von Smart-Metern und der Abwicklung von Marktprozessen kaum durchdringen werden kann.

Ein **Kommunikationsnetz** kann unterschiedliche Arten von Rechnern miteinander verbinden, sodass diese digitale Daten austauschen können. Einfache Sensoren können dann z. B. Daten an Workstations unterschiedlicher Hersteller übermitteln. Ziel ist immer eine zuverlässige, sichere und effiziente Datenübertragung zwischen heterogenen Kommunikationspartnern. Die beteiligten Kommunikationspartner müssen sich dabei auf eine Vielzahl von Dingen einigen, z. B. wie ein einzelnes Bit auf dem **Übertragungsmedium** codiert wird, wie Nachrichten in einem Netz aus mehreren Teilnehmern weiter vermittelt werden und wie eine Anfrage an einen Dienst gestellt werden kann.

Damit eine solche Kommunikation erfolgreich abgewickelt werden kann, werden **Kommunikationsstandards** benötigt. Solche Standards definieren z. B. die Steckverbindungen von Kabeln, die elektrischen Signalpegel oder die Darstellung von Nachrichten. Als Modell für solche Standards wurde das **ISO/OSI-Referenzmodell** der digitalen Kommunikation definiert (siehe [2]). Es gibt einen Rahmen für die Entwicklung von Kommunikationsstandards vor, indem die Aufgaben der digitalen Kommunikation auf insgesamt sieben Schichten mit eng begrenzten Aufgaben aufgeteilt werden (siehe [13]).

Das ISO/OSI-Modell selbst stellt dabei keine konkreten Realisierungen dar. Es definiert aber ein Rahmenwerk für die Umsetzung von Kommunikationsstandards und legt dabei fest, welche Aufgaben generell übernommen werden müssen bzw. könnten, um eine erfolgreiche Datenübertragung zu gewährleisten. Auf jeder Schicht können dabei voneinander unabhängige, austauschbare Technologien und Protokolle eingesetzt werden. In manchen Szenarien werden auch nicht alle Schichten implementiert.

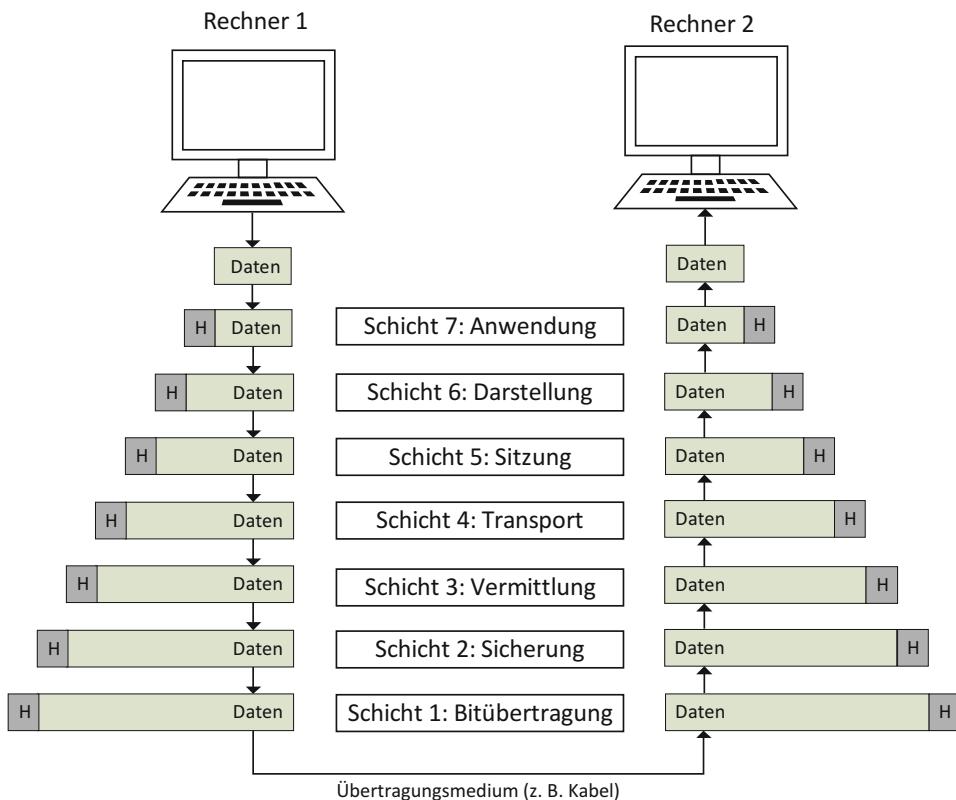
Wird auf beiden Seiten der Kommunikation, also bei Sender und Empfänger auf den einzelnen Schichten derselbe Satz von Standards verwendet, können Daten erfolgreich übertragen werden. Die Aufgaben innerhalb der Schichten sind wie folgt definiert:

Während der Datenübertragung werden die Protokolle der sieben OSI-Schichten auf beiden Seiten vertikal durchlaufen. Auf der Seite des Senders von Schicht 7 bis hinunter auf Schicht 1, auf der Seite des Empfängers dann von Schicht 1 bis Schicht 7. Abb. 2.19 stellt den Kommunikationsablauf zwischen Sender und Empfänger nach dem ISO/OSI-Modell dar.

Die reinen Nutzdaten einer Schicht werden auch als *Data Units* bezeichnet. In dieser Struktur sind die zu übertragenden Informationen in einer wohl definierten Bitsequenz abgebildet. Diese Daten werden dann beim Versand der nächstniedrigeren Schicht übergeben. Dort werden sie in einer neuen *Data Unit* verpackt und um weitere Informationen

Tab. 2.1 Die sieben Schichten des ISO/OSI-Referenzmodells für Datenkommunikation

Schicht	Name der Schicht	Aufgabe der Schicht
1	Bitübertragungsschicht	Definition des Übertragungsmediums, von Signalpegeln und Steckverbindungen. Bereitstellen eines ungesicherten Bitstroms
2	Sicherungsschicht	Steuerung des Medienzugriffs, Fehlererkennung und -korrektur
3	Vermittlungsschicht	Adressierung und Routing
4	Transportschicht	Auf- und Abbau einer logischen Verbindung, Korrektur der Reihenfolge und Quittierung von Paketen, Duplikaterkennung
5	Sitzungsschicht	Ablaufsteuerung von Dialogen
6	Darstellungsschicht	Systemunabhängige Darstellung von Daten, Kompression und Verschlüsselung
7	Anwendungsschicht	Definition von Informationsmodellen und Diensten auf den Daten

**Abb. 2.19** Kommunikationsablauf im 7-Schicht-ISO/OSI-Modell

der jeweiligen Schicht angereichert. Die Größe der sog. Application Protocol Data Unit (APDU), also die Nutzdaten der Anwendungsschicht, ist daher meist deutlich kleiner als z. B. die Nutzdaten der Transportschicht, also die Transport Protocol Data Unit (TPDU). Auf der Empfängerseite werden die *Data Units* dann Schicht für Schicht wieder entpackt und die zusätzlichen Informationen der unterliegenden Schichten entfernt. Die Anwendungsschicht des Empfängers erhält entsprechend auch wieder nur genau die APDU, die der Sender verschickt hat.

Die **Bitübertragungsschicht** (*Physical Layer*) ist die unterste Schicht des OSI-Modells (Schicht 1). Standards auf dieser Ebene stellen der darüberliegenden Schicht einen ungesicherten Bitstrom zur Verfügung. Dafür müssen z. B. Übertragungsmedien sowie entsprechende Bauelemente, wie Antennen und Steckverbindungen, definiert werden. Zu den Aufgaben dieser Schicht gehört auch die sog. Leitungscodierung. Die zu übertragenden Bits müssen in konkrete Signalpegel (z. B. elektrisch auf einem Kupferkabel) abgebildet werden. Das einfachste Verfahren, um Bits in Form von elektrischen Signalen auf einem Kabel darzustellen, wird als Non Return to Zero (NRZ) bezeichnet (siehe z. B. [32, S. 64 f.]). Bei diesem Verfahren werden den beiden Bits jeweils konstante Leitungszustände zugeordnet. Die „0“ wird auf einem Kupferkabel z. B. über den Spannungswert 3 bis 15 Volt, die „1“ über den Spannungswert -3 bis -15 Volt abgebildet. Das Verfahren hat allerdings Nachteile. Der resultierende Signalverlauf ist nicht gleichanteilsfrei und auch der Übertragungstakt kann nicht aus dem Signal extrahiert werden. Entsprechend existieren andere Verfahren, wie z. B. das Manchester-Verfahren oder die 4B/5B-Codierung, welche diese Nachteile kompensieren (siehe [13, S. 203]).

Bei den Übertragungsmedien unterscheidet man zwischen kabelloser (Funk) und kabelgebundener Übertragung. Rechner und Übertragungsstrecken können dabei auf unterschiedliche Weise angeordnet werden. Die Anordnung wird auch als **Netztopologie** bezeichnet. Die einfachste Anordnung ist die **Punkt-zu-Punkt-Verbindung**, bei der zwei Teilnehmer über ein für sie exklusives Übertragungsmedium, z. B. eine Leitung, miteinander verbunden sind. Andere Topologien sind z. B. die **Bus-, Ring- oder Sterntopologie**. Die Wahl der Netztopologie hat u. a. Einfluss auf die Ausfallsicherheit des Netzes. Existieren mehrere alternative Wege von einem Sender zu einem Empfänger, kann die Datenübertragung mitunter selbst bei Ausfall einer Übertragungsstrecke aufrechterhalten werden. Bei der Datenübertragung per Funk existiert grundsätzlich nur ein gemeinsam genutztes Übertragungsmedium. Dies ist dann mit der Bustopologie gleichzusetzen. Abb. 2.20 zeigt die unterschiedlichen Netztopologien.

Beim Aufbau von komplexen Netztopologien kommen auch aktive Netzkopplungselemente zum Einsatz, welche die Aufgabe haben, Nachrichten weiterzuleiten. Der Bitübertragungsschicht sind dabei die sog. Repeater und Hubs zuzuordnen. Ein Repeater ist ein Signalverstärker, der die Reichweite von Nachrichten auf einem Übertragungsmedium erhöht. Eine ähnliche Aufgabe besitzt auch ein Hub, der bei kabelbasierten Übertragungsmedien mehrere Netzketten sternförmig miteinander verbindet.

Auf der zweiten Schicht befindet sich die **Sicherungsschicht** (*Data Link Layer*). Während sich die Bitübertragungsschicht der Übertragung einzelner Bits widmet, befasst

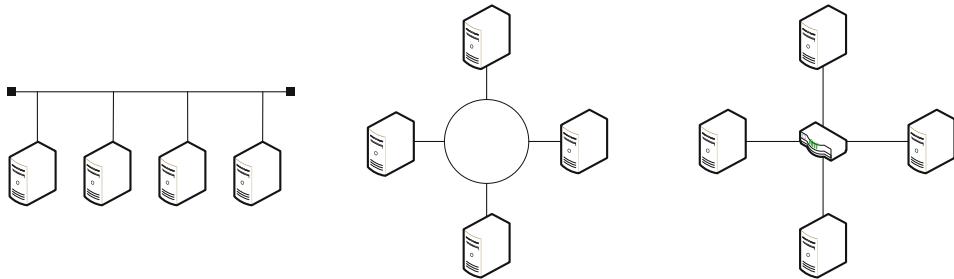


Abb. 2.20 Bus-, Ring- und Sterntopologie von Kommunikationsnetzen

sich die Sicherungsschicht mit der Übertragung einer Gruppe von Bits, meist als *Frames* bezeichnet. Die Sicherungsschicht hat dabei die Aufgabe, eine weitgehend fehlerfreie Datenübertragung auf einem potenziell gestörten Übertragungsmedien zu gewährleisten.

Für die Erkennung und ggf. auch die Korrektur von **Übertragungsfehlern** werden den Frames meist zusätzliche, redundante Daten hinzugefügt, was auch als **Kanalcodierung** bezeichnet wird. Das einfachste Verfahren zur Erkennung von Übertragungsfehlern ist die Verwendung sog. Paritätsbits. Dabei wird nach einer Menge von übertragenen Bits, z. B. 8 Bits, ein weiteres Bit hinzugefügt, sodass insgesamt eine gerade (*Even*) oder ungerade (*Odd*) Anzahl von 1 übertragen wird. Trifft beim Empfänger dann dennoch eine andere Anzahl von 1 ein, wurde der Frame nicht korrekt übertragen und kann neu angefordert werden. Mehr als ein Übertragungsfehler kann dadurch allerdings nicht erkannt werden. Komplexere Verfahren, wie die **zyklische Redundanzprüfung** (Cyclic Redundancy Check (CRC)), können im Idealfall sogar die Übertragungsfehler korrigieren und somit eine erneute Übertragung vermeiden.

Auf der Sicherungsschicht kann das Übertragungsmedium potenziell bereits durch mehrere Kommunikationsteilnehmer gleichzeitig genutzt werden. Die Teilnehmer müssen daher durch Vergabe einer physischen Adresse (sog. MAC-Adresse) eindeutig identifiziert werden können. Ein **Switch**, ein aktives Netzkopplungselement, kann diese Adressen auswerten, um den Datenverkehr (im Gegensatz zum Hub) intelligent zwischen mehreren Übertragungsmedien zu vermitteln.

Eine wichtige Aufgabe der Sicherungsschicht ist die **Zugriffs- oder Medienkontrolle**. Es muss sichergestellt werden, dass auf einem gemeinsam genutzten Übertragungsmedium (z. B. einem Bus) lediglich ein Teilnehmer pro Zeiteinheit Daten sendet, da sich Signale sonst überlagern können und es dann zu **Kollisionen** kommt. Für die Zugriffskontrolle existiert eine Vielzahl unterschiedlicher Verfahren. Dabei kann die Kommunikation grundsätzlich so gesteuert werden, dass Kollisionen von vornherein auszuschließen sind, oder Kollisionen können behandelt werden, wenn sie wirklich eintreten. Für die **Kollisionsvermeidung** existieren die folgenden Möglichkeiten:

- Es wird nur genau einem Teilnehmer erlaubt, spontan Nachrichten auf ein gemeinsam genutztes Medium zu senden. Alle anderen dürfen nur antworten. Dies wird auch als

Unbalanced Communication bezeichnet, da die Kommunikationspartner dabei nicht gleichberechtigt sind.

- Eine andere Möglichkeit der Kollisionsvermeidung ist, Zeitscheiben auf dem Übertragungsmedium zu vergeben, in denen die Teilnehmer senden dürfen. Dies setzt allerdings eine sehr genaue Zeitsynchronisation der Teilnehmer voraus.

Auch für die Kollisionsbehandlung sind mehrere Varianten bekannt. Hier werden Verfahren, wie z. B. CSMA/CD oder CSMA/CA, eingesetzt. Diese Verfahren werden immer dann eingesetzt, wenn mehrere Teilnehmer gleichberechtigt sind und prinzipiell jederzeit ein gemeinsames Übertragungsmedium zur Datenübertragung nutzen sollen. Bei beiden Verfahren prüft ein Sender zunächst, ob das Übertragungsmedium frei ist. Im ungünstigsten Fall sorgt aber die Laufzeit der Signale auf dem Medium dafür, dass ein zweiter Teilnehmer fast gleichzeitig zu senden beginnt und es zu einer Kollision kommt.

Das Verfahren mit dem Namen **Carrier Sense Multiple Access/Collision Detect (CSMA/CD)** kann eine solche Kollision erkennen und behandeln. Dieses Verfahren wird z. B. durch Ethernet genutzt. Im Falle einer Kollision brechen alle Teilnehmer die Übertragung unverzüglich ab. Anschließend wartet jeder einzelne Teilnehmer eine zufällig gewählte Zeitdauer, um dann zu prüfen, ob eine neue Datenübertragung starten kann. Durch die zufällig gewählte Wartezeit sind Systeme, die CSMA/CD einsetzen, nicht deterministisch, d. h., es existieren keine harten Zeitgarantien für den Versand von Nachrichten. Systeme, die CSMA/CD einsetzen, sind daher nicht echtzeitfähig (siehe [13, S. 203 ff.]).

Das Verfahren mit dem Namen **Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA)** ist ein etwas komplizierteres Verfahren der Kollisionsbehandlung. Es wird z. B. durch das Bussystem KNX genutzt (siehe Abschn. 5.2). Voraussetzung ist, dass auf dem Übertragungsmedium ein sog. dominantes gegenüber einem rezessiven Bit existiert. Das dominante Bit setzt sich gegenüber dem rezessiven Bit bei einer Kollision durch. Bei KNX ist das dominante Bit die „0“, welches sich bei einer Kollision gegenüber der „1“ durchsetzt. Ein Sender, der die „1“ überträgt, auf dem Medium aber eine „0“ liest, erkennt dadurch eine Kollision und kann dann seine Datenübertragung einstellen. Der Vorteil dieses Verfahrens ist, dass die Nachricht mit den dominanten Bits am Anfang immer vollständig übertragen werden kann. Durch bewusste Nutzung der dominanten Bits im Kopf einer Nachricht kann so auch die Priorität gegenüber Kollisionen erhöht werden.

Die dritte Schicht ist die **Vermittlungsschicht** (*Network Layer*). Hier wird die Übertragung von Daten zwischen Kommunikationspartnern in einem Netzwerk aus vielen Teilnehmern organisiert. Entsprechend müssen Adressen definiert werden, damit eine Nachricht zum Ziel gelangen kann. Bei der paketorientierten Datenübertragung schließt dies auch die Wegesuche (Routing) mit ein. Netzkoppelemente wie Router oder Switches müssen in die Lage versetzt werden, Pakete aus einem Netzsegment korrekt an ein anderes Netzsegment übergeben zu können. Für die Adressierung können Protokolle auf dieser Schicht unterschiedliche Konzepte einführen. Bei der sog. Unicast-Adressierung ist eine Nachricht für genau einen Empfänger bestimmt, mithilfe von sog. Multicast-Adressen

kann eine Nachricht direkt an eine Gruppe von Empfängern versendet werden. Ein sog. Broadcast dient dazu, alle Empfänger in bestimmten Netzwerksegmenten auf einmal zu erreichen. Für bestimmte Arten von Nachrichten (z. B. Alarmmeldungen) kann der Nachrichtenversand als Multi- oder Broadcast sehr effektiv sein, da nicht alle Empfänger explizit mit einer eigenen Nachricht adressiert werden müssen. Bekannte Protokolle dieser Schicht sind z. B. das Internetprotokoll (IP) oder das Internet Control Message Protocol (ICMP).

Die vierte Schicht, die **Transportschicht** (*Transport Layer*), führt für die darüber liegenden Schichten Dienste für eine gesicherte Prozess-zu-Prozess-Kommunikation ein. Dazu gehört vor allem das Konzept einer verbindungsorientierten Kommunikation, bei der eine (logische) Verbindung zwischen (zwei) Teilnehmern explizit auf- und wieder abgebaut wird. Auf Basis einer solchen Verbindung kann z. B. beim Empfänger eine falsche Reihenfolge eingehender Datenpakete korrigiert, fehlende Datenpakete können neu angefragt oder Paketduplicata aussortiert werden, ohne dass die darüberliegenden Schichten sich darum kümmern müssten. Das Internetprotokoll Transmission Control Protocol (TCP) realisiert eine solche verbindungsorientierte Kommunikation.

Auf der fünften Schicht, der **Sitzungsschicht** (*Session Layer*), wird der Ablauf eines Dialogs zwischen Teilnehmern organisiert. Beispiele für solche Dialoge sind Anfragen an einen Dienst oder die Authentifizierung eines Teilnehmers. Dafür wird eine sinnvolle und erlaubte Reihenfolge von Nachrichten definiert, die auch die Festlegung von Zeitbeschränkungen (*Timeout*) umfasst.

Bei einem solchen Dialog haben die Teilnehmer meist eine klare Rollenverteilung. Das bekannteste Konzept einer solchen Rollenverteilung wird als **Client/Server-Modell** bezeichnet. Ein Client baut eine Verbindung zu einem Server auf, stellt eine Anfrage und wartet auf die Antwort. Diese Art des Ablaufs wird auch als synchrone Kommunikation bezeichnet. Das Client/Server-Prinzip wird im Kontext der meisten Internetdienste (z. B. WWW oder E-Mail), aber auch bei Maschine-zu-Maschine-Kommunikation auf Basis von Webservices genutzt. Das **Publish/Subscribe**-Konzept ist ein davon unterschiedliches Modell. Ein *Subscriber* meldet sich bei einem *Publisher* für bestimmte Themen an und erhält dann in Zukunft automatisch Benachrichtigungen vom *Publisher*, sobald Informationen zu einem der abonnierten Themen verfügbar sind. Eine Nachricht des *Publisher*s ist in diesem Verfahren von vielen Empfängern gleichzeitig empfangbar. Eine Antwort auf ein Abonnement erhält der *Subscriber* allerdings oft erst sehr viel später, also asynchron. Dieser Ablauf ist für Alarm- oder Statusmeldungen oft besser geeignet, da Nachrichten so sehr viel effizienter an mehrere Empfänger verteilt werden können. Zudem können so Dienste auch leicht auf mehrere Rechner verteilt werden. Der Publish/Subscribe-Mechanismus wird z. B. durch den Java Message Service (JMS) als nachrichtenorientierte Infrastruktur umgesetzt. Sowohl Webservices als auch JMS werden in der Energiewirtschaft für die Integration von Anwendungssystemen genutzt (siehe dazu Abschn. 3.1).

Die sechste Schicht ist die **Darstellungsschicht** (*Presentation Layer*). Hier werden systemspezifische Anwendungsdaten, z. B. Objekte oder Nachrichten der Anwendungs-

schicht, auf eine systemunabhängige Darstellungsform in Form von Einsen und Nullen abgebildet. Für eine solche Abbildung existieren im Prinzip beliebig viele Möglichkeiten und jeder Kommunikationsteilnehmer in einem Netzwerk kann hier unterschiedliche Präferenzen haben. Welche Variante gewählt wird, kann von vornherein fest definiert sein oder durch die Darstellungsschicht während des Kommunikationsaufbaus zwischen Teilnehmern ausgehandelt werden. Für die Beschreibung einer Abbildung von Anwendungsobjekten auf Bitströme werden in Kommunikationsstandards oft Methoden, wie Abstract Syntax Notation One (ASN.1) oder External Data Representation (XDR), genutzt (siehe [1]). Ist eine Nachricht mithilfe von ASN.1 beschrieben, wird durch die sog. *Encoding Rules* die Abbildung der Nachricht auf einen Bitstring definiert. Eine mögliche Abbildung wird durch die *Basic Encoding Rules* beschrieben, dabei wird jeder Abschnitt einer Nachricht aus den drei Teilen *Tag*, *Length* und *Value* aufgebaut (TLV). Der *Tag* ist ein eindeutiger Bitstring, der den Anfang eines Nachrichtenteils einleitet, die Länge zeigt die Anzahl der Bits an, die diesen Teil der Nachricht ausmacht, und der Wert beinhaltet die eigentlichen Nutzdaten der Nachricht. BACnet nutzt z. B. dieses Verfahren, um eine eindeutige Abbildung von Daten auf Bistings in Nachrichten zu erreichen (siehe Abschn. 5.2).

Die oberste und siebte Schicht ist die **Anwendungsschicht** (*Application Layer*). Hier werden Funktionen und Schnittstellen bereitgestellt, die durch spezifische Anwendungen genutzt werden können. Bekannte Beispiele von Protokollen auf dieser Schicht sind das Hypertext Transfer Protocol (HTTP) oder das Simple Mail Transfer Protocol (SMTP).

Anwendungsprotokolle beinhalten häufig die explizite oder implizite Definition eines Informationsmodells. Um eine Funktion zu definieren, die in der Lage ist, eine E-Mail zu versenden, muss eine E-Mail als Objekt mitsamt ihren Eigenschaften zunächst klar definiert werden. Dies gilt auch für Anwendungsprotokolle in der Energiewirtschaft. Ein Kern des Kommunikationsstandards ISO/IEC 61850 ist das Informationsmodell, mit dessen Hilfe Schaltanlagen und verteilte, regenerative Energieanlagen abgebildet werden können (siehe Abschn. 5.1). KNX und BACnet beinhalten Informationsmodelle, auf deren Basis Komponenten einer Gebäudeautomation dargestellt werden können (siehe Abschn. 5.2). COSEM definiert ein Informationsmodell für Smart Meter (siehe Abschn. 5.3). Die Anwendungsschicht muss weiterhin Funktionen und Dienste anbieten, um Zugriff auf die Objekte des Informationsmodells über das Kommunikationsnetz zu gewähren.

Das ISO/OSI-Modell beschreibt, welche Aufgaben für eine erfolgreiche Datenübertragung auf den einzelnen Schichten zu bewältigen sind. Für jede Schicht existiert in der Praxis eine schier unüberschaubare Menge von Alternativen. Welche Standards und Technologien sinnvoll sind, hängt allerdings von den konkreten Anforderungen an die Datenübertragung ab:

- die räumliche Ausdehnung des Netzes: Kurzstrecke, Lokales (LAN) oder globales Netz,

- die Art der Kommunikationsteilnehmer: Mikrorechner oder vollwertige Rechnersysteme,
- die Qualitätsanforderungen an die Datenübertragung: Übertragungsgeschwindigkeit, Latenz, Determinismus und Sicherheit.

Auf der Bitübertragungsschicht werden Daten meist seriell, also nacheinander übertragen. Beispiele für entsprechende Standards sind **RS-232** oder RS-485. Mithilfe von RS-232 können Punkt-zu-Punkt-Verbindungen zwischen zwei Rechnern oder externen Peripheriegeräten hergestellt werden (siehe [23]). RS-232 nutzt dafür meist drei Leitungen, auf denen die Daten auf Basis des NRZ-Codes übertragen werden. Die drei Leitungen werden als GND (Erde, *Ground*), RxD (Empfangsdaten, *Receive Data*) und TxD (Sendedaten, *Transmit Data*) bezeichnet. Durch die getrennten Datenleitungen für Sender und Empfänger ist eine Vollduplexübertragung möglich, d. h., Daten können in beide Richtungen gleichzeitig übertragen werden. RS-232 ist untrennbar mit einem elektronischen Bauteil namens **Universal Asynchronous Receiver Transmitter (UART)** verbunden. Dieses Bauteil übersetzt ein zu übertragendes Datenwort, z. B. ein Byte, in einen seriellen Bitstrom und fügt noch weitere Informationen, z. B. Start- und Stop-Bits, und Informationen zur Fehlerkorrektur (Parity-Bit) hinzu. Durch die zusätzlichen Start- und Stop-Bits können die Daten asynchron, d. h. ohne gemeinsamen Takt, über die Leitung übertragen werden. RS-232 und UART erfüllen demnach die Aufgaben der ersten beiden ISO/OSI-Schichten (Bitübertragungs- und Sicherungsschicht). RS-232 besitzt allerdings deutliche Einschränkungen. Die Übertragungsgeschwindigkeit ist i. d. R. auf maximal 115.000 Bits/s begrenzt und die Leitungslänge darf 15 m meist nicht übersteigen. Einige Nachteile von RS-232 wurden mit **RS-485** beseitigt. Im Gegensatz zur NRZ-Codierung werden die Bits auf einer Zweidrahtleitung anhand einer Spannungsdifferenz codiert, was die Datenübertragung robuster macht. Zudem werden Abschlusswiderstände vorgeschrieben, um Signalreflexionen zu vermeiden. Dies erlaubt wesentlich höhere Übertragungsraten, Leitungslängen und die Verkettung (wird als *Daisy Chain* bezeichnet) mehrerer Sender- und Empfänger auf einer gemeinsamen Leitung. Sowohl RS-232 als auch RS-485 findet sich heute vor allem in Bestandssystemen z. B. im Bereich der Prozessautomation (siehe Abschn. 5.1).

In der Bürokommunikation hat sich bei den kabelgebundenen Netzen in den letzten Jahren der **Ethernet**standard etabliert. Dieser wird auf der Bitübertragungs- und der Sicherungsschicht eingeordnet. Fast alle lokalen Netze (Local Area Network (LAN)) werden heute auf Basis von Ethernet aufgebaut. Ethernet wird aber auch zunehmend in industriellen Anlagen als Kommunikationsstandard eingesetzt. Ethernet ist dabei eigentlich ein Synonym für mehrere Standards, die unter dem Dach der Arbeitsgruppe 802.3 des Institute of Electrical and Electronics Engineers (IEEE) verantwortet werden. Ethernet nutzt meist mehrere verdrillte Zweidrahtleitungen (*Twisted Pair*, TP) als Übertragungsmedium. Die Kabel sind meist mit sog. RJ-45-Steckern ausgestattet. Die dazu passenden Buchsen finden sich an Netzwerkkarten oder Switches.

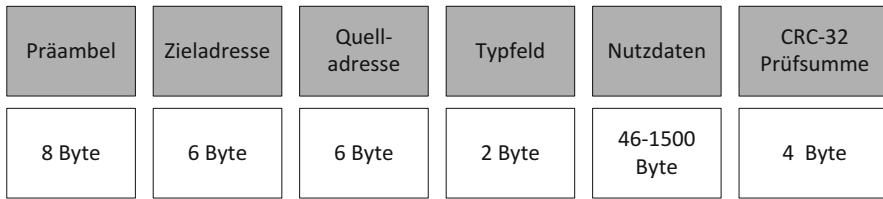


Abb. 2.21 Aufbau eines Ethernetframes

Für die Leitungscodierung kommt bei Ethernet entweder die Manchester-Codierung oder 4B/5B zum Einsatz. Diese Verfahren haben den Vorteil, dass der Empfänger den Übertragungstakt mithilfe der Pegelwechsel auf dem Übertragungsmedium selbst erkennen kann. Dadurch wird keine eigene Leitung zur Synchronisation des Taktes benötigt. Daten werden bei Ethernet in Form von einzelnen Datenpaketen, sog. Frames, übertragen. Der Aufbau eines solchen Frames wird in Abb. 2.21 dargestellt.

Ursprünglich war Ethernet auf eine Bustopologie ausgelegt, bei der ein gemeinsames Übertragungsmedium, z. B. ein Koaxialkabel, genutzt wurde. Datenframes werden spontan, also asynchron von einem Sender auf dem gemeinsamen Bus gesendet. Der Empfänger fischt dann genau die Nachrichten aus dem Strom heraus, die an ihn adressiert sind. Heute werden Ethernetnetze meist nicht mehr in einer Bustopologie, sondern in einer Sterntopologie betrieben. Die Datenpakete werden dann durch einen Switch im Netz weitervermittelt (*Switched Media*). Dies erlaubt den Aufbau von fast unbegrenzt großen Netzen auf Basis von Ethernet.

Das Internet ist bekannt als ein weltweiter Zusammenschluss von Tausenden Einzelnetzwerken. Über das Internet sind heterogene Rechnersysteme in der Lage, global Daten auszutauschen und Dienste anzubieten und zu nutzen. Dies wird durch einen gemeinsamen Satz von Standards sichergestellt. Der Kern dieser Standards wird durch die beiden Protokolle TCP und IP gebildet. Diese Protokolle finden sich auf den ISO/OSI-Ebenen Vermittlung (Ebene 3) und Transport (Ebene 4). Fast der gesamte Datenverkehr, der z. B. für das World Wide Web (WWW) und für E-Mails abgewickelt wird, fußt auf der Nutzung von TCP/IP. Der Erfolg dieser Protokolle wird durch mehrere spezielle Eigenschaften begründet.

Während der Datenkommunikation wird ein Datenstrom in mehrere voneinander unabhängige Pakete zerteilt. Jedes Datenpaket wird um einen Kopf (Header) erweitert, in dem die eindeutige Adresse der Quelle und des Ziels vermerkt ist. Jeder Kommunikationsteilnehmer im Internet wird anhand einer eindeutigen Adresse, der IP-Adresse identifiziert. Eine solche IP-Adresse besteht (bei IPv4) aus 4 Byte, z. B. „87.143.44.25“. Größere Netze können anhand von Subnetzen gruppiert werden, einer Hochschule können z. B. bestimmte Adressbereiche zugewiesen werden. Netzkopplungselemente wie Router oder Switches können die Pakete anhand dieser Informationen individuell weitervermitteln, was als **Routing** bezeichnet wird. Durch das Routing kann jedes Paket einen individuellen Weg zum Ziel nehmen. Dabei kann z. B. die Auslastung des Netzes berücksichtigt

werden, sodass besonders stark frequentierte Teilstrecken entlastet werden. Dies macht die Datenübertragung sehr robust.

Mithilfe des TCP-Protokolls wird eine logische Ende-zu-Ende-Verbindung realisiert. Sender und Empfänger können darüber Daten in beide Richtungen gleichzeitig übertragen (Vollduplex). Dabei wird aber nicht wie bei einer klassischen Telefonleitung eine exklusive Leitung reserviert, die Verbindung existiert rein virtuell. Sowohl Sender als auch Empfänger vermerken, dass sie miteinander in Verbindung stehen. Dabei wird die Verbindung eindeutig durch zwei Endpunkte (IP-Adresse + Port) identifiziert. Das TCP-Protokoll sorgt dann dafür, dass eingehende Datenpakete in der richtigen Reihenfolge wieder zusammengesetzt werden. Eventuell verloren gegangene Datenpakete werden erneut angefordert. TCP gleicht somit die Unsicherheiten unterliegender Schichten aus.

Die OSI-Ebenen 5–7 (Sitzung, Darstellung und Anwendung) werden im Internet durch spezifische Anwendungen ausgefüllt. Hier findet sich z. B. mit SMTP eine Anwendung zum Versand von E-Mails oder mit File Transfer Protocol (FTP) ein Protokoll zur Übertragung von Dateien. Die bekannteste Anwendung des Internets ist sicher das WWW. Das WWW besteht aus vielen verteilten Webseiten, die anhand einer Adresse, der sog. Unified Resource Locator (URL), auf einem Rechner im Internet eindeutig lokalisiert werden können. Die Webseiten können über die URLs miteinander verlinkt werden, wodurch die namensgebende, spinnennetzartige Struktur des WWW entsteht. Der Inhalt der Webseiten basiert auf einer textbasierten Auszeichnungssprache, der sog. Hypertext Markup Language (HTML), die XML recht ähnlich ist. Ein Webbrower kann die Kommunikationsinfrastruktur des Internets nutzen, um solche Dokumente von einem entfernten Rechner abzurufen. Dazu formuliert ein Client (Browser) eine Anfrage gegenüber einem Webserver anhand des Anwendungsprotokolls HTTP. HTTP definiert dabei nur einige wenige Befehle, z. B. GET, um ein Dokument abzurufen, oder POST, um Daten vom Client an den Server zu schicken. HTTP ist ein zustandsloses Protokoll, nachdem der Webserver die Antwort auf eine Anfrage gesendet hat, wird die Verbindung wieder geschlossen. Ein Webserver kennt daher i. d. R. nicht die Historie aller Anfragen von einem bestimmten Client.

HTML-Dokumente sind nicht dazu geeignet, automatisiert von Maschinen ausgewertet zu werden. Die grafische Darstellung einer Webseite im Webbrower ist ausschließlich dazu gedacht, von Menschen interpretiert zu werden. Das HTTP-Protokoll kann aber nicht nur dazu genutzt werden, um HTML-Dokumente zu übertragen. Mithilfe von HTTP können auch maschinenlesbare Datenformate, wie z. B. XML oder JavaScript Object Notation (JSON), übertragen werden. Aus dieser Idee entspringen die **Webservices**. Webservices dienen der Maschine-zu-Maschine-Kommunikation, sodass Rechner automatisiert Daten über das Internet austauschen können. Auch in der Energiewirtschaft werden Webservices genutzt, z. B. im Kontext von Smart-Metern (siehe Abschn. 5.3).

Zu Diagnosezwecken und zur Fehlerbehebung ist es mitunter interessant, das Zusammenspiel zwischen den Protokollen der unterschiedlichen ISO/OSI-Ebenen sichtbar zu machen. Mit **Wireshark** existiert ein entsprechendes Werkzeug für alle gängigen PC-Plattformen (siehe [66]). Wireshark ist ein Open-Source-Softwarepaket, welches

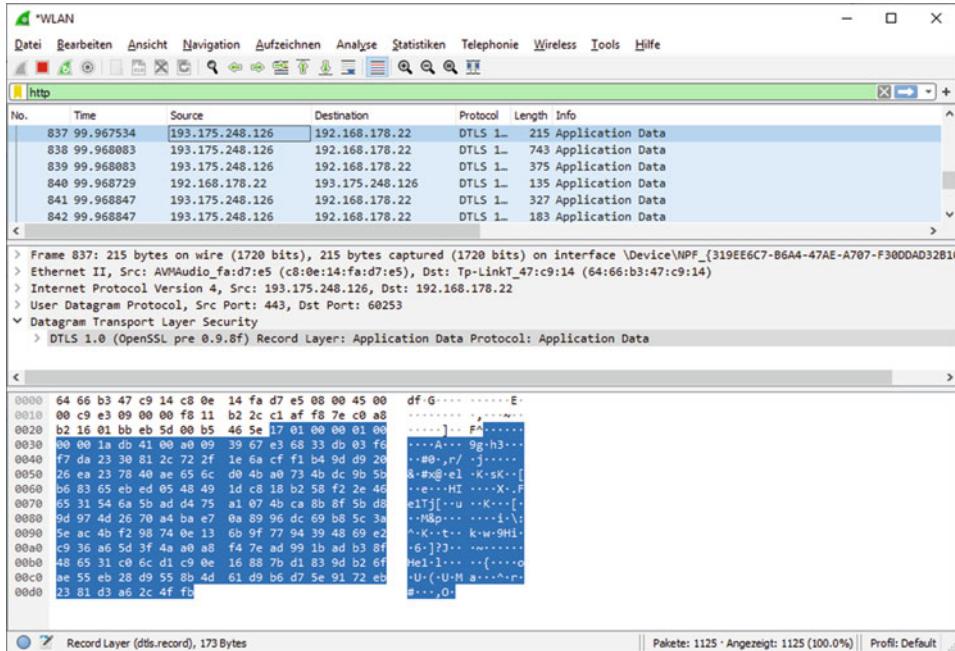


Abb. 2.22 Darstellung einer Netzwerkkommunikation in Wireshark

bei der Analyse und grafischen Darstellung von Datenprotokollen hilft. Es ermöglicht, die Datenkommunikation an einer Netzwerkschnittstelle (z. B. Ethernet oder WLAN) aufzuzeichnen und zu analysieren. Wireshark ist von Haus aus in der Lage, viele gängige Kommunikationsprotokolle, wie z. B. IP und TCP, zu parsen und die Bestandteile der Nachrichten übersichtlich darzustellen. Darüber hinaus werden aber auch *exotischere* Kommunikationsprotokolle erkannt, darunter solche, die im weiteren Verlauf dieses Lehrbuchs behandelt werden, z. B. MMS, KNX, LON oder BACNet (siehe Abschn. 5.1 und 5.2). Mithilfe von Filtern können zudem gezielt bestimmte Protokolle oder Adressen aus dem Datenstrom extrahiert werden. Abb. 2.22 zeigt ein Bildschirmfoto von Wireshark. Im oberen Fensterbereich werden einige Ethernetframes aufgelistet, im mittleren Fenster sind die erkannten Protokolle zu finden, im unteren Bereich werden die Nutzdaten der ausgewählten Protokollebene aufgeführt.

2.6 Automationstechnik und Feldbussysteme

Um eine verlässliche Energieversorgung sicherzustellen, werden umfangreiche energietechnische Anlagen benötigt. Dazu gehören z. B. Kraftwerke, regenerative Anlagen und Speicher. Für die Versorgung mit elektrischer Energie müssen diese Anlagen so betrieben werden, dass die Erzeugung zu jedem Zeitpunkt der Entnahme entspricht

(siehe Abschn. 2.1). Es werden zudem Energienetze benötigt, um elektrische Energie zu übertragen und zu verteilen. Als Teil dieser Netze werden dazu Schaltanlagen und Umspannwerke betrieben. In Erdgasnetzen existieren zudem Verdichterstationen, die den Druckabfall in den Leitungen kompensieren.

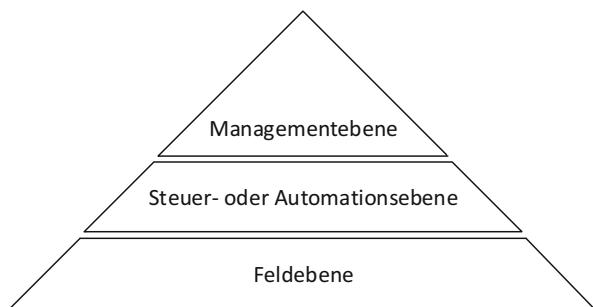
Auch in Gebäuden finden sich mittlerweile umfangreiche Anlagen. Gewerke, wie Heizung, Lüftung, Klimatisierung und Beleuchtung, werden zunehmend automatisch gesteuert und geregelt. Für den operativen Betrieb solcher Anlagen werden schon seit längerer Zeit Systeme aus der **Automations-** bzw. **Automatisierungstechnik** eingesetzt. Dabei müssen mitunter in Echtzeit Zustandsdaten erfasst, übertragen, ausgewertet und visualisiert werden sowie entsprechende Aktionen in Form von Steuerbefehlen aus Leitwarten an die Anlagen übermittelt werden.

Systeme der Automatisierungstechnik sind im Allgemeinen hierarchisch organisiert. Meist wird zwischen der Feldebene, der Steuer- oder Automationsebene und der Managementebene unterschieden (siehe [40] oder [32, S. 51]). Eine solche Struktur findet sich sowohl bei der Prozesssteuerung von Netzen und Energieanlagen, aber auch in der Gebäudetechnik. Jede dieser Ebenen bringt dabei eigene Anforderungen an die Informationsverarbeitung und Kommunikation mit sich. Auf den Ebenen kommen daher unterschiedliche Standards, Kommunikationsprotokolle und IT-Systeme zum Einsatz. Abb. 2.23 zeigt die Ebenen der Automatisierungspyramide.

Auf der **Feldebene** findet die Interaktion mit dem eigentlichen Prozess statt. Durch Sensoren werden Messgrößen, wie z. B. elektrische Spannung, Strom, Netzfrequenz, Temperatur, Anwesenheit, Helligkeit usw., erfasst und digitalisiert. Aktoren, wie z. B. Regler, Stellglieder, Schalter, Antriebe oder Pumpen, empfangen Befehle und setzen diese um. Steuer- und Regelaufgaben werden in der **Automationsebene** abgebildet. In der industriellen Prozesssteuerung werden hier sog. speicherprogrammierbare Steuerungen (SPS), im Englischen als Programmable Logic Controller (PLC) bezeichnet, eingesetzt. Messwerte und Schaltbefehle müssen zwischen Feld- und Automationsebene ausgetauscht werden, dazu müssen beide Ebenen über Datenübertragungswege miteinander verbunden sein.

Die Feld- und Automationsebene sind nicht immer klar voneinander zu unterscheiden. Oft sind Geräte verfügbar, die Aufgaben der Feld- und Automationsebene miteinander

Abb. 2.23 Die Ebenen der Automatisierungspyramide



integrieren. In Schaltanlagen spricht man von sog. Intelligent Electronic Device (IED), in der Gebäudeautomation werden Komponenten, die lokale Anwendungen umsetzen, als Direct Digital Control (DDC) bezeichnet.

Die Automationsebene kann, muss aber nicht, an eine **Managementebene** angebunden sein. Anlagen können in dieser Ebene überwacht, gesteuert und konfiguriert werden. Die Managementebene dient oft auch dazu, Anlagenteile unterschiedlicher Technologien oder Gewerke miteinander zu integrieren. In der Managementebene werden meist auch höherwertige Automationsaufgaben realisiert. Typische Aufgaben sind dann z. B. das Energiemanagement oder die Instandhaltungsplanung.

Die Aufgaben der Überwachung und Steuerung von Anlagen auf der Managementebene werden auch unter dem Sammelbegriff Supervisory Control and Data Aquisition (SCADA) zusammengefasst (siehe [4]). Dazu wird meist ein (Netz- oder Gebäude-)Leitsystem (auch SCADA-System) eingesetzt. Dieses stellt eine geeignete Benutzeroberfläche, das sog. Human Machine Interface (HMI), bereit. Im SCADA-System werden alle Informationen der angebundenen Anlagen integriert dargestellt, z. B. die aktuellen Betriebszustände, Alarne und Trends. Dabei werden Informationen aus der Feld- und Steuerungsebene aggregiert und grafisch visualisiert (siehe [6, S. 4–6]). Für die Darstellung der Informationen existieren entsprechende Standards (siehe [65]). Abb. 2.24 zeigt eine typische Netzteleitwarte eines Verteilnetzbetreibers.



Abb. 2.24 Netzteleitwarte der Energie Steiermark

Ein SCADA-System bietet darüber hinaus meist viele erweiterte Funktionen, z. B. die Historisierung und Archivierung der Daten, das Logging von Meldungen und die Reporterstellung und Benachrichtigungen im Fehlerfall (siehe [43]). Meist können die Anlagen auch aus dem SCADA-System heraus ferngesteuert werden. Entsprechende Befehle müssen dann an die Feld- und Steuerebene weitergegeben werden können. Entsprechend müssen zwischen der Feld- und Steuerungsebene und dem SCADA-System kontinuierlich Daten ausgetauscht werden. Dabei sind mitunter größere Distanzen zu überwinden. Die dazu notwendige Kommunikationstechnik wird dann auch als **Fernwirktechnik** bezeichnet.

Daten in Form von Schaltbefehlen und Messwerten müssen innerhalb und zwischen den Ebenen der Automatisierungspyramide ausgetauscht werden. Die Automatisierungstechnik stellt dabei oft hohe Anforderungen an die digitale Kommunikation, z. B. Determinismus, Echtzeit, Robustheit und Verfügbarkeit. Tab. 2.2 zeigt die unterschiedlichen Anforderungen, die sich aus der digitalen Kommunikation im Kontext von Industrie- oder Büroanwendungen ergeben (siehe dazu auch [8]).

Bei der digitalen Kommunikation in der Automatisierungstechnik wird vor allem zwischen der Feld- und Automationsebene oft die Echtzeitfähigkeit der Datenübertragung benötigt. Historisch gesehen war es üblich, Sensoren und Aktoren mit der Automationsebene per Direktverkabelung zu verbinden. Der hohe Verkabelungsaufwand sorgte zunehmend dafür, dass die Direktverkabelung durch Feldbusssysteme abgelöst wurde.

Ein **Feldbus** ist ein digitaler, serieller Datenbus für die Kommunikation zwischen Geräten der industriellen Automatisierungstechnik, der eine sichere und deterministische Datenübertragung garantiert. In der Energiewirtschaft werden solche Systeme insbesondere im Kontext der Automation von Energieanlagen eingesetzt. Es existieren viele unterschiedliche Feldbusstandards, die untereinander meist nicht kompatibel zueinander

Tab. 2.2 Vergleich der Anforderungen an Kommunikationssysteme im Büro- bzw. Industrie Einsatz

	Büro	Industrie
Installation	Feste Grundinstallation im Gebäude, variabler Geräteanschluss an Standardarbeitsplätzen, überwiegend sternförmige Verkabelung	Anlagenabhängige Verkabelung und Kabelführung, feldkonfektionierbare Steckverbindungen bis IP67, redundante Verkabelung, häufig Ringstrukturen
Daten	Große Datenpakete, mittlere Netzverfügbarkeit, hauptsächlich azyklische Übertragung, kein Echtzeitverhalten notwendig	Kleine Datenpakete, sehr hohe Netzverfügbarkeit, hauptsächlich zyklische Übertragung, Echtzeitverhalten notwendig
Umwelt	Normaler Temperaturbereich, wenig Staub, Feuchtigkeit und Erschütterungen, kaum mechanische oder chemische Belastung, geringe EMV-Belastung	Erweiterter Temperaturbereich, Staub, Feuchtigkeit und Erschütterungen möglich, Gefahr durch mechanische Beschädigungen oder chemische Belastung, hohe EMV-Belastung

sind, z. B. Modbus, Profibus, DeviceNet, ControlNet, CC-Link, Interbus und viele mehr. Mit IEC 61158 und IEC 61784 existieren Normreihen, die sich mit Feldbussystemen auseinandersetzen. Darin werden allerdings alle gängigen Feldbussysteme in Form von sog. Protokollfamilien aufgeführt, eine normative Funktion haben die Normreihen daher nicht. Ein Feldbussystem wird meist nicht nach unabhängigen Kriterien ausgewählt. Die Hersteller von speicherprogrammierbaren Steuerungen und Prozessleitsystemen präferieren i. d. R. *ihre* Feldbussysteme.

Ein Beispiel für einen Feldbusstandard ist **Modbus**. Modbus ist bereits recht alt, seine Entwicklung begann in den späten 1970er-Jahren. Modbus wurde ursprünglich von dem US-Unternehmen Modicon entwickelt, damit die hauseigenen speicherprogrammierbaren Steuerungen Daten austauschen konnten. Heute gehört Modbus zu den weltweit am meisten eingesetzten Feldbussystemen. Dies liegt auch daran, dass Modbus ein recht einfacher, offener Standard ist, der lizenzzfrei eingesetzt werden kann (siehe [26]). Modbus existiert in mehreren Varianten. Die einfachste Variante nutzt eine serielle Datenübertragung über RS-232 oder RS-485. Dabei können die Daten entweder binär codiert werden (wird als Modbus RTU bezeichnet) oder auf der Basis von American Standard Code for Information Interchange (ASCII). Der letztere Fall wird auch als Modbus ASCII bezeichnet (siehe [15]). Das Modbus-Protokoll kann allerdings auch auf Basis von TCP/IP-Netzen genutzt werden. Mit Modbus+ existiert zudem eine proprietäre Variante der Firma Modicon mit einer angepassten Sicherungs- und Bitübertragungsschicht. Abb. 2.25 zeigt die Abbildung der Modbus-Varianten auf das ISO/OSI-Modell.

Bei der seriellen Datenübertragung wird bei Modbus ein gemeinsamer Bus genutzt. Dabei muss sichergestellt werden, dass nur ein Gerät den gemeinsamen Bus pro Zeiteinheit zur Datenübertragung nutzt (Zugriffsschutz). In Modbus wird dies dadurch sichergestellt, dass zwischen den zwei Gerätetypen Master und Slave unterschieden wird. Auf einem Bus existiert genau ein Master, der mit mehreren Slaves in Verbindung stehen kann. Diese

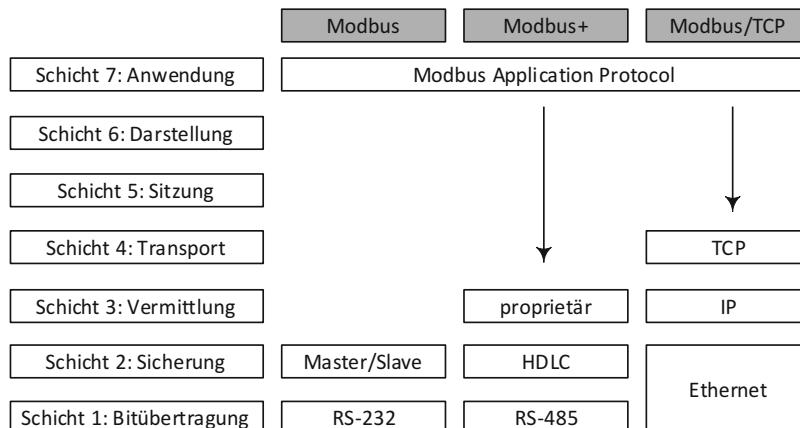


Abb. 2.25 Abbildung des Modbus-Standards auf die Schichten des ISO/OSI-Modells

Kommunikationsstrategie wird als *Unbalanced Communication* bezeichnet. Modbus-Slaves werden meist durch sog. SPS realisiert. An diese Mikrorechner sind Sensoren oder Aktoren angeschlossen (siehe Kap. 5). Nur der Master darf auf dem Bus unaufgefordert Nachrichten verschicken, ein Slave darf lediglich auf Anfragen eines Masters reagieren. Daten der Slaves liest oder schreibt der Master der Reihe nach im sog. Polling-Verfahren. Ändern sich in einem Slave die Daten zwischen zwei Abrufen mehrfach, gehen diese Änderungen verloren. Die maximale Polling-Rate ist davon abhängig, wie viele Geräte am Bus angeschlossen sind, wie hoch die verwendete Übertragungsgeschwindigkeit ist und wie viele Daten zu übertragen sind.

Modbus definiert ein eigenes Nachrichtenformat, welches auch als Modbus Application Protocol (MBAP) bekannt ist. Alle Modbus-Varianten nutzen dieses Nachrichtenformat, setzen aber mitunter andere Technologien auf den unterliegenden OSI-Schichten ein. Abb. 2.26 zeigt das Nachrichtenformat von Modbus.

Eine Nachricht kann direkt adressiert bzw. als Rundnachricht (*Broadcast*) von einem Master an alle Slaves gerichtet sein. Eine Nachricht besteht aus der Zieladresse, dem sog. Funktionscode, den Nutzdaten und einer Prüfsumme. Die Zieladresse „0“ ist reserviert für einen Broadcast, Geräte dürfen die Adressen „1–247“ belegen. Der Funktionscode repräsentiert die gewünschte Funktion, 02h steht z. B. für die Funktion *Coil lesen*. Der Slave sendet dann die Daten in Form einer Nachricht zurück an den Master. Zwischen zwei Nachrichten muss eine Pause von mindestens 3,5 Zeichen eingehalten werden.

Das Informationsmodell eines Modbus-Slaves ergibt sich eher implizit aus dem Nachrichtenformat. Man kann es sich als Arbeitsblatt einer Tabellenkalkulation vorstellen, welches bis zu 40.000 Zeilen besitzt. Jede Zeile besitzt eine eindeutige numerische Adresse. In jeder Speicherstelle können zwei unterschiedliche Arten von Daten abgelegt werden. Sogenannte Coils repräsentieren digitale Schaltzustände, 1 Bit = An/Aus. Register sind jeweils 16 Bit breit und können z. B. analoge Messwerte aufnehmen. Dabei ist nicht festgelegt, wie die 16 Bit interpretiert werden sollen. Es ist zudem üblich, größere Daten auf mehrere Register aufzuteilen. Sowohl Coils als auch Register können jeweils nur lesbar oder les- und schreibbar sein. Durch die Kombination des Funktionscodes und der Adresse des Coils bzw. des Registers wird in einer Nachricht eindeutig definiert, welche Daten des Slaves gelesen oder geschrieben werden sollen. Der Funktionscode 04h zeigt z. B. das Lesen eines *Analog-Output-Registers* an. Zusätzlich muss dann die Adresse des entsprechenden Registers übergeben werden. Welche Coils oder Register in einem

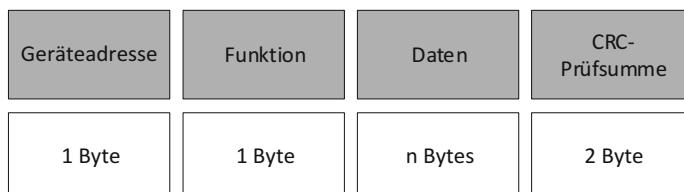


Abb. 2.26 Nachrichtenformat von Modbus

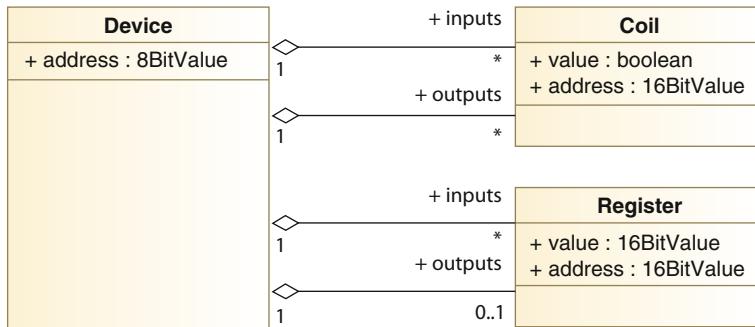


Abb. 2.27 Modbus-Informationsmodell als Klassendiagramm der UML

Slave existieren, an welcher Adresse etwas abgelegt ist und was die Coils und Register letztendlich repräsentieren, kann durch die Entwickler frei festgelegt werden. Durch diese Freiheiten unterscheiden sich modbus-fähige Geräte allerdings meist stark von Hersteller zu Hersteller. Ein Modbus-Slave kann also nicht ohne größeren Änderungsaufwand einfach durch ein anderes Gerät ersetzt werden. Abb. 2.27 zeigt das Informationsmodell von Modbus als Klassendiagramm der UML.

Ein weitaus komplexereres Beispiel für einen Feldbus ist der **Profibus** (*Process Field Bus*). Dieser wird durch die Profibus-Nutzerorganisation definiert und vor allem im Kontext von Automatisierungsanlagen der Firmen Siemens und ABB eingesetzt. Diese verfügen über einen signifikanten Weltmarktanteil im Bereich der Kraftwerksleittechnik (siehe [20]). Profibus wird als Systembaukasten bezeichnet. Die Profibus-Module auf den OSI-Schichten 1,2 und 7 können zu unterschiedlichen Systemvarianten zusammengestellt werden (siehe [5]). Eine besonders beliebte Variante ist Profibus DP, wobei DP für *dezentrale Peripherie!* steht. Profibus DP ist in vielen Teilen zu Modbus recht ähnlich. Auch hier wird ein gemeinsamer serieller Bus auf Basis von RS-485 eingesetzt. Im Gegensatz zu Modbus sind aber Steckverbindungen genau definiert. Darüber hinaus werden viele weitere Vorgaben gemacht, z. B. zur Verkabelung und zum Timing. Profibus kann im Gegensatz zu Modbus nicht über ein Modem genutzt werden. Bei Profibus DP wird ebenfalls das Master/Slave-Verfahren für den Zugriffsschutz eingesetzt. Als Erweiterung gegenüber Modbus darf aber mehr als ein Master auf einem Bus existieren. In diesem Fall wird die Zugriffsberechtigung zwischen mehreren Mاستern in Form eines *Tokens* reihum nach dem Prinzip des *Token-Passing* vergeben. Auf der Anwendungsschicht bietet Profibus DP drei Versionen eines Nachrichtenprotokolls an.

- Mit **DP-V0** können Daten zyklisch zwischen Master und Slave ausgetauscht werden (wie Modbus).
- **DP-V1** führt zusätzlich azyklischen Datenaustausch ein. Dieser Modus ist die Voraussetzung für die Parametrisierung und Kalibrierung der Feldgeräte im laufenden Betrieb und die Einführung bestätigter Alarmmeldungen.

- **DP-V2** ermöglicht u. a. den sog. Slave-Querverkehr mittels Broadcast-Meldungen, taktisochrone Nachrichten und eine Zeitsynchronisierung.

Weitere Details zu Profibus finden sich z. B. in [30] oder [31].

Neben den Feldbussystemen wird im industriellen Umfeld zunehmend Ethernet als Übertragungsstandard eingesetzt. Hierfür existieren mehrere gute Gründe. Im Vergleich zu Feldbussystemen erlaubt Ethernet sehr hohe Übertragungsraten. Es ist ein offenes Protokoll und Netzkomponenten sind preiswert. Da Ethernet im Büroumfeld sehr weitverbreitet ist, erlaubt dies zudem die vertikale Integration der Automatisierungssysteme in die Managementebene. Die erweiterten Umweltanforderungen werden im Industriuumfeld durch angepasste Hardware gelöst. Es existiert ein großes Angebot von robusten Netzkomponenten (z. B. für die Hutschienenmontage), Kabeln und Steckverbindungen. Industrielle Anwendungen stellen allerdings hohe Anforderungen an die Datenübertragung, z. B. hohe Netzverfügbarkeit und Echtzeitverhalten. Die Datenübertragung auf Basis von Ethernet allein ist allerdings nicht deterministisch (siehe [17]). Über Switches können Frames im Netz unterschiedliche Wege nehmen. Die Übertragungszeit hat dann für unterschiedliche Frames eine hohe Varianz (großer Jitter). Es kann auch zu Paketkollisionen oder verworfenen Paketen kommen. Um eine sichere Datenübertragung zu garantieren, ist Ethernet zudem auf Protokolle höherer Schichten angewiesen, z. B. TCP/IP. Für echtzeitfähige Anwendungen ist dies nicht akzeptabel. Ohne Weiteres kann Ethernet also nicht auf der Feldebene in Automatisierungssystemen eingesetzt werden. Auf der Grundlage von Ethernet wurden aber verschiedene Standards für die Datenübertragung im industriellen Umfeld geschaffen. Diese gleichen mithilfe unterschiedlicher Verfahren die Nachteile (vor allem den fehlenden Determinismus) aus. Diese Standards werden unter dem Sammelbegriff **Industrial Ethernet** zusammengefasst. Beispiele für entsprechende Standards sind EtherNet/IP, EtherCAT, Sercos III und Powerlink.

Ein weitverbreiteter Standard aus dem Bereich von Industrial Ethernet ist **Profinet**. Dieser wird wie Profibus von der Profibus-Nutzerorganisation verwaltet. Profinet zerfällt in zwei Unterstandards:

- **Profinet CBA** (*Component-based Automation*) nutzt Ethernet und TCP/IP. Dieser Teil ist für die Kommunikation zwischen größeren Automatisierungsmodulen gedacht und ist nur bedingt echtzeitfähig.
- **Profinet IO** dient der Anbindung von dezentraler Peripherie (*Devices*) an eine Steuerung (*Controller*) auf der Feldebene. Viele Devices kommunizieren dabei zyklisch/azyklisch mit einem Controller. Profinet IO kann als ethernetbasierter Nachfolger von Profibus DP gesehen werden.

Um bei Profinet IO eine sichere und deterministische Datenübertragung sicherzustellen, wird Ethernet um drei Konzepte erweitert:

- **Synchronisation**

Profinet IO sorgt für eine deterministische Datenübertragung durch Synchronität. Die Übertragung der Daten erfolgt dabei synchron durch gleich große Ethernetframes in äquidistanten Zeitintervallen. Für alle Kommunikationsteilnehmer ist dadurch genau abschätzbar, wann sie als Nächstes Daten senden können. Dieses Verfahren wird als Isochronous Real Time (IRT) bezeichnet und ersetzt den nichtdeterministischen Zugriffsschutz CSMA/CD, der normalerweise bei Ethernet genutzt wird. Durch die synchrone Übertragung können sehr kleine Varianzen in den Übertragungszeiten sichergestellt werden. Man spricht dann auch von geringem *Jitter*. Die Voraussetzung für eine zeitsynchrone Datenübertragung ist die sehr genaue Synchronisierung aller Uhren im Netzwerk. Dazu wird das Precision Time Control Protocol (PTCP) verwendet, das als spezielle Variante auch in IEC 61588 definiert ist (siehe [14]).

- **Bandbreitenreservierung**

Bei der sog. Bandbreitenreservierung wird eine festgelegte Bandbreite des Netzwerks für hoch priorisierte Nachrichten genutzt. Diese Bandbreite wird dann als Echtzeit-autobahn für Daten bezeichnet. Der Rest des Zeitintervalls dient als offener Kanal für sonstige Nachrichten. Dieses Verfahren setzt allerdings spezielle Switches voraus, die in der Lage sind, die hoch priorisierten Nachrichten weiterzuleiten, andere Daten hingegen zwischenspeichern.

- **Scheduling**

Um Nachrichten effizient im Netz zu verteilen, kennt ein Controller die genaue Netztopologie. Weiter entfernte Elemente erhalten Nachrichten zuerst, sodass Laufzeiten ausgeglichen werden.

2.7 Standards der Energieinformatik

Im Energiesystem interagiert eine große Menge von unterschiedlichen Akteuren miteinander. Ziel ist die verlässliche Abwicklung des physikalisch-technischen und des kaufmännischen Energieflusses. Die technische Infrastruktur besteht dabei aus einer Vielzahl von Komponenten unterschiedlichster Hersteller. Eine nahtlose Zusammenarbeit kann nur dann gewährleistet werden, wenn gemeinsame Standards existieren und diese entsprechend eingehalten werden. An den zentralen Schnittstellen der Energiewirtschaft existieren ausreichend viele geeignete Standards, um die Digitalisierung voranzutreiben. Die Herausforderung liegt vielmehr darin, die Verbreitung der Standards und den einheitlichen Einsatz voranzutreiben (siehe [50, S. 4]). Eines der Hauptziele dieses Lehrbuchs ist es, dem Leser einen ersten Überblick über einige relevante Standards der Energieinformatik zu verschaffen.

Ein **Standard** bzw. eine **Norm** ist ein Dokument, welches einvernehmlich erstellt und durch eine anerkannte Organisation verabschiedet wurde. Für eine gemeinsame und wiederholte Nutzung werden durch einen Standard Regeln, Leitlinien oder Merkmale für Tätigkeiten und deren Ergebnisse festgelegt, sodass in einem bestimmten Kontext ein

Optimum erreicht werden kann (siehe [28]). Standards legen eine gemeinsame Sprache für alle Beteiligten fest und können sich dabei unterschiedliche Abstraktionsniveaus zu eigen machen. Manche Standards legen lediglich die funktionalen oder qualitativen Anforderungen an Systeme oder Abläufe fest. Andere Standards beinhalten exakte technische Spezifikationen. Ein Standard besteht dabei meist aus normativen und informativen Bestandteilen. Die informativen Teile, wie Beispiele oder Empfehlungen, helfen dem Anwender des Standards, die beschriebenen Konzepte zu verstehen. Die normativen Teile eines Standards haben verbindlichen Charakter und müssen eingehalten werden, wenn z. B. eine Zertifizierung angestrebt wird. Oft werden auch Testszenarien festgeschrieben, die erfüllt werden müssen, wenn ein bestimmter Standard eingehalten werden soll.

Organisationen, die Standards verabschieden, existieren auf unterschiedlichen Ebenen. Auf der internationalen Ebene sind vor allem die Internationale Organisation für Normung (ISO), die Internationale Elektrotechnische Kommission (IEC) und die Internationale Fernmeldeunion (ITU) relevant, auf der europäischen Ebene hat das Europäische Komitee für Normung (CEN), das Europäische Komitee für elektrotechnische Normung (CENELEC) und das Europäische Institut für Telekommunikationsnormen (ETSI) entsprechende Aufgaben. Auf nationaler Ebene existieren mit dem Deutschen Institut für Normung (DIN) und insbesondere der Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE (DKE) Organisationen, die wichtige Standardisierungsaufgaben mit Bezug zur Energieinformatik übernehmen.

Alle diese Gremien organisieren ihre Arbeiten in weiteren Untergruppen, z. B. Arbeitsgruppen bzw. Komitees. Die IEC betreibt aktuell mehr als 200 solcher technischen Komitees (Technical Committee (TC)). Auf nationaler Ebene existieren oft sog. Spiegelkomitees zu den internationalen Arbeitsgruppen. Diese überführen internationale Normen in nationale Regelungen, koordinieren die Normung mit nationalen Stellen oder setzen eigene Impulse bei der Normung. In diesen Prozess sind meist viele weitere Organisationen involviert. In Deutschland spielt z. B. der BDEW oder der Deutscher Verein des Gas- und Wasserfaches (DVGW) eine wichtige Rolle. Auch staatliche Organisationen sind am Normungsprozess beteiligt. Als Beispiel sei das Bundesamt für Sicherheit in der Informationstechnik (BSI) genannt, welches mithilfe von technischen Richtlinien z. B. die Anforderungen an intelligente Messsysteme in Deutschland normiert (siehe Abschn. 5.3).

Die Themen der Energieinformatik sind äußerst vielfältig. Entsprechend viele Gremien und Organisationen sind auch an der Standardisierung beteiligt. Die Abhängigkeiten zwischen den Organisationen sind dabei leider schwer zu überschauen. Abb. 2.28 zeigt den Versuch, die Abhängigkeiten zwischen den Normungsgremien und Organisationen darzustellen. Der Fokus liegt dabei auf dem Thema Smart Grid.

Der aktuelle Stand der Standardisierung und Normung wird für Deutschland durch DIN und DKE in Form mehrerer **Normungsroadmaps** dargestellt. Die Themen der Roadmaps sind dabei keinesfalls überschneidungsfrei. Es gibt teilweise große Überlappungen, z. B. bei den Themen E-Energy/Smart Grid, Smarthome/Smart Building und IT-Sicherheit. Für die Energieinformatik sind folgende Dokumente besonders interessant:

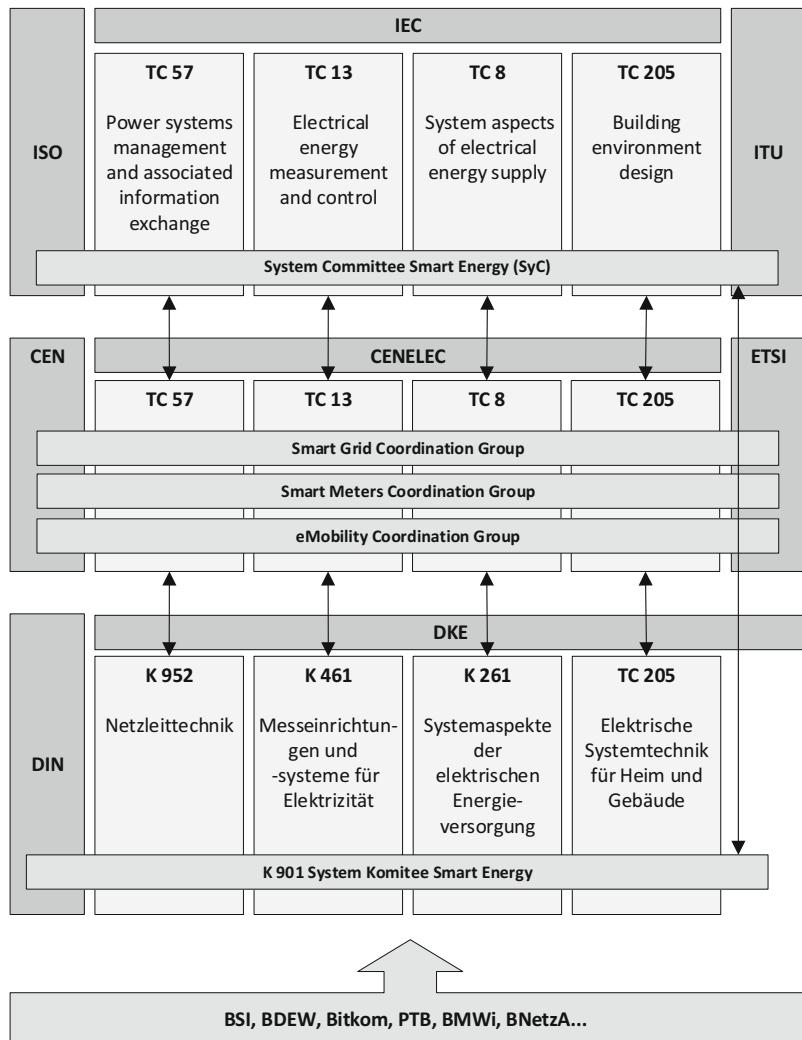


Abb. 2.28 Normungsgremien und Organisationen im Kontext des Smart Grid

- Deutsche Normungsroadmap E-Energy/Smart Grids 2.0 (siehe [41]),
- Deutsche Normungsroadmap Smart Home + Building 2.0 (siehe [29]),
- Deutsche Normungsroadmap Energieeffizienz (siehe [45]),
- Deutsche Normungsroadmap Industrie 4.0 (siehe [46]),
- Deutsche Normungsroadmap IT-Sicherheit (siehe [36]).

Die Deutsche Normungsroadmap E-Energy/Smart Grid 2.0 stellt den Normungsstand im Bereich der Digitalisierung des Stromnetzes dar. Die aktuelle Version stammt aus dem Jahr 2017.

Ein Thema der Roadmap ist die Darstellung des Normungsprozesses in diesem Bereich. Dieser ist zyklisch aufgebaut und beginnt mit der Sammlung von sog. Anwendungsfällen (*Use Cases*), aus denen schrittweise die Anforderungen für Standards abgeleitet werden können. Eine wichtige Rolle in diesem Prozess spielt das sog. Smart Grid Architecture Model (SGAM). Das SGAM hat das Ziel, die unterschiedlichen Aspekte eines Smart Grid technologienutral zu beschreiben. Es stellt alle Akteure, Komponenten und Abstraktionsebenen des Smart Grid dar, sodass Anwendungsfälle gruppiert und darauf aufbauend Standards entwickelt werden können. Das SGAM wurde durch die **Smart Grid Coordination Group** des CENELEC im Rahmen des M/490 Smart Grid Standardization Mandate ausgearbeitet (siehe [21]). Es bildet eine wichtige Grundlage, um Systemarchitekturen für das Smart Grid zu entwickeln (siehe [18]).

Das SGAM teilt die Konzepte eines Smart Grid in drei Achsen ein. Auf der X-Achse finden sich die sog. Anwendungsdomänen, die im wesentlichen den Akteuren des physikalisch-technischen Energieflusses entsprechen. Auf der Y-Achse finden sich die Integrationszonen, die einen direkten Bezug zur Automatisierungspyramide besitzen. Die dritte Achse bezeichnet die sog. Interoperabilitätsebenen. Hier werden Anforderungen an die Interoperabilität der Systeme auf insgesamt fünf Ebenen hierarchisiert. Auf der Komponentenebene wird die physikalische Verteilung der beteiligten Komponenten dargestellt. Die Kommunikationsebene beinhaltet Protokolle und Mechanismen für den Informationsaustausch. Die Informationsebene definiert die Semantik von Daten anhand von Informationsmodellen. Auf der Funktionsebene werden Funktionen und Dienste auf den Informationsmodellen beschrieben. In der Geschäftsebene werden Geschäftsmodelle und ökonomische und regulatorische Aspekte dargestellt. Auf allen Ebenen existieren Standards, um die nahtlose Integration (*Seamless Integration*) von Komponenten zu ermöglichen. Dieses Lehrbuch fokussiert sich dabei hauptsächlich auf die Ausgestaltung der Informations- und der Kommunikationsebene. Dort finden sich Standards, wie das CIM bzw. IEC 61850, die als Kernstandards des Smart Grid angesehen werden. Beide Standards werden in den nachfolgenden Kapiteln behandelt (siehe Abschn. 3.1 und 5.1). Abb. 2.29 zeigt das SGAM.

Durch Themen wie Energie- und Lastmanagement existieren zwischen Smart Grid und der Automation von Gebäuden starke Überschneidungen. Der Stand der Normung im Bereich der Digitalisierung von Wohn- und Geschäftsgebäuden wird aber in einer eigenen Normungsroadmap dargestellt. Die aktuelle Version des Dokuments stammt aus dem Jahr 2015.

Auch in diesem Bereich wurde ein Architekturmodell entworfen, welches sich sehr eng am SGAM orientiert. Das sog. Home Building Architecture Model (HBAM) dient dazu, Anwendungsfälle aus dem Bereich Smarthome und Smart Building zu gruppieren. Das Modell basiert auf denselben Interoperabilitätsebenen und ähnlichen Integrationszonen wie das SGAM. Bei den Anwendungsdomänen wird zwischen Kommunikation und Unterhaltung, Betriebsschutz und Sicherheit, Komfort, Gesundheitsunterstützung und Energiemanagement unterschieden. Auf der Kommunikations- und Informationsebene werden dabei Technologien wie KNX, LonWorks oder BACnet eingesetzt. Diese bilden

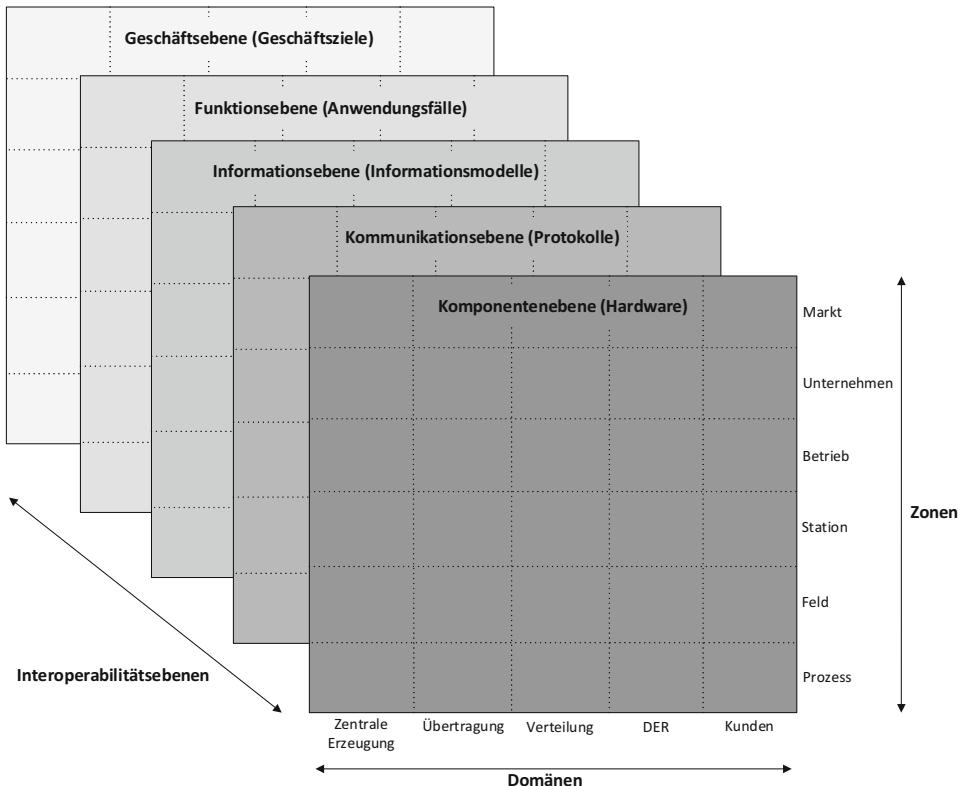


Abb. 2.29 Das Smart Grid Architecture Model

die Voraussetzung, dass Informationen mit Sensoren und Aktoren ausgetauscht werden können. Diese Standards werden im Abschnitt Gebäudeautomation näher betrachtet (siehe Abschn. 5.2). Im Bereich Energiemanagement wird auf Standards wie DLMS und COSEM verwiesen, die im Kontext des Smart-Meters eingesetzt werden. Diese Standards werden im Abschnitt Smart Meter dargestellt (siehe Abschn. 5.3).

Die Erhöhung der Energieeffizienz wird neben dem Einsatz von regenerativen Energieformen als zweite Säule der Energiewende gesehen. Ein wichtiger Beitrag dazu liefert das **Energiemanagement**, bei dem energietechnische Erzeuger und Verbraucher so ausgelegt und betrieben werden, dass Ressourcen und somit das Klima geschont und Kosten eingespart werden. Ohne belastbare Daten, die den Energieverbrauch in Produktionsanlagen oder Gebäuden detailliert aufschlüsseln, kann aber i. d. R. kein wirksames Energiemanagement betrieben werden. In der Normungsroadmap Energieeffizienz finden sich dazu einige Aussagen. Das aktuelle Dokument ist vom März 2018.

Bezüglich der Ausgestaltung des Energiemanagements wird darin auf Normen verwiesen, die Messkonzepte und Messtechnik beschreiben (siehe [47]). Auf dieser Basis können in Energiedatenmanagementsystemen Energiekennzahlen, die *Energy Performance*-

ce Indicators, abgeleitet werden. Die Norm gibt dazu allerdings wenig Hinweise, wie die Datenverarbeitung für das Energiemanagement umgesetzt werden soll. Da Energiedaten meist auf Zeitreihen basieren, wird der Umgang mit solchen Daten in Abschn. 3.3 dargestellt.

Im Kontext der Automatisierung von Anlagen zur Erzeugung und Verteilung von Energie bestehen gewisse Überschneidungen zum Thema Industrie 4.0. Der entsprechende Normungsstand wird in der Normungsroadmap Industrie 4.0 des DKE dargestellt. Die aktuelle Version ist vom März 2018.

Von entscheidender Bedeutung ist dabei die hinreichende Abbildung der physischen Welt in die Informationswelt (siehe [46, S. 33]). Dies wird auch als digitaler Schatten bezeichnet. Auch in diesem Bereich wurde mit dem Reference Architecture Model for Industry 4.0 (RAMI 4.0) ein Ordnungsrahmen vorgelegt, welcher Komponenten, funktionale und hierarchische Ebenen strukturiert. Das RAMI 4.0 ist im Aufbau sehr ähnlich zum SGAM und HBAM.

In der Kommunikations- und Informationsebene wird OPC UA als Standard vorgeschlagen. Dieser realisiert ein domänenunabhängiges Schnittstellen- und Informationsmodell, mit dessen Hilfe die Kommunikation zwischen den Ebenen umgesetzt werden kann. Dieser Standard wird in Abschn. 5.2 dargestellt. Die Normungsroadmap Industrie 4.0 verweist mit dem DIN 77005-1 zudem auf einen Standard zur Abbildung von Lebenszyklusinformationen. Dieser kann im Energieumfeld für das Asset-Management von Anlagen genutzt werden. Hier existieren Überschneidungen zum Informationsmodell CIM, welches in Abschn. 3.1 vorgestellt wird.

Dass IT-Sicherheit auch für die Energiewirtschaft eine wichtige Rolle spielt, schlägt sich mittlerweile in einer ganzen Reihe von Standards und Regularien nieder. Der Normungsstand bzgl. der IT-Sicherheit wird ebenfalls in einer eigenen Normungsroadmap des DKE dargestellt. Die aktuelle Version ist vom Juni 2017.

Im Kontext des Smart Grid wurden im Rahmen des M/490-Mandats durch die Smart Grid Security Group (SGIS) verschiedene Standards analysiert und erweitert. Kernbestandteil der Arbeit des SGIS sind die Analyse und Integration existierender Sicherheitsstandards mit dem SGAM. Für jede Domäne, Zone und Ebene werden Abbildungen auf existierende Standards aufgezeigt und evtl. vorhandene Lücken benannt.

Dabei werden zwei Klassen von Sicherheitsstandards unterschieden. Die *Requirements Standards* beschreiben tendenziell eher, welche Assets zu schützen sind. Hier wird insbesondere der IEC 27019 gesehen, der auch im deutschen Rechtsrahmen für die Energiewirtschaft eine wichtige Rolle spielt. Im Gegensatz dazu definieren die *Solution Standards* hingegen konkrete Maßnahmen zum Schutz von Komponenten. Hier werden vor allem der IEC 62351 bzgl. der Kommunikationssicherheit bei der Netzautomation und der IEC 62056-5-3 für die Absicherung von DLMS/COSEM bei Smart-Metern genannt (siehe Kap. 6).

2.8 Zusammenfassung

In diesem Kapitel wurden für beide Zielgruppen dieses Buches einige wichtige Grundlagen vermittelt, die für das weitere Verständnis dieses Lehrbuchs wichtig sind.

Für InformatikerInnen wurde zunächst die Wertschöpfungskette der Energiewirtschaft vorgestellt. Dabei wurde gezeigt, dass unsere moderne Gesellschaft von der verlässlichen Versorgung mit leitungsgebundener Energie abhängig ist. Diese Versorgung sicherzustellen, ist ein komplexes Unterfangen und äußert sich in den Abläufen des technisch-physikalischen und des kaufmännischen Energieflusses. Die Gesamtaufgabe einer sicheren und kostenoptimalen Energieversorgung ist auf mehrere unterschiedliche Rollen, wie z. B. Übertragungsnetzbetreiber oder Erzeuger aufgeteilt. Auf allen Ebenen fallen dabei komplexe Aufgaben an, z. B. die Planung und der Betrieb von Netzen, der kostenoptimale Einsatz von Kraftwerken, die Bewirtschaftung eines Energieportfolios oder der börsliche Handel mit Energieprodukten. Dabei besteht ein hoher Kommunikations- und Abstimmungsbedarf, dem heute lediglich mithilfe moderner Informations- und Kommunikationstechnologie begegnet werden kann.

Es wurde gezeigt, dass die Energiewirtschaft aktuell enormen Veränderungsprozessen unterworfen ist. Ein großer Treiber von Veränderungen ist die Energiewende, die das Ziel hat, den Ausstoß von Klimagassen durch fossile Energieträger zugunsten von regenerativen Quellen zurückzudrängen. Die Energiewende hat enormen Einfluss auf den physikalisch-technischen und den kaufmännischen Energiefluss. Großkraftwerke werden zukünftig mit hoher Wahrscheinlichkeit weiter durch dezentrale und regenerative Erzeuger ersetzt. Neben der Energiewende tragen vor allem die zunehmenden Möglichkeiten der Digitalisierung zu einem großen Veränderungsdruck in der Energiewirtschaft bei. Sowohl Speicherplatz als auch Rechenkraft sind heute quasi unbegrenzt verfügbar. Durch Cloud-Systeme, Blockchain und Big Data ergeben sich auch in der Energiewirtschaft gänzlich neue Möglichkeiten.

Drei weitere Abschnitte richteten sich hauptsächlich an EnergiewirtschaftlerInnen. Dabei wurden einige Begriffe, wie z. B. Daten und Informationsmodell, präzisiert, die für das Verständnis weiterer Abschnitte relevant sind. Auf Basis des ISO/OSI-Referenzmodells wurden die Aufgaben der digitalen Datenübertragung dargestellt und erläutert. Anhand diverser Standards, wie z. B. RS-232, Ethernet und dem Internetprotokollstapel, wurde zudem gezeigt, wie diese Aufgaben anhand konkreter Technologien und Protokolle umgesetzt werden können.

Energietechnische Anlagen können heute nur unter dem Einsatz von umfangreichen Automatisierungssystemen verlässlich und sicher betrieben werden. Es wurde gezeigt, dass solche Automatisierungssysteme hierarchisch aufgebaut sind und dass jede Ebene eigene Anforderungen an die digitale Kommunikation stellt. Wegen der besonderen Kommunikationsanforderungen solcher Systeme werden auf der Feld- und Automationsebene meist Feldbusssysteme oder Industrial Ethernet eingesetzt. Mit Modbus, Profibus und Profinet wurden entsprechende Technologien vorgestellt.

Damit Systeme unterschiedlicher Hersteller erfolgreich integriert werden können, sind Standards von essenzieller Bedeutung. Eine standardisierte Systemarchitektur beschreibt dabei zunächst auf hohem Abstraktionsniveau, welche Aufgaben überhaupt anfallen und welche Komponenten auf welchen Ebenen daher miteinander zu integrieren sind. In der Energiewirtschaft existiert hier mit dem SGAM eine entsprechende Architektur, welche die Aufgaben einer verlässlichen Energieversorgung anhand von Domänen, Zonen und Interoperabilitätsebenen strukturiert.

Insbesondere auf der Informations-, der Kommunikations- und der funktionalen Ebene des SGAM werden aber weitere konkrete Standards benötigt, die Informationsmodelle, Kommunikationsprotokolle und Schnittstellen ausgestalten. Es wurde gezeigt, dass hier insbesondere die Arbeitsgruppen der IEC in Erscheinung treten und relevante Standards im Kontext der Energieversorgung definieren und verabschieden.

2.9 Kontrollfragen

- Wie hat sich der Endenergieverbrauch in Deutschland seit 1990 insgesamt entwickelt? In welchen Sektoren ist der Energieverbrauch gesunken, in welchen gestiegen?
- Welche gemeinsamen Eigenschaften haben elektrische Energie und Erdgas als Energieträger?
- Welchen Unterschied gibt es zwischen elektrischer Energie und Erdgas im Bezug zur Speicherbarkeit? Welche Konsequenzen hat das?
- Welche Bereiche der leitungsgebundenen Energieversorgung sind reguliert?
- Nach welchen Kriterien variiert die Nachfrage nach elektrischer Energie?
- Besitzen Übertragungsnetzbetreiber eigene Erzeugungseinheiten, um damit Regelergie zu erbringen?
- Was ist das Kerngeschäft eines Stadtwerks?
- Auf welchen Marktplätzen können Energieprodukte gehandelt werden?
- Welchen Änderungen ist die Energiewirtschaft derzeit unterworfen?
- Was bedeutet der Begriff Sektorenkopplung?
- Warum ist es sinnvoll, unser Energiesystem auf die Nutzung von regenerativen Energiequellen umzubauen?
- Wie hoch ist der Anteil erneuerbarer Energien an der Nettostromerzeugung aktuell?
- Aus welchen Energiequellen wurde in Deutschland der Ausstieg beschlossen bzw. ist geplant?
- Was wird mit dem Begriff Internet of Things verbunden?
- Was ist die Aufgabe eines Informationsmodells?
- Was ist der Unterschied zwischen einem Daten- und einem Informationsmodell?
- Aus welchen Elementen besteht ein Klassendiagramm der UML?
- Was ist der Unterschied zwischen einer Assoziation und einer Vererbungsbeziehung?
- Was wird durch den Begriff Multiplizität beschrieben?
- Wofür steht die Abkürzung SQL?

- Wie viele Schichten hat das ISO/OSI-Referenzmodell der digitalen Kommunikation?
- Welche Bedeutung kommt der sog. Zugriffskontrolle in der Sicherungsschicht zu?
- Erläutern Sie den Unterschied zwischen den Verfahren CSMA/CD und CSMA/CA.
- Was ist eine Netztopologie? Nennen Sie Beispiele für Netztopologien. Was wird benötigt, um unterschiedliche Topologien überhaupt realisieren zu können?
- Wozu dient ein sog. UART?
- Welche Aufgaben fallen auf der Anwendungsschicht an? Nennen Sie Beispiele für Standards, die auf dieser Ebene zu finden sind.
- Zwischen welchen Hierarchieebenen wird in der Automationstechnik unterschieden?
- Was ist ein SCADA-System?
- Warum werden in der Automationstechnik oft Feldbusse eingesetzt und nicht ein Büronetz, wie z. B. Ethernet?
- Auf Basis welcher Elemente werden Daten bei Modbus repräsentiert?
- Kann Modbus ereignisorientierte Kommunikation umsetzen?
- Um welche Konzepte erweitert Profinet IO den Ethernetstandard?
- Nennen Sie Organisationen auf internationaler, regionaler und nationaler Ebene, die für die Energieinformatik relevante Standards verabschieden.
- Was ist der Unterschied zwischen einem informativen und einem normativen Standard?
- Aus welchen Ebenen besteht das sog. SGAM?

Literatur

1. N. Craig Meyers und Gary Chastek. *The Use of ASN.1 and XDR for Data Representation of Real-Time Distributed Systems*. Okt. 1993. URL: https://resources.sei.cmu.edu/asset_files/TechnicalReport/1993_005_001_16178.pdf (besucht am 25.06.2019).
2. ISO/IEC 7498-1:1994-11 – *Informationstechnik – Kommunikation Offener Systeme – Basis-Referenzmodell*. ISO - International Organization for Standardization, Nov. 1994.
3. Richard Y. Wang und Diane M. Strong. „Beyond Accuracy: What Data Quality Means to Data Consumers“. In: *Journal of Management Information Systems* 12.4 (1996), S. 5–33. issn: 0742-1222. <https://doi.org/10.1080/07421222.1996.11518099>.
4. A. Daneels und W. Salter. „What is SCADA?“ In: *Proceedings of 7th Biennial International Conference on Accelerator and Large Experimental Physics Control Systems*. Trieste, 1999. URL: <http://cds.cern.ch/record/532624/files/mc1i01.pdf> (besucht am 10.07.2019).
5. *Kurzbeschreibung von PROFIBUS in deutscher Sprache zusammengestellt für DKE*. Unter Mitarb. von PROFIBUS Nutzerorganisation e.V. Okt. 2002. URL: <https://www.dke.de/resource/blob/1614942/0cbe92e5bfed4036cb6d62bfe9c27d98/typ3profibus-data.pdf> (besucht am 25.06.2019).
6. David Bailey und Edwin Wright. *Practical SCADA for industry*. Amsterdam: Elsevier, 2003. 288 S. ISBN: 978-0-7506-5805-8.
7. A. Pras und J. Schoenwaelder. *On the Difference between Information Models and Data Models*. 2003. URL: <https://tools.ietf.org/html/rfc3444#page4> (besucht am 19.02.2020).
8. *Warum industrielles Ethernet?* 2004. URL: http://www.networktraining.de/downloads/Warum_industrielles_Ethernet.pdf (besucht am 25.06.2019).
9. Tatjana Kostic, Otto Preiss und Christian Frei. „Understanding and using the IEC 61850: a case for meta-modelling“. In: *Computer Standards & Interfaces* 27 (Juni 2005), S. 679–695.

- issn: 09205489. <https://doi.org/10.1016/j.csi.2004.09.008>. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0920548904001023> (besucht am 30.06.2019).
10. Peter Mertens. *Grundzüge der Wirtschaftsinformatik*. 9., überarbeitete Auflage. Springer-Lehrbuch. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 2005. ISBN: 3-540-23411-X. URL: <https://doi.org/10.1007/b138341>.
11. Bericht der Bundesnetzagentur über die Systemstörung im deutschen und europäischen Verbundsystem am 4. November 2006. Bonn: Bundesnetzagentur (BNetZA), Feb. 2007. URL: https://www.bundesnetzagentur.de/Shared.Docs/Downloads/DE/Sachgebiete/Energie/Unternehmen_Institutionen/Versorgungssicherheit/Berichte_Fallanalysen/Bericht_9.pdf (besucht am 24.06.2020).
12. Felix Naumann. *Datenqualität*. 2007. URL: <https://gi.de/informatiklexikon/datenqualitaet/> (besucht am 25.06.2019).
13. Gerd Küveler und Dietrich Schwoch. *Informatik für Ingenieure und Naturwissenschaftler* 2. 5., vollständig überarbeitete und aktualisierte Auflage. Viewegs Fachbücher der Technik. Wiesbaden: Friedr. Vieweg & Sohn Verlag | GHV Fachverlage GmbH Wiesbaden, 2007. ISBN: 978-3-8348-9191-4. URL: <https://doi.org/10.1007/978-3-8348-919-14>.
14. IEC 61588:2009 – *Precision clock synchronization protocol for networked measurement and control systems*. International Electrotechnical Commission (IEC), Feb. 2009. URL: <https://www.vde-verlag.de/iec-normen/216317/iec-61588-2009.html> (besucht am 20.02.2020).
15. S. Mohagheghi, J. Stoupis und Z. Wang. „Communication protocols and networks for power systems-current status and future trends“. In: *2009 IEEE/PES Power Systems Conference and Exposition*. 2009 IEEE/PES Power Systems Conference and Exposition. März 2009, S. 1–9. <https://doi.org/10.1109/PSCE.2009.4840174>.
16. Niels Ehlers. „Strommarktdesign angesichts des Ausbaus fluktuierender Stromerzeugung“. Berlin: Technische Universität Berlin, 2011. 198 S. URL: https://www.ensys.tu-berlin.de/fileadmin/fg8/Downloads/Publications/Dissertation_Ehlers_2011.pdf (besucht am 25.06.2019).
17. Mark Lochhaas. *Understanding Ethernet Speed and Determinism*. 2011. URL: <https://www.automationworld.com/article/technologies/networking-connectivity/ethernet-tcp-ip/understanding-ethernet-speed-and> (besucht am 25.06.2019).
18. Hans-Jürgen Appelrath u. a., Hrsg. *IT-Architekturentwicklung im Smart Grid*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. ISBN: 978-3-642-29207-1. URL: <https://doi.org/10.1007/978-3-642-29208-8>.
19. *Ausbau der Verteilnetze muss nächste Priorität sein*. Berlin: Bundesverband der Energie- und Wasserwirtschaft e.V. (BDEW), 11. Dez. 2012. URL: https://www.vewsaar.de/fileadmin/dokumente/Aktuell/Presse/BDEW_Verteilnetzstudie_dena.pdf (besucht am 25. 06. 2019).
20. Michael Maurer. „Anforderungen an ein modernes Prozessleitsystem (PLS)“. In: *Energie aus Abfall*. Hrsg. von Karl J. Thomé-Kozmiensky und Michael Beckmann. Bd. 9. Neuruppin: TK-Verlag, 2012, S. 271–292. ISBN: 978-3-935317-78-8. URL: http://www.vivis.de/phocadownload/Download/2012_eaa/2012_EaA_271_290_Maurer.pdf.
21. *Smart Grid Reference Architecture*. CEN-CENELEC-ETSI Smart Grid Coordination Group, Nov. 2012. URL: https://ec.europa.eu/energy/sites/ener/files/documents/xpert_group1_reference_architecture.pdf (besucht am 11.06.2020).
22. *Fifth Assessment Report*. 2013. URL: <https://www.ipcc.ch/assessment-report/ar5/> (besucht am 25.06.2019).
23. Lou Frenzel. *What's The Difference Between The RS-232 And RS-485 Serial Interfaces?* 6. Apr. 2013. URL: <https://www.electronicdesign.com/what-s-difference-between/what-s-difference-between-rs-232-and-rs-485-serial-interfaces> (besucht am 23.09.2019).
24. Gerd Balzer und Christian Schorn. *Asset Management für Infrastrukturanlagen*. 2. Aufl. VDI-Buch. Berlin: Springer Vieweg, 2014. 373 S. ISBN: 978-3-642-54939-7.

25. Jens Büchner u. a. *Moderne Verteilernetze für Deutschland – Studie im Auftrag des Bundesministeriums für Wirtschaft und Energie (BMWi)*. Forschungsprojekt Nr. 44/12. 12. Sep. 2014. URL: <https://www.bmwi.de/Redaktion/DE/Publikationen/Studien/verteilernetzstudie.html> (besucht am 23.06.2020).
26. Alberto Marvão. *Modbus communication protocol*. 2014. URL: <https://africa.cummins.com/sites/za/files/11%20November%202018%20-%20Modbus%20communication%20protocol.pdf> (besucht am 25.06.2019).
27. Antwort der Bundesregierung auf die kleine Anfrage der Abgeordneten Bärbel Höhn, Oliver Krischer, Annalena Baerbock, weiterer Abgeordneter und der Fraktion BÜNDNIS 90/DIE GRÜNEN. Berlin: Deutscher Bundestag, 24. Juni 2015. URL: <http://dip21.bundestag.de/dip21/btd/18/053/1805323.pdf> (besucht am 18.02.2020).
28. Signe Annette Bøgh, Hrsg. *A World Built on Standards – A Textbook for Higher Education*. Nordhavn: Danish Standards Foundation, 2015. ISBN: 978-87-7310-963-2. URL: <https://www.bsigroup.com/globalassets/localfiles/en-gb/standards/bsol-academic/A%20world%20built%20on%20standards.pdf>.
29. *Deutsche Normungsroadmap Smart Home + Building Version 2.0*. Frankfurt: DKE – Deutsche Kommission Elektrotechnik Elektronik Informationstechnik, Aug. 2015. URL: <https://www.dke.de/resource/blob/778214/6ec4d037024b61a63d14544d181c638a/deutsche-normungs-roadmap-smart-home---building--version-2-0-data.pdf> (besucht am 09.06.2020).
30. Max Felser. *PROFIBUS Handbuch*. 1. Aufl. Berlin: epubli GmbH, 2015. Online-Ressource. ISBN: 978-3-7375-5470-1.
31. Dietmar Brück. *Profibus DP*. 18. Apr. 2016. URL: https://www.htwsaar.de/ingwi/fakultaet/personen/profile/jannick-morsch/unterlagen-labore/versuche-automatisierungslabor/asi-p-dp-versuch/Praktikum_PROFIBUS_DP.pdf (besucht am 25.06.2019).
32. Hermann Merz, Thomas Hansemann und Christof Hübner. *Gebäudeautomation* 3., aktualisierte Auflage. München: Carl Hanser Fachbuchverlag, 2016. 1310 S. ISBN: 978-3-446-44662-5. URL: <https://doi.org/10.3139/9783446447721>.
33. Norbert Schwieters u. a. *Deutschlands Energieversorger werden digital*. 2016. URL: <https://www.pwc.at/de/publikationen/pwc-studie-digitalisierung-energiewirtschaft-01-2016-screen.pdf> (besucht am 25.06.2019).
34. Kaile Zhou, Chao Fu und Shanlin Yang. „Big data driven smart energy management: From big data to big insights“. In: *Renewable and Sustainable Energy Reviews* 56 (1. Apr. 2016), S. 215–225. ISSN: 1364-0321. <https://doi.org/10.1016/j.rser.2015.11.050>. URL: <http://www.sciencedirect.com/science/article/pii/S1364032115013179> (besucht am 23.06.2020).
35. *Der Mensch in Watt*. 2017. URL: <https://blog.wienenergie.at/2017/03/22/der-mensch-in-watt/> (besucht am 24.01.2019).
36. *Deutsche Normungsroadmap IT-Sicherheit – Version 3.0*. DKE - Deutsche Kommission Elektrotechnik Elektronik Informationstechnik, 21. Juni 2017. URL: <https://www.dke.de/resource/blob/778258/44e2d336c2702f285ba669ee5cd47b10/deutsche-normungs-roadmap-it-sicherheit-version-3-0-data.pdf> (besucht am 04.06.2020).
37. „Energiewende und Dezentralität“. In: (2017). URL: https://www.agora.energiewende.de/fileadmin2/Projekte/2016/Dezentralitaet/Agora_Dezentralitaet_WEB.pdf
38. Simon Göß. *Dreaming of the copper plate: How physics and power trading diverge* 26. Jan. 2017. URL: <https://blog.energybrainpool.com/en/physical-flows-vs-commercial-schedules-in-cross-border-power-trading/> (besucht am 09.06.2020).
39. Byunghun Lee u. a. „Model transformation between OPC UA and UML“. In: *Computer Standards & Interfaces* 50 (1. Feb. 2017), S. 236–250. issn: 0920-5489. <https://doi.org/10.1016/j.csi.2016.09.004>. URL:<http://www.sciencedirect.com/science/article/pii/S0920548916300769> (besucht am 07.04.2020).

40. Tobias Meudt, Malte Pohl und Joachim Metternich. *Die Automatisierungspyramide – Ein Literaturüberblick*. 7. Juni 2017. URL: <https://tuprints.ulb.tudarmstadt.de/6298/> (besucht am 25.06.2019).
41. *Normungsroadmap E-Energy / Smart Grids 2.0 – Status, Trends und Perspektiven der Smart Grid-Normierung*. Frankfurt: Verband der Elektrotechnik Elektronik Informationstechnik e.V. (VDE), 23. Jan. 2017. URL: <https://www.dke.de/de/arbeitsfelder/energy/deutsche-normungsroadmap-e-energy-smart-grids-2-0> (besucht am 09.06.2020).
42. Steffen Schattner. *Asset Management von Versorgungsnetzen*. 2017. URL: <https://www.pwc.de/de/energiewirtschaft/pwc-studie-assetmanagement-versorgungsnetze.pdf> (besucht am 25. 06. 2019).
43. Thomas Teodorowicz. „Comparison of SCADA protocols and implementation of IEC 104 and MQTT in MOSAIK“. Bachelorarbeit. Münster: University of Münster, 15. Aug. 2017. URL: https://www.uni-muenster.de/imperia/md/content/informatik/agremke/comparison_of_scada_protocols_and_implementation_of_iec_104_and_mqtt_in_mosaik.pdf (besucht am 07. 08. 2019).
44. Bruno Burger. *Stromerzeugung in Deutschland im ersten Halbjahr 2018*. 2018. URL: https://www.ise.fraunhofer.de/content/dam/ise/de/documents/publications/studies/daten-zu-erneuerbaren-energien/ISE_Stromerzeugung_2018_Halbjahr.pdf (besucht am 25. 06. 2019).
45. *Deutsche Normungsroadmap Elektrische Energieeffizienz 2.0*. Verband der Elektrotechnik Elektronik Informationstechnik e.V. (VDE), 16. März 2018. URL: <https://www.dke.de/resource/blob/950182/9d6b9b67cf0c967fb1e7d9e417c009e0/hr-energieeffizienz-v2-de-data.pdf> (besucht am 11.06.2020).
46. *Deutsche Normungsroadmap Industrie 4.0 – Version 3.0*. Deutsches Institut für Normung (DIN), März 2018. URL: <https://www.din.de/blob/95954/97b71e1907b0176494b67d8d6392c54/aktualisierte-roadmap-i40-data.pdf> (besucht am 11.06.2020).
47. DIN EN ISO 500001: *Energiemanagementsysteme – Anforderungen mit Anleitung zur Anwendung*. Deutsches Institut für Normung (DIN), Dez. 2018. <https://doi.org/10.31030/2852891>. URL: <https://www.beuth.de/de/-/289820323> (besucht am 12.06.2020).
48. *Global Warming of 1.5 °C* 2018. URL: <https://www.ipcc.ch/sr15/> (besucht am 25.06.2019).
49. Christoph Kost u. a. *Stromgestehungskosten erneuerbare Energien*. FraunhoferInstitut für Solare Energiesysteme ISE, März 2018. URL: https://www.ise.fraunhofer.de/content/dam/ise/de/documents/publications/studies/DE2018_ISE_Studie_Stromgestehungskosten_Erneuerbare_Energien.pdf
50. Elie-Lukas Limbacher und Philip Richard. *Schnittstellen und Standards für die Digitalisierung der Energiewende – Übersicht, Status Quo und Handlungsbedarf*. Deutsche Energie-Agentur (dena), Jan. 2018. URL: https://www.dena.de/fileadmin/dena/Dokumente/Pdf/9240_Schnittstellen_und_Standards_fuer_die_Digitalisierung_der_Energiewende.pdf.
51. Malte Stomberg, Alexander Stuckenholz und Torsten Cziesla. „SolChain“. In: *BWK – Das Energiefachmagazin* 11 (2018), S. 48–51.
52. *Uhren gehen nach: Streit am Balkan soll Ursache sein*. 2018. URL: <https://www.zeit.de/news/2018-03/07/uhren-gehen-nach-streit-am-balkan-soll-ursache-sein-180307-99-376293> (besucht am 25.06.2019).
53. Michael Utecht und Tobias Zierau. *SAP für Energieversorger*. 1. Auflage. SAP PRESS. Bonn: Rheinwerk Verlag, 2018. 975 S. ISBN: 978-3-8362-6260-6.
54. Andreas Wilkens. *Power to Gas: Netzbetreiber planen 100-MW-Anlage*. Okt. 2018. URL: <https://www.heise.de/newsticker/meldung/Power-to-Gas-Netzbetreiber-planen-100-MW-Anlage-4191465.html> (besucht am 25.06.2019).
55. *Abschlussbericht der Kommission für Wachstum, Strukturwandel und Beschäftigung*. Berlin: Bundesministerium für Wirtschaft und Energie (BMWi), Jan. 2019. URL: <https://www.bmwi.de>

- de/Redaktion/DE/Downloads/A/abschlussbericht-kommission-wachstum-strukturwandel-und-beschaeftigung.pdf (besucht am 13.05.2020).
56. Oliver Bendel. *Digitalisierung*. 2019. URL: <https://wirtschaftslexikon.gabler.de/definition/digitalisierung-54195/version-277247> (besucht am 25.06.2019).
57. *Bilanzkreisvertrag Strom*. Bundesnetzagentur (BNetzA), 12. Apr. 2019. URL: https://www.bundesnetzagentur.de/DE/Service-Funktionen/Beschlusskammern/BK06/BK6_83_Zug_Mess/838_bilanzkreisvertrag/BK6-18-061_Standardbilanzkreisvertrag_01_08_2020.pdf (besucht am 23.06.2020).
58. Lion Hirth u. a. *Kosten- oder Marktbasiert? Zukünftige Redispatch-Beschaffung in Deutschland*. 7. Okt. 2019. URL: <https://www.bmwi.de/Redaktion/DE/Publikationen/Studien/zukuenftige-redispatch-beschaffung-in-deutschland.pdf> (besucht am 24.06.2020).
59. Daniel Mohr. *Deutsche Börse wird noch mehr zum Tech-Konzern*. 2019. URL: <https://www.faz.net/aktuell/finanzen/finanzmarkt/deutsche-boerse-wird-noch-mehr-zum-tech-konzern-16041643.html> (besucht am 25.06.2019).
60. Uwe Pagel. *MaKoChain: Lieferantenwechsel in Sekunden*. 2019. URL: <https://edna-bundesverband.de/news/makochain-lieferantenwechsel-in-sekunden/> (besucht am 25.06.2019).
61. Ulf Meinke. *Amprion, OGE, Thyssenkrupp und RWE – Große Wasserstoff-Pläne*. 15. Juni 2020. URL: <https://www.waz.de/wirtschaft/amprion-oge-thyssenkrupp-und-rwe-grosse-wasserstoff-plaene-id229304936.html> (besucht am 23.06.2020).
62. *MODELIO – The open source extensible modeling environment*. 2020. URL: <https://www.modelio.org/> (besucht am 19.02.2020).
63. *Strompreise könnten 2020 wegen höherer EEG-Umlage steigen*. 15. Okt. 2020. URL: <https://www.wiwo.de/energie-strompreise-koennten-2020-wegen-hoherer-eeg-umlage-steigen/25117122.html> (besucht am 23.06.2020).
64. *Power Systems Analysis Software*. URL: https://wiki.openelectrical.org/index.php?title=Power_Systems_Analysis_Software (besucht am 19.02.2019).
65. *VDI/VDE 3699 Blatt 1 – Prozessführung mit Bildschirmen – Begriffe*. URL: <https://www.vdi.de/richtlinien/details/vdivde-3699-blatt-1-prozessfuehrung-mit-bildschirmen-begriffe> (besucht am 27.02.2020).
66. *Wireshark*. URL: <https://www.wireshark.org/> (besucht am 19.02.2020).



Daten

3

Ein Hauptzweck der Energieinformatik ist die automatisierte Datenverarbeitung, sodass auf Basis von IKT-Systemen intelligente Entscheidungen getroffen werden können. Wie wir bereits gesehen haben, können Daten in solchen Systemen nur dann sinnvoll verarbeitet werden, wenn ihnen ein Informationsmodell zugrunde liegt (siehe Abschn. 2.4).

Im nachfolgenden Abschnitt wollen wir uns daher mit einem allgemeinen Informationsmodell der Energiewirtschaft befassen, dem sog. CIM. Dieses Modell hilft dabei, ansatzweise alle Komponenten des physikalisch-technischen und des kaufmännischen Energiesystems mit ihren Eigenschaften abzubilden. Daten können in der Energiewirtschaft so zwischen Anwendungssystemen unterschiedlicher Hersteller und über Unternehmensgrenzen hinweg ausgetauscht werden.

Anlagen der Energiewirtschaft besitzen üblicherweise einen Standort, sind also mit räumlichen Angaben verknüpft. Räumliche Daten spielen für viele Anwendungsfälle der Energiewirtschaft eine herausragende Rolle, z. B. für die Netzplanung, die Instandhaltung von Anlagen oder die Visualisierung unterschiedlichster Betriebszustände. Räumliche Daten benötigen allerdings besondere Verfahren der Speicherung, Verarbeitung, Auswertung und Visualisierung. Diesen Themen widmen wir daher einen eigenen Abschnitt (siehe Abschn. 3.2).

Neben den räumlichen Daten trifft man in der Energiewirtschaft besonders häufig auf sog. Zeitreihen. Sämtliche Messwerte, Zustandsdaten, Preiskurven oder Prognosen äußern sich überlicherweise in dieser Form. In Abschn. 3.3 werden wir daher auf die effiziente Speicherung, Verarbeitung und Darstellung von Zeitreihen mithilfe von Standardwerkzeugen eingehen.

Das Kapitel wird durch einige Hinweise zu öffentlich zugänglichen Datenquellen abgerundet, die sich insbesondere dafür eignen, eigene Energiesystemmodelle aufzubauen (siehe Abschn. 3.4).

3.1 Common Information Model

Ohne ein gemeinsames Datenmodell ist die Integration von Anwendungssystemen in der Energiewirtschaft oft mühsam und aufwendig. Eine solche Integration ist aber sowohl innerhalb von Organisationen als auch bei übergreifenden Prozessen über die Grenzen von Organisationen hinweg unerlässlich. Die Daten der proprietären Systeme basieren auf unterschiedlichen Informations- und Datenmodellen. Bei einem Datenaustausch müssen diese Formate dann übersetzt werden. Bei n eingesetzten Systemen müssen dabei maximal $\frac{n(n-1)}{2}$ Datenflüsse gepflegt werden. Dies ist oft schwierig, kompliziert und fehleranfällig. Abb. 3.1 zeigt die Anzahl von Kommunikationsverbindungen zwischen nur vier Anwendungssystemen, die jeweils gepflegt werden müssen.

Das **Common Information Model (CIM)** ist ein Ansatz, ein gemeinsames und übergreifendes Datenmodell für große Teile der Energiewirtschaft zu definieren. CIM zielt darauf ab, den Integrationsaufwand zwischen Anwendungssystemen deutlich zu reduzieren. CIM bildet dabei quasi eine gemeinsame Sprache, die durch alle Systeme verstanden werden kann. Abb. 3.2 zeigt die Anwendungsintegration mittels eines gemeinsamen Datenmodells wie CIM. Der Integrationsaufwand zwischen Anwendungssystemen sinkt, da ein gemeinsames Datenmodell genutzt wird und Schnittstellen nicht individuell angepasst werden müssen.

Der Kern von CIM ist ein sehr umfangreiches objektorientiertes Informationsmodell. Dieses wird in mehreren Abschnitten des Standards IEC 61970 definiert (siehe [21]). Das Modell basiert auf der Unified Modelling Language (UML) und wird auf Basis des Werkzeugs Enterprise Architect von Sparx Systems in mehreren Modelldateien gepflegt. Neben dem IEC kümmert sich vor allem die *CIM User Group* um die Verbreitung und Anwendung des Standards. Aktuell gültige und zukünftige Versionen der Standards können

Abb. 3.1 Kommunikationsverbindungen bei der direkten Integration von Anwendungssystemen

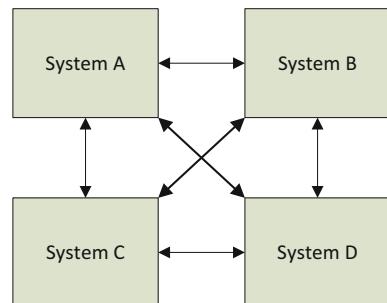
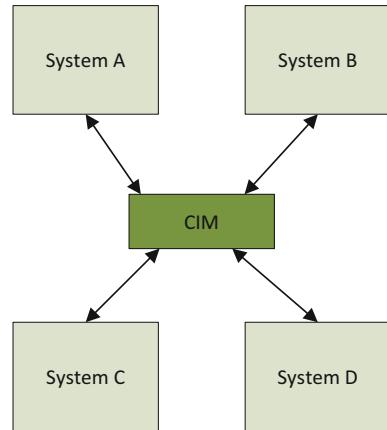


Abb. 3.2 Anwendungsintegration mittels CIM



dort bezogen werden. Für den Zugriff ist allerdings eine kostenpflichtige Mitgliedschaft notwendig.

CIM besteht (in Version 17 aus dem Jahr 2014) in seiner Gesamtheit aus 1616 Klassen, mit 8011 Attributen in 146 Paketen (siehe [10]). Das CIM-Informationsmodell stellt Klassen für ansatzweise alle Komponenten bereit, die ein Energiesystem ausmachen. Mithilfe von CIM können physische Anlagen, wie z. B. Leitungen, Transformatoren und Generatoren, dargestellt werden, aber auch virtuelle Konzepte, wie Rechnungen, Auktionen und Angaben zu Betrieb und Wartung. Es wird daher möglich, in der Energiewirtschaft alle möglichen Arten von Daten einheitlich darzustellen und auszutauschen. Die Klassen der CIM-Standards werden auf Pakete mit unterschiedlichem Fokus aufgeteilt. Beim Basisstandard IEC 61970 sind dies z. B. die Pakete:

- **Domain**

Das Paket Domain definiert grundlegende Datentypen und Einheiten für den Rest des CIM-Modells. Typische Klassen des Pakets sind z. B. UnitSymbol, ActivePower, CurrentFlow, Voltage oder Impedance.

- **Core**

Das Paket Core stellt Basisklassen bereit, die grundlegende Komponenten und ihre Eigenschaften festlegen. Typische Klassen des Pakets sind Equipment, ConductingEquipment, PowerSystemRessource oder Company.

- **Wires**

Das Paket Wires stellt Klassen bzgl. der elektrischen Charakteristika von Übertragungs- oder Verteilnetzen bereit. Typische Klassen des Pakets sind Fuse, Breaker, Line, PowerTransformer, Switch oder TapChanger.

- **Generation**

Das Paket Generation beinhaltet Klassen, um unterschiedliche Generatoren, Turbinen und Kessel abzubilden. Typische Klassen sind GenerationUnit, HydroPowerPlant oder ThermalGenerationUnit.

Eine der wichtigsten Anwendungsfälle für CIM ist es, das **funktionale Modell** einer Netztopologie abzubilden. Dieses beschreibt die Funktion der einzelnen Netzkomponenten und wird z. B. für die Lastflussberechnung verwendet. Entsprechende Modellelemente für Übertragungsnetze sind im Standard IEC 61970 definiert. Dabei wird i. d. R. von einem symmetrischen Netz ausgegangen, d. h., die drei Phasen des Drehstromnetzes werden zu einer Leitung zusammengefasst. Für Verteilnetze wird dieser Standard in IEC 61968 erweitert (siehe z. B. [5]). Dadurch können auch Netze dargestellt werden, bei denen die drei Phasen nicht symmetrisch sein müssen.

CIM nutzt extensiv die Möglichkeiten der Objektorientierung. Neben Assoziationen, Aggregation und Komposition wird vor allem die Vererbung genutzt. Als Beispiel sei hier der Leistungsschalter (Klasse Breaker) aus dem Paket Wires genannt. Ein Leistungsschalter besitzt grundsätzlich die Fähigkeit, ein- oder ausgeschaltet zu sein. Damit ist er eine spezielle Unterart eines Schalters (Klasse Switch). Ein Schalter wiederum ist ein stromleitendes Element (Klasse ConductingElement), welches wiederum einen physischen Anlagenteil darstellt (Klasse Equipment). Diese Beziehungen werden in CIM in Form einer Vererbungshierarchie abgebildet. Abb. 3.3 zeigt die Vererbungshierarchie eines Leistungsschalters als Ausschnitt des CIM.

Die Komponenten eines Stromnetzes werden in CIM nicht direkt miteinander verbunden. Stattdessen kann jeder stromleitende Anlagenteil (Klasse ConductingElement) mehrere Anschlusspunkte (Klasse Terminal) besitzen. Ein Anschlusspunkt gehört dabei immer genau zu einem stromleitenden Anlagenteil. Ein Anschlusspunkt wiederum kann mit genau einem Anschlussknoten (Klasse ConnectivityNode) verbunden sein. Anschlussknoten sind wiederum mit vielen Anschlusspunkten verbunden. Abb. 3.4 zeigt einen Ausschnitt aus dem CIM, welches die Beziehungen zwischen stromleitenden Elementen mit ihren Anschlusspunkten darstellt.

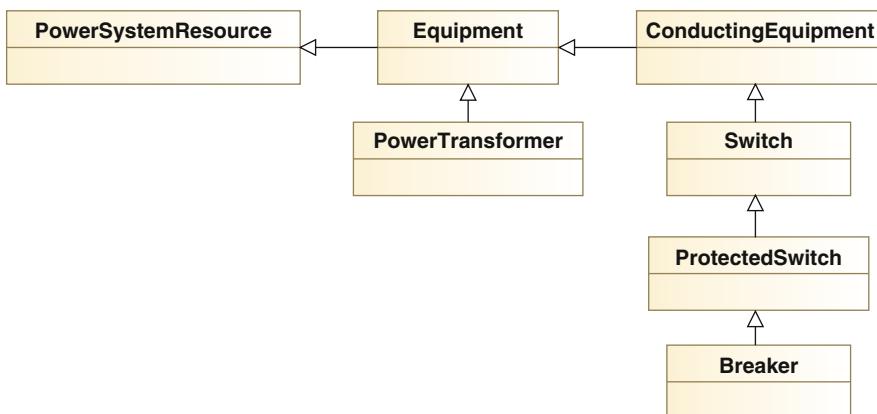


Abb. 3.3 Vererbungshierarchie eines Leistungsschalters im CIM-Informationsmodell

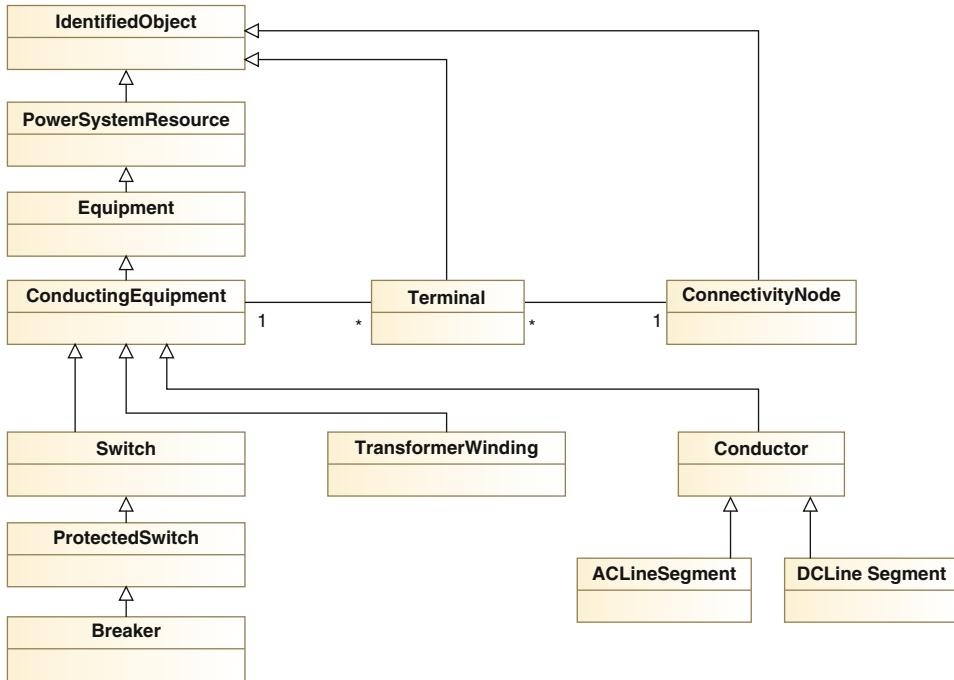


Abb. 3.4 Beziehung von stromleitenden Elementen zu Anschlusspunkten im CIM-Datenmodell

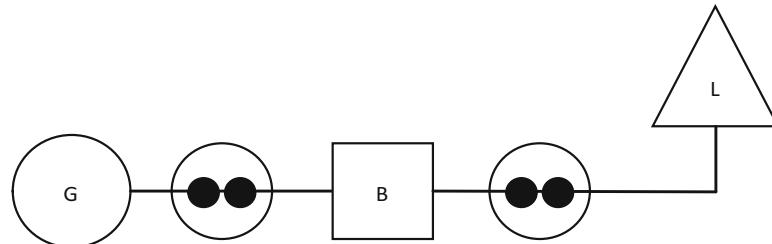


Abb. 3.5 Beispiel eines einfachen Stromnetzes aus Generator, Klemmen, Leistungsschalter und Last

Beispiel 3.1.1 Abb. 3.5 zeigt ein einfaches elektrisches Netz in Form eines Einliniendiagramms. Das Netz besteht lediglich aus einem Generator, zwei Klemmen, einem Leistungsschalter und einer Last.

Dieses Netz soll nun auf Basis von CIM-Klassen auf ein Objektmodell abgebildet werden. Dazu müssen Objekte der Klassen **Load**, **Terminal**, **ConnectivityNode**, **Breaker** und **SynchronousMaschine** genutzt werden. Das Ergebnis ist in Abb. 3.6 als Objektdiagramm der UML abgebildet.

Die Abbildung eines Transformators in CIM wird über die Klasse **PowerTransformer** abgebildet, die wiederum von der Klasse **ConductingElement** ableitet. Jeder

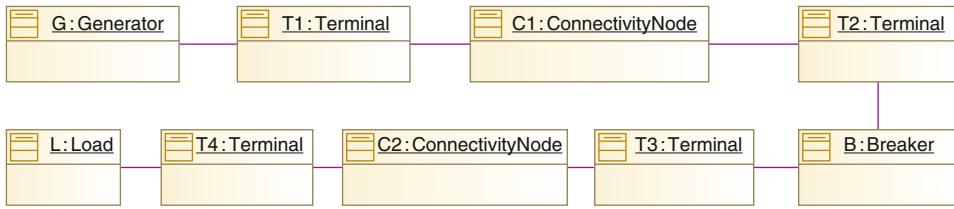


Abb. 3.6 Objektdiagramm des Beispielnetzes auf Basis von CIM-Objekten

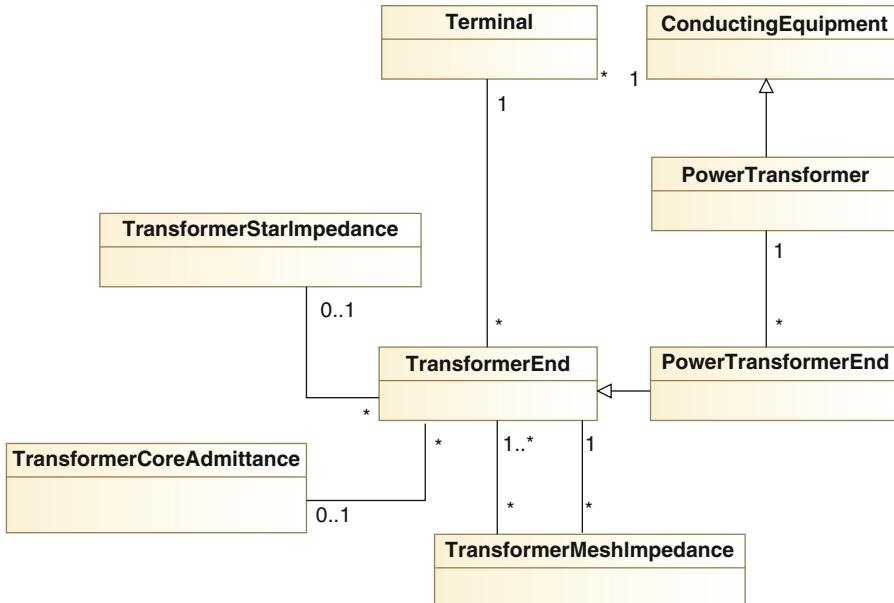


Abb. 3.7 Objektdiagramm des Beispielnetzes auf Basis von CIM-Objekten

Transformator verfügt dabei über Wicklungen, deren Anschlüsse über Objekte der Klasse **TransformerEnd** realisiert sind. Ein **TransformerEnd** kann dann über zugeordnete Terminals an das Netz angeschlossen werden. Über Objekte der Klassen **TransformerStarImpedance**, **TransformerCoreAdmittance** und **TransformerMeshImpedance** können verschiedene elektrische Kennwerte für den Transformator definiert werden. Abb. 3.7 zeigt die relevanten Klassen des CIM-Datenmodells, um einen Transformator abzubilden.

CIM unterscheidet zwischen einem physischen Asset und seinen funktionalen Aufgaben. Beide Perspektiven werden im Gesamtmodell miteinander in Beziehung gesetzt (siehe [9]). Ein Asset kann durchaus existieren, ohne mit der Netztopologie assoziiert zu sein, z. B. weil ein Leistungsschalter gerade im Lager liegt. Auf der anderen Seite kann

ein physisches Asset aber durchaus auch mehrere Aufgaben erfüllen. Wie wir in Abschn. 5.1 sehen werden, integrieren *Intelligent Electronic Devices* oft mehrere Funktionen, z. B. Lastschalter und Zähler in einem Gerät. Die Trennung der funktionalen von der topologischen Perspektive hat daher Vorteile. Ein Austausch eines physischen Assets bedarf dann keiner Änderung des funktionalen Modells. Ein funktionsgleicher Lastschalter kann in einem Netzmodell ersetzt werden, ohne dass die Topologie angepasst werden muss.

Neben dem funktionalen und dem Asset-Modell sind noch weitere Perspektiven des Energiesystems in CIM abbildungbar. Mit IEC 62325 existiert z. B. eine Erweiterung für den Energiehandel (siehe z. B. [16]). Marktteilnehmern wird dadurch der Datenaustausch in unterschiedlichen Phasen des Energiehandels ermöglicht. Obschon die Netzphysik überall auf der Erde gleich ist, unterscheiden sich die Marktmodelle zwischen einzelnen Ländern ganz erheblich. Daher werden im CIM in diesem Bereich mehrere unterschiedliche Informationsmodelle entwickelt. Durch die entso-e wird das europäische Marktmodell als ein eigenes CIM-Profil mit dem Namen European Style Market Profile (ESMP) gepflegt (siehe [12]).

CIM ist zunächst ein reines Informationsmodell. Um Daten auf Basis von CIM zwischen Anwendungssystemen oder Rechnern austauschen zu können, müssen CIM-basierte Objekte aber systemunabhängig in Form eines Datenstroms abgebildet werden. Mit **CIM RDF** existiert eine entsprechende Abbildung von CIM auf Extensible Markup Language (XML). Die Objekte eines CIM-Modells werden dabei auf XML-Tags abgebildet. Beziehe zu anderen Objekten werden über eindeutige Schlüssel hergestellt. CIM-Modelle können so in Form von XML-Dokumenten serialisiert und über Systemgrenzen hinweg ausgetauscht werden. Der folgende Ausschnitt stellt ein solches CIM-basiertes RDF-Dokument dar. Das Modell beinhaltet zwei Objekte, ein Objekt der Klasse `ConnectivityNode`, welches auf andere Objekte der Klasse `Terminal` Bezug nimmt.

```
1 <cim:Terminal rdf:id="T1">
2   <cim:Terminal.ConductingEquipment rdf:resource="#_L1"/>
3   <cim:Terminal.ConnectivityNode rdf:resource="#_C1"/>
4   <cim:IdentifiedObject.name>Terminal of Generator</cim:
      IdentifiedObject.name>
5 </cim:Terminal>
6 <cim:ConnectivityNode rdf:id="C1">
7   <cim:ConnectivityNode.Terminalsrdf:resource="#_T1"/>
8   <cim:ConnectivityNode.Terminalsrdf:resource="#_T2"/>
9 </cim:ConnectivityNode>
```

Mithilfe von CIM sind Daten der Energiewirtschaft einheitlich definiert und können auf Basis von CIM/RDF auch in einem Datenformat serialisiert werden. Bei einer Anfrage an einen CIM-basierten Datenspeicher müssen aber auch die gewünschte Operation auf den Daten und die entsprechenden Antworten und Fehlermeldungen formuliert werden können. Ein Beispielszenario wäre die Integration eines Asset-Management-Systems

mit einem SCADA-System. Das Asset-Management-System benötigt für die Planung von Wartungsmaßnahmen z. B. Informationen, wie oft ein bestimmter Leistungsschalter ausgelöst wurde. Die Anfrage an das SCADA-System und seine Antwort müssen dabei in Form von Nachrichten ausgedrückt werden können. In IEC 61968-100 wurde daher für die CIM-basierte Anwendungsintegration ein spezielles Nachrichtenformat für solche Anfragen eingeführt. Eine solche Nachricht besteht aus den folgenden Teilen:

- **Header**

Der Nachrichtenkopf beinhaltet das *Verb*, welches die Art der Nachricht vorgibt (z. B. `get`, `create`, `change` oder `delete`) und beschreibt, was mit der Nachricht zu tun ist. Das *Noun* identifiziert die betroffenen CIM-Objekte, die in der Nutzlastnachricht aufgeführt sind.

- **Request**

Der Request-Teil der Nachricht beinhaltet zusätzliche Informationen für eine Anfrage, z. B. die eindeutigen Schlüssel der angefragten CIM-Objekte.

- **Reply**

Der Reply-Teil der Nachricht beinhaltet Statusinformationen bzw. Fehlermeldungen bei Antworten auf eine Anfrage.

- **Payload**

Der Payload-Teil der Nachricht beinhaltet die eigentlichen CIM-Daten, die mithilfe der Nachricht ausgetauscht werden sollen.

Das folgende XML-Dokument stellt einen Ausschnitt aus einer solchen Nachricht dar. Inhaltlich wird dadurch mitgeteilt, dass zwei Schalter einen neuen Zustand eingenommen haben (siehe [11]).

```
1 <ns0:EventMessage >
2   <ns0:Header>
3     <ns0:Verb>changed</ns0:Verb>
4     <ns0:Noun>Switches</ns0:Noun>
5     <ns0:Revision>1</ns0:Revision>
6   </ns0:Header>
7   <ns0:Payload>
8     <m:Switches>
9       <m:Switch>
10        <m:mRID>363482488448</m:mRID>
11        <m:normalOpen>false</m:normalOpen>
12      </m:Switch>
13      <m:Switch>
14        <m:mRID>894094949444</m:mRID>
15        <m:normalOpen>true</m:normalOpen>
16      </m:Switch>
17    </m:Switches>
18  </ns0:Payload>
19 </ns0:EventMessage>
```

Auf Basis dieses Nachrichtenformats ist es nun technisch möglich, dass Anwendungssysteme der Energiewirtschaft in einer gemeinsamen Sprache Informationen austauschen. Die Nachrichten können dabei auf Basis unterschiedlicher Mechanismen transportiert werden. In IEC 61968-100 werden hauptsächlich Webservices und Java Message Service (JMS) als Integrationstechnologien genannt (siehe Abschn. 2.5).

Trotz des großen Potenzials wird das CIM in der Praxis bislang nur selten genutzt. Über die Gründe kann hier nur spekuliert werden. Möglicherweise sehen die Hersteller von IT-Lösungen im Energieumfeld die Integrationsfähigkeit ihrer Anwendungssysteme bislang nicht als wichtigen Wert oder verstehen sie sogar als Gefahr, da durch die einfache Integrierbarkeit modularer Systeme der Konkurrenzdruck steigen könnte. Man kann nur hoffen, dass das CIM durch die IT-Dienstleister der Energiewirtschaft in Zukunft stärker beachtet wird.

Den größten Verbreitungsgrad hat CIM aktuell auf der Ebene der Übertragungsnetzbetreiber. In Europa wurde durch entso-e der sog. Common Grid Model Exchange Standard (CGMES) definiert (siehe [7]). Europäische Übertragungsnetzbetreiber nutzen dieses CIM-Profil dazu, die nächsttägliche Netzauslastung in Form einer Lastflussanalyse zu bewerten. Dieser Prozess wird als Day Ahead Congestion Forecast (DACF) bezeichnet. Dazu erstellt jeder Übertragungsnetzbetreiber mehrere CIM-basierte Dateien, die täglich an zentraler Stelle zu einem globalen Modell zusammengeführt und bewertet werden.

Im Bezug zu CIM entsteht aktuell eine interessante Plattform im Rahmen des **openKonsequenz**-Konsortiums. Das Konsortium hat das Ziel, eine offene und modulare Softwareplattform für Netzbetreiber zu entwickeln. Unterschiedliche Module wie SCADA, ERP, GIS oder CRM werden über eine zentrale Kommunikationsinfrastruktur in Form eines Enterprise Service Bus (ESB) miteinander integriert. Die Nachrichten auf dem ESB basieren dabei auf dem CIM-Datenmodell. Abb. 3.8 zeigt die Integrationsarchitektur der Plattform (siehe [15], S. 21]).

Aktuell sind fünf Module entweder fertiggestellt oder in Entwicklung, darunter das Störungsinformationstool, Bereitschaftsplanung, geplante Netzmaßnamen, Betriebstagebuch und Einspeisemanagement. Grundlage der Entwicklung sind genaue Architekturvorgaben, die anhand eines entsprechenden Handbuchs spezifiziert wurden. Sie legen fest, auf welcher technischen Basis Module zu entwickeln sind und wie diese miteinander kommunizieren sollen (siehe [15]). Die openKonsequenz-Plattform kann die Basis für eine moderne, modulare und CIM-basierte Anwendungsarchitektur für Energieversorger bilden. Es bleibt abzuwarten, wie sich die Plattform in Zukunft weiterentwickeln wird.

Für die Arbeit mit CIM-Daten existieren mehrere kommerzielle Werkzeuge. **CIMSpy** gehört hier sicher zu den ausgereiftesten Werkzeugen am Markt (siehe [24]). CIMSpy bietet umfassende Möglichkeiten, CIM-Daten zu verarbeiten, zu editieren, zu exportieren, zu validieren und zu visualisieren. Abb. 3.9 zeigt ein Bildschirmfoto des Werkzeugs, bei dem ein CIM-basiertes Netzmodell in unterschiedlichen Ansichten dargestellt wird.

Mit **CIMDesk** steht zudem eine spezielle Version von CIMSpy für den CGMES der entso-e zur Verfügung (siehe [22]). Laut Aussagen des Herstellers ist für alle Über-

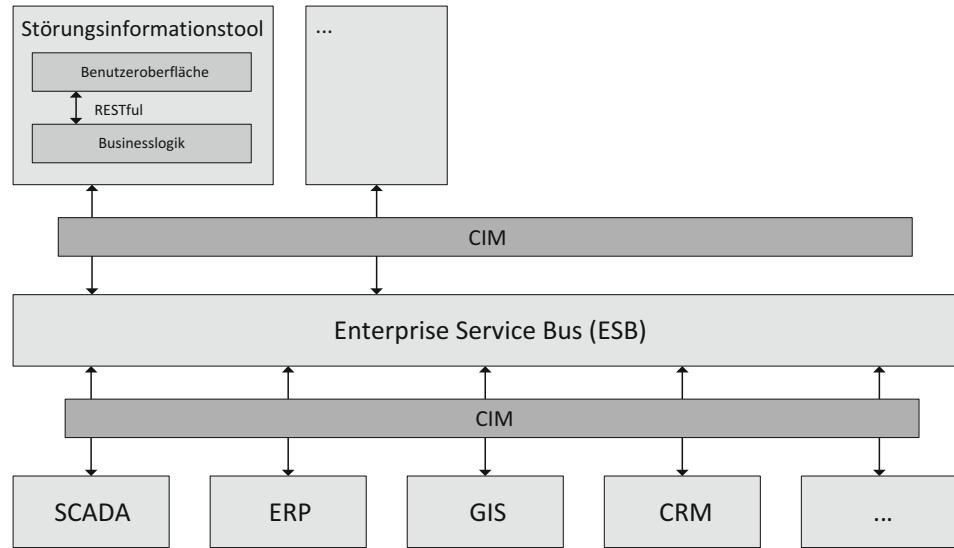


Abb. 3.8 Integrationsarchitektur der openKonsequenz-Plattform

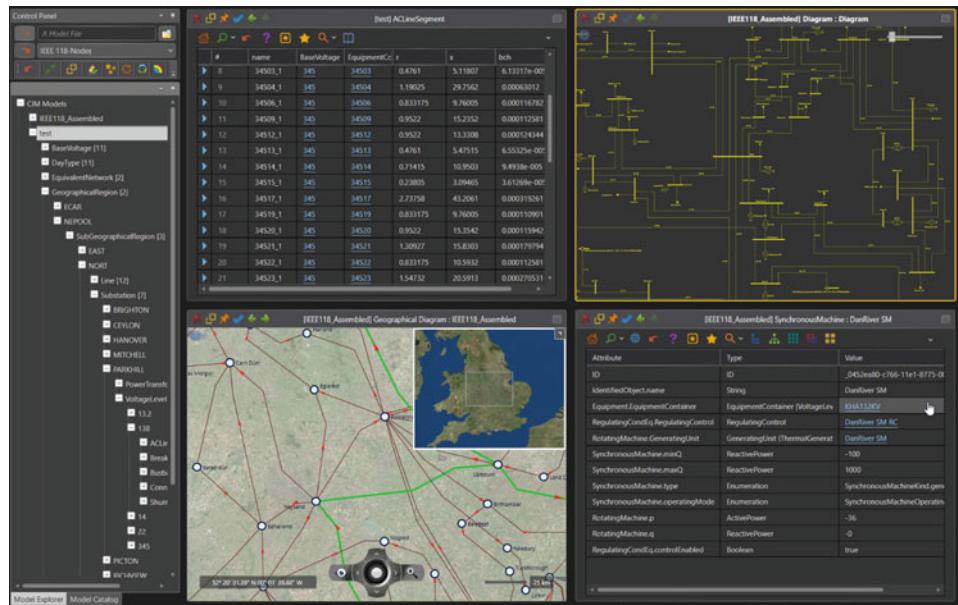


Abb. 3.9 Bildschirmfoto des Werkzeugs CIMSpy. Bildquelle: <https://www.powerinfo.us/CIMSpy.html>

tragungsnetzbetreiber in Europa die Validierung der CIM-Daten mithilfe von CIMDesk verpflichtend, bevor diese übermittelt werden.

Um CIM-kompatible Anwendungssysteme zu entwickeln, bedarf es entsprechender Programmierbibliotheken. Die CIM-Klassen müssen dann in einer konkreten Programmiersprache realisiert sein, z. B. in Java oder C#. Zudem werden Funktionen benötigt, um CIM-Objekte in das standardisierte XML-Format zu schreiben und dieses zu lesen. Leider existieren in diesem Kontext nur wenige freie Lösungen. Ein Beispiel ist die C++-Bibliothek libcimpp, welche im Rahmen des Projekts CIM++ entwickelt wurde (siehe [17]). Das Projekt bildet das CIM-Modell in Form von C++-Klassen ab, zudem können XML/RDF-Daten gelesen und geschrieben werden. Die Bibliothek kann allerdings für kommerzielle Projekte nicht kostenlos eingesetzt werden. Eine freie Alternative unter der MIT-Lizenz ist das Projekt PyCIM. Mithilfe dieser Python-Bibliothek können leicht eigene CIM-Modelle erstellt und in den zuvor beschriebenen XML-Dateien geschrieben oder daraus gelesen werden. Die Bibliothek ist dank der MIT-Lizenz auch für kommerzielle Projekte einsetzbar, offenbar wird das Projekt aber nicht mehr aktiv gepflegt.

Beispiel 3.1.2 Der folgende Programmcode erzeugt ein CIM-basiertes Objektmodell, welches das einfache Beispielnetz von oben mit PyCIM umsetzt. Anschließend wird das Modell in eine XML/RDF-Datei geschrieben.

```
1 g1 = SynchronousMachine(name="Generator", UUID=str(uuid.uuid4()))
2 t1 = Terminal(name="Terminal 1", UUID=str(uuid.uuid4()))
3 t2 = Terminal(name="Terminal 2", UUID=str(uuid.uuid4()))
4 t3 = Terminal(name="Terminal 3", UUID=str(uuid.uuid4()))
5 t4 = Terminal(name="Terminal 4", UUID=str(uuid.uuid4()))
6 c1 = ConnectivityNode(name='Node 1', UUID=str(uuid.uuid4()))
7 c2 = ConnectivityNode(name='Node 2', UUID=str(uuid.uuid4()))
8 b1 = Breaker(name="Breaker", UUID=str(uuid.uuid4()))
9 l1 = EnergyConsumer(name="Consumer", UUID=str(uuid.uuid4()))
10
11 g1.addTerminals(t1)
12 c1.addTerminals(t1, t2)
13 b1.addTerminals(t2, t3)
14 c2.addTerminals(t3, t4)
15 l1.addTerminals(t4)
16
17 d = {}
18 for elem in [g1, t1, t2, t3, t4, c1, c2, b1, l1]:
19     d[elem.UUID] = elem
20
21 cimwrite(d, "simple_model.xml")
```

3.2 Räumliche Daten

Im letzten Abschnitt haben wir mit CIM ein allgemeines Informationsmodell für die Energiewirtschaft behandelt. Eine der Möglichkeiten von CIM ist die Darstellung von funktionalen Netzmodellen oder die Abbildung von Energieanlagen, um das Asset-Management zu unterstützen. Solche Energieanlagen sind nicht nur durch Attribute wie den Hersteller oder das Baujahr gekennzeichnet, sie haben meist auch einen räumlichen Bezug. Sowohl für die Netzplanung als auch für die Instandhaltung ist der Standort von Anlagen im Bezug zu sonstigen geografischen Gegebenheiten, wie z. B. Gebirgen oder Flussverläufen, ein wichtiges Kriterium (siehe auch [6]). Auch für den Betrieb von Netzen und Erzeugungseinheiten sind räumliche Daten wichtig, da z. B. eine Visualisierung in Form einer Landkarte viele Vorteile bietet. Für die Speicherung und Visualisierung von räumlichen Daten werden allerdings besondere Werkzeuge benötigt. Hier kommen häufig sog. räumliche Datenbanksysteme sowie geografische Informationssysteme (GIS) zum Zug. Der folgende Abschnitt soll diese Systeme und ihre Anwendung in der Energiewirtschaft näher erläutern.

Auch in CIM können geografische Informationen abgebildet werden. Dazu stehen die Klassen `Location`, `PositionPoint` und `CoordinateSystem` zur Verfügung. Die niedrigste Abstraktionsebene wird durch die Klasse `Location` dargestellt. Ein Objekt dieses Typs beschreibt einen einzelnen Punkt in einem Koordinatensystem, welcher mindestens eine X- und eine Y-Koordinate besitzt. Einem logischen Ort, der über die Klasse `Location` dargestellt wird, können mehrere solcher Punkte zugewiesen werden. Abb. 3.10 zeigt einen Ausschnitt aus dem CIM-Informationsmodell mit den relevanten Klassen zur Darstellung von geografischen Daten.

Beispiel 3.2.3 Ein Leiter für Wechselstrom wird in CIM durch ein Objekt der Klasse `ACLineSegment` dargestellt. Einem solchen Leiter werden sinnvollerweise zwei Objekte der Klasse `PositionPoint` zugeordnet, jeweils ein Objekt für jedes Ende des Leiters.

Die Bedeutung der Koordinaten wird durch den Bezug auf ein Koordinatensystem festgelegt. Es erscheint logisch, dass in CIM vor allem physischen Assets ein geografischer Ort zugeordnet werden kann, wodurch in der Praxis der genaue Standort einer Anlage dargestellt wird. Darüber hinaus ist es aber ebenfalls möglich, Komponenten des funktionalen Modells, z. B. einem Objekt der Klasse `PowerSystemResource`, einen Ort zuzuordnen. Bei einem Datenaustausch kann so z. B. der geografische Ort von funktionalen Elementen übermittelt werden, ohne auch die Details zu den Assets übertragen zu müssen. In der Planungsphase ist der Aufstellungsort eines Assests zudem mitunter bereits bekannt, nicht aber, um welche konkrete Realisierung (Hersteller usw.) es sich handeln wird.

Für die Darstellung von räumlichen CIM-Daten stehen verschiedene Werkzeuge zur Verfügung. Die bereits erwähnten Anwendungssysteme `CIMSpy` und `Cimphony` sind beide in der Lage, Netztopologien im Bezug zu einer Landkarte darzustellen (siehe

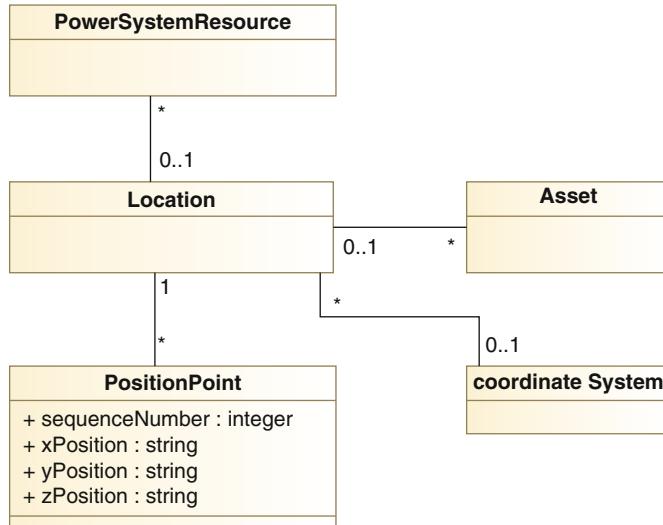


Abb. 3.10 Klassen für die Abbildung des geografischen Orts in CIM

Abschn. 3.1). Mit CIMSpace wird durch das Schweizer Unternehmen 9code eine kostenlos nutzbare Lösung zur Verfügung gestellt (siehe [23]). CIMSpace ist ein webbasiertes Werkzeug, um Netztopologien auf Basis von CIM in einer Landkarte zu visualisieren. CIMSpace nutzt dabei Kartenmaterial von OpenStreetmap als Basis. Die darzustellende Netztopologie kann aus einer CIM/RDF-Datei importiert oder mithilfe von CIMSpace in einer Landkarte neu erstellt werden. Das Netz wird dann in einer überlagerten Ebene entsprechend dargestellt. Abb. 3.11 zeigt die Darstellung eines Verteilnetzes in Kombination mit geografischen Daten in CIMSpace.

Räumliche Datenbanken (*Spatial Databases*) sind in der Lage, Daten mit Raumbezug zu speichern und zu verarbeiten. Solche Datenbanksysteme können nicht nur Standarddatentypen wie Ganzzahl oder Text in Datensätzen verarbeiten, sondern bieten zudem Datentypen an, um geografische Angaben abzubilden. Auf geografischen Datentypen werden zudem Funktionen angeboten, um z. B. die Entfernung zwischen zwei Punkten auf der Erdoberfläche zu bestimmen. Auf Basis einfacher Datentypen mit Raumbezug wie einem Punkt oder einer Linie ist man zudem in der Lage, komplexe Objekte, wie Ländergrenzen oder Flussverläufe, darzustellen und zu speichern. Ihre besondere Stärke spielen räumliche Datenbanken durch die Möglichkeit aus, räumliche Daten in Kombination mit regulären Daten zu verarbeiten und auszuwerten.

Ein bekannter Open-Source-Vertreter solcher Datenbanksysteme ist PostgreSQL mit der Erweiterung **PostGIS**. Die relationale Datenbank PostgreSQL wird durch PostGIS um zusätzliche Datentypen und Funktionen erweitert, die sich nahtlos in SQL integrieren. Sowohl PostgreSQL und PostGIS sind Open-Source-Systeme und für die gängigen Plattformen wie Windows, Linux und OS X verfügbar. PostgreSQL/PostGIS

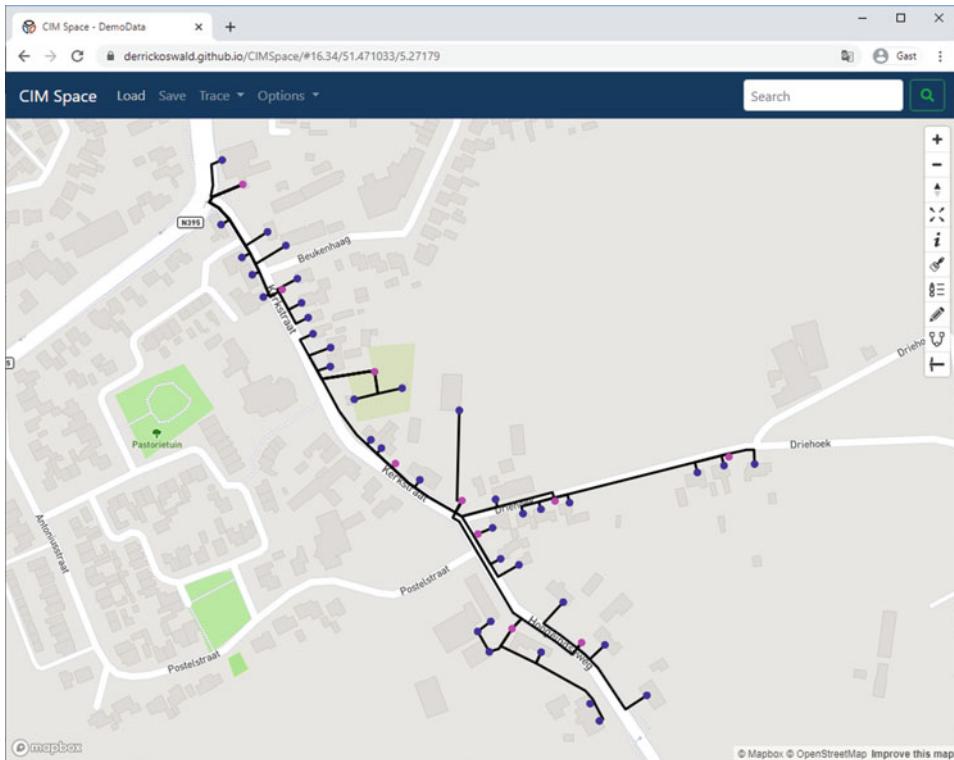


Abb. 3.11 Darstellung eines Verteilnetzes in Kombination mit geografischen Daten in CIMSpace

kann mit überschaubarem Aufwand als eigenständige Anwendung installiert werden. Der Installationsassistent von PostgreSQL ermöglicht auch die Installation von PostGIS in einem nachgelagerten Schritt. Hilfreich ist auch die Installation des Werkzeugs pgAdmin, welches die browserbasierte Administration einer PostgreSQL-basierten Datenbank erlaubt. Alle zuvor genannten Werkzeuge finden sich auch im linux-basierten Livesystem OSGeoLive. Dieses beinhaltet neben PostGIS auch diverse weitere freie Werkzeuge, wie z. B. QGis, die für die Arbeit mit Geodaten hilfreich sind.

Um in einer existierenden PostgreSQL-Datenbank räumliche Daten verarbeiten zu können, muss zunächst die PostGIS-Erweiterung aktiviert werden. Dies kann mithilfe des folgenden SQL-Kommandos erreicht werden:

```
1 CREATE EXTENSION postgis;
```

In pgAdmin sollte danach die PostGIS-Erweiterung im Bereich Extensions angezeigt werden. Zudem wird dadurch die Tabelle `spatial_ref_sys` angelegt. Diese enthält eine Liste von sog. Koordinatenreferenzsystemen, auf die sich alle Geodaten in der Datenbank beziehen.

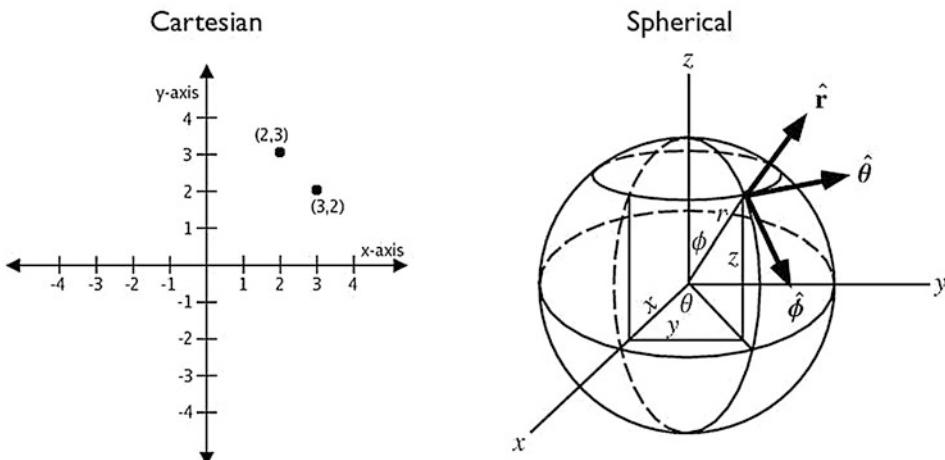


Abb. 3.12 Unterschied zwischen kartesischem und sphärischem Koordinatensystem. Bildquelle: Paul Ramsey und Mark Leslie. <http://postgis.net/workshops/postgisintro/geography.html>, <https://creativecommons.org/licenses/by-sa/3.0/legalcode>

PostGIS und viele andere räumliche Datenbanken können Geodaten entweder auf Basis eines kartesischen Koordinatensystems oder auf Basis eines sphärischen Koordinatensystems interpretieren. Im kartesischen Koordinatensystem repräsentiert der Datentyp `Geometry` einen Punkt in einer planaren Fläche. Der Vorteil dieser Darstellung eines Ortes ist, dass viele Berechnungen in der ebenen Fläche sehr einfach sind. Der Abstand zwischen zwei Punkten kann dann leicht mithilfe des Satzes von Pythagoras errechnet werden. Allerdings hat diese Darstellung auch erhebliche Nachteile. Durch die Krümmung der Erdoberfläche entstehen vor allem für weit entfernte Orte große Ungenauigkeiten, wenn z. B. Abstände berechnet werden sollen. Für solche Fälle ist es also sinnvoller, ein sphärisches Koordinatensystem zu nutzen. Im sphärischen Koordinatensystem repräsentiert der Datentyp `Geography` einen Punkt auf einer Kugeloberfläche, sodass Ortsangaben auf der Erdoberfläche sehr viel präziser dargestellt werden können. Leider werden dadurch einige Berechnungen sehr viel komplexer, sofern sie von der Datenbank überhaupt noch angeboten werden. Abb. 3.12 zeigt den grundsätzlichen Unterschied zwischen den beiden Koordinatensystemen.

Der einfachste Datentyp von PostGIS ist der Punkt. Dieser besitzt mindestens eine X- und eine Y-Koordinate, z. B. eine geografische Breite und eine Länge. In Structured Query Language (SQL) wird ein solcher Punkt mithilfe der Funktion `ST_Point` wie folgt erzeugt:

```
1 SELECT ST_Point(0, 0);
```

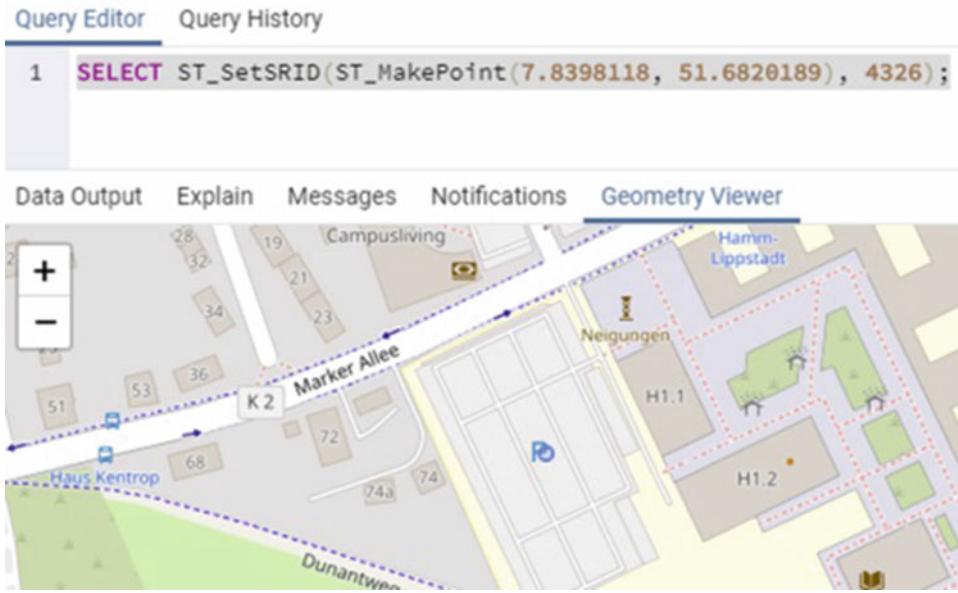


Abb. 3.13 Darstellung eines einzelnen Punktes in einer Karte von pgAdmin

Standardmäßig ist ein solcher Punkt eine `Geometry`, wird also in einem kartesischen Koordinatensystem interpretiert. Ein Punkt in einem sphärischen Koordinatensystem wird wie folgt erzeugt:

```
1 SELECT ST_Point(7.8398118, 51.6820189)::geography;
```

Die Bedeutung der Koordinaten wird durch ein Bezugssystem (*Spational Reference System*) vorgegeben. Meist wird heute das World Geodetic System 1984 (WGS 84) verwendet, welches in der PostGIS-Systematik die Spational Reference ID (SRID) mit der Nummer 4326 verwendet. Ein Koordinate auf der Erdoberfläche kann dann mithilfe des folgenden SQL-Kommandos erzeugt werden:

```
1 SELECT ST_SetSRID(ST_Point(7.8398118, 51.6820189)::geography,
4326);
```

Da pgAdmin eine integrierte Ansicht für geografische Angaben besitzt, kann ein solcher Punkt bereits in einer Landkarte dargestellt werden. Abb. 3.13 zeigt die Darstellung des Punktes in pgAdmin. Die Koordinate liegt auf dem Parkplatz der Hochschule Hamm-Lippstadt in Hamm.

Wenn Funktionen auf geometrische Daten angewandt werden sollen, müssen sich diese auf dasselbe Bezugssystem beziehen. Liegen mehrere Objekte nicht im selben Bezugssys-

tem, kann das Bezugssystem des Objekts „g“ mithilfe der Funktion `ST_Transform(g, srid)` transformiert werden.

Beispiel 3.2.4 Das Empire State Building liegt im WGS 84 ungefähr an der Position -73.9858108 geografischer Länge und 40.7484676 geografischer Breite. Wir können diesen Punkt nun z. B. in das System EPSG:26918 umwandeln, welches in Nordamerika benutzt wird. Dazu wird der folgende Befehl eingesetzt:

```
1 select ST_AsText(st_transform('srid=4326;point(-73.9858108  
40.7484676)::geometry, 26918));
```

Auf den räumlichen Datentypen bietet PostGIS eine Vielzahl von Funktionen an, z. B. Flächen-, Abstands- und Längenberechnung. Die Funktion `ST_distance(p1, p2)` berechnet z. B. den Abstand zwischen zwei Punkten p1 und p2. Dabei müssen beide Punkte auf demselben Basistyp basieren (also entweder `Geometry` oder `Geography`) und sich auf dasselbe Bezugssystem beziehen.

Beispiel 3.2.5 Der folgende SQL-Befehl berechnet den Abstand der beiden Punkte (0,0) und (2,2).

```
1 select st_distance(st_point(0,0), st_point(2,2))
```

Da beide Punkte im kartesischen Koordinatensystem liegen, kann der Abstand über den Satz von Pythagoras bestimmt werden.

Um komplexere geometrische Objekte aufzubauen, unterstützt PostGIS, wie viele andere Geodatenbanken auch, eine spezielle Auszeichnungssprache namens Well-Known-Text (WKT). Ein Linienzug aus mehreren Strecken kann in WKT wie folgt ausgedrückt werden:

```
1 LINESTRING (7.841909 51.6833296, 7.8388372 51.6824185)
```

Ein geschlossenes Polygon wird darin wie folgt ausgedrückt:

```
1 POLYGON (  
2   (7.84117093166529 51.6820616308458,  
3   7.84147230488234 51.6821487749086,  
4   7.8416357 51.6819363612556,  
5   7.84132888027904 51.6818474016915,  
6   7.84117093166529 51.6820616308458))
```

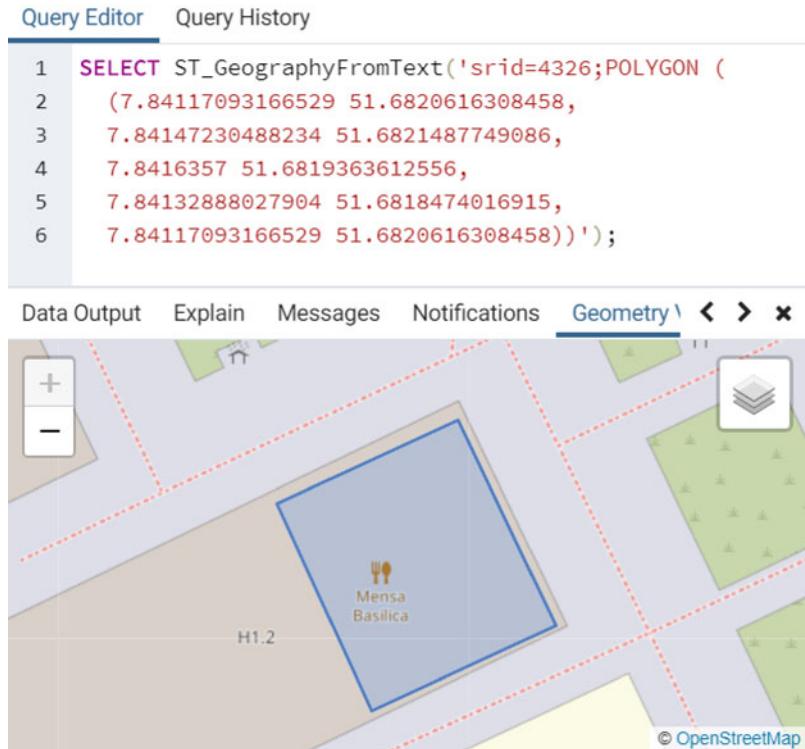


Abb. 3.14 Darstellung eines Polygons in pgAdmin

PostGIS ist in der Lage, diese Ausdrücke zu parsen und als geometrische oder geografische Objekte zu behandeln. Dafür werden die beiden Funktionen `ST_GeometryFromText(text)` und `ST_GeographyFromText(text)` angeboten. Mithilfe des folgenden SQL-Befehls kann in pgAdmin das Polygon bereits in einer Landkarte dargestellt werden.

```

1 SELECT ST_GeographyFromText('srid=4326;POLYGON (
2   (7.84117093166529 51.6820616308458,
3   7.84147230488234 51.6821487749086,
4   7.8416357 51.6819363612556,
5   7.84132888027904 51.6818474016915,
6   7.84117093166529 51.6820616308458))';

```

In Abb. 3.14 wird das Ergebnis dargestellt, wie es in der Oberfläche von pgAdmin zu sehen ist. Es handelt sich bei diesem Polygon um eine recht interessante Fläche, die Mensa der Hochschule Hamm-Lippstadt in Hamm.

Ein Linienzug besitzt eine Länge, der mithilfe der Funktion `ST_Length(g)` auf dem Objekt g bestimmt werden kann. Polygone hingegen besitzen einen Flächeninhalt,

der über die Funktion `ST_Area(g)` berechnet werden kann. Die Fläche des oben dargestellten Polygons liegt bei ca. 600 m².

Beispiel 3.2.6 Wir wollen im Folgenden den Umgang mit räumlichen Daten in einer PostGIS-Datenbank demonstrieren. Die Open Power Systems Data Initiative stellt eine Liste aller regenerativen Erzeugungsanlagen in Deutschland in Form einer CSV-Datei bereit (siehe dazu Abschn. 3.4). Die CSV-Datei kann z. B. mithilfe eines Python-Skripts in die Tabelle `renewable_power_plants` einer PostGIS-Datenbank geladen werden. Danach werden die Längen- und Breitengrade der Anlagen mithilfe von SQL in einen Punkt vom Typ `Geography` umgewandelt und in der neuen Tabellenspalte `geo` abgelegt.

```
1 alter table renewable_power_plants add column geo geography;
2 update renewable_power_plants set
3     geo=ST_SetSRID(ST_MakePoint(lon, lat), 4326);
```

Auf dieser Tabelle können nun die räumlichen Funktionen von PostGIS angewendet werden. Abb. 3.15 zeigt das Ergebnis, wenn die ersten tausend Anlagen aus der Tabelle in der Kartenansicht von pgAdmin angezeigt werden.

Auf den eben bereitgestellten Daten können bereits sehr komplexe Abfragen umgesetzt werden. Als Beispiel sollen genau diejenigen Onshore-Windanlagen ausgegeben werden, die im Umkreis von 10 km um die Hochschule in Hamm zu finden sind. Dazu kann erneut die Funktion `ST_distance` genutzt werden. Da die Tabellenspalte `geo` auf dem Datentyp `Geography` basiert, wird dieser Abstand auf der sphärischen Erdoberfläche berechnet. Die Abfrage kann wie folgt formuliert werden:

```
1 select * from renewable_power_plants
2 where technology='Onshore' and
3     ST_distance(geom, ST_SetSRID(
4         ST_Point(7.8398118, 51.6820189)::geography, 4326)) <= 10000
```

Als Ergebnis werden 23 Anlagen ausgegeben. Die Datenbank benötigt dafür nur wenige Millisekunden.

Räumliche Datenbanken wie PostGIS dienen dazu, geografische Objekte zu speichern. Sie stellen zudem über spezielle SQL-Funktionen Funktionalität bereit, um die Topologie auszuwerten und abzufragen. Die **GIS** können solche Datenbanken als Datenquelle nutzen. Sie ergänzen räumliche Datenbanken um Funktionen zur Erfassung, Bearbeitung, Organisation und Präsentation der Daten in Form von Karten. Es existiert eine Vielzahl von kommerziellen Produkten, darunter ArcGIS von ESRI oder Map3D von Autodesk. Vergleichbare quelloffene Systeme sind GRASS GIS und QGis.

QGis ist ein quelloffenes geografisches Informationssystem, welches als lokale Desktop-anwendung für alle gängigen Betriebssysteme zur Verfügung steht. QGis steht unter der GNU General Public License und wird unter dem Dach der Open Source Geospatial

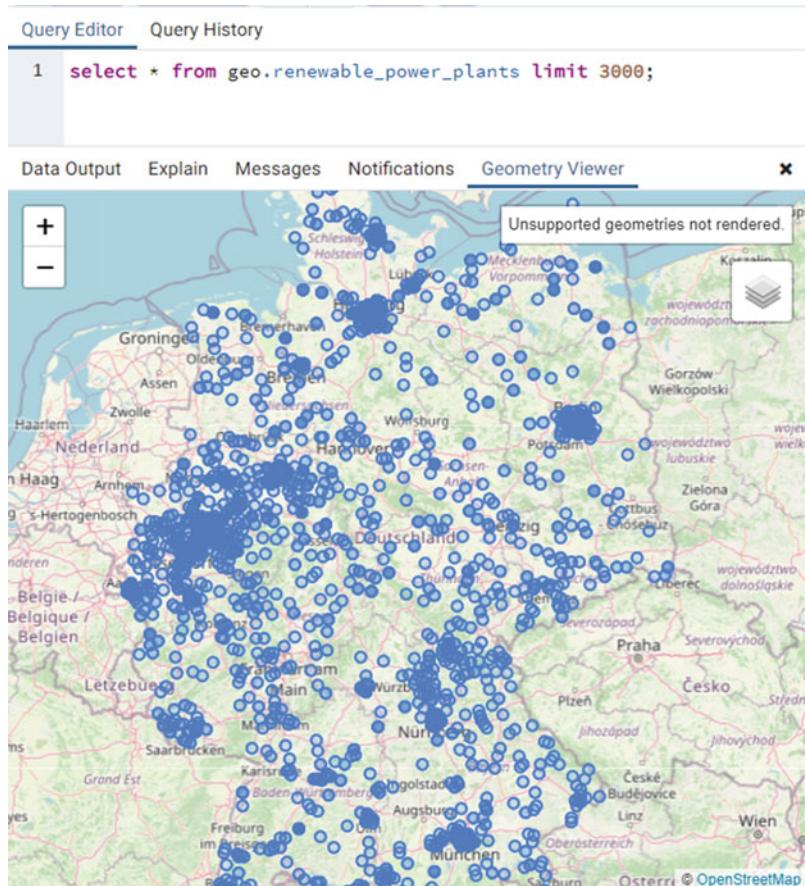


Abb. 3.15 Die Standorte einiger regenerativer Energieanlagen als Abfrageergebnis von PostGIS

Foundation (OSGeo) weiterentwickelt. QGis unterstützt eine große Vielfalt unterschiedlicher räumlicher Datenformate und Datenbanken, z. B. Shapefile, GeoTIFF, PostGIS u. v. m. Daten aus diesen unterschiedlichen Quellen können importiert, transformiert und gemeinsam, meist in einer Landkarte, visualisiert werden. QGis stellt dafür einen enormen Funktionsumfang bereit, u. a. auch ein internes python-basiertes Scripting. Darüber hinaus existiert aber eine große Vielfalt von externen Plug-ins, um mit räumlichen Daten zu arbeiten. Abb. 3.16 zeigt ein Bildschirmfoto von QGis. Auf einer Hintergrundkarte auf Basis von OpenStreetMap werden dabei die Standorte erneuerbarer Energieanlagen dargestellt.

QGis organisiert geografische Daten in verschiedenen Ebenen, die übereinander z. B. in einer Landkarte dargestellt werden. Jede Ebene kann dabei auf einer eigenen Datenquelle

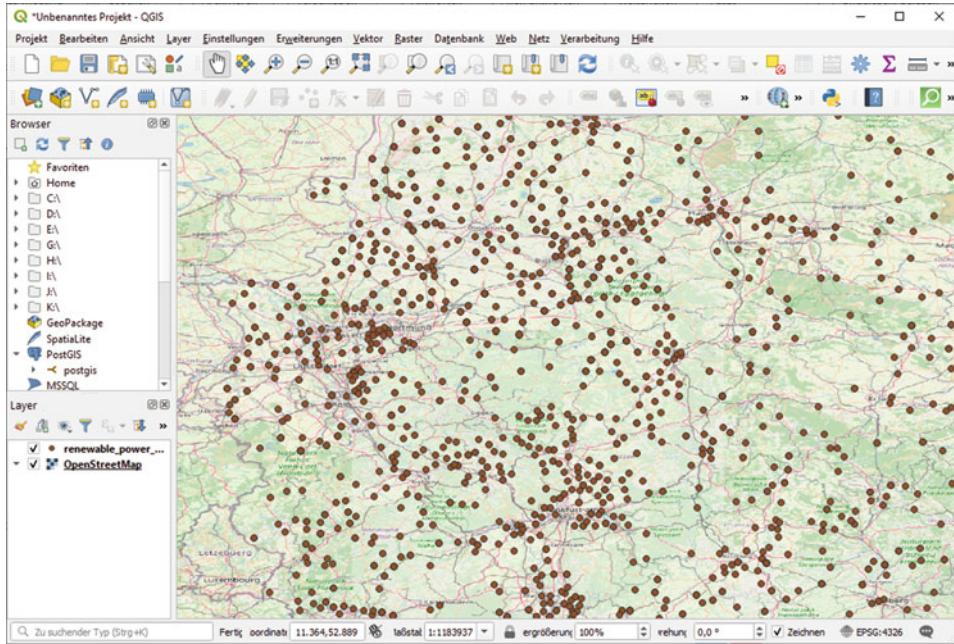


Abb. 3.16 Darstellung von regenerativen Energieanlagen in einer Kartenansicht von QGis

basieren. Die Darstellung der Daten ist dabei ansatzweise beliebig konfigurierbar. Aus der Kombination von räumlichen Datenbanken und den Funktionen eines Geoinformationsystems wie QGis entstehen vielfältige Möglichkeiten, um räumliche Daten auszuwerten und zu visualisieren. Abb. 3.17 zeigt die Visualisierung aller Windkraftanlagen in Deutschland mithilfe von QGis. Die Nennleistung der Anlagen wurde dabei anhand der Kreisgröße dargestellt.

3.3 Zeitreihen

Im letzten Abschnitt wurde der Umgang mit räumlichen Daten dargestellt. Solche Daten spielen insbesondere im Asset-Management oder im operativen Betrieb von Energieanlagen eine wichtige Rolle. Daten mit Raumbezug stellen in der Energiewirtschaft tendenziell eher Stammdaten dar. Sie gehören zu Grundinformationen einer Anlage und ändern sich nur selten.

Energiedaten hingegen zählen zu den Bewegungsdaten. Sie entstehen durch den Betrieb der Anlagen bzw. durch die Geschäftstätigkeit der Energieunternehmen. Energiedaten äußern sich meist als Zeitreihen mit einem irgendwie gearteten Energiebezug (siehe

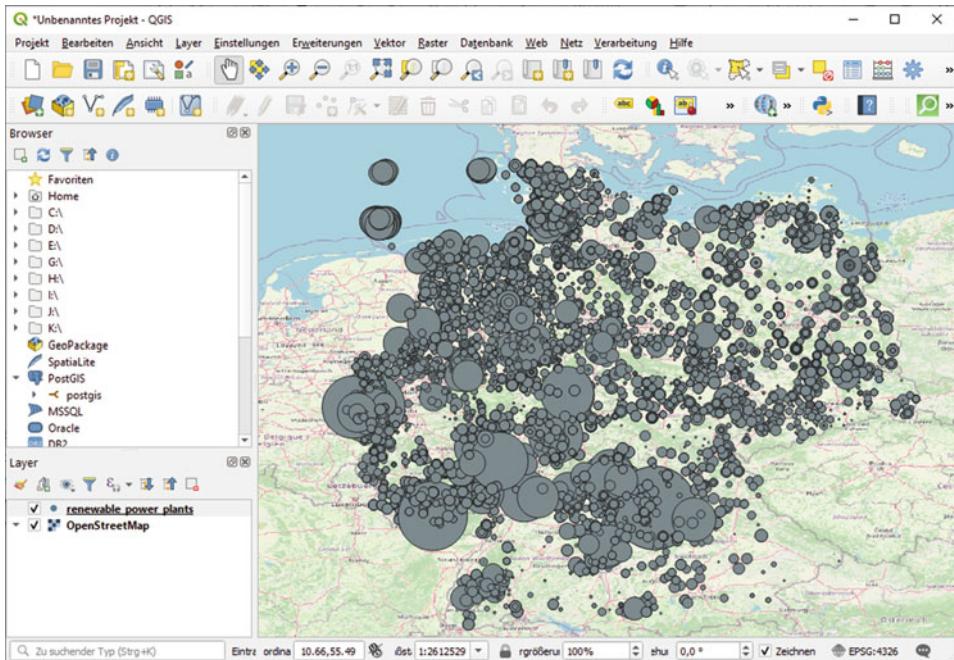


Abb. 3.17 Darstellung von Windkraftanlagen in QGis. Die Größe der Kreise wird in Abhängigkeit der Nennleistung variiert

[13, S. 27]). Beispiele für solche Energiedaten sind Energieverbräuche, Einspeisemengen, Kraftwerks- oder Netzfahrpläne. Im weiteren Sinne können aber auch Wetterdaten oder Marktpreise gemeint sein.

Ansatzweise alle Energieunternehmen müssen mit Energiedaten umgehen. Dies führt zum **Energiedatenmanagement (EDM)** und beinhaltet die Erfassung, Speicherung, Analyse, Visualisierung und Übertragung von Energiedaten (siehe [13, S. 35, 66 f.]). Das EDM ist eine Kernaufgabe eines jeden Unternehmens der Energiewirtschaft und ist in der gesamten Wertschöpfungskette zu finden (siehe [4, S. 28]). EDM sorgt dafür, dass der komplette Datenbestand aktuell und konsistent gehalten wird. In der Literatur wird der Begriff Energiedatenmanagement mitunter mit Marktkommunikation gleichgesetzt. Natürlich müssen auch im Kontext von Marktprozessen wie GPKE, MABIS etc. Energiedaten ausgetauscht werden (siehe Abschn. 5.4). Das EDM sollte aber eher als technische Aufgabe für alle Teile der Energiewertschöpfungskette gesehen werden.

Für die Realisierung des EDM werden IKT-Systeme oder Komponenten benötigt, die durch EnergieinformatikerInnen entworfen und realisiert werden. In viele Systeme ist das EDM dabei als Querschnittsfunktion integriert. Es existieren dabei zwei Ansätze (siehe [4, S. 35 f.]).

Integrierte Systeme bündeln die gesamte Funktionalität in nur einem Produkt. Dieses Vorgehen hat den Vorteil, dass ein hoher Integrationsgrad bei den Prozessen möglich ist

und einheitliche Benutzerführung vorherrscht. Nachteilig sind die hohe Abhängigkeit von nur einem Hersteller und die oft komplexe und damit teure Anpassung an eigene Prozesse. Solche Systeme sind häufiger bei kleineren und mittleren Energieunternehmen zu finden.

Im Gegensatz zu den integrierten Systemen bestehen **Best-of-Breed**-Lösungen aus vielen Einzelkomponenten von mitunter unterschiedlichen Herstellern. In einer solchen Systemlandschaft können die Speicherung, die Auswertung, Verarbeitung, Visualisierung und Übermittlung von Energiedaten durch jeweils eigene Komponenten unterschiedlicher Hersteller realisiert werden. Dies hat den Vorteil, dass die eigene Systemlandschaft leichter an die spezifischen Anforderungen und Prozesse des Unternehmens angepasst werden kann. Der Nachteil entsteht allerdings durch einen hohen Integrationsaufwand. Auch die meist mehrfache Datenhaltung und die unterschiedlichen Benutzeroberflächen können zu Problemen führen. Solche Systemlandschaften sind bei größeren Energieunternehmen vorherrschend und dort oft gar nicht anders umzusetzen.

Eine **Zeitreihe** besteht aus einer geordneten Folge von diskreten Datenpunkten mit einem Bezug zu einem Zeitpunkt bestehend aus Datum und Uhrzeit. Meist besitzt eine Zeitreihe zudem weitere Metadaten, wie die physikalische Einheit, die Abtastung, die Datenquelle oder Qualitätsmerkmale. In der Energiebranche spielen viertelstündliche (Strom) bzw. stündliche Zeitreihen (Gas) eine herausragende Rolle. Beispiele für Zeitreihen sind Lastgänge, Fahrpläne, Gebote oder Marktdaten. Für die Angabe von Zeitpunkten existiert die ISO Norm 8601 (siehe [2]). Ein Beispiel für eine solche Angabe ist „2018-10-05T11:00:00Z“. Dabei steht das Kürzel „Z“ für die sog. universelle Weltzeit (UTC). Die Verwendung eines globalen Zeitstandards wie UTC sollte gegenüber einer lokalen Zeitangabe vorgezogen werden. Bei der Zeitsynchronisation von Prozesssteuerungsanlagen in der Energiewirtschaft existieren dazu auch entsprechende Empfehlungen (siehe [18, Abschn. 12.4.4]).

Generell besteht das Informationsmodell von Zeitreihen aus mindestens zwei Klassen, die über eine Assoziation miteinander verbunden sind. Eine Klasse repräsentiert dabei die Zeitreihe selbst (z. B. `TimeSeries` oder `Curve`) und beinhaltet Attribute wie den Namen der Reihe, die physikalische Einheit der Daten oder Qualitätsmerkmale. Eine zweite Klasse repräsentiert dann einzelne Datenpunkte (z. B. `CurveData` oder `Point`). Das Informationsmodell macht meist noch weitere Vorgaben, z. B. ob es sich um eine äquidistante oder nichtäquidistante Zeitreihe handelt. Bei äquidistanten Zeitreihen sind die zeitlichen Abstände zwischen den Messpunkten immer gleich. Für eine äquidistante Zeitreihe müssen daher nur der Anfangszeitpunkt und eine Sequenznummer für die Messungen gespeichert werden. Auch für Interpolation der Werte zwischen zwei Zeitpunkten existieren unterschiedliche Möglichkeiten. Ein Wert kann z. B. bis zum Auftreten des nächsten Wertes Gültigkeit besitzen (Treppenstufen) oder es kann linear interpoliert werden (Liniendiagramm). Darüber hinaus existieren aber viele weitere Möglichkeiten. Welche Interpretation hier gewählt wird, muss zusätzlich festgelegt werden.

Um zeitreihenbasierte Energiedaten abzubilden, finden sich auch einige Klassen im CIM (siehe Abschn. 3.1). Das Kernpaket des IEC 61970 beinhaltet einige Klassen, um allgemeine Zeitreihen oder Fahrpläne abzubilden. Eine einfache Datenreihe oh-

ne weiteren Bezug zu anderen Objekten wird z. B. durch die Klassen `Curve` und `CurveData` definiert. Dadurch kann eine Reihe von Gleitkommazahlen mit bis zu zwei Y- und einer X-Achse abgebildet werden. Echte Zeitreihen, also Werte mit einem Zeitbezug, können anhand der Klassen `IrrgularIntervalSchedule` und `RegularIntervalSchedule` dargestellt werden. Eine äquidistante Zeitreihe wird dabei durch ein `RegularIntervalSchedule` repräsentiert. Abb. 3.18 zeigt die Typhierarchie für äquidistante Zeitreihen im Kernmodell des CIM.

Das Kernmodell von CIM wurde durch IEC 62325 so erweitert, dass auch Energiemärkte und -produkte abgebildet werden können (siehe Abschn. 3.1). In diesem Kontext müssen häufig Zeitreihen, z. B. Gebote, Preise oder Fahrpläne zwischen Handelspartnern ausgetauscht werden. Als Teil des ESMP sind im Paket `MarketManagement` des CIM dazu einige Klassen vorhanden. Ein Objekt der Klasse `MarketDocument` repräsentiert einen Geschäftsvorfall, z. B. ein Angebot, welches übermittelt werden kann. Das `MarketDocument` kann aus einer oder mehreren Zeitreihen in Form von `TimeSeries`-Objekten bestehen. Eine Zeitreihe besitzt beliebig viele Datenpunkte (Klasse `Point`) in einer oder mehreren Perioden. Abb. 3.19 zeigt den Zusammenhang zwischen den Klassen `MarketDocument`, `TimeSeries` und `Point` im IEC 62325.

Damit beim Datenaustausch von Marktdokumenten die Zeitreihen korrekt interpretiert werden können, wurden durch die entso-e verschiedene **Kurventypen** definiert (siehe [19]). Der Kurventyp der Zeitreihe wird im Attribut `CurveType` der Klasse `TimeSeries` als Zeichenkette festgelegt. Tab. 3.1 zeigt die Bedeutung der fünf unterschiedlichen Kurventypen.

Für die Speicherung von Zeitreihen bieten sich erneut relationale Datenbanksysteme an. Im einfachsten Fall kann dazu das Informationsmodell aus Abb. 3.18 bzw. Abb. 3.19 auf das Tabellschema der Datenbank abgebildet werden. Dabei entsteht mindestens eine Tabelle, um die Metadaten der Zeitreihen zu verwalten, und eine weitere, um die Zeitreihenwerte zeilenweise aufzunehmen. Die zeilenweise Speicherung der Zeitreihenwerte hat

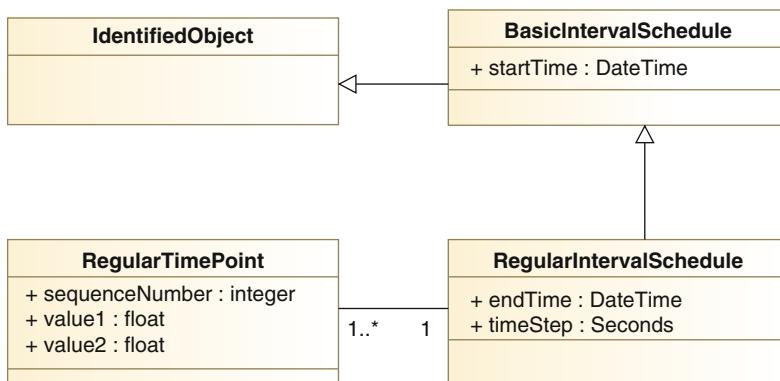


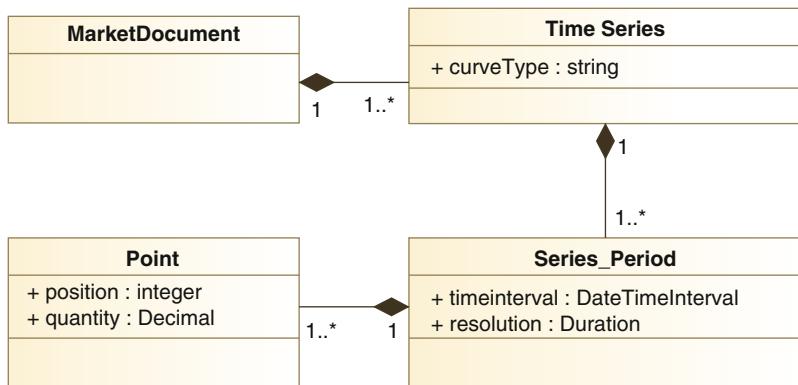
Abb. 3.18 Typhierarchie einer äquidistanten Zeitreihe im CIM-Basismodell

Tab. 3.1 Die Bedeutung der Kurventypen von Zeitreihen des entso-e-ESMP-Profil

Kurventyp	Bedeutung
A01	Die Zeitreihenwerte sind äquidistant und über die gesamte Periode gültig. Dies führt in der Darstellung zu einer Art Treppenstufendiagramm. Der Typ A01 ist der Standardtyp
A02	Die Zeitreihenwerte sind äquidistant und nur für den gegebenen Zeitpunkt gültig
A03	Der Typ A03 ist ähnlich zu A01, nur müssen die Werte nicht äquidistant sein
A04	Der Typ A04 kann nichtäquidistante Werte besitzen. Zwischen zwei Werten findet eine lineare Interpolation statt. Um Sprünge abbilden zu können, sind auch zwei Werte zum selben Zeitpunkt erlaubt
A05	Der Typ A05 ist ähnlich zu A04, es sind aber keine Sprünge erlaubt

dabei den Vorteil, dass mithilfe der SQL-eigenen Mechanismen leicht Aggregate, wie z. B. der Durchschnittswert oder das Minimum/Maximum, für einzelne Zeitperioden berechnet werden können. Durch die weite Verbreitung relationaler Datenbanken ist zudem die Integration in Anwendungssysteme sehr einfach möglich. In der Programmiersprache Python werden nur wenige Anweisungen benötigt, um Zeitreihendaten aus einer PostgreSQL-Datenbank zu lesen und als grafischen Plot auszugeben.

Beispiel 3.3.1 In einer lokalen PostgreSQL-Datenbank mit dem Namen `energieinformatik` wurden in der Tabelle `wettermessung` Temperaturdaten des Deutschen Wetterdienstes abgelegt. Das folgende Python-Script lädt die Wetterdaten für Düsseldorf aus der Datenbank und erzeugt einen Plot über die minimale und maximale Tagestemperatur:

**Abb. 3.19** Zusammenhang zwischen den Klassen MarketDocument, TimeSeries und Point im IEC 62325

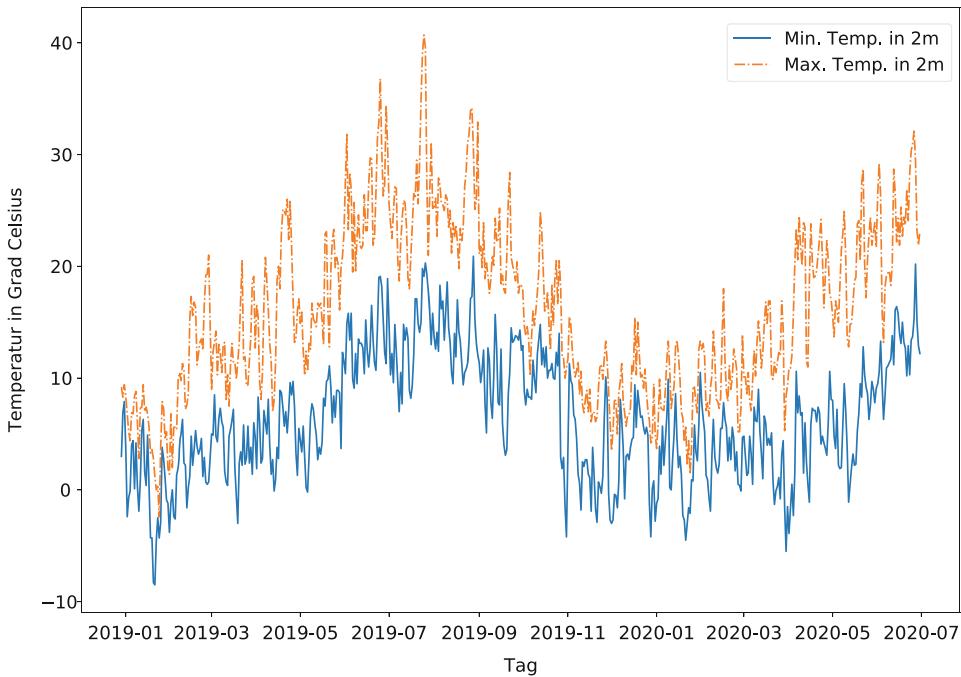


Abb. 3.20 Der Verlauf der höchsten und niedrigsten Tagestemperaturen in Düsseldorf als Ergebnis einer Abfrage aus einem Python-Script

```

1 import pandas as pd
2 from sqlalchemy import create_engine
3 import matplotlib.pyplot as plt
4 engine = create_engine('postgresql://postgres:geheim@localhost
   :5432/energieinformatik')
5 sql = 'select * from wettermessung where stations_id=1078 order by
   Datum;'
6 df = pd.read_sql(sql, engine)
7 df.plot(x='datum', y=['min_2m', 'max_2m'])
8 plt.show()

```

Abb. 3.20 zeigt das Ergebnis des Plots.

Die Speicherung von Zeitreihendaten in einer relationalen Datenbank hat einige Vor-, aber auch Nachteile. Relationale Datenbanken sind in Unternehmen weitverbreitet. Viele MitarbeiterInnen verfügen daher über die Fähigkeiten, mithilfe von SQL mit solchen Datenquellen umzugehen. Relationale Datenbanken integrieren sich zudem leicht in viele Systeme, um die Daten weiter verarbeiten zu können (z. B. Python, Excel oder R). Relationale Datenbanken zeigen allerdings bei der Verwaltung großer Mengen von Zeitreihendaten aus unterschiedlichen Gründen mitunter eine eher unbefriedigende

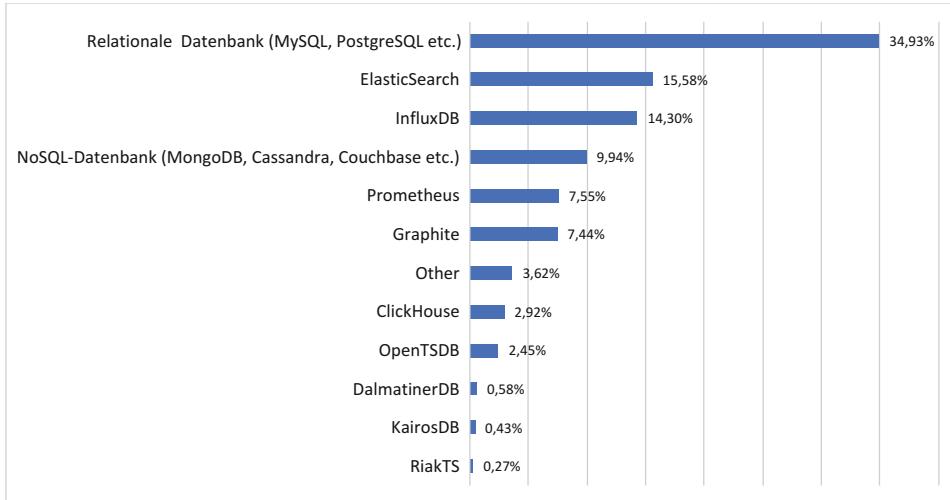


Abb. 3.21 Nutzung verschiedener Datenbanksysteme für die Verwaltung von Zeitreihendaten

Leistung. Zeitreihendaten haben ein anderes Zugriffsmuster als transaktionale Daten. Einmal gespeicherte Zeitreihendaten werden meist kaum wieder geändert oder gelöscht. Der Datenbestand wächst meist aber kontinuierlich weiter, wodurch die Tabellen, welche die Werte der Zeitreihen speichern, schnell vergleichsweise groß werden. Der Zugriff auf die Daten erfolgt zudem meist in zeitlichen Blöcken. Für die Speicherung und Verarbeitung von Zeitreihendaten bieten sich verteilte Systeme und Programmiermodelle, wie z. B. MapReduce, an (siehe [3]). Relationale Datenbanken können allerdings nicht gut auf mehrere Rechner verteilt werden, da sie Datenkonsistenz anstreben (siehe [1]).

Die zuletzt genannten Gründe sorgen dafür, dass Zeitreihendaten zunehmend in speziellen **Zeitreihendatenbanken** verwaltet werden. Eine Umfrage der Firma Persona LLC unter ca. 1500 Teilnehmern bestätigt diesen Trend (siehe [32]). Relationale Datenbanken werden von den Teilnehmern der Umfrage in nur noch 35 Prozent der Fälle eingesetzt (siehe Abb. 3.21).

Graphite ist ein Datenbanksystem, welches speziell auf Zeitreihendaten ausgelegt ist (siehe [27]). Es wurde ursprünglich für die Überwachung von Serverinfrastrukturen entwickelt, da auch in diesem Kontext viele Zeitreihendaten, wie z. B. die CPU-Auslastung oder der Speicherverbrauch, aufgezeichnet und überwacht werden müssen. Graphite nutzt für die Speicherung der Daten ein eigenes Persistenz-Backend mit dem Namen *Carbon*, welches die Daten in Binärdateien speichert. Graphite stellt zudem ein webbasiertes Frontend bereit, um auf die Daten zuzugreifen.

Eine Zeitreihe wird in Graphite als **Metrik** bezeichnet. Eine Metrik hat einen eindeutigen Namen, der aus hierarchisch gegliederten Teilen bestehen kann, z. B. `energy.power.consumption`. Als zusätzliche Metadaten können einer Metrik beliebig viele *Tags* angehängt werden. Als Tag wird dabei ein Schlüssel-Wert-Paar

Tab. 3.2 Beispiele für Graphite Functions

Function	Bedeutung
diffSeries	Erzeugt eine neue Zeitreihe als Differenz zweier Metriken
aggregate	Aggregiert eine Metrik auf Basis einer Aggregationsfunktion wie sum(), min() oder max()
exponentialMovingAverage	Berechnet den gleitenden Mittelwert über eine Metrik
holtWintersForecast	Berechnet eine Prognose für die Metrik mithilfe der sog. exponentiellen Glättung (siehe Abschn. 4.1)

bezeichnet, z. B. `User=123` oder `power_station=99`. In der Konfiguration müssen für jede Metrik die zeitliche Granularität und die Aufbewahrungsfrist (*Retention*) der Daten definiert werden. Eine Zeitreihe kann z. B. für die letzten 10 Tage eine minütliche Granularität besitzen, danach sollen nur noch stündliche Werte gespeichert werden. Rutschende Werte aus einem Zeitfenster, werden sie automatisch aggregiert. Bei der Aggregationsmethode kann zwischen Summe und Durchschnitt gewählt werden.

Graphite kann ausschließlich auf Linux-Plattformen betrieben werden. Am einfachsten ist die Installation über den offiziellen **Docker**-Container (siehe [25]). Nach der Installation ist die Weboberfläche unter `http://localhost` erreichbar. Daten können auf vielfältige Weise in Graphite importiert werden. Am einfachsten gelingt dies auf der Konsole mit dem Programm `netcat`.

Beispiel 3.3.2 Im folgenden Beispiel wird einer lokalen Graphite-Instanz auf Port 2003 der Wert 4 für den aktuellen Zeitpunkt an die Metrik `energy.power.consumption` übergeben.

```
1 echo "energy.power.consumption 4 `date +%s'" | nc localhost 2003
```

Neben Netcat stehen aber viele weitere Werkzeuge und APIs bereit. In der Weboberfläche können Metriken in Form von Graphen visualisiert werden. Jeder Graph wird auch als Bilddatei unter einer eindeutigen URL bereit gestellt, sodass dieser auch in anderen Umgebungen integriert werden kann.

Eine interessante Funktionalität von Graphite sind die sog. *Functions*, die dazu genutzt werden können, Zeitreihendaten zu transformieren, zu aggregieren oder sonstige Berechnungen auf ihnen durchzuführen. Functions können in der Weboberfläche mithilfe des *Graphite Composer* auf Metriken angewandt werden, sie können aber auch per Hand an die Web-URL eines Graphen angehängt werden. Abb. 3.22 zeigt die Anwendung einiger Functions in der Weboberfläche von Graphite.

Einige Beispiele für Graphite Functions sind in Tab. 3.2 dargestellt. Eine Übersicht über alle Funktionen findet sich in der Dokumentation des Systems (siehe [28]).

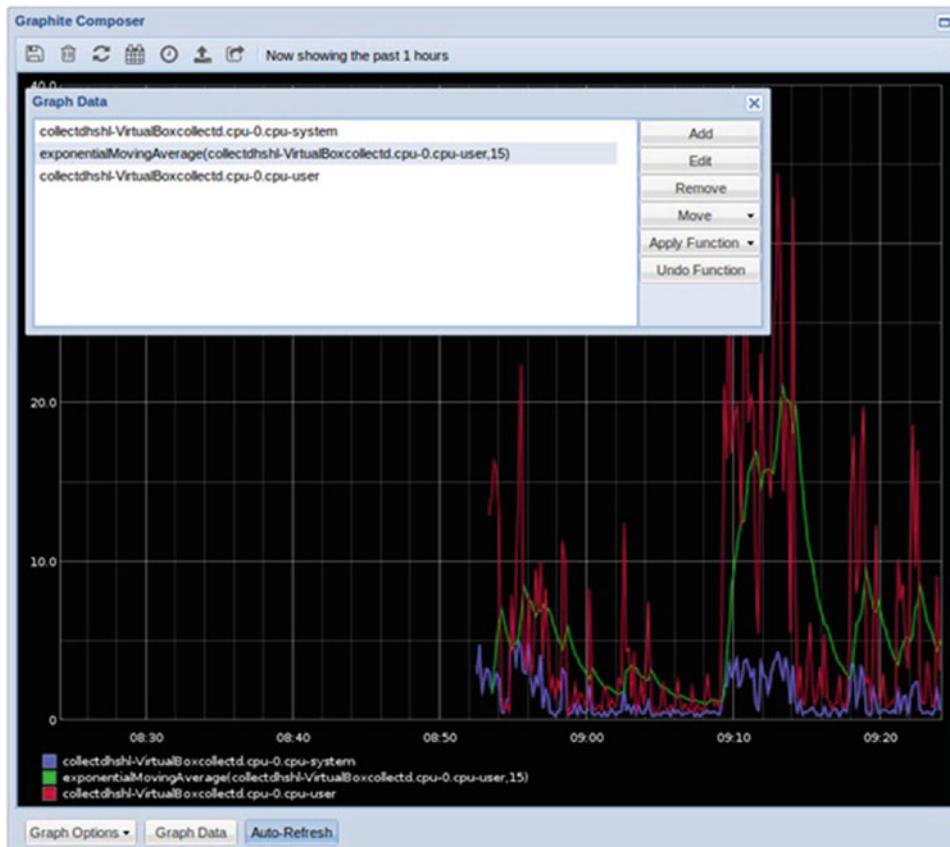


Abb. 3.22 Anwendung von Graphite Functions auf eine Metrik

OpenTSDB ist ein weiteres Datenbanksystem, welches speziell der Verarbeitung von Zeitreihendaten dient (siehe [30]). Auch OpenTSDB ist Open Source und wird z. B. bei Unternehmen wie Yahoo oder Salesforce dazu genutzt, um die Serverinfrastruktur zu überwachen. OpenTSDB organisiert Zeitreihen wie Graphite als Metriken, die ebenfalls hierarchisch aufgebaute Namen besitzen, z. B. `energy.consumption.electric`. Metriken besitzen auch hier mindestens einen Tag, z. B. `customer=1`. Es existieren aber auch einige Unterschiede zu Graphite. In OpenTSDB muss keine Granularität für Zeitreihen festgelegt werden. Daten werden nicht automatisch aggregiert, sondern im Rohformat dauerhaft gespeichert. Im Gegensatz zu Graphite nutzt OpenTSDB das System Apache HBASE als unterliegendes Speichersystem. Apache HBASE ist ein hoch verteilbares Speichersystem auf Basis von Hadoop und HDFS und kann auf ansatzweise beliebig viele Rechner verteilt werden. Dadurch skaliert das System linear durch Hinzufügen weiterer Knoten.

Um OpenTSDB betreiben zu können, wird zunächst ein HBase-System benötigt, dessen Installation sich mitunter als etwas schwierig darstellt. Am einfachsten kann OpenTSDB daher auf Basis eines vorkonfigurierten Docker-Containers betrieben werden (siehe [31]). Daten können in OpenTSDB auf unterschiedlichem Weg importiert werden. Es kann z. B. eine Telnet-Verbindung mit einem OpenTSDB-Server auf dem Port 4242 aufgebaut werden, wodurch dann einzelne Datenpunkte mit dem PUT-Kommando abgesetzt werden können:

```
1 put energy.electric 1356998400 42.5 customer=01
```

Mit der Konsolenanwendung `import` steht ein Werkzeug bereit, um große Datens Mengen im Batch zu importieren. Der einfachste Weg, um Daten von OpenTSDB zu lesen, ist, die existierende HTTP-Schnittstelle zu nutzen. Mithilfe eigener GET-Anfragen können Daten einer bestimmten Metrik abgerufen werden. Dem Request können dabei Zusatzinformationen zu Zeiträumen, Tags und Aggregationen mitgegeben werden, z. B.:

```
1 http://localhost:4242/api/query?start=5h-ago&m=sum:energy.electric
```

Das Resultat der Anfrage wird dann in Form von JSON-codiertem Text zurückgegeben:

```
1 [ {  
2   "metric": "energy.electric",  
3   "tags": {"customer": "1"},  
4   "aggregateTags": [],  
5   "dps": {"1539845880": 20, "1539845940": 30}  
6 } ]
```

In den meisten EDM-Systemen müssen Zeitreihendaten visualisiert werden. Viele Eigenschaften der Zeitreihen, z. B. Trend, Zyklen oder Ausreißer, lassen sich in Form von Charts oder Graphen sehr schnell erkennen. Einige der vorgestellten Systeme bieten dazu bereits rudimentäre Funktionen an. Neben Excel und R bieten auch Graphite und OpenTSDB entsprechende Funktionalität an.

Ein Standardwerkzeug zur Visualisierung von Zeitreihendaten ist **Grafana** (siehe [26]). Grafana ist ein Open-Source-System, kann auf allen Standardbetriebssystemen betrieben werden und verfügt über eine webbasierte Oberfläche. Grafana kann Daten auf 39 verschiedene Arten darstellen, z. B. als Liniendiagramm, Tabelle, Kuchendiagramm, Heatmap und viele mehr. Die unterschiedlichen Darstellungen werden in *Dashboards* zu einem Bericht zusammengefasst. In einem Dashboard kann eine beliebige Anzahl von Visualisierungen angelegt und organisiert werden. Grafana kann dabei aktuell 47 verschiedene Datenquellen einbinden, u. a. auch relationale Datenbanksysteme wie PostgreSQL, aber auch Graphite oder OpenTSDB. In einem gemeinsamen Dashboard

können dabei Daten aus unterschiedlichen Quellen miteinander kombiniert werden. Je nach Datenquellen müssen für eine Darstellung unterschiedliche Parameter vorgegeben werden. Für Zeitreihendatenbanken, wie z. B. Graphite oder OpenTSDB, müssen z. B. die Namen der Metriken, Tags und ggf. Funktionen übergeben werden.

Ist die Datenquelle eine relationale Datenbank, wie z. B. PostgreSQL, wird eine SQL-Abfrage benötigt. Das Ergebnis der SQL-Abfrage muss dabei mindestens zwei Datenspalten besitzen. Die erste Spalte muss den Namen `time` tragen und die Zeitpunkte der Zeitreihe in Form von Unix-Zeitstempeln beinhalten. Alle weiteren Spalten werden dann als Datenspalten behandelt, die jeweils einzeln z. B. als Graph dargestellt werden. Die Daten müssen zudem aufsteigend nach den Zeitpunkten sortiert sein. Für die Nutzung in den SQL-Abfragen bietet Grafana einige Makros an, die sonst nicht zum Umfang von SQL gehören. Das Makro `\$__timeFilter()` dient beispielsweise dazu, dass der in der Grafana-Oberfläche ausgewählte Datumsbereich an die SQL-Abfrage übergeben wird und entsprechende Daten geladen werden, sobald der Datumsbereich geändert wird.

Beispiel 3.3.3 Eine Grafana-Abfrage, um die Wetterdaten aus dem vorhergehenden Beispiel abzurufen, sieht wie folgt aus:

```
1 select \$__time(Datum), Min_2m, Mittel_2m, Max_2m from
      wettermessung
2 where Stations_ID=1078 and \$__timeFilter(Datum) order by Datum;
```

Abb. 3.23 zeigt das Ergebnis in einem Grafana-Dashboard. Am oberen rechten Rand ist auch der aktuell ausgewählte Datumsbereich zu sehen, der für alle Visualisierungen in diesem Dashboard gültig ist.

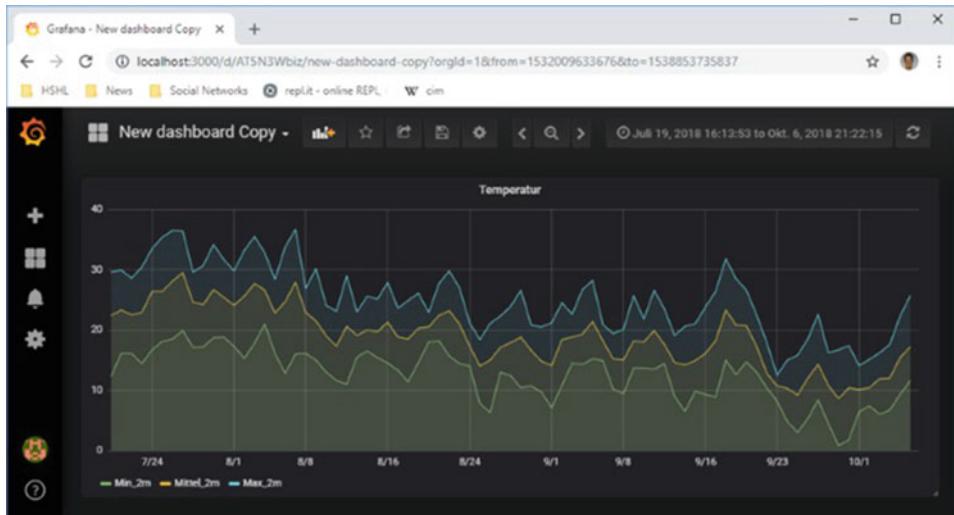


Abb. 3.23 Temperatur für Düsseldorf als Liniendiagramm in Grafana

3.4 Datenquellen

Daten werden nicht nur organisationsintern benötigt. Daten zur Struktur des Energiesystems liefern wichtige Basisinformationen für die Erstellung von Prognose- und Simulationsmodellen. Eine kurzfristige Strompreisprognose kann heute nur sinnvolle Ergebnisse liefern, wenn z. B. die Windeinspeisung berücksichtigt wird. Dazu muss aber die installierte Leistung von Windkraftanlagen in einem geografischen Areal bekannt sein. Dann kann diese Information mit einer entsprechenden Wetterprognose verknüpft werden. Energiebezogene Stamm- und Bewegungsdaten müssen also frei zugänglich und maschinell zu verarbeiten sein. Der Wert solcher Daten wurde mittlerweile von mehreren Institutionen erkannt. Im Folgenden sollen mehrere Beispiele für Datenquellen dargestellt werden.

Das Marktstammdatenregister (MaStR) ist ein umfassendes behördliches Register des Strom- und Gasmarktes (siehe [14]), welches von der Bundesnetzagentur auf Basis von § 111e und § 111f des Energiewirtschaftsgesetzes (EnWG) und der Verordnung der Registrierung energiewirtschaftlicher Daten (MaStRV) betrieben wird. Das MaStR ist ein webbasiertes System, welches im Dezember 2018 online gegangen ist (siehe [29]). Im MaStR sind vor allem die Stammdaten zu Strom- und Gaserzeugungsanlagen zu registrieren. Darüber hinaus sind Angaben zu Marktakteuren, wie Anlagenbetreibern, Netzbetreibern und Energielieferanten, zu finden. Bewegungsdaten, die mit der energiewirtschaftlichen Aktivität eines Marktakteurs oder den Vorgängen innerhalb von Anlagen verbunden sind, werden dort nicht erfasst. Anlagenbetreiber sind dazu verpflichtet ihre Anlagen dort selbst zu registrieren und die Daten aktuell zu halten. Für bestehende Anlagen existiert hier eine Frist bis zum 31.01.2021, neue Anlagen müssen dort innerhalb von einem Monat registriert werden. Die Daten stehen unter der Datenlizenz Deutschland 2.0 und können frei, auch für kommerzielle Zwecke, genutzt werden.

Die Daten des MaStR basieren auf einem eigenen Informationsmodell, welches nicht durch ein Klassendiagramm, sondern durch ein Objektdiagramm dokumentiert wird. Abb. 3.24 zeigt das Informationsmodell des Marktstammdatenregisters aus der offiziellen Dokumentation.

Die Attribute und Datentypen der Objekte werden in mehreren Excel-Sheets weiter spezifiziert. Im Mittelpunkt stehen die drei Klassen Marktakteur, Einheit und Netz. Diese Klassen gehen dann weitere Beziehungen ein, z. B. zu Bilanzierungsgebiet, Dokument oder Ticket. Das Modell ist dabei sehr viel weniger detailreich als CIM. Ein Netz wird nicht anhand seiner Topologie beschrieben. Auch für Erzeugungseinheiten werden nur wenige grundlegende Attribute eingeführt. Der geografische Ort einer Einheit wird durch ein zugeordnetes Objekt vom Typ Lokation beschrieben.

Für die automatisierte und maschinenbasierte Abfrage und Eintragung der Daten wird eine Programmierschnittstelle (*Application Programming Interface (API)*) angeboten. Die Schnittstelle bietet über 50 Funktionen für den lesenden Zugriff auf Akteure und Anlagen.

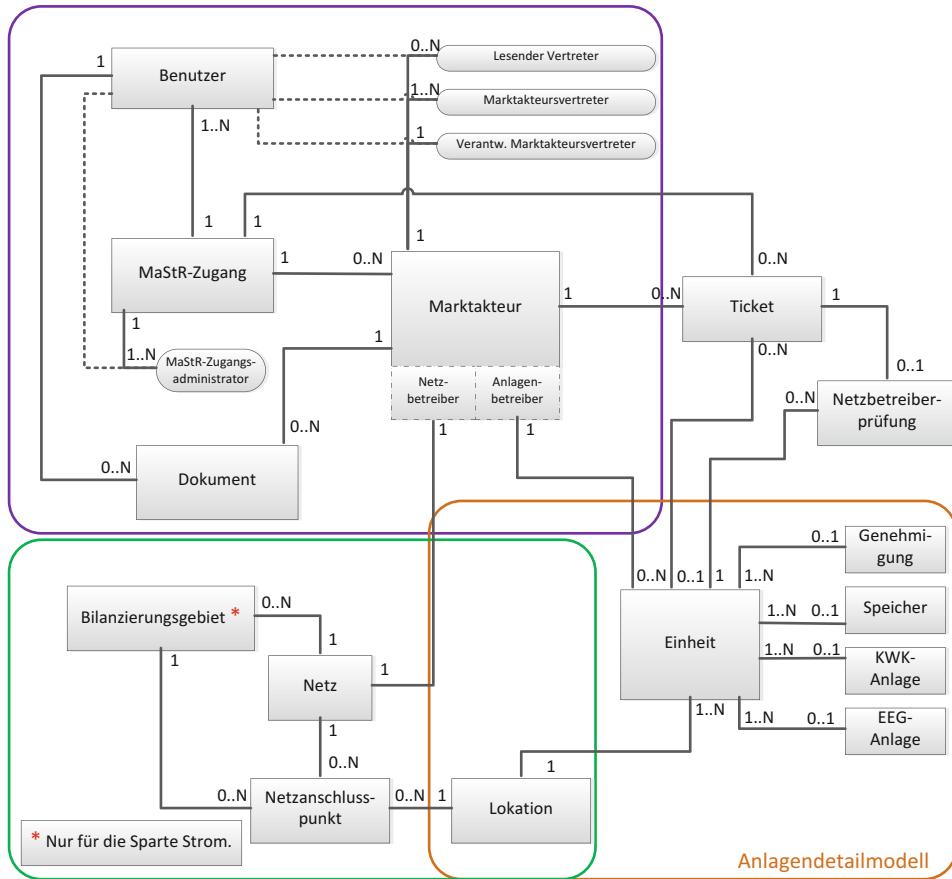


Abb. 3.24 Informationsmodell des Marktstammdatenregisters. Bildquelle: Dokumentation des MaStr unter <https://www.marktstammdatenregister.de>

Für Netzbetreiber werden weitere 37 Funktionen zur Eintragung von Daten angeboten. Für die Abfrage der Daten ist eine Registrierung notwendig. Dabei wird ein sog. API-Key, ein 512 Zeichen langer benutzereindeutiger Schlüssel vergeben, der zur Authentifizierung genutzt wird. Technisch basiert die Schnittstelle auf dem sog. SOA-Standard, bei dem die Daten XML-basiert über HTTP-Aufrufe ausgetauscht werden. Die Schnittstelle wird durch eine XML-basierte Datei im sog. Web-Service-Description-Language-(WSDL-)Format spezifiziert. Diese Datei beschreibt alle Funktionen mitsamt ihren Parametern und Ergebnistypen, welche durch die API zur Verfügung gestellt werden. Moderne Programmiersprachen können mithilfe der WSDL-Datei Serviceendpunkte generieren, z. B. Visual Studio mit C#. Die Schnittstelle kann dann leicht in ein eigenes System

eingebunden werden. Der Webdienst des MaStR ist auf den Webseiten des Marktstamm-datenregisters ausführlich beschrieben.

Das Datenmodell des MaStR wurde offenbar spezifisch neu entwickelt. Ein Standard wie CIM wurde nicht als Basis genutzt. Hier wäre der Rückgriff auf CIM mit der Extraktion eines eigenen Profils evtl. die bessere Alternative gewesen. Die Spezifikation des Datenmodells durch das Objektdiagramm ist weder eindeutig noch vollständig. Es wird z. B. nicht direkt klar, ob eine Stromerzeugungseinheit aus mehreren anderen Einheiten bestehen kann. Da die Daten allerdings ausschließlich über die angebotene Schnittstelle veröffentlicht werden, ergibt sich durch die WSDL-Datei eine eindeutige Spezifikation der Daten. Die Qualität der Daten ist zum aktuellen Zeitpunkt (März 2019) noch recht überschaubar. Viele Einträge sind noch nicht durch die Netzbetreiber geprüft und enthalten offensichtlich falsche Angaben. Eine verlässliche Datenquelle kann das MaStR wohl erst nach Ablauf der Eintragungsfristen bilden.

Eine weitere öffentlich zugängliche Quelle von Anlagendaten wird durch die Open Power Systems Data Initiative (OPSD) bereitgestellt. OPSD wird durch ein Konsortium aus mehrere Forschungseinrichtungen, darunter Neon (Neue Energieökonomik), der Technischen Universität Berlin, der ETH Zürich und anderen betrieben. Das Ziel der Initiative ist es, Stamm- und Bewegungsdaten für die Erstellung von Simulations- und Prognosemodellen zur Verfügung zu stellen. Eine größere Anzahl von wissenschaftlichen Veröffentlichungen bezieht sich dabei explizit auf OPSD als Datenquelle. Zurzeit sind auf den Webseiten von OPSD verschiedene Pakete von Stamm- und Bewegungsdaten zu finden, z. B.:

- Daten zu konventionellen und regenerativen Erzeugungseinheiten in Europa,
- Lastdaten,
- Wind- und Solareinspeisung,
- Wetterdaten und Marktpreise.

Die Daten werden von OPSD zu sog. Datenpaketen (*Data Package*) zusammengefasst. Ein solches Paket besteht i. d. R. aus einem ZIP-Archiv, welches neben den eigentlichen Nutzdaten (meist im CSV-Format) zusätzliche Metadaten beinhaltet. Diese Metadaten stellen eine genaue Spezifikation der Nutzdaten dar und basieren auf dem sog. Data-Package-Format. Das Data-Package -format wurde durch die *Frictionless Data Initiative* definiert und dient dem Austausch offener Daten (*Open Data*). Die Metadaten werden dabei in einer JSON-Datei serialisiert.

Die Datenpakete können auf unterschiedliche Arten bezogen werden. Eine einfache Variante ist der direkte Download der Pakete von den OPSD-Seiten. Darüber hinaus existiert zu jedem Data Package auch ein Python-Script, welches die Daten aus den Originalquellen bezieht und entsprechend aufbereitet. Dadurch kann der Datenimport auch in eigene Prozesse integriert werden.

Die Vorgehensweise, die bei der OPSD gewählt wurde, um Daten aus externen Quellen zu sammeln und aufzubereiten, basiert auf offenen Standards, wie dem Data-Package-

Format. Dies erleichtert die automatisierte Verarbeitung und stellt wichtige Metadaten in einem geeigneten Format bereit. Durch die zur Verfügung gestellten Python-Skripte sind der Import und die Aggregation der Daten transparent und können in lokale Prozesse integriert werden. Die Datenqualität ist als hoch zu bewerten, die Anzahl der hochwertigen Veröffentlichungen, die auf OPSD-Daten basieren, gibt hierfür ein gewisses Indiz. Fraglich ist mitunter die Lizenz der Daten. Die Daten stammen ursprünglich aus diversen Quellen, z. B. von der Bundesnetzagentur, dem Umweltbundesamt und diversen Unternehmen. Im Zweifel muss hier vor einer Verwendung die Lizenz der Daten genau geprüft werden.

Seit Mitte 2017 veröffentlicht die Bundesnetzagentur auf ihrer Webseite mit dem Namen SMARD zu Transparenzzwecken eine Vielzahl von Strommarktdaten. Diese Strommarktdaten sind Bewegungsdaten in Form von Zeitreihen. Die Daten stammen ursprünglich von entso-e, dem Verband der europäischen Übertragungsnetzbetreiber, und besitzen insgesamt eine hohe Qualität. Die Daten stehen unter der Creative-Commons-Namensnennung-4.0-International-Lizenz und können unter Nennung der Quelle auch kommerziell verwendet werden. Folgende Daten sind, teilweise seit Anfang 2015 verfügbar:

- realisierte und prognostizierte Erzeugung nach Erzeugungsart getrennt,
- installierte Erzeugungsleitung nach Erzeugungsart getrennt,
- realisierter und prognostizierter Stromverbrauch,
- Großhandelspreise, kommerzieller Außenhandel und physikalischer Stromfluss für unterschiedliche Marktgebiete,
- Daten zum Thema Systemstabilität, wie z. B. die Regelleistungsmenge verschiedener Qualitäten, Ausgleichsenergie und die Gesamtosten für diese Systemleistungen.

Die Daten können online als Graph visualisiert oder in Rohform als CSV- oder Excel-Dateien heruntergeladen werden.

Um eine räumliche Bewertung von Energieanlagen durchführen zu können, werden auch offene geografische Grunddaten benötigt, z. B. Verläufe von Flüssen, Gebirgen, Länder- und Gebietsgrenzen oder ganze Landkarten. Auch für solche Daten stehen einige Datenquellen zur Verfügung, die frei genutzt werden können. Das Bundesamt für Kartographie und Geodäsie betreibt beispielsweise eine Open-Data-Plattform, über die viele Daten in Form von Shape Files heruntergeladen werden können. Auch ESRI, der Hersteller von ArcGis, betreibt eine Open-Data-Plattform, über die viele Daten frei (Open Data Datenlizenz Deutschland) zugänglich sind. Eine der bekanntesten Plattformen für frei zugängliche Geodaten ist aber sicherlich OpenStreetMap.

OpenStreetMap (OSM) ist ein Projekt, welches darauf abzielt, Geodaten zu sammeln und zur freien Verfügung bereitzustellen. Es soll dabei ein möglichst realistisches und detailliertes Abbild der physischen Welt geschaffen werden. Die Daten umfassen dabei die genauen Standorte und Verläufe von Gebäuden, Straßen, Flüssen und vielen anderen Objekten. Die Daten stehen unter der sog. Open Data Commons Open Database License

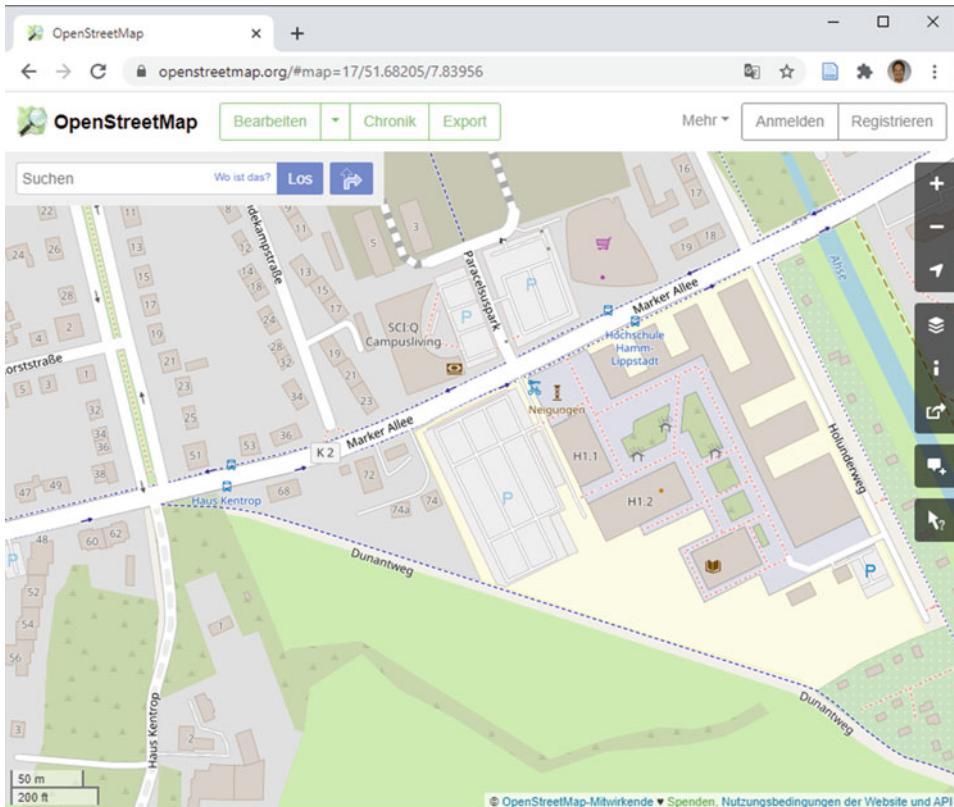


Abb. 3.25 Eine Kartendarstellung der Daten von OpenStreetMap im Browser

(ODbL) und können unter Nennung der Quelle auch kommerziell genutzt werden. Die Daten werden durch Freiwillige gesammelt und bearbeitet (sog. *Crowdsourcing*) und können in Form von Rohdaten oder als Kartenmaterial in Grafikform verwendet werden. Abb. 3.25 zeigt die typische Kartenansicht einer OpenStreetMap-Karte in einem Webbrowser.

Die Rohdaten von OpenStreetMap basieren dabei auf einem relativ einfachen Informationsmodell. Es besteht aus lediglich vier Grundelementen Node, Way, Relation und Tag. Ein Objekt vom Typ Node repräsentiert einen Punkt auf der Erdoberfläche. Er wird durch einen Längen- und Breitengrad und eine eindeutige ID definiert. Ein Way ist eine geordnete Liste von Nodes (sog. Polyline). Er dient z. B. dazu, Flüsse oder Straßen bzw. die Begrenzung von Flächen zu beschreiben. Eine Relation ist eine Mehrzweckstruktur, die dazu dient, mehrere Nodes und/oder Ways miteinander in ein Verhältnis zu setzen, um dadurch z. B. komplexe Gebilde abbilden zu können. Dabei kann jedes Kindelement der Relation eine bestimmte Rolle einnehmen, z. B. ein Way als Außenhülle eines Gebäudes. Nodes, Ways und Relations werden durch Tags weiter

beschrieben. Sie definieren die Bedeutung von Nodes, Ways oder Relations, indem diese Elemente um Schlüssel-Wert-Paare erweitert werden. Ein Node kann z. B. durch `amenity=cafe` als Café gekennzeichnet werden. Tags können jederzeit neu definiert werden, es existiert keine feste Definition, welche Tags zu benutzen sind.

OSM-Daten werden üblicherweise in einem eigenen XML-Format repräsentiert und ausgetauscht. In diesem Format kann auch der Gesamtbestand der Daten (aktuell ca. 76 Gb) heruntergeladen werden. Die folgenden OSM-Daten repräsentieren das Mensagebäude der Hochschule Hamm-Lippstadt in Hamm als Objekt vom Typ Way als OSM-Daten.

```
1 <way id="237712560" visible="true">
2   <nd ref="2455904215"/>
3   <nd ref="2537194799"/>
4   <nd ref="3911092686"/>
5   <nd ref="2455904238"/>
6   <nd ref="2455904207"/>
7   <nd ref="3911092678"/>
8   <nd ref="2455904243"/>
9   <nd ref="3911092681"/>
10  <nd ref="2455904215"/>
11  <tag k="building" v="university"/>
12  <tag k="description" v="Mensa, Campus Office, Seminarraeume"/>
13  <tag k="name" v="H1.2"/>
14 </way>
```

Für die Verarbeitung solcher OSM-Daten existieren viele hilfreiche Werkzeuge. OSM-Daten können z. B. leicht in andere geografische Datenformate umgewandelt (z. B. ShapeFile oder GeoJSON) oder in eine PostGIS-Datenbank importiert werden.

Wie wir in Abschn. 3.2 gesehen haben, erlauben räumliche Datenbanken, räumliche Daten nach bestimmten Kriterien abzufragen oder zu aggregieren. Mit der *Overpass API* steht eine serverbasierte Möglichkeit auch für die Daten von OpenStreetMap zur Verfügung. Mit der Overpass API können alle Elemente (Nodes, Ways, Relation) sehr gezielt abgefragt, gefiltert und in fast beliebigen Formaten exportiert werden. Das folgende Listing zeigt ein Beispiel einer solchen Abfrage, die alle Elemente zurückliefert, die mit dem Power-Tag ausgezeichnet sind.

```
1 [bbox:{bbox}];
2 ( node["power"];
3   way["power"];
4   relation["power"]; );
5 out center;
```

Mit der Webanwendung Overpass-Turbo können solche Abfragen getestet und die Ergebnisse visualisiert werden. Mithilfe des Plug-ins QuickOSM kann Overpass auch

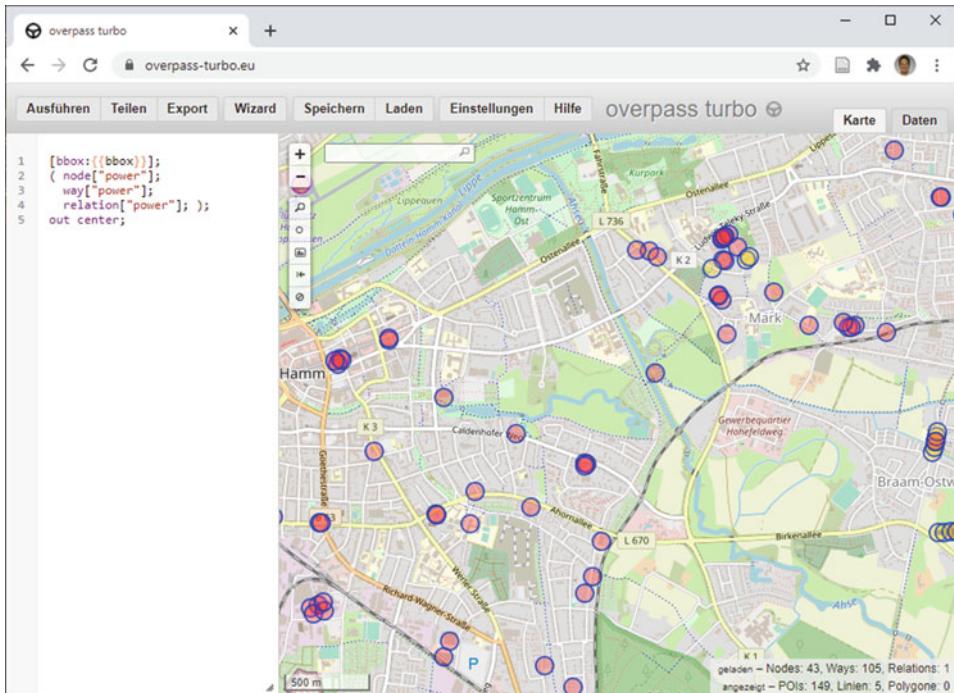


Abb. 3.26 Das Ergebnis der Abfrage aller Elemente mit dem Power-Tag in Hamm mithilfe der Overpass API

direkt in QGis integriert werden. Abb. 3.26 zeigt das Ergebnis für die zuvor gezeigte Abfrage für die Gegend um die Stadt Hamm.

OpenStreetMap beinhaltet auch Standortdaten zu einigen Anlagen der Energieinfrastruktur. Mithilfe des Power-Tags können Objekte z. B. als Strommast (Tower), Umspannwerk (Substation) oder Kraftwerk (Plant) ausgewiesen werden. Aktuell umfasst die Datenbasis von OpenStreetMap z. B. mehr als 11 Millionen Strommasten. Die Anlagendaten sind allerdings weder vollständig noch qualitativ besonders hochwertig. Dieses Problem versucht u. a. das Projekt **OpenGridMap** zu lösen (siehe [8]). Das Projekt wird am Lehrstuhl für Anwendungs- und Middleware-Systeme an der Technischen Universität München entwickelt. Es zielt darauf ab, Anlagendaten auf Ebene von Übertragungs- und Verteilnetzen zu sammeln (Crowdsourcing), wenn nötig automatisiert abzuleiten und zu verifizieren. Das Ziel ist die Erstellung von Netzmodellen, die für eine Simulation genutzt werden können. Dieses Ziel verfolgen auch einige weitere Projekte. Hier ist insbesondere auch SciGRID zu nennen, welches neben den elektrischen auch Erdgasnetze einbezieht (siehe z. B. [20]).

3.5 Zusammenfassung

Die Energiewirtschaft ist auf verlässliche Informations- und Kommunikationssysteme angewiesen. In diesen Systemen werden Daten verarbeitet und ausgetauscht, um z. B. Netze, Erzeugungsanlagen oder Gebäude zu steuern und zu regeln oder um Energieprodukte auf Großhandelsmärkten zu handeln. Um diese Aufgaben zu bewältigen, werden qualitativ hochwerte Stamm- und Bewegungsdaten benötigt. Welche Daten in der Energiewirtschaft hauptsächlich verarbeitet werden und auf welchen Informationsmodellen diese basieren, war daher Hauptaugenmerk dieses Kapitels.

Ein Abschnitt wurde dem *Common Information Model* gewidmet. Dieses realisiert ein allgemeines Informationsmodell für die Energiewirtschaft, auf dessen Basis sowohl Objekte des physikalisch-technischen als auch des kaufmännischen Energiesystems abgebildet und beschrieben werden können. Wie wir gesehen haben, können CIM-Daten auf Basis von XML/RDF ausgetauscht werden, es existieren zudem ein standardisiertes Nachrichtenformat und Empfehlungen für Integrationstechnologien, um Anwendungssysteme der Energiewirtschaft miteinander zu integrieren.

Energieanlagen, die mithilfe von CIM beschrieben werden können, haben meist einen räumlichen Bezug. Räumliche Daten spielen daher in der Energiewirtschaft eine besondere Rolle. Es wurden daher Verfahren und Werkzeuge vorgestellt, wie diese Daten gespeichert, verarbeitet und dargestellt werden können. Systeme, wie PostGIS oder QGis, bieten entsprechende räumliche Datentypen und Funktionen an, um die Daten auszuwerten und komplexe Abfragen zu erstellen.

Räumliche Daten zählen in der Energiewirtschaft meist zu den Stammdaten. Bewegungsdaten äußern sich hingegen oft in Form von Zeitreihen, entsprechend wurde der Speicherung, Verarbeitung und Visualisierung von Zeitreihen ein entsprechender Abschnitt gewidmet. Wie wir gesehen haben, können Zeitreihen auch mithilfe des CIM abgebildet werden, entsprechende Klassen finden sich im Kernmodell des CIM als auch in der Markterweiterung des IEC 62325. Für die Speicherung von Zeitreihen werden noch häufig relationale Datenbanken verwendet, bei sehr großen Datenbeständen werden diese aber zunehmend durch spezielle Zeitreihendatenbanken wie Graphite oder OpenTSDB abgelöst. Für die Visualisierung von Zeitreihendaten wird in der Praxis häufig Grafana eingesetzt, da sich dieses System mit einer großen Anzahl von Datenbanksystemen verbinden lässt und viele Möglichkeiten der Darstellung bietet.

Viele Ansätze der Energieinformatik kommen ohne frei zugängliche Daten nicht aus. Für die Modellierung von Energiesystemen zum Zwecke der Prognose oder Simulation werden entsprechende Daten benötigt. Dies können z. B. Stammdaten zu Energieanlagen, aber auch Bewegungsdaten, wie Marktpreise oder Wetterdaten, sein. Es wurden daher einige nützliche Datenquellen vorgestellt, aus denen entsprechende Daten bezogen werden können. Neben dem Marktstammdatenregister als quasi offizielle Quelle für Stammdaten und der SMARD-Plattform als Quelle für Marktdaten existiert eine schier unüberschaubare Vielzahl von Initiativen, wie z. B. OPSD, die Daten im Kontext von Energiesystemen

zur Verfügung stellen. In vielen Szenarien besteht zudem der Bedarf für Geodaten, wie z. B. Landkarten oder Angaben zu Standorten und Verläufen von Erzeugungsanlagen oder Netzen. Mit OpenStreetMap und OpenGridMap existieren auch in diesem Bereich Plattformen, die als freie Datenquelle genutzt werden können.

3.6 Kontrollfragen

- In welchen Standardreihen wird das CIM eingeführt?
- Auf welchem Paradigma basiert das Informationsmodell des CIM?
- Beschreiben Sie mit eigenen Worten, was der Zweck des CIM ist.
- Nennen Sie einen konkreten Anwendungsfall, in dem das CIM aktuell genutzt wird.
- Welche unterschiedlichen Perspektiven auf das Energiesystem sind über das CIM abbildungbar? Welche Anwendungsfälle gibt es für diese Modelle? Nennen Sie Beispiele für Elemente, die zu den unterschiedlichen Modellen gehören.
- Wie heißt der Basistyp des CIM, von dem jeder stromleitende Anlagenteil ableitet?
- Über Objekte welchen Typs werden Anschlusspunkte eines stromleitenden Anlagen-teils im CIM realisiert?
- Warum werden im CIM mehrere unterschiedliche Varianten zur Abbildung von Marktmodellen gepflegt? Nennen Sie ein Beispiel mit Namen.
- Welches Datenformat wird standardmäßig dazu genutzt, um CIM-basierte Daten zu serialisieren?
- Nennen Sie Werkzeuge, mit denen CIM-Daten bearbeitet und visualisiert werden können.
- Existieren Programmzbibliotheken, um die Verarbeitung von CIM-Daten in eigenen Anwendungssystemen zu integrieren? Wenn Ja, welche?
- Warum spielen räumliche Daten in der Energiewirtschaft eine besondere Rolle?
- Sind räumliche Daten in der Energiewirtschaft eher Stamm- oder Bewegungsdaten?
- Können im CIM auch räumliche Daten abgebildet werden? Wenn ja, wie?
- Was ist PostGIS?
- In welchen unterschiedlichen Koordinatensystemen können die Datentypen von PostGIS basieren? Erläutern Sie deren Vor- und Nachteile für unterschiedliche Anwendungsfälle.
- Was ist das World Geodetic System 1984?
- Wie lautet der SQL-Befehl, um in PostGIS einen Punkt zu erzeugen?
- In welcher Auszeichnungssprache können räumliche Strukturen ausgedrückt werden?
- Welche SRID besitzt das WGS84 in PostGIS?
- Nennen Sie Beispiele für Funktionen, die von PostGIS auf räumlichen Daten angeboten werden.
- Welche zusätzlichen Möglichkeiten bietet ein geografisches Informationssystem wie QGis gegenüber PostGIS?

- Was für Daten werden in der Energiewirtschaft verarbeitet, wenn von Energiedatenmanagement die Rede ist?
- Welche generelle Struktur hat das Informationsmodell von Zeitreihen?
- Nennen Sie Beispiele für Kurventypen, die durch entso-e im Kontext von CIM-basierten Zeitreihen eingeführt wurden.
- Welche Vor- und Nachteile hat die Speicherung von Zeitreihendaten in einem relationalen Datenbanksystem?
- Nennen Sie Beispiele für NoSql-basierte Zeitreihendatenbanken.
- Wie werden Zeitreihen in NoSql-basierten Zeitreihendatenbanken üblicherweise genannt?
- Welches System bietet sich für die Visualisierung von Zeitreihendaten an?
- Nennen Sie freie Datenquellen für Stamm- und Bewegungsdaten der Energiewirtschaft.
- Wie genau ist ein sog. Data Package der Frictionless Data Initiative intern aufgebaut?
- Aus welchen Grundelementen besteht das Informationsmodell der OpenStreetMap-Initiative?
- Wie werden Energieassets in OSM-Daten codiert?

Literatur

1. Eric A. Brewer. „Towards robust distributed systems“. In: *Proceedings of the nineteenth annual ACM symposium on Principles of distributed computing*. PODC '00. Portland, Oregon, USA: Association for Computing Machinery, 16. Juli 2000, S. 7. ISBN: 978-1-58113-183-3. URL: <https://doi.org/10.1145/343477.343502> (besucht am 21.02.2020).
2. DIN ISO 8601:2006-09 – Datenelemente und Austauschformate – Informationsaustausch – Darstellung von Datum und Uhrzeit. Beuth Verlag GmbH, 2006. <https://doi.org/10.31030/9730781>. URL: <https://www.beuth.de/de/-/-/90754621> (besucht am 18.09.2019).
3. Jeffrey Dean und Sanjay Ghemawat. „MapReduce: simplified data processing on large clusters“. In: *Communications of the ACM* 51.1 (1. Jan. 2008), S. 107–113. issn: 0001-0782. URL: <https://doi.org/10.1145/1327452.1327492> (besucht am 04.06.2020).
4. Christiana Köhler-Schute, Hrsg. *Energiedaten effizient managen*. Berlin: KSEnergy-Verl., 2008. 190 S. ISBN: 978-3-00-025766-7.
5. IEC 61968-1:2012 – Application integration at electric utilities – System interfaces for distribution management – Part 1: Interface architecture and general recommendations. International Electrotechnical Commission (IEC), Okt. 2012.
6. Bill Meehan. *Modeling electric distribution with GIS*. New York: Esri Press, 2013.
7. Common Grid Model Exchange Standard (CGMES). Version 2.4. European Network of Transmission System Operators for Electricity (entso-e), 28. Mai 2014. URL: https://docstore.entsoe.eu/Documents/CIM_documents/Grid_Model_CIM/140528_ENTSOE_CGMES_v2.4.14.pdf (besucht am 20.02.2020).
8. José Rivera u. a. „OpenGridMap: An Open Platform for Inferring Power Grids with Crowd-sourced Data“. In: *Energy Informatics*. Hrsg. von Sebastian Gottwalt, Lukas König und Hartmut Schmeck. Bd. 9424. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2015, S. 179–191. ISBN: 978-3-319-25875-1. https://doi.org/10.1007/978-3-319-25876-8_15.

9. J. Simmins. *Common Information Model Primer*. Juni 2015. URL: <https://www.epri.com/#/pages/product/3002006001/?lang=en-US>.
10. André Göring. *CIM – Common Information Model*. 2016. URL: <https://wiki.eclipse.org/images/7/7b/201604201-21-oK-ACQC-Top-8-CIM.pdf>
11. José Antonio Rodríguez Mondéjar und José María Oyarzabal Moreno. „WP 2 – Innovative Distribution Grid Use Cases and Functions“. In: (30. Dez. 2016). URL: <https://ec.europa.eu/research/participants/documents/downloadPublic?documentIds=080166e5af3ae0ff&appId=PPGMS>.
12. *Common information model (CIM) – European style market profile – User guide*. European Network of Transmission System Operators for Electricity (entso-e), 3. Jan. 2017. URL: https://docstore.entsoe.eu/Documents/EDI/Library/cim_based/20170110_ENTSO-E_CIM.pdf (besucht am 20.02.2020).
13. Fabian Rigoll. „Nutzerorientiertes Energiedatenmanagement“. Dissertation. Karlsruhe: KIT-Bibliothek, 2017. URL: <http://digibib.ubka.uni-karlsruhe.de/volltexte/1000068109>.
14. *Das Marktstammdatenregister*. 2018. URL: https://www.bundesnetzagentur.de/SharedDocs/Downloads/DE/Sachgebiete/Energie/Unternehmen_Institutionen/DatenaustauschUndMonitoring/MaStR/MaStR%20-%20Gesamtkonzept_170301.pdf?__blob=publicationFile&v=4 (besucht am 25.06.2019).
15. Andre Göring. *openKonsequenz Architecture Committee Handbook*. 26. Nov. 2018. URL: <https://www.openkonsequenz.de/das-konsortium/download/402/49/17> (besucht am 25.06.2019).
16. *IEC 62325-301:2018 – Framework for energy market communications – Part 301: Common information model (CIM) extensions for markets*. International Electrotechnical Commission (IEC), März 2018.
17. Lukas Razik u. a. „Automated deserializer generation from CIM ontologies“. In: *Computer Science – Research and Development* 33.1-2 (2018), S. 93–103. ISSN: 1865-2034. <https://doi.org/10.1007/s00450-017-0350-y>.
18. *DIN EN ISO/IEC 27019:2019-12 – Informationssicherheitsmaßnahmen für die Energieversorgung*. DIN - Deutsches Institut für Normung, Dez. 2019. URL: <https://www.din.de/de/wdc-beuth:din21:312706732> (besucht am 28.05.2020).
19. European Network of Transmission System Operators for Electricity (ENTSO-E). *The introduction of different time series possibilities (curvetype) with ENTSO-E electronic documents*. 28. März 2019. URL: https://docstore.entsoe.eu/Documents/EDI/Library/depreciated/Timeseries-curve-types_v1_2.pdf (besucht am 17.09.2019).
20. A. Pluta u. a. „SciGRID gas – A topological open source model of the European Gas Transmission Network“. In: *3. Jahrestreffen Forschungsnetzwerk Energiesystemanalyse*. Aarhus, Denmark, Mai 2019.
21. *IEC 61970:2020 SER – Energy management system application program interface (EMS-API) – ALL PARTS*. International Electrotechnical Commission (IEC), Jan. 2020.
22. *CIMDeskURL*: <https://www.powerinfo.us/CIMdesk.html> (besucht am 20.02.2020).
23. *CIMSpace*. URL: <https://derrickoswald.github.io/CIMSpace/> (besucht am 20.02.2020).
24. *CIMSpy*. URL: <https://www.powerinfo.us/CIMSpy.html> (besucht am 20.02.2020).
25. *Docker Image for Graphite & Statsd*. URL: <https://hub.docker.com/r/graphiteapp/docker-graphite-statsd/> (besucht am 21.02.2020).
26. *GrafanaURL*: <https://grafana.com/> (besucht am 21.02.2020).
27. *GraphiteURL*: <https://graphiteapp.org/> (besucht am 21.02.2020).
28. *Graphite Functions*. URL: <https://graphite.readthedocs.io/en/latest/functions.html> (besucht am 14.07.2020).
29. *Marktstammdatenregister*. URL: <https://www.marktstammdatenregister.de/> MaStR (besucht am 14.07.2020).
30. *OpenTSDB*. URL: <http://opentsdb.net/> (besucht am 21.02.2020).

31. *Ready to run OpenTSDB container.* URL: <https://hub.docker.com/r/petergrace/opentsbdbuilder/> (besucht am 21.02.2020).
32. Peter Zaitsev. *Percona Blog Poll: What Database Engine Are You Using to Store Time Series Data?* URL: <https://www.percona.com/blog/2017/02/10/percona-blog-poll-database-engine-using-store-time-series-data/> (besucht am 18.09.2019).



Algorithmen

4

Mit dem Common Information Model (CIM) haben wir uns bereits ausgiebig mit einem Informationsmodell der Energiewirtschaft befasst (siehe Abschn. 3.1). Eine Hauptaufgabe dieses Modells ist die Beschreibung des strukturellen Zusammenhangs zwischen Einzelkomponenten des Energiesystems. Mithilfe von Klassen, Assoziationen und Vererbung wird darin der Zusammenhang zwischen Konzepten, wie z. B. Transformator, Leitung, und Standort spezifiziert. Diese strukturelle Beschreibung ist meist nur eine Perspektive eines Systems. Nach einer allgemeinen Systemtheorie können Systeme auch anhand einer hierarchischen oder einer funktionalen Perspektive beschrieben werden (siehe [8, S. 7]).

Das folgende Kapitel widmet sich besonders der funktionalen Perspektive des Energiesystems. Dabei geht es um die Wirkzusammenhänge zwischen Eingangs- und Ausgangsgrößen, die einen Effekt oder ein Phänomen beschreiben. Für eine solche Beschreibung kommen erneut Modelle zum Einsatz. Im Gegensatz zu einem Strukturmodell wie CIM, bildet das funktionale Modell eines Systems Zusammenhänge zwischen Ursache und Wirkung ab. Dabei werden gewisse Annahmen (Hypothesen) getroffen und Vereinfachungen vorgenommen. Bei der Erstellung eines solchen Modells hilft der Methodenapparat der Mathematik oder der Informatik. Das Ergebnis ist dann eine formale Beschreibung, z. B. ein Gleichungssystem, welches analysiert und bewertet werden kann. Ein solches Modell kann für vielerlei Anwendungsfälle genutzt werden, z. B. für die Vorhersage von zukünftigen Entwicklungen (Prognose), für die optimale Auslegung technischer Systeme (Optimierung) oder die rechnergestützte Analyse von möglichen Szenarien (Simulation).

Ein Beispiel könnten die Modellierung des Strom- oder Gasmarkts sein. Das System der Preisfindung auf diesen Märkten ist überaus komplex. Im Allgemeinen richtet sich der Preis nach Angebot und Nachfrage. Dabei wirken auf das Angebot zahlreiche Faktoren, wie z. B. das Wetter, Brennstoffpreise oder die Verfügbarkeiten von Kraftwerken. Die Nachfrage wird durch die Tageszeit, den Wochentag und die Außentemperatur beeinflusst.

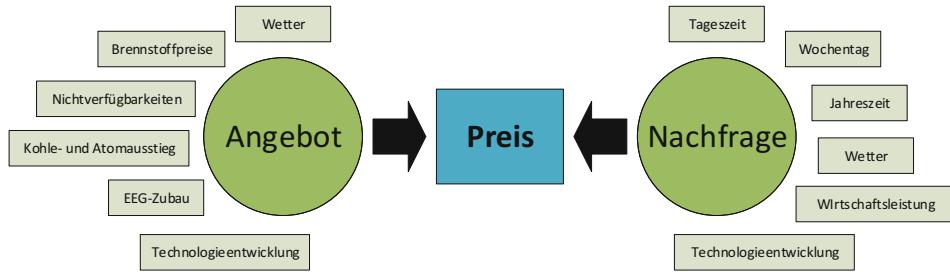


Abb. 4.1 Einflussfaktoren auf die Strompreisfindung

Einige Faktoren wirken dabei eher kurz-, andere langfristig. Abb. 4.1 zeigt einen Überblick über die Wirkzusammenhänge bei der Strompreisfindung.

Sind die Wirkzusammenhänge zwischen den Faktoren bekannt, kann das System in Form eines funktionalen Modells abgebildet werden. Mithilfe eines solchen Modells könnten dann z. B. Prognosen erstellt werden. Darauf basierend kann dann z. B. ein Energiehändler bessere Handelsentscheidungen treffen.

Für die Modellbildung stehen im Allgemeinen ganz unterschiedliche Methoden bereit (siehe [7, S. 72 ff.]).

- **Fundamentalanalytische Modelle**

Ein fundamentalanalytisches Modell versucht ein System auf Basis fundamentaler Einflüsse zu erklären. Für die Strompreisprognose würde dazu ein Modell eines realen Energieversorgungssystems mit technischen Charakteristika, wie z. B. Kraftwerken, Brennstoffkosten usw. abgebildet.

- **Finanzmathematische Modelle**

In finanzmathematischen Modellen wird die Entwicklung von Zielgrößen anhand von stochastischen Prozessen modelliert. Diese Modelle werden häufig für die Risikoanalyse genutzt.

- **Spieltheoretische Modelle**

Im deregulierten Energiemarkt können die Marktakteure durch spieltheoretische Ansätze modelliert werden. Ziel dieser Modellierung ist es, die Marktmacht einzelner Marktakteure zu bewerten.

- **Statistische und ökonometrische Modelle**

Statistische und ökonometrische Modelle basieren stärker auf empirischen Erkenntnissen. Dabei werden indirekte Faktoren, beim Strompreis z. B. der Zusammenhang mit der Temperatur herangezogen.

Welche Methoden sinnvoll sind, hängt von vielen Faktoren ab, z. B. der gewünschte Zeithorizont, die Genauigkeit der Daten, die Kenntnisse der Wirkzusammenhänge usw.

In den folgenden Abschnitten werden wir uns mit der Erstellung verschiedener funktionaler Modelle für die Energiewirtschaft auseinandersetzen und zeigen, mit welchen

Verfahren diese Modelle bewertet werden können. Im Abschn. 4.1 werden funktionale Modelle für die Prognose zukünftiger Verläufe, z. B. der Last aufgebaut. Neben dem einfachen ökonometrischen Modell der Standardlastprofile werden wir dort einige Methoden der Zeitreihenanalyse vorstellen. Im Abschn. 4.2 werden Modelle erstellt, um optimale Konfigurationen von Systemen zu bestimmen. Hier werden wir uns hauptsächlich mit Verfahren der linearen, ganzzahligen Optimierung auseinandersetzen, um das in der Energiewirtschaft wichtige *Unit-Commitment*-Problem lösen zu können.

4.1 Prognose

Ein wichtiger Grund, in der Energiewirtschaft Wirkzusammenhänge in Form eines Systemmodells zu beschreiben, ist die Möglichkeit zur Prognose. Das Ziel einer Prognose ist es, begründete Aussagen über zukünftige Ereignisse, Zustände oder Entwicklungen zu treffen. Viele Einflussfaktoren in der Energiewirtschaft sind extrem volatil, z. B. das Wetter, Marktpreise für Brennstoffe oder die Last. Je besser die zukünftige Entwicklung solcher Größen vorausgesagt werden kann, desto effizienter kann aber darauf reagiert werden. Bessere Entscheidungen treffen zu können, bedeutet in einer Wettbewerbswirtschaft immer auch mehr Profit.

In der Energiewirtschaft existieren manche Größen, die sich im Mittel recht gut prognostizieren lassen, z. B. der Energieverbrauch und die daraus resultierende Last. Ob gute Prognosen erstellt werden können, richtet sich im Allgemeinen nach vielen Faktoren:

- Wie gut sind die Einflüsse auf das System verstanden?
- Wie viele Daten sind verfügbar?
- Hat die Prognose evtl. wieder Einfluss auf das Ereignis selbst?

Die möglichen Verfahren können nach verschiedenen Kriterien unterschieden werden, z. B. nach ihrem Prognosehorizont. Die möglichen Ansätze werden dabei oft in Kurz- und Langfristprognosen eingeteilt.

Ein Energieversorger benötigt eine möglichst genaue Vorhersage der Lastkurve seiner Kunden. Entsprechend dieser Lastkurve kauft der Versorger die Energiemenge ein oder erzeugt sie in Kraftwerken selbst. Weicht die vorhergesagte Last von der tatsächlich bereitgestellten Energiemenge ab, gleicht der Übertragungsnetzbetreiber dies aus, was zu erheblichen Kosten führt (siehe Abschn. 2.1). Die überwiegende Menge der Kunden eines Energieversorgungsunternehmens (EVU) besitzt allerdings lediglich Energiemengenzähler, die nur selten (z. B. jährlich) abgelesen werden. Dennoch muss für diese Kunden eine Lastkurve mit viertelstündlicher Auflösung erstellt werden, um den entsprechenden Bilanzkreis zu bewirtschaften. Ein beliebtes Verfahren, um hier zu einer gesicherten Lastprognose zu kommen, ist die Verwendung von **Standardlastprofilen**.

Bei diesem Verfahren ist die Modellannahme, dass ein Kunde eines Energieversorgers einer bestimmten Kundenkategorie zugeordnet werden kann und der Lastkurve einer

bestimmten Kundengruppe ein klares Muster unterliegt. Eine solche normierte Lastkurve für eine bestimmte Kategorie von Kunden wird als Standardlastprofil bezeichnet. Anhand der Zuordnung von Kunden zu den Kategorien und einer Skalierung der Standardlastprofile auf den geschätzten Jahresverbrauch kann eine Prognose der Gesamtlast über alle entsprechenden Kunden erzeugt werden.

Die Profile werden durch den Verband der Elektrizitätswirtschaft (VDEW) erstellt. Aktuell existieren sieben gewerbliche (G0-G7), drei landwirtschaftliche (L0-L2) und ein Haushaltsprofil (H0). Die Lastkurven können z. B. beim Bundesverband der Energie- und Wasserwirtschaft e. V. (BDEW) in Form eines Excel-Sheets heruntergeladen werden (siehe [18]). Ein Standardlastprofil enthält je eine Zeitreihe von Lastwerten (96 Viertelstundenwerte) für einen Werktag, einen Samstag und einen Sonn- bzw. Feiertag. Zudem wird noch zwischen Sommer, Winter und Übergangszeit unterschieden. Die Lastprofile sind dabei auf einen Jahresverbrauch von 1000 kWh normiert.

Für die Anwendung der Standardlastprofile hat der VDEW eine genaue Anleitung herausgegeben (siehe [2]). Die wichtigsten Schritte sind dabei:

- Anwendbarkeit prüfen,
- Kundengruppe charakterisieren,
- Erstellung eines Jahreslastverlaufs.

Bei dem Erstellen eines Jahreslastverlaufs sind einige Dinge zu beachten:

- Das jeweilige Standardlastprofil muss auf den geschätzten Jahresverbrauch des Kunden skaliert werden.
- Es sind bundesweite und regionale Feiertage zu beachten.
- Die Zeitumstellung zwischen Sommer- und Winterzeit muss besonders behandelt werden.
- Es müssen Schaltjahre beachtet werden.

In der Praxis kann die Last der meisten Endkunden mithilfe der Standardlastprofile zumindest grob abgeschätzt werden. Es ergeben sich allerdings zunehmend Schwierigkeiten. Die Standardlastprofile basieren auf Messungen aus Westdeutschland aus den Jahren 1986 und 1987. Seitdem haben sich allerdings manche Gewohnheiten gewandelt und auch der Einfluss dezentraler Energieanlagen bzw. der Elektromobilität macht sich zunehmend bemerkbar. Es existieren daher eine Reihe von Studien, die weitere Differenzierungen von Lastprofilen vorschlagen. Ein Vergleich verschiedener Modelle wird in [15] angestrengt. Auch wird in der Arbeit ein eigener Ansatz vorgestellt, um Lastprognosen für Haushaltskunden zu erstellen. Im Rahmen einer Kooperation mehrerer Forschungseinrichtungen entstand auf dieser Basis das Open-Source-System LoadProfileGenerator, welches kostenfrei genutzt werden kann.¹

¹<https://www.loadprofilegenerator.de/>.

Die Last großer Industriekunden kann nicht mithilfe von Standardlastprofilen prognostiziert werden, da sie recht individuelle Lastverläufe zeigen. Solche Stromkunden fallen unter die sog. registrierende Leistungsmessung (RLM) und besitzen dafür einen Energiemengenzähler, der die Messwerte zeitnah an den Versorger übermitteln kann. Auch anhand dieser Daten kann eine Lastprognose abgeleitet werden, wofür dann Verfahren der **Zeitreihenanalyse** genutzt werden. Dabei muss angenommen werden können, dass die Zeitreihe (z. B. eine Lastkurve) sich in der Zukunft ähnlich verhält wie in der Vergangenheit. Ein solches Modell (i. Allg. eine mathematische Funktion) soll durch nur wenige Parameter beschrieben werden können, z. B. durch Steigung und Achsverschiebung einer Geraden. Die Parameter des Modells werden dann anhand der Zeitreihenwerte aus der Vergangenheit gewählt. Dabei wird die Abweichung zwischen der Prognose und den realen Zeitreihendaten möglichst minimiert.

Im Nachhinein (ex post) lässt sich die Prognosegüte, also die Abweichung der Prognose von den tatsächlichen Werten, bewerten. Für diese Bewertung wurden unterschiedliche **Fehlermaße** eingeführt. Ein Fehlermaß bildet die Differenz zwischen Prognose und Realität auf einen einzelnen Wert ab. Je kleiner dieser Wert, desto besser die Prognose. Bekannte Fehlermaße sind:

- der mittlere quadratische Fehler (Mean Squared Error (MSE)),
- der mittlere absolute Fehler (Mean Absolute Error (MAE)),
- der mittlere absolute prozentuale Fehler (Mean Absolute Percentage Error (MAPE)).

Reine Zufallsprozesse (*Random Walk*) lassen sich kaum vorhersagen, ein Beispiel solcher Prozesse sind Aktienkurse. Je mehr wiederkehrende Muster aber erkennbar sind, desto eher kann ein Prognosemodell erstellt werden. In der Energiewirtschaft hat man es meist mit Zeitreihen zu tun, in denen bestimmte Muster zu finden sind. Ein solches Muster wird auch als **Zeitreihenkomponente** bezeichnet.

Häufig wird z. B. zwischen dem Level, dem Trend, der Saisonalität und der Restkomponente bzw. den Residuen unterschieden. Ein Level L_t existiert, wenn die Zeitreihe einen langfristigen Mittelwert ungleich null besitzt. Ein Trend T_t existiert genau dann, wenn ein lang anhaltendes Steigen oder Absinken der Zeitreihe zu beobachten ist. Ein oder mehrere saisonale Muster S_t existieren, wenn die Zeitreihe von saisonalen Faktoren (z. B. der Tages- oder Jahreszeit) abhängen. Eine Saison unterliegt dabei immer einer fixen und bekannten zeitlichen Länge. Die Restkomponente R_t beschreibt den zufälligen Modellfehler. Die Komponenten einer Zeitreihe können additiv oder multiplikativ aufeinander wirken:

- Additive Komponenten schreibt man als $y_t = L_t + T_t + S_t + R_t$.
- Multiplikative Komponenten schreibt man als $y_t = L_t \cdot T_t \cdot S_t \cdot R_t$.

Abb. 4.2 zeigt einen fiktiven Lastverlauf, der die zuvor genannten Zeitreihenkomponenten aufweist.

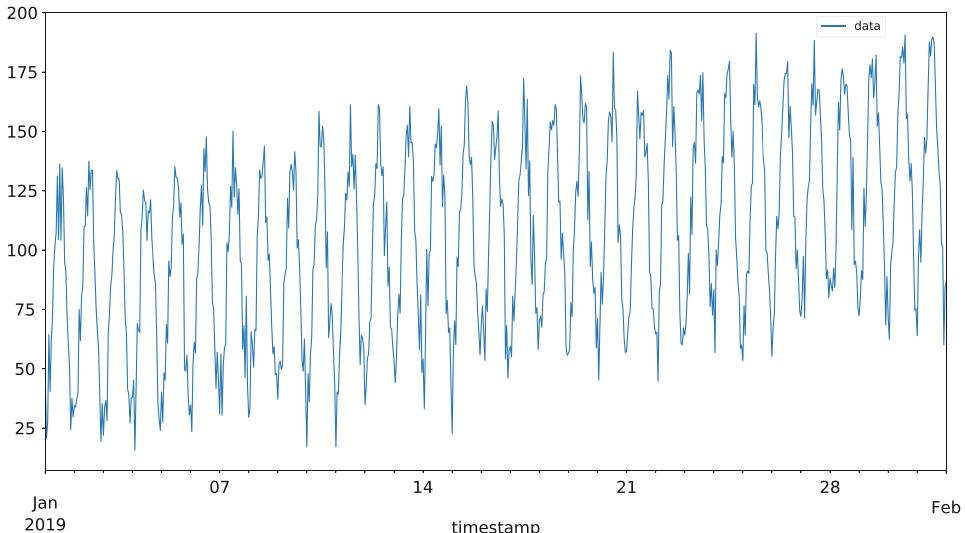


Abb. 4.2 Beispiel einer Zeitreihe, die auf mehreren Zeitreihenkomponenten basiert

Für die Prognose ist es oft hilfreich, die Komponenten einer Zeitreihe jeweils mit eigenen Modellen zu beschreiben. Jede Komponente kann dann leichter für sich allein und mit unterschiedlichen Verfahren prognostiziert werden. Um zu einem geeigneten Gesamtmodell zu kommen, muss die vorhandene Zeitreihe allerdings erst in ihre Komponenten zerlegt werden. Eine solche Zerlegung nennt man **Dekomposition**. Für die Dekomposition existieren mehrere Verfahren.

Eine recht einfache Möglichkeit ist die Verwendung des **gleitenden Mittelwerts** (Moving Average (MA)). Der gleitende Mittelwert berechnet das arithmetische Mittel über ein gleitendes Fenster von Zeitreihenwerten (siehe auch [17, S. 161–167]). Die mathematische Definition des gleitenden Mittelwerts lautet dabei wie folgt:

$$ma^n(t) = \frac{1}{n} \sum_{i=0}^{n-1} x(t-i) \quad (4.1)$$

Dabei sind $x(t)$ die ursprünglichen Werte der Zeitreihe und n die **Fenstergröße**, über die der Mittelwert berechnet wird. Der MA erzeugt eine neue Zeitreihe, die um die Fenstergröße n weniger Werte besitzt als die Originalzeitreihe. Der MA hat eine glättende Wirkung auf die Zeitreihe, je größer die Fenstergröße n gewählt wird, desto weniger Schwankungen zeigt die neue Zeitreihe. Der MA zeigt dann nur noch die langfristigen Änderungen der Originaldaten. Oft wird die MA-Zeitreihe dabei auf die Originalzeitreihe zentriert, d. h., die erzeugte Zeitreihe wird um $\frac{n}{2}$ Datenpunkte verschoben.

Um den MA auf eine Zeitreihe anwenden zu können, muss der Algorithmus entsprechend der zuvor dargestellten Rechenvorschrift implementiert werden. In der Realität wird

man dazu bereits existierende Werkzeuge, wie z. B. Excel oder R nutzen. In Python hilft das Pandas-Paket dabei, den gleitenden Mittelwert über einer Zeitreihe zu berechnen. Liegen die Daten in einer Spalte eines Data-Frames, können die Funktionen `rolling()` und `mean()` genutzt werden. Der folgende Python-Code berechnet den gleitenden Mittelwert über die Spalte „data“.

```

1 import matplotlib.pyplot as plt
2 data["ma"] = data["data"].rolling(window=24).mean(center=True)
3 data.plot(x="timestamp", y=["data", "ma"])
4 plt.show()

```

Die berechneten Daten werden in der Spalte „ma“ abgelegt. Die Fenstergröße des MA (hier 24) wird als Parameter an die Funktion `rolling()` übergeben. Die beiden Zeitreihen werden zudem in einem gemeinsamen Chart dargestellt. Abb. 4.3 zeigt den resultierenden Chart mit den Originaldaten und dem gleitenden Mittelwert.

Der gleitende Mittelwert kann für die Dekomposition einer Zeitreihe in Komponenten genutzt werden. Ein entsprechendes Verfahren wird z. B. in [17] beschrieben. Der gleitende Mittelwert kann dabei die Trendkomponente T_t einer Zeitreihe abschätzen. Bei einem additiven Komponentenmodell ergibt sich: $S_t + R_t = y_t - T_t$. Die Originaldaten, also y_t , sind bekannt, ebenso die Trendkomponente T_t . Es bleibt also die von den zufälligen Restschwankungen überlagerte Saisonalkomponente übrig. Die Saisonalkomponente kann nun von diesen Restschwankungen befreit werden. Die bereinigte Saisonalkomponente für eine Periode ergibt sich als Mittelwert über alle Perioden. Die Restschwankungen

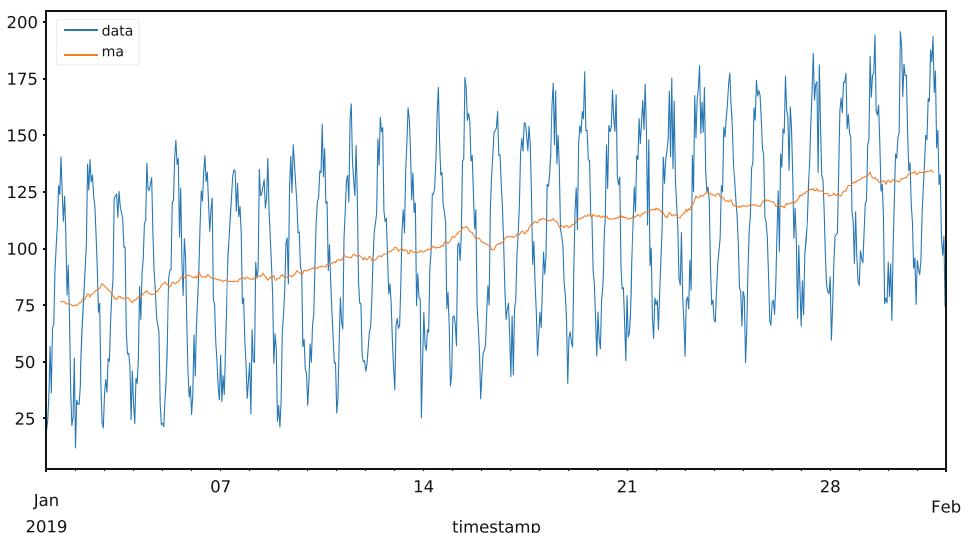


Abb. 4.3 Glättung einer Zeitreihe mithilfe des gleitenden Mittelwerts

R_t ergeben sich dann als Differenz der Fortschreibung dieser Periode über den gesamten Zeitraum der Daten.

In Python kann mithilfe des Moduls `StatsModels` eine solche Dekomposition auf Basis des gleitenden Mittelwerts durchgeführt werden. Die Methode `seasonal_decompose()` geht dabei von einem saisonalen Zeitreihenmodell aus. Es wird die Trendkomponente extrahiert und danach werden wie beschrieben die Saisonalkomponente und die Residuen bestimmt. Der folgende Beispielcode erzeugt eine Dekomposition saisonaler Daten in Teilkomponenten.

```
1 from statsmodels.tsa.seasonal import seasonal_decompose
2 data = data.set_index("timestamp")
3 result = seasonal_decompose(data, model='additive', freq=24)
```

Als Parameter muss die angenommene Intervallbreite der Saison (Frequenz) angegeben werden und ob ein additives oder multiplikatives Modell angenommen wird. Abb. 4.4 zeigt das Ergebnis der Dekomposition mithilfe des Python-Moduls `StatsModels`.

Ob eine solche Dekomposition gelungen ist, kann insbesondere anhand der Residuen bewertet werden (siehe dazu auch [17]). Das Zeitreihenmodell als Überlagerung von Trend- und Saisonalkomponente ist genau dann brauchbar, wenn

- die Residuen unkorreliert sind und
- die Residuen einen Mittelwert von „0“ besitzen.

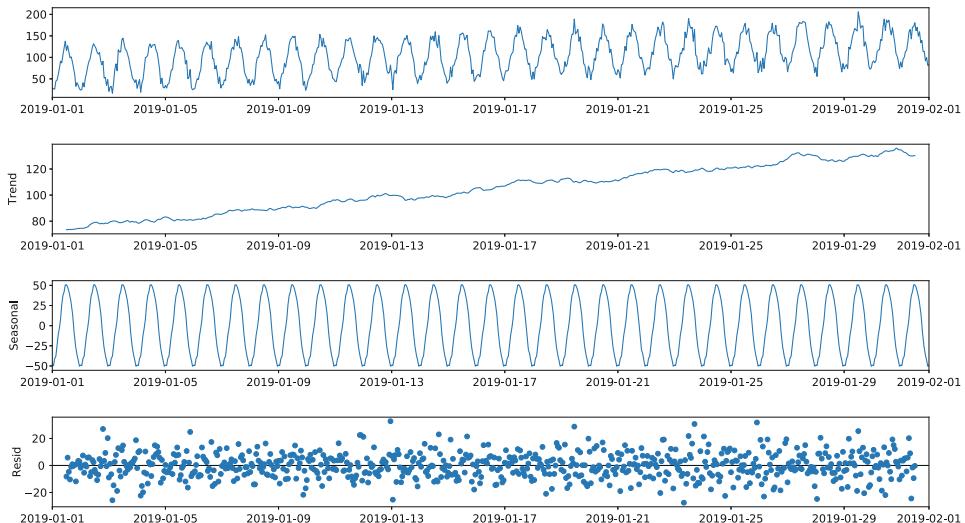


Abb. 4.4 Resultat der Dekomposition einer Zeitreihe in die einzelnen Komponenten

Ist diese Situation gegeben, können die Zeitreihenkomponenten auch für eine Prognose eingesetzt werden. Dafür wird der Trend weiter in die Zukunft fortgeführt und mit der Saisonalkomponente entsprechend überlagert. Die Trendkomponente kann dafür auch mithilfe der linearen Regression weiter parametrisiert werden.

Neben der Komponentenzerlegung mithilfe des gleitenden Mittelwerts existieren viele weitere Verfahren, wie z. B. das sog. Loess-Verfahren (siehe z. B. [12]). Auf diese werden wir hier nicht weiter eingehen.

Ein Prognoseverfahren mit hoher Praxisrelevanz wird als exponentielles Glätten bezeichnet. Nach ihren Erfindern (Charles C. Holt und Peter Winters) ist es auch als Holt-Winters-Verfahren bekannt. Das exponentielle Glätten kann insbesondere dazu genutzt werden, um die zukünftige Last aus vergangenen Verläufen vorherzusagen (siehe z. B. [6]).

Die einfachste Form des exponentiellen Glättens wird als exponentielles Glätten erster Ordnung bezeichnet und geht von einer Zeitreihe ohne Trend- und Saisonalkomponente aus. Für die Prognose \hat{y} wird der sog. Level l am Ende der Zeitreihe für h Schritte in die Zukunft fortgesetzt:

$$\hat{y}_{t+h} = l_t \quad (4.2)$$

Der Level ergibt sich dabei rekursiv aus den vergangenen Werten der Zeitreihe:

$$l_t = \alpha y_t + (1 - \alpha)l_{t-1} \quad (4.3)$$

In den Level fließt der sog. Glättungsfaktor (*Smoothing Factor*) α ein. Er besitzt einen Wert zwischen null und eins und beeinflusst die Gewichtung weiter zurückliegender Zeitreihenwerte. Je kleiner α ist, desto stärker werden weiter zurückliegende Werte berücksichtigt. Mit der Berechnung der Levelwerte l_t kann am Anfang der Zeitreihe begonnen werden. Das Ergebnis ist eine neue Zeitreihe, die sich möglichst nah an die Originaldaten anschmiegen sollte:

$$\begin{aligned} l_1 &= \alpha y_1 + (1 - \alpha)l_0 \\ l_2 &= \alpha y_2 + (1 - \alpha)l_1 \\ l_3 &= \alpha y_3 + (1 - \alpha)l_2 \\ &\dots \end{aligned} \quad (4.4)$$

Um die Levelwerte zu berechnen, sind zwei Parameter festzulegen:

- der Glättungsfaktor α , der zwischen null und eins liegen muss,
- der Startwert l_0 .

Beide Parameter sollten so festgelegt werden, dass ein Fehlermaß (z. B. SSE) zwischen den Originaldaten und den angepassten Werten möglichst klein wird. Im Allgemeinen ist dies allerdings ein nichtlineares Optimierungsproblem, für dessen Lösung man entsprechende Werkzeuge benötigt.

Das exponentielle Glätten erster Ordnung kann in Python mithilfe des Moduls `Statsmodel` durchgeführt werden. Der folgende Programmcode erzeugt eine solche Prognose für 20 Schritte in die Zukunft. Dabei werden die Werte für l_0 bereits so bestimmt, dass ein möglichst kleiner Fehler für die angepassten Werte entsteht.

```

1 from statsmodels.tsa.holtwinters import ExponentialSmoothing
2 fit = ExponentialSmoothing(data["data"]).fit()
3 fcast = fit.forecast(48)
4 fcast.plot()
5 fit.fittedvalues.plot()
6 plt.show()
```

Abb. 4.5 zeigt das Ergebnis der Vorhersage mittels exponentiellen Glättens erster Ordnung.

Das exponentielle Glätten wurde im Jahr 1957 durch Holt um eine Trendkomponente erweitert. Das Modell einer Zeitreihe mit Trend basiert dabei auf einer einfachen Geradegleichung $y = bx + l$. Dabei ist l der Level und b die Steigung der Geraden. Entsprechend ergibt sich die Prognose für h Schritte in die Zukunft als:

$$\hat{y}_{t+h} = hb_t + l_t \quad (4.5)$$

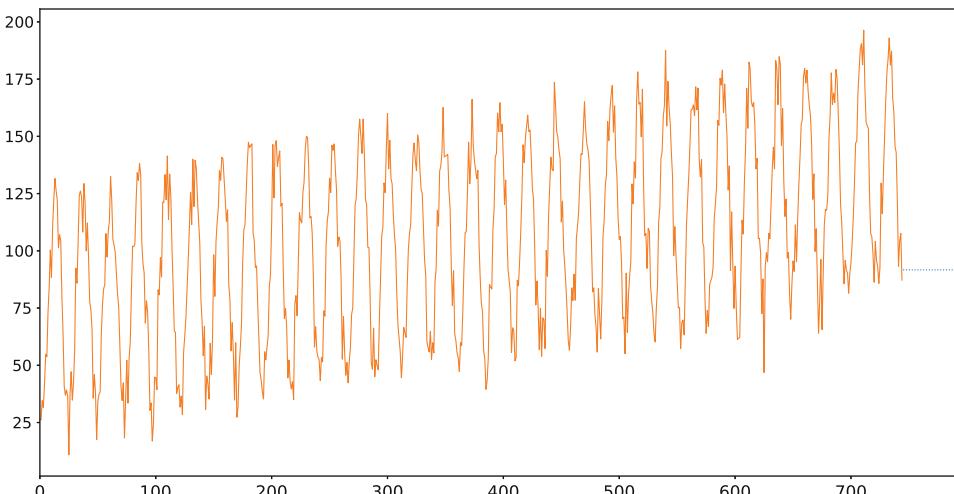


Abb. 4.5 Exponentielles Glätten erster Ordnung

Sowohl für den Level l_t als auch für die Steigung b_t wird nun exponentielles Glätten eingesetzt:

$$\begin{aligned} l_t &= \alpha y_t + (1 - \alpha)(l_{t-1} + b_{t-1}) \\ b_t &= \beta(l_t - l_{t-1}) + (1 - \beta)b_{t-1} \end{aligned} \quad (4.6)$$

Dabei wird ein zweiter Glättungsfaktor β eingeführt, der auf die Berechnung der Steigung des Trends wirkt. Kleine Werte von β deuten darauf hin, dass sich die Steigung der Trendgeraden über die Zeit kaum ändert. Man bezeichnet dieses Verfahren folgerichtig als exponentielles Glätten zweiter Ordnung. Auch beim exponentiellen Glätten zweiter Ordnung ist es günstig, am Beginn der Zeitreihe mit der Berechnung von b_t zu starten:

$$\begin{aligned} b_1 &= \beta(l_1 - l_0) + (1 - \beta)b_0 \\ b_2 &= \beta(l_2 - l_1) + (1 - \beta)b_1 \\ b_3 &= \beta(l_3 - l_2) + (1 - \beta)b_2 \\ &\dots \end{aligned} \quad (4.7)$$

Für das exponentielle Glätten zweiter Ordnung werden bereits vier Parameter benötigt, die das Modell beeinflussen:

- der Glättungsfaktor α und der Startwert l_0 für den Level,
- der Glättungsfaktor β und der Startwert b_0 für den Trend.

Auch für das exponentielle Glätten mit Trend kann das Python-Modul `Statsmodels` genutzt werden. Der folgende Programmcode erzeugt eine entsprechende Prognose für 20 Schritte in die Zukunft. Dabei wird beim Trend von einem additiven Komponentenmodell ausgegangen.

```

1 from statsmodels.tsa.holtwinters import ExponentialSmoothing
2 fit = ExponentialSmoothing(data["data"], trend='add').fit()
3 fcast = fit.forecast(48)
4 fcast.plot()
5 fit.fittedvalues.plot()
6 plt.show()
```

Abb. 4.6 zeigt das Ergebnis der Vorhersage mittels exponentiellen Glättns zweiter Ordnung. Im Vergleich zu Abb. 4.5 besitzt die Prognose eine steile Steigung.

Die letzte Erweiterung, die uns beim exponentiellen Glätten fehlt, ist die Berücksichtigung der Saisonalität. Tatsächlich haben Holt und Winter das Modell entsprechend erweitert. Man nennt dieses Verfahren auch das exponentielle Glätten dritter Ordnung. Dabei besteht das Zeitreihenmodell nun aus drei Komponenten. Dem Level l_t , dem Trend

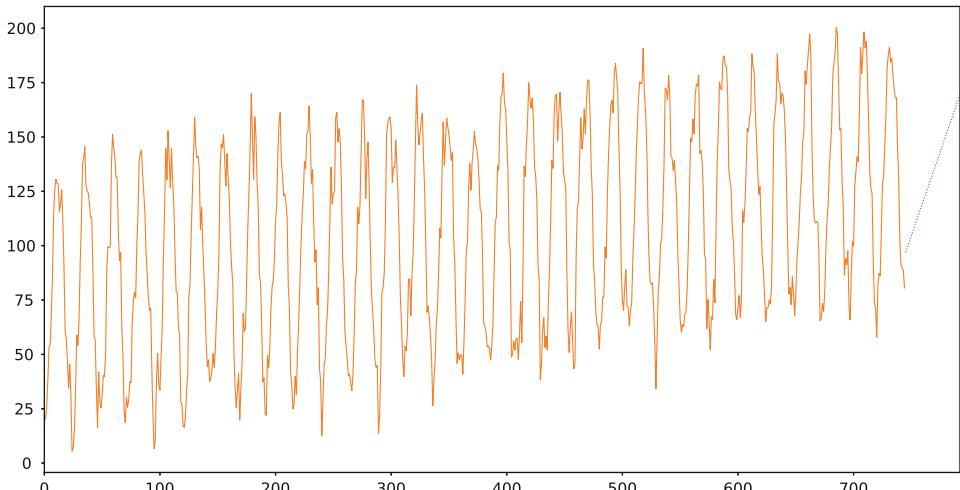


Abb. 4.6 Exponentielles Glätten zweiter Ordnung

b_t und der Saison s_t . Die Prognose für h Schritte in die Zukunft ergibt sich dann zu:

$$\hat{y}_{t+h} = l_t + h b_t + s_{t-L+1+(h-1)modL} \quad (4.8)$$

Dabei muss die Saisonlänge L , also die Anzahl der Datenpunkte, nach der sich die Saison wiederholt, bekannt sein. Entsprechend existieren nun drei Komponenten, in denen exponentielles Glätten eingesetzt wird:

$$\begin{aligned} l_t &= \alpha(y_t - s_{t-L}) + (1 - \alpha)(l_{t-1} + b_{t-1}) \\ b_t &= \beta(l_t - l_{t-1}) + (1 - \beta)b_{t-1} \\ s_t &= \gamma(y_t - l_{t-1} - b_{t-1}) + (1 - \gamma)s_{t-L} \end{aligned} \quad (4.9)$$

Der Glättungsfaktor γ nimmt dabei Einfluss auf die Saisonalität und liegt wieder zwischen null und $1 - \alpha$. Das exponentielle Glätten dritter Ordnung benötigt schon sechs verschiedene Modellparameter ($\alpha, \beta, \gamma, l_0, b_0, s_0$). Diese müssen so gewählt werden, dass die angepassten Werte (*Fitted Values*) sich möglichst den Originaldaten annähern.

Eine solche Prognose kann wieder recht einfach mithilfe von Python durchgeführt werden, wobei die Belegung der sechs Modellparameter bereits optimiert wird. Der folgende Programmcode erstellt eine entsprechende Prognose für 48 Schritte in die Zukunft:

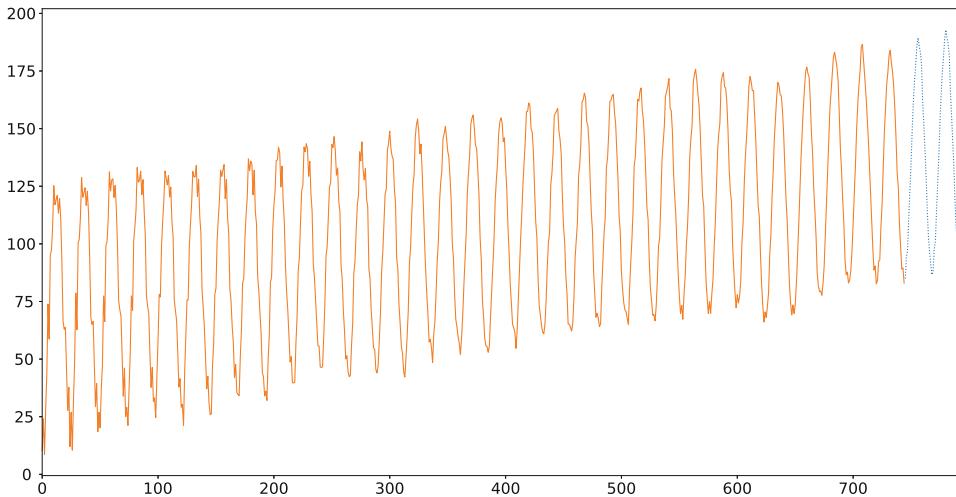


Abb. 4.7 Exponentielles Glätten dritter Ordnung

```
1 from statsmodels.tsa.holtwinters import ExponentialSmoothing
2 fit = ExponentialSmoothing(
3     data["data"], seasonal_periods=24,
4     trend='add', seasonal='add').fit()
5 fcast = fit.forecast(48)
6 fcast.plot()
7 fit.fittedvalues.plot()
8 plt.show()
```

Abb. 4.7 zeigt das Ergebnis der Vorhersage mittels exponentiellen Glättens dritter Ordnung. Der Verlauf der Prognose (blauer Teil der Kurve) beinhaltet nun auch die zu beobachtende saisonale Schwankung. Die Last eines RLM-Kunden kann mit diesem Verfahren gut prognostiziert werden, wobei für unterschiedliche Tage (Wochentag, Wochenende) evtl. unterschiedliche Prognosemodelle aufgesetzt werden müssen.

In Excel existiert im Menü „Daten“ seit der Version 2016 das sog. Prognoseblatt. Dort kann ebenfalls eine Prognose für Zeitreihendaten erzeugt werden. Das unterliegende Verfahren basiert ebenfalls auf dem exponentiellen Glätten dritter Ordnung. Wie in Excel üblich, steht dort auch eine grafische Oberfläche zur Verfügung, um das Prognoseverfahren zu konfigurieren. Abb. 4.8 zeigt den zugehörigen Assistenten in Excel, um eine Prognose zu konfigurieren.



Abb. 4.8 Das Prognoseblatt von Excel, mit dessen Hilfe Prognosen auf Basis exponentieller Glättung dritter Ordnung durchgeführt werden können

4.2 Optimierung

Im letzten Abschnitt haben wir uns mit Prognosemodellen beschäftigt. Dabei ging es z. B. um die Frage, wie viel Last in einem Bilanzkreis in der Zukunft zu erwarten ist. Aus dieser Erkenntnis müssen bei einem Energieversorger dann entsprechende Handlungen abgeleitet werden:

- Auf der Basis einer solchen Lastprognose sollen Kraftwerke möglichst kostenoptimal eingesetzt werden.
- Ein Pumpspeicherwerk soll gegenüber den vorausgesagten Börsenpreisen optimal bewirtschaftet werden.
- Die Residuallast in einem Bilanzkreis soll kostenoptimal in Standardbörsenprodukte zerlegt werden.
- Ein Verteilnetz soll so ausgelegt werden, dass es die Energieflüsse optimal bewältigen kann.

In all diesen Fragestellungen ist man auf der Suche nach optimalen Konfigurationen eines Systems. Solche Aufgaben nennt man **Optimierungsprobleme**. Der Prozess, der letztendlich zu einem optimalen Ergebnis führt, wird in die Schritte Planung, Entscheidung, Durchführung und Kontrolle unterteilt (siehe [14, S. 1]). Der Wissenschaftsdisziplin des *Operations Research* (OR) befasst sich dabei mit dem Teilproblem der Planung. Das Ziel der Planung ist eine Vorbereitung von optimalen Entscheidungen. Entsprechend muss das Entscheidungsproblem anhand eines Optimierungs- oder Simulationsmodells abgebildet werden. Anschließend kann das Problem auf Basis mathematischer Verfahren gelöst und die Lösung bewertet werden. Im Allgemeinen besteht ein Optimierungsmodell aus den folgenden Bestandteilen (siehe [14, S. 4]):

- eine Zielfunktion $z = F(x)$,
- eine Menge von Nebenbedingungen $g_i(x) \geq 0$.

Dabei ist x ein Vektor von Variablen x_1, x_2, \dots, x_n , für die meist gilt: $x_i \geq 0$. Aufgabe des Optimierungsverfahrens ist es dann, Belegungen für die Variablen z zu finden, sodass einerseits die Nebenbedingungen nicht verletzt werden und andererseits der Wert der Zielfunktion minimiert (z. B. Kosten) oder maximiert (z. B. Ertrag) wird.

Von **linearer Optimierung** spricht man, wenn die Zielfunktion und alle Nebenbedingungen lineare Funktionen sind. Solche Modelle spielen in der Energiewirtschaft eine besondere Rolle und können mit gängigen Verfahren gelöst werden. Oft sind die gegebenen Zusammenhänge von Natur aus linear, z. B. Kosten als lineare Summe mehrerer Bestandteile, oder können stückweise linearisiert werden. Lineare Modelle haben die folgende Form:

- Zielfunktion (z. B. zu maximieren): $z = c^T x$,
- Nebenbedingungen: $Ax \leq b$.

Die beiden Vektoren $c, b \in \mathbb{R}^n$ und die Matrix $A \in \mathbb{R}^{m \times n}$ sind sog. Koeffizienten des Modells.

Bei einem deterministischen Modell setzt man die Koeffizienten als bekannt voraus. Ein solches Modell wird bei Entscheidungen bei Sicherheit verwendet. Bei einem stochastischen Modell hingegen sind die Parameter als Zufallsvariable mit einer entsprechenden

Verteilung zu interpretieren. Diese Modelle werden bei Entscheidung unter Unsicherheit genutzt. Wir werden uns hier ausschließlich mit deterministischen Modellen befassen.

Beispiel 4.2.1 Im Folgenden wollen wir ein einfaches Beispiel entwickeln. Dabei nehmen wir an, wir wären Kraftwerksbetreiber und im Besitz eines Steinkohle- und eines Gaskraftwerks mit einer maximalen Leistung von 200 MW pro Kraftwerk. Nun haben wir insgesamt 300 MWh an der Strombörse verkauft und müssen diese Energie mit den beiden Kraftwerken über eine Stunde lang erzeugen, indem wir die Kraftwerke mit einer entsprechenden Leistung laufen lassen. Dabei wollen wir aber möglichst viel an unseren Kraftwerken verdienen. Der Gewinn aus beiden Kraftwerken ist unterschiedlich hoch. Wir verdienen mit jeder MWh aus dem Steinkohlekraftwerk 10 €, mit derselben Energiemenge aus einem Gaskraftwerk jedoch nur 2 €. Wir dürfen zudem nur maximal 230 t CO₂ produzieren. Das Steinkohlekraftwerk stößt für jede MWh ca. 1 t CO₂ aus, das Gaskraftwerk mit 575 kg CO₂ pro MWh deutlich weniger. Wir müssen nun entscheiden, mit welcher Leistung die beiden Kraftwerke gefahren werden müssen, sodass

- wir einerseits unseren Profit maximieren
- und andererseits die Kraftwerke innerhalb der erlaubten Bereiche betrieben werden.

Für das Optimierungsmodell stellen wir zunächst die Zielfunktion auf. Da wir an einem möglichst hohen Profit interessiert sind, handelt es sich um ein Maximierungsproblem. Der Gesamtprofit ergibt sich als Summe der Profite der beiden Kraftwerke. Diese sind abhängig davon, wie viele MWh in jedem Kraftwerk erzeugt werden. Dabei können wir ausnahmsweise den Wert der Energie gleich dem Wert der Leistung setzen, weil wir einen Optimierungszeitraum von einer Stunde betrachten. Es ergibt sich die folgende lineare Zielfunktion, die von den zwei Entscheidungsvariablen KohleMWh und GasMWh abhängen:

$$z \text{ €} = 10 \text{ €/MWh} \cdot KohleMWh + 2 \text{ €/MWh} \cdot GasMWh \quad (4.10)$$

Da wir unseren Profit maximieren wollen, müssen für die Entscheidungsvariablen nun Belegungen gefunden werden, die einen möglichst hohen Wert erzeugen. Dabei können die Entscheidungsvariablen aber keine beliebigen hohen Werte annehmen. Gäbe es keine weiteren Restriktionen an die Entscheidungsvariablen, wäre das Optimierungsproblem unbeschränkt. Die Maximalleistungen der beiden Kraftwerke stellen aber technische Restriktionen dar, die nicht verletzt werden dürfen. Darüber hinaus können Kraftwerke natürlich auch keine negativen Leistungswerte ansteuern. Es ergeben sich dadurch die folgenden Nebenbedingungen:

$$\begin{aligned}
 & KohleMWh + GasMWh = 300 \text{ MWh} \\
 & KohleMWh \leq 200 \text{ MWh} \\
 & GasMWh \leq 200 \text{ MWh} \\
 & KohleMWh \geq 0 \text{ MWh} \\
 & GasMWh \geq 0 \text{ MWh}
 \end{aligned} \tag{4.11}$$

Eine weitere, nichttriviale Nebenbedingung ergibt sich durch die Grenze des CO₂-Ausstoßes. Wir formulieren die Nebenbedingungen wie folgt:

$$KohleMWh + 0,575 \cdot GasMWh \leq 230t \tag{4.12}$$

Das Optimierungsproblem aus unserem Beispiel hängt nur von zwei Entscheidungsvariablen ab. In einem solchen Fall kann das Optimierungsproblem auch grafisch dargestellt und gelöst werden (siehe [11, S. 33–36]). Jede Nebenbedingung entspricht einer Geraden in der Ebene, die den Rand des zulässigen Lösungsraums beschreibt. Der zulässige Lösungsraum ist dann ein zweidimensionaler Polyeder. Es ist einleuchtend, dass eine optimale Lösung des Problems auf dem Rand dieses Polyeders liegen muss. Abb. 4.9 zeigt die grafische Darstellung des zulässigen Bereichs für unser Optimierungsproblem.

Auch die Zielfunktion kann in der Lösungsebene als Gerade dargestellt werden. Diese Gerade wird als **Isogewinngerade** bezeichnet. Um das Problem rein grafisch zu lösen, wird der Zielfunktionswert auf einen Startwert festgelegt, z. B. 0. Erhöht man nun den Zielfunktionswert schrittweise, so verschiebt sich die Isogewinngerade parallel über den Lösungsraum. Dies kann so lange fortgeführt werden, bis eine äußere Ecke des Polyeders berührt wird. Dort ist dann die optimale Lösung zu finden.

Nicht alle Modelle, die aufgestellt werden können, sind auch lösbar. An den zulässigen Bereich, der durch die Nebenbedingungen gegeben ist, sind einige Bedingungen geknüpft, damit ein lösbares Optimierungsproblem entsteht.

- Er darf nicht leer sein.
- Er muss beschränkt sein.
- Er muss konvex sein, d. h., eine Gerade, die zwei Punkte des zulässigen Bereichs verbindet, muss auch vollständig im zulässigen Bereich liegen.

Die Anzahl der Entscheidungsvariablen gibt die Dimension des zulässigen Bereichs vor. Im Allgemeinen ist daher der Lösungsraum keine Ebene im zweidimensionalen Bereich. Bei n Variablen entsteht ein **n -Polyeder**. In diesem Fall kann das Modell nicht mehr rein grafisch gelöst werden. Es wird ein mathematisches Verfahren, ein Algorithmus zur Lösungssuche benötigt.

Das älteste und bekannteste Verfahren nennt sich die **Simplex-Methode**. Die Grundidee wurde von Georg B. Dantzig im Jahr 1947 vorgestellt und zur Lösung von Planungs-

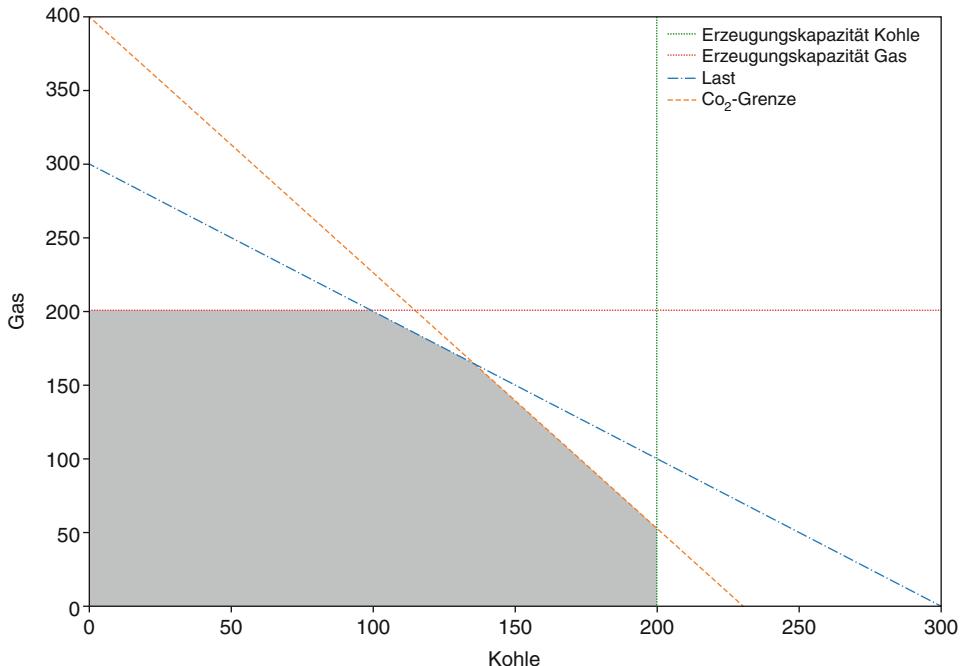


Abb. 4.9 Grafische Darstellung des zulässigen Bereichs für die Optimierung des Kraftwerkseinsatzes

problemen bei der U.S. Airforce genutzt (siehe [1]). Die Simplex-Methode besteht dabei aus zwei Schritten ([11, S. 44–56]). In der ersten Phase wird eine zulässige Basislösung bestimmt, die auf dem Rand des Polyeders liegen muss. In der zweiten Phase werden benachbarte Ecken besucht. Der Zielfunktionswert muss sich dabei entweder verbessern oder zumindest nicht weiter verschlechtern. Dies wird so lange wiederholt, bis keine Verbesserung mehr erzielt werden kann. In diesem Fall ist das Optimum gefunden.

Das Simplex-Verfahren garantiert, dass nach endlich vielen Schritten eine exakte und global optimale Lösung des Problems gefunden wird oder die Unlösbarkeit feststeht. Dabei hat der Simplex-Algorithmus eine *Worst-Case*-Komplexität von $O(k^n)$ (siehe [5]). Mit dem Interior-Point-Verfahren existiert noch ein weiterer Algorithmus zur Lösung von linearen Optimierungsproblemen. Dieses hat eine *Worst-Case*-Komplexität von $O(\frac{1}{\epsilon} \sqrt{n})$, ist also theoretisch effizienter als das Simplex-Verfahren. Für typische Problemgrößen erzielt das Simplex-Verfahren in der Praxis aber meist eine bessere Laufzeit. Da Algorithmen existieren, die ein lineares Optimierungsproblem effizient lösen können, gehören diese Probleme zur Klasse P , die in polynomieller Zeit lösbar sind.

Den Simplex-Algorithmus selbst zu implementieren, ist wenig sinnvoll, da sehr gute Implementierungen am Markt existieren. Für die Lösung von linearen (ganzzähligen) Optimierungsproblemen existieren hervorragende Programmpakete, sog. Solver. Im kommerziellen Umfeld gehören der GUROBI Optimizer bzw. das IBM Ilog CPLEX

Optimization Studio sicherlich zu den bekanntesten und besten Lösungen. Neben der Lösung von linearen Optimierungsaufgaben sind diese Solver in der Lage, auch weitere Optimierungsaufgaben zu lösen, z. B. auch ganzzahlige Probleme oder Modelle mit quadratischen Bestandteilen. Neben den kommerziellen Lösungen existiert aber auch eine Vielzahl von freien und quelloffenen Sovern. Hier ist das GNU Linear Programming Kit (GLPK) oder SYMPHONY zu erwähnen. Allerdings sind diese freien Solver z. T. signifikant weniger performant als die kommerziellen Lösungen (siehe [9]).

Um unser Optimierungsproblem mit einem Solver lösen zu können, muss es in einer Sprache ausgedrückt werden, welche durch einen Solver verstanden wird. Ansatze jeder Solver besitzt allerdings seine eigene Modellierungssprache, um Optimierungsprobleme darzustellen. Darüber hinaus existieren aber auch solver-unabhängige Modellierungssprachen, wie z. B. General Algebraic Modeling System (GAMS) oder A Mathematical Programming Language (AMPL). Diese Sprachen erlauben es, ein mathematisches Problem wie in einer Programmiersprache zu formulieren, damit ein Solver es lösen kann. Ein großer Vorteil dieser Sprachen ist, dass das Optimierungsmodell von den Eingabedaten (z. B. Marktpreise) getrennt dargestellt werden kann (siehe z. B. [10, S. 46]). Besonders AMPL ist beliebt, da die Sprache sehr nah an der mathematischen Notation ist (siehe [3]).

Für Python existiert mit dem Modul **Pyomo** eine objektorientierte Modellierungssprache, um Optimierungsprobleme zu formulieren (siehe [16]). Mithilfe von Pyomo können nicht nur lineare Optimierungsprobleme, sondern auch viele andere Problemklassen, z. B. quadratische oder stochastische Modelle, abgebildet werden. Pyomo verfolgt dabei einen ähnlichen Ansatz wie GAMS oder AMPL, der Vorteil liegt aber darin, dass Pyomo auf Python aufsetzt und somit auch alle anderen Fähigkeiten von Python genutzt werden können. Wurde das Optimierungsproblem mit Pyomo abgebildet, kann das Problem an einen Solver zur Lösung übergeben werden. Pyomo unterstützt dazu viele Solver direkt (z. B. CPLEX, Gurobi, glpk), kann aber ein Optimierungsmodell auch in Form von GAMS oder AMPL exportieren.

In Pyomo ist ein Optimierungsmodell (meist) ein Objekt der Klasse `ConcreteModel`. Um das Modell aufzubauen, wird das Objekt schrittweise mit Angaben zu den Entscheidungsvariablen, der Zielfunktion und den Nebenbedingungen erweitert. Entscheidungsvariablen werden dem Modell über Objekte der Klasse `Var` hinzugefügt. Der Konstruktor der Klasse erlaubt diverse Parameter, z. B. die Definition von Gültigkeitsbereichen bzw. einer Vorbelegung. Der folgende Programmcode erstellt ein neues Pyomo-Modell und definiert die Entscheidungsvariablen `Kohle` und `Gas` unseres einfachen Kraftwerksmodells.

```
1 from pyomo.environ import *
2 model = ConcreteModel()
3 model.Kohle = Var(domain=NonNegativeReals, bounds=(0,200))
4 model.Gas = Var(domain=NonNegativeReals, bounds=(0,200))
```

Beide Entscheidungsvariablen gehören dabei zur Domäne der nichtnegativen Realzahlen. Der Gültigkeitsbereich der Variablen wird durch den Parameter bounds weiter eingeschränkt.

Die Zielfunktion wird anhand der Klasse `Objective` definiert. Die zu optimierende Funktion kann als Ausdruck im Parameter `expr` übergeben werden. Der folgende Programmcode fügt dem Modell die Zielfunktion für unser Kraftwerkmodell hinzu.

```
1 model.profit = Objective(expr = 10 * model.Kohle + 2 * model.Gas,
                           sense = maximize)
```

Nebenbedingungen werden durch die Klasse `Constraint` abgebildet. Die einzuhaltende Bedingung ist ein boolescher Ausdruck und wird im Parameter `expr` übergeben. Der folgende Programmcode fügt dem Optimierungsmodell die beiden Nebenbedingungen hinzu, die in unserem Modell nicht verletzt werden dürfen.

```
1 model.Demand = Constraint(expr = model.Kohle + model.Gas == 300)
2 model.Co2Bound = Constraint(expr = model.Kohle + 0.575 * model.Gas
                             <= 230)
```

Um das Modell an einen Solver zu übergeben, wird ein Objekt der Klasse `SolverFactor` erzeugt. Dazu muss ein entsprechender Solver auf dem System installiert sein. Mithilfe der Methode `solve()` kann dann der Optimierungslauf gestartet werden. Im folgenden Programmbeispiel wird davon ausgegangen, dass das Solver-Paket „glpk“ auf dem Rechner installiert ist.

```
1 opt = SolverFactory('glpk')
2 result = opt.solve(model)
3 result.write()
```

Die Ergebnisse des Optimierungslaufs werden automatisch in das Modell zurückgeladen. Die Entscheidungsvariablen können dann aus dem Modell ausgelesen und weiterverarbeitet werden. Der folgende Programmcode gibt den Inhalt der beiden Entscheidungsvariablen auf der Konsole aus.

```
1 print("Kohle = ", model.Kohle.value)
2 print("Gas = ", model.Gas.value)
```

Abb. 4.10 zeigt das Ergebnis eines Optimierungslaufs mit Pyomo und dem Solver glpk. Es wurde ein optimales Ergebnis gefunden. Der größte Profit wird genau dann erzeugt, wenn das Kohlekraftwerk mit 135,29 MW und das Gaskraftwerk mit 164,71 MW betrieben wird.

```
# -----
#   Solution Information
# -----
Solution:
- number of solutions: 0
  number of solutions displayed: 0
KohleMWh = 135.294117647059
GasMWh = 164.705882352941
[ 0.08] Creating model
[ 0.08] Applying solver
[ 0.11] Processing results
Number of solutions: 1
Solution Information
  Gap: 0.0
  Status: feasible
  Function Value: 1682.352941176472
Solver results file: results.json
[ 0.12] Applying Pyomo postprocessing actions
[ 0.12] Pyomo Finished
```

Abb. 4.10 Ergebnis eines Optimierungslaufs mit Pyomo und dem Solver glpk

Im Folgenden soll nun ein etwas komplexeres Optimierungsproblem gelöst werden. Wir nutzen dazu erneut Pyomo, um das Optimierungsmodell aufzubauen. Nehmen wir an, wir besäßen ein Pumpspeicherkraftwerk. Abb. 4.11 zeigt ein Foto eines der ältesten Pumpspeicherkraftwerke in Deutschland, das Koepchenwerk in Herdecke.

Ein Pumpspeicherkraftwerk stellt einen Energiespeicher dar. Wenn elektrische Energie günstig ist, kann Wasser in ein Oberbecken gepumpt werden. Ist Energie teuer, kann durch Ablassen des Wassers in einem Generator wieder elektrische Energie erzeugt werden. Ein solcher Speicher kann z. B. dazu genutzt werden, die Preisunterschiede zwischen Base-Load und Peak-Load auszunutzen (siehe auch Abschn. 2.1).

Sowohl Pumpe als auch Generator besitzen im Koepchenwerk eine maximale Leistung von 153 MW. Das Oberbecken hat einen maximalen Arbeitsinhalt von 590 MWh, der Gesamtwirkungsgrad wird mit 80 % angenommen. Wir besitzen eine stündliche Preisprognose für elektrische Energie. Eine solche Prognose wird auch als Hourly Price Forward Curve (HPFC) bezeichnet. Wir wissen daher, was wir für das Befüllen des Speichers bezahlen müssen sowie wie viel wir verdienen können, wenn wir das Wasser wieder in elektrische Energie zurückwandeln. Unsere Aufgabe ist es nun, den Füllstand des Speichers an einem Tag zu bewirtschaften. Dabei wollen wir natürlich erneut möglichst viel Geld verdienen. Die Frage ist nun, wann das Speicherbecken gefüllt und wann wieder entleert werden muss.

Als Optimierungsziel wollen wir möglichst viel Profit erwirtschaften, d. h., die Preisdifferenz zwischen verkaufter Energie im Generatorbetrieb und gekaufter elektrischer Energie im Pumpbetrieb muss möglichst groß sein. Abb. 4.12 zeigt die Strompreiskurve, gegen die wir im Folgenden das Pumpspeicherkraftwerk optimieren wollen.



Abb. 4.11 Das Koepchenwerk in Herdecke

In unserem Optimierungsmodell führen wir für jede Stunde die Entscheidungsvariablen $pump_i$ und $generator_i$ ein. Die Variablen $pump_i$ zeigen die Leistungen in MW an, mit der in der Stunde i Wasser in das Oberbecken gepumpt wird. Die Variablen $generator_i$ zeigen die Leistungen in MW an, mit der in der Stunde i Energie im Generatorbetrieb erzeugt wird. Für diese Variablen müssen später günstige Belegungen gefunden werden.

Multiplizieren wir $pump_i$ mit dem Preis $preis_i$ in der jeweiligen Stunde, kommen wir zu den Kosten für den Pumpbetrieb in Stunde i . Multiplizieren wir $generator_i$ mit dem Preis $preis_i$, erhalten wir den Ertrag. Dabei sind jedoch die 20 % Speicherverluste zu berücksichtigen. Es ergibt sich dann die folgende, zu maximierende Zielfunktion, die einen Profit in € ergibt. Die Einheiten der Elemente werden der Einfachheit halber nicht aufgeführt.

$$profit = \sum_{i=1}^{24} 0,8 \cdot generator_i \cdot preis_i - \sum_{i=1}^{24} pump_i \cdot preis_i \quad (4.13)$$

Wir müssen natürlich wieder einige Nebenbedingungen einhalten. Diese Nebenbedingungen spannen den n -Polyeder auf, auf dessen Ecke die optimale Lösung liegt. Zunächst muss die Pump- und Generatorleistung in jeder Stunde innerhalb der zulässigen Grenzen, also zwischen 0 MW und 153 MW, liegen.

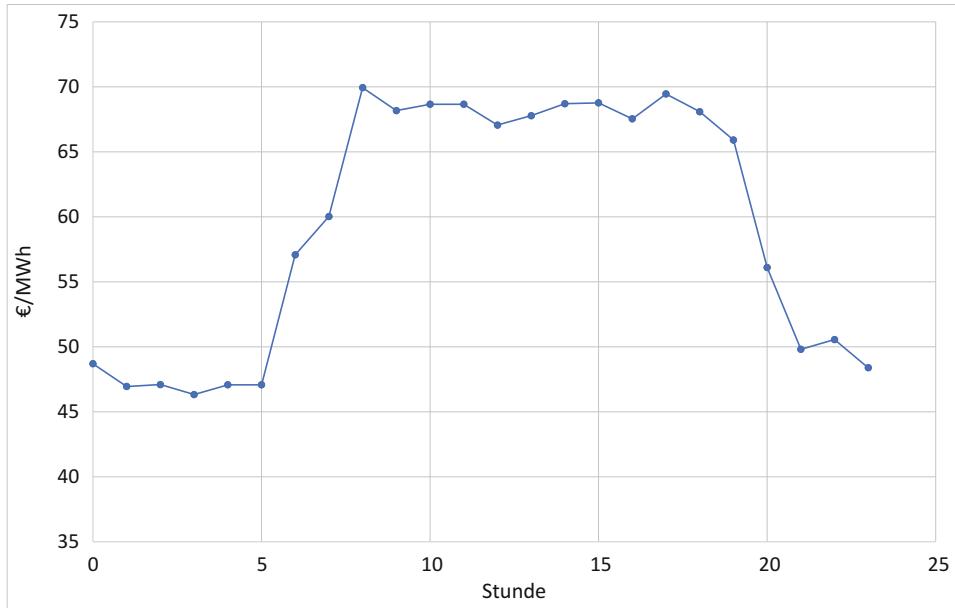


Abb. 4.12 Die beispielhafte HPFC als Eingangsdaten für die Bewirtschaftung eines Pumpspeicher-kraftwerks

$$\begin{aligned} 0 \leq pump_i &\leq 153 \text{ MW} \\ 0 \leq generator_i &\leq 153 \text{ NW} \end{aligned} \tag{4.14}$$

Zudem darf auch der Füllstand keine beliebigen Werte annehmen. Um allerdings Bedingungen an den Füllstand zu stellen, müssen wir diesen erst durch Entscheidungsvariablen $fill_i$ greifbar machen. Der Füllstand in jeder Stunde ergibt sich aus dem Füllstand der letzten Stunde und der Pump- und Generatorleistung. Zusätzlich fordern wir, dass der Füllstand nie negativ werden und der Füllstand auch nicht über 590 MWh ansteigen darf. Zu guter Letzt können wir noch festlegen, dass der Füllstand am Anfang und am Ende eines Tages einen bestimmten Wert erreichen muss. Mit einem halb vollen Becken hat man am nächsten Tag eine gute Ausgangsbasis. Es ergeben sich dann die folgenden Nebenbedingungen:

$$\begin{aligned} fill_i &= fill_{i-1} + pump_i - generator_i \\ 0 \leq fill_i &\leq 590 \text{ MWh} \\ fill_1 &= 300 \text{ MWh} \\ fill_{24} &= 300 \text{ MWh} \end{aligned} \tag{4.15}$$

Im Gegensatz zum ersten Beispiel suchen wir hier nach mehreren Zeitreihen als Optimierungsergebnis, der jeweils stündlichen Generatorleistung, der Pumpleistung und dem Füllstand. Wir könnten dazu für jede Stunde eigene Entscheidungsvariablen einführen, was entsprechend viel Tipparbeit darstellt. Ein solches Modell kann mit Pyomo aber sehr viel einfacher aufgebaut werden. Entscheidungsvariablen können mit einem oder mehreren Indizes dimensionaliert werden. Im Ergebnis entsteht dann ein Feld (Array) von Entscheidungsvariablen.

Zur Erstellung von Indizes kann die Klasse `RangeSet` von Pyomo genutzt werden. Für unseren Pumpspeicher handelt es sich bei den Indizes um die Stunden. Den Entscheidungsvariablen können wieder direkt Gültigkeitsgrenzen übergeben werden. Diese gelten dann für alle Indizes. Im folgenden Programmausschnitt werden die notwendigen, indizierten Entscheidungsvariablen mithilfe von Pyomo definiert:

```

1 from pyomo.environ import *
2 model = ConcreteModel()
3 model.hours = RangeSet(1, 24)
4 model.pump = Var(model.hours, domain=NonNegativeReals, bounds
5   =(0,153))
6 model.generator = Var(model.hours, domain=NonNegativeReals, bounds
7   =(0,153))
6 model.filling = Var(RangeSet(0, 24), domain=NonNegativeReals,
8   bounds=(0,590))
```

Eine Basisinformation für unser Optimierungsmodell sind die stündlichen Energiepreise. Diese könnten wir fest im Modell hinterlegen, was aber keine gute Lösung wäre, da sich dieser Teil des Modells sicher häufig ändert. In Pyomo werden solche Daten als Parameter bezeichnet. Sie können mithilfe der Klasse `Param` im Modell definiert werden. Spätestens, wenn Nebenbedingungen oder Zielfunktionen sich auf solche Daten beziehen, müssen die Daten geladen worden sein. In unserem Fall handelt es sich um stündliche Preise. Der folgende Programmausschnitt lädt die benötigte HPFC aus einer CSV-Datei und legt die Daten als Objekt der Klasse `Param` im Modell ab.

```

1 data = pd.read_csv('prices.csv', sep=',', index_col=0)
2 prices = data.to_dict()['price']
3 model.price = Param(model.hours, initialize=prices)
```

Unsere Entscheidungsvariablen sind Felder. Entsprechend müssen wir Nebenbedingungen an einige oder alle Indizes der Felder richten. Für die Füllstandsbedingung am Anfang und am Ende des Tages ist dies noch sehr leicht:

```

1 model.filling_start_limit = Constraint(expr = model.filling[0] ==
300)
2 model.filling_end_limit = Constraint(expr = model.filling[24] ==
300)

```

Komplexer ist hingegen der Ausdruck, der den Füllstand überhaupt erst erzeugt. Hierfür kann eine Python-Funktion erstellt werden, die als Regel an eine Nebenbedingung übergeben wird.

```

1 def filling_limit(model, h):
2     return model.filling[h] == model.pump[h] - model.generator[h]
3                                         + model.filling[h-1]
4 model.filling_limit = Constraint(model.hours, rule=filling_limit)

```

Auch für die Zielfunktion wird eine Python-Funktion erstellt. Diese berechnet den Profit über die aktuelle Belegung der Generator- und Pumpenwerte. Die Funktion wird der Klasse `Objective` bei der Erstellung der Zielfunktion übergeben.

```

1 def objective_rule(model):
2     revenue = sum(model.generator[i] * 0.8 * model.price[i] for i
3                   in model.hours)
4     costs = sum(model.pump[i] * model.price[i] for i in model.
5                  hours)
6     return revenue - costs
7
8 model.obj = Objective(rule=objective_rule, sense=maximize)

```

Das Modell kann dann wie zuvor an einen Solver übergeben und gelöst werden.

```

1 opt = SolverFactory('glpk')
2 result = opt.solve(model)
3 result.write()

```

Nach der Optimierung werden die Ergebnisse des Solvers wieder zurück ins Modell gelesen. Die Daten können dann in Python weiterverarbeitet werden. Es bietet sich z. B. an, die Zeitreihen mithilfe von Matplotlib darzustellen. Dazu werden die Daten aus dem Modell zunächst in ein Data-Frame zurückgelesen.

```

1 for h in model.hours:
2     data.at[h, 'filling'] = model.filling[h].value
3     data.at[h, 'pump'] = model.pump[h].value
4     data.at[h, 'generator'] = model.generator[h].value
5
6 data.plot(y=['filling','pump','generator'])
7 plt.show()

```

Abb. 4.13 zeigt das Ergebnis des Solvers nach der Optimierung des Pumpspeichers. Wasser soll offenbar vor allem am Anfang des Tages, wenn elektrische Energie günstig ist, in das Oberbecken gepumpt und gegen Mittag wieder abgelassen werden.

Viele praktische Probleme können bereits mit einer rein linearen Modellierung gelöst werden. Oft jedoch müssen diskrete Zustände (Ja/Nein) im Modell abgebildet werden. Auch aussagenlogische Ausdrücke (wenn, dann) sind keine Seltenheit. Um solche Zustände in einem Optimierungsmodell abbilden zu können, werden Entscheidungsvariablen benötigt, die nur bestimmte ganzzahlige Werte annehmen dürfen, z. B. nur 0 oder 1. Gelten solche Bedingungen nur für einen Teil der Entscheidungsvariablen, spricht man auch von gemischtganzzahligen Optimierungsproblemen (Mixed-Integer Linear Programming (MILP)).

Sowohl der Simplex- als auch der Interior-Point-Algorithmus können lediglich lineare Optimierungsprobleme lösen. Dabei sind alle Entscheidungsvariablen kontinuierlich. MILP-Probleme sind wesentlich schwieriger zu lösen. Man kann zeigen, dass solche

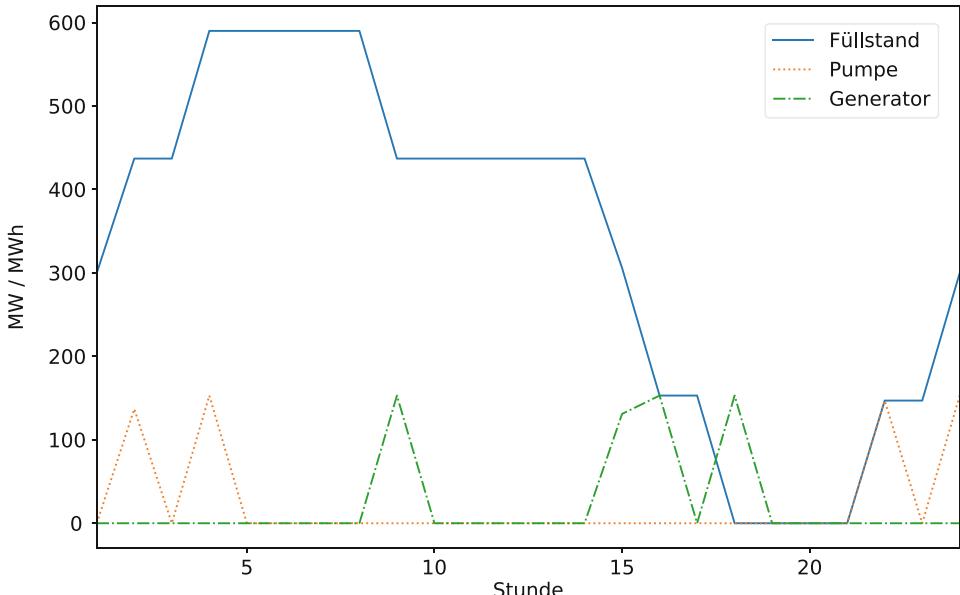


Abb. 4.13 Das Ergebnis der Optimierung der Bewirtschaftung des Pumpspeichers

Probleme zur Klasse NP, also den nicht in polynomialer Zeit lösbar Problemen, gehören. Zur Lösung solcher Probleme werden in den Solvoren unterschiedliche Verfahren eingesetzt, z. B. Branch and Bound oder Branch and Cut (siehe z. B. [11, S. 148–155]). Dabei wird das Optimierungsproblem zunächst so gelöst, als seien für alle Entscheidungsvariablen kontinuierliche Werte erlaubt, was als **lineare Relaxation** bezeichnet wird. Danach wird ein **Suchbaum** aufgespannt, der mithilfe von **Tiefensuche** durchlaufen wird. Jeder Knoten steht dabei für ein um weitere Nebenbedingungen erweitertes Optimierungsproblem. Das Ziel ist es dabei, eine Belegung zu finden, welche die ursprünglichen Ganzzahligkeitsbedingungen erfüllt.

Im Folgenden greifen wir wieder unser Beispiel vom Anfang auf. Dieses Mal besitzen wir ein Braunkohle-, ein Steinkohle- und ein Gaskraftwerk. Für jede Stunde des Tages wollen wir eine vorgegebene Menge Energie aus diesem Kraftwerkspark erzeugen. Als Optimierungsergebnis wünschen wir uns also für alle Kraftwerke die anzusteuernden Betriebspunkte für jede Stunde h des Tages. Diese Entscheidungsvariablen haben einen kontinuierlichen Wertebereich. Der Solver muss diese im Rahmen der Nebenbedingungen so festlegen, dass minimale Betriebskosten entstehen. Wir müssen aber noch einige weitere Nebenbedingungen einhalten:

- Wenn ein Kraftwerk genutzt wird, kann es nur in bestimmten Leistungsbereichen betrieben werden.
- Wird ein Kraftwerk angefahren, fallen Anfahrtskosten an.
- Die Leistung eines Kraftwerks kann nicht beliebig schnell steigen oder sinken, es müssen sog. Rampen eingehalten werden.

Etwas knifflig ist die Forderung, dass ein Kraftwerk nur in bestimmten Leistungsbereichen betrieben werden darf. Ein Kraftwerk ist demnach entweder ausgeschaltet ($\text{Erzeugung}=0$) oder befindet sich zwischen einem minimalen Leistungswert (P_{\min}) und einem maximalen Leistungswert (P_{\max}). Dieses Modellierungsproblem wird als **Schwellwertproblem** bezeichnet (siehe [11, S. 100 ff.]). Im Optimierungsmodell führen wir für jede Stunde und für jedes Kraftwerk eine binäre Entscheidungsvariable `hasgeneration` ein, die lediglich die Werte „0“ oder „1“ annehmen können. Diese bilden ab, ob das jeweilige Kraftwerk in Stunde h produziert oder nicht. In Pyomo führen wir dazu indizierte Entscheidungsvariablen ein, die als „binär“ deklariert werden:

```
1 model.hasgeneration = Var(model.plants, RangeSet(0, 24), within=Binary)
```

Über Nebenbedingungen erzwingen wir nun die korrekte Belegung dieser Variablen. Soll ein Kraftwerk eingesetzt werden, kann die binäre Variable `hasgeneration` nur den Wert 1 annehmen. Das Kraftwerk darf dann nur zwischen bestimmten Betriebspunk-

ten eingesetzt werden. Entsprechend werden dafür zwei Nebenbedingungen eingeführt, jeweils für Pmin und Pmax.

```

1 def fullfills_p_max(model, p, h):
2     return model.generation[p, h] <= model.p_max[p] * model.
3         hasgeneration[p, h]
4
5 model.fullfills_p_max = Constraint(model.plants, model.hours, rule
6         =fullfills_p_max)
7
8 def fullfills_p_min(model, p, h):
9     return model.generation[p, h] >= model.p_min[p] * model.
10        hasgeneration[p, h]
11
12 model.fullfills_p_min = Constraint(model.plants, model.hours, rule
13         =fullfills_p_min)

```

Der Solver belegt bei der Optimierung die binären Entscheidungsvariablen *hasgeneration* in jeder Stunde und für jedes Kraftwerk mit den Werten „0“ oder „1“. Diese Information kann nun dazu genutzt werden, um die Startkosten eines Kraftwerks zu modellieren. Ein Kraftwerksstart in Stunde h tritt genau dann auf, wenn ein Kraftwerk in der letzten Stunde ($h - 1$) nicht am Netz war, in Stunde i aber schon. Stellt man sich die binären Variablen $hasgeneration_i$ als logische Literale vor (wahr/falsch), kann das Vorhandensein einer Kraftwerksanfahrt in Stunde i als logische Aussage formuliert werden:

$$\neg hasgeneration_{i-1} \wedge hasgeneration_i \rightarrow hasstartup_i \quad (4.16)$$

Solche aussagenlogischen Ausdrücke können in Nebenbedingungen eines Optimierungsproblems überführt werden (siehe [11, S. 112–116]). Jede Aussage muss dazu zunächst in die konjunktive Normalform (KNF) überführt werden. Eine solche logische Formel ist eine Konjunktion von Disjunktionstermen (siehe [4, S. 267]). Ein Beispiel einer solchen KNF ist:

$$(A \vee B \vee C) \wedge (\neg A \vee \neg B \vee C) \quad (4.17)$$

Eine solche Formel kann in Form von MILP-Ungleichungen dargestellt werden, indem zunächst für jedes Atom (z. B. A, B und C) eine 0/1-Variable eingeführt wird. Jede Konjunktion der KNF muss einzeln wahr werden, um insgesamt wahr zu werden. Daher wird für jede Konjunktion eine eigene Ungleichung erstellt, deren Summe ≥ 1 ist. In dieser Summe wird jedes positive Literal als eine 0/1 Variable x_i dargestellt, jedes negative Literal wird durch $1 - x_i$ ersetzt. Aus dem Beispiel entstehen dadurch die folgenden Ungleichungen:

$$\begin{aligned} a + b + c &> 1 \\ (1 - a) + (1 - b) + c &> 1 \end{aligned} \tag{4.18}$$

In unserem Beispiel lautete die logische Aussage:

$$\neg \text{hasgeneration}_{i-1} \wedge \text{hasgeneration}_i \rightarrow \text{hasstartup}_i \tag{4.19}$$

Diese Aussage muss nun zunächst in die KNF-Form überführt werden. Dazu wenden wir die beiden folgenden Regeln an:

$$\begin{aligned} A \rightarrow B &\Leftrightarrow \neg A \vee B \\ \neg(A \wedge B) &\Leftrightarrow \neg A \vee \neg B \end{aligned} \tag{4.20}$$

Für unsere Aussage ergibt sich dadurch die KNF-Form:

$$\text{hasgeneration}_{i-1} \vee \neg \text{hasgeneration}_i \vee \text{hasstartup}_i \tag{4.21}$$

Diese Aussage besteht nur aus einer einzigen Disjunktion. Entsprechend muss nur eine einzige Ungleichung formuliert werden:

$$\text{hasgeneration}_{i-1} + (1 - \text{hasgeneration}_i) + \text{hasstartup}_i > 1 \tag{4.22}$$

Wenn $\text{hasgeneration}_{i-1}$ den Wert 0 annimmt und hasgeneration_i aber mit „1“ belegt ist, muss hasstartup_i mit „1“ belegt werden, damit die Ungleichung erfüllt ist. In Pyomo kann diese Nebenbedingung wie folgt formuliert werden:

```

1 def fullfill_generation(model, p, h):
2     return model.has_generation[p, h-1] + (1 - model.
3         has_generation[p, h]) + model.startup[p, h] >= 1
4 model.fullfill_generation = Constraint(model.plants,
5     model.hours, rule=fullfill_generation)

```

In der Praxis existieren bei der Optimierung des Kraftwerkseinsatzes noch viele weitere Nebenbedingungen. Die Nebenbedingungen sind meist durch die technischen Restriktionen der Kraftwerke gegeben (siehe [13]). Zudem existieren aber auch Vertragskonstellationen, die z. B. beim Bezug von Brennstoffen eingehalten werden müssen (z. B. sog. Take-or-Pay-Verträge). Die Erstellung eines Fahrplans für ein Kraftwerk nennt man **Kraftwerkseinsatzplanung** (*Unit Commitment Problem*). Das Ergebnis eines solchen Optimierungsproblems ist der sog. Einsatzfahrplan (*Dispatch*) für die Kraftwerke. Dieser Fahrplan muss dann durch die Kraftwerke eingehalten werden.

Eine Kraftwerkseinsatzplanung auf Basis von MILP wird meist im Kurzfristbereich für die nächsten 2–3 Tage durchgeführt. Für längere Planungshorizonte werden die Optimierungsmodelle mitunter so komplex, dass aktuell selbst gute Solver Ergebnisse nicht in überschaubarer Zeit erzeugen können. Für eine Langfristplanung des Kraftwerkseinsatzes im Rahmen z. B. von Investitionsplanungen oder der Absicherung des Energiehandels werden meist einfachere Modelle genutzt, die auf einer Preisprognose des Strompreises, z. B. einer HPFC, basieren. Es wird dann oft vereinfachend angenommen, dass ein Kraftwerk am Netz ist, wenn seine Bruttomarge positiv ist, und ansonsten stillsteht. Die Bruttomarge eines Kraftwerks kann dann vereinfachend aus Brennstoffpreis, Wirkungsgrad, Strompreis und CO₂-Preis berechnet werden. Die theoretische Bruttomarge eines Gaskraftwerks bezeichnet man dann auch als *Clean Spark Spread*, die Marge eines Kohlekraftwerks wird als *Clean Dark Spread* bezeichnet.

Im realen Einsatz wird ein Solver als eine Kernkomponente genutzt, um das eigentliche Optimierungsproblem zu lösen. Der Solver ist dann in ein umgebendes System eingebettet, welches die Geschäftsprozesse und Anwendungsfälle im Unternehmen optimal unterstützt, z. B. die Bewirtschaftung eines Pumpspeicherkraftwerks. Solche Systeme werden auch als **Decision-Support-Systeme** bezeichnet. In der Realität besitzen derartige Systeme eine benutzerfreundliche Bedienoberfläche mit der Möglichkeit, Marktpreise schnell zu aktualisieren und die Kraftwerksstammdaten anzupassen. Die Systeme parametrisieren daraus dann die Optimierungsmodelle, übergeben sie an den Solver und visualisieren die Ergebnisse, z. B. in grafischer Form. Meist können auch unterschiedliche Szenarien gegeneinander gestellt und miteinander verglichen werden. Der Entwurf und die Realisierung derartiger Decision-Support-Systeme sind eine Aufgabe von EnergieinformatikerInnen.

Moderne Solver basieren auf jahrzehntelanger Forschung. Sie verwenden neben abgewandelten Formen von Simplex oder Interior Point noch Verfahren, wie Preprocessing, Heuristiken und andere spezielle Techniken. In der Praxis ist das Erstellen eines Optimierungsmodells daher meist wichtiger als die genaue Kenntnis, wie der Solver das Problem löst (vgl. [10, S. 59]). Dabei müssen bestimmte Situationen, wie Schwellwerte und logische Aussagen im Optimierungsmodell abgebildet werden. Die Anwendung solcher **Modellierungstechniken** setzt auch Erfahrung voraus.

Oft können mehrere unterschiedliche Varianten für ein Optimierungsmodell gefunden werden. Dann sollte diejenige Variante bevorzugt werden, welche durch einen Solver leichter zu lösen ist. Dies ist dann gegeben, wenn die Differenz zwischen linearer Relaxation und MILP-Problem (*Duality Gap*) möglichst klein ist.

4.3 Zusammenfassung

Die Vorhersage zukünftiger Verläufe volatiler Faktoren wie Marktpreise, Last oder Wetter spielt in der Energiewirtschaft eine wichtige Rolle. Ohne über eine stundengenaue Prognose der Last zu verfügen, kann z. B. ein Bilanzkreis nicht sinnvoll bewirtschaftet werden. Für die Erstellung einer Prognose können dabei recht unterschiedliche Model-

larten und Verfahren zum Tragen kommen. Auf der einen Seite werden fundamentale, ökonometrische Modelle eingesetzt, welche die elementaren Wirkzusammenhänge eines Systems abbilden. Auf der anderen Seite können auch Verfahren der Zeitreihenanalyse eine Rolle spielen, die eine Prognose auf Basis historischer Daten umsetzen.

In diesem Kapitel wurden daher einige elementare Grundbegriffe der Prognose im Kontext der Energiewirtschaft vorgestellt. Als Beispiel für ein einfaches ökonometrisches Prognosemodell wurde die Anwendung der sog. Standardlastprofile präsentiert. Danach wurde die Lastprognose auf Basis verschiedener Methoden der Zeitreihenanalyse fortgesetzt. Exemplarisch wurden in diesem Kapitel die Dekomposition einer Zeitreihe in Zeitreihenkomponenten sowie die Prognose mithilfe des exponentiellen Glättens vorgestellt. Ein Augenmerk wurde dabei darauf gelegt, wie die Verfahren mithilfe von Python umgesetzt werden können.

An vielen Stellen der Energiewirtschaft besteht die Anforderung, dass die Parameter eines Systemmodells so gewählt werden sollen, dass sich ein optimaler Zustand einstellt. Meist sucht man nach Konfigurationen, die geringe Kosten oder hohe Gewinnmargen sicherstellen. Ein Beispiel dafür ist die kostenoptimale Erzeugung von Energie aus einem Kraftwerkspark. In diesem Kapitel haben wir uns daher mit der Formulierung von solchen Optimierungsmodellen befasst. Dabei haben wir uns auf lineare, ganzzahlige Optimierungsprobleme beschränkt, die im Kontext der Energiewirtschaft häufig auftreten. Es wurde gezeigt, dass mithilfe von Solvern solche Probleme auf Basis des Simplex-Algorithmus mit überschaubarem Aufwand gelöst werden können. Zur Darstellung eines Optimierungsproblems setzen viele Solver eigene Modellierungssprachen ein. In diesem Kapitel wurde das Python-Modul Pyomo als Modellierungssprache genutzt, welches eine ganze Reihe von Solvern einbinden kann. Als rein lineares, praxisrelevantes Beispiel wurde die kostenoptimale Bewirtschaftung eines Pumpspeicherwerk dargestellt. Im Rahmen der Kraftwerkseinsatzplanung wurde zudem die Modellierung ganzzahliger Modellparameter vorgeführt, die sich z. B. im Rahmen von Schwellwerten niederschlagen.

4.4 Kontrollfragen

- Nach welchen Perspektiven kann ein System nach einer allgemeinen Systemtheorie beschrieben werden? Zu welcher dieser Perspektiven gehört das CIM?
- Welche Zusammenhänge beschreibt ein funktionales Modell?
- Welche Gründe kann es haben, Teile des Energiesystems in Form eines funktionalen Modells abzubilden?
- Welche Faktoren beeinflussen die Preisfindung am Strommarkt?
- Mithilfe welcher unterschiedlicher Methoden können funktionale Modelle erstellt werden?
- Wozu benötigt ein Energieversorger eine möglichst genaue Prognose der Last seiner Endkunden?
- Erläutern Sie die Prognose der Last mithilfe von Standardlastprofilen.

- Welche Modellannahme liegt den Standardlastprofilen zugrunde?
- Mithilfe welcher Fehlermaße lassen sich Prognosen im Nachhinein bewerten?
- In welche Komponenten lassen sich Zeitreihen häufig zerlegen?
- Mithilfe welcher Methode lässt sich eine solche Zerlegung bewerkstelligen? Wie funktioniert die Zerlegung genau?
- Welche Indikatoren dienen zur Bewertung, ob eine solche Zerlegung gelungen ist?
- Wie kann diese Zerlegung mithilfe der Programmiersprache Python durchgeführt werden?
- Wie viele Modellparameter besitzt das exponentielle Glätten dritter Ordnung? Wie werden diese bestimmt?
- Aus welchen Bestandteilen besteht ein Optimierungsmodell?
- Wann kann ein lineares Optimierungsmodell grafisch gelöst werden?
- Welche Algorithmen sind in der Lage, lineare Optimierungsprobleme zu lösen?
- Was ist ein Solver? Warum sollte man einen Solver einsetzen?
- Nennen Sie ein Beispiel für mathematische Modellierungssprachen, in denen Optimierungsprobleme abgebildet werden können. Was ist der Vorteil einer solchen Modellierungssprache?
- Welcher Datentyp wird im Pyomo-Framework dazu verwendet, um eine Entscheidungsvariable abzubilden?
- Was ist ein Pumpspeicherkraftwerk? Wie funktioniert es?
- Wofür steht die Abkürzung HPFC?
- Wann spricht man von sog. MILP-Problemen? Sind diese im Vergleich zu linearen Optimierungsproblemen leichter oder schwieriger zu lösen?
- Wie werden logische Ausdrücke in Nebenbedingungen ganzzahliger Optimierungsprobleme überführt?
- Was ist die konjunktive Normalform aussagenlogischer Formeln?
- Was ist das Ergebnis einer Kraftwerkseinsatzplanung?
- In welchem Zeithorizont wird eine Kraftwerkseinsatzplanung auf Basis von MILP-Modellen i. d. R. durchgeführt?
- Wie zeigen sich die technischen Restriktionen der Kraftwerksblöcke im MILP-Optimierungsmodell?
- Wie wird die Einsatzwahrscheinlichkeit von Kraftwerken im Langfristbereich bewertet?
- Wie wird die theoretische Bruttomarge von Gas- bzw. Kohlekraftwerken bezeichnet?

Literatur

1. George B. Dantzig. „Origins of the simplex method“. In: *A history of scientific computing*. New York, NY, USA: Association for Computing Machinery, 1. Juni 1990, S. 141–151. ISBN: 978-0-201-50814-7. URL: <https://doi.org/10.1145/87252.88081> (besucht am 26.02.2020).

2. Christian Fünfgeld und Remo Tiedemann. *Anwendung der Repräsentativen VDEW-Lastprofile step-by-step*. 31. Jan. 2000. URL: https://www.bdew.de/media/documents/2000131_Anwendung-repraesentativen_Lastprofile-Step-by-step.pdf (besucht am 25.06.2019).
3. P. Kaminsky und D. Rajan. *Introduction to AMPL*. TU Chemnitz, 13. Sep. 2000. URL: https://www.tu-chemnitz.de/mathematik/part_dgl/teaching/WS2009_Grundlagen_der_Optimierung/amplguide.pdf (besucht am 25.06.2019).
4. Hartmut Ehrig, Hrsg. *Mathematisch-strukturelle Grundlagen der Informatik*. 2. Aufl. Springer-Lehrbuch. Berlin: Springer, 2001. 621 S. ISBN: 978-3-540-41923-5.
5. Christian Wagenknecht. *Algorithmen und Komplexität*. Carl Hanser-Verlag, 2003.
6. J. W. Taylor. „Short-term electricity demand forecasting using double seasonal exponential smoothing“. In: *Journal of the Operational Research Society* 54.8 (2003), S. 799–805. issn: 0160-5682. <https://doi.org/10.1057/palgrave.jors.2601589>.
7. Dominik Möst. *Zur Wettbewerbsfähigkeit der Wasserkraft in liberalisierten Elektrizitätsmärkten*. 1. Aufl. Bd. 3220. Europäische Hochschulschriften Reihe 5, Volks- und Betriebswirtschaft. Frankfurt am Main u. a.: Lang, 2006. 252 S. ISBN: 978-3-631-55701-3.
8. Günter Ropohl. *Allgemeine Technologie: eine Systemtheorie der Technik*. 3., überarbeitete Auflage. Karlsruhe: Universitätsverlag Karlsruhe, 2009. 360 S. ISBN: 978-3-86644-374-7. URL: <https://core.ac.uk/download/pdf/14504340.pdf> (besucht am 04.10.2019).
9. Jared L. Gearhart u. a. „Comparison of Open-Source Linear Programming Solvers“. In: (Okt. 2013). URL: <https://prod.sandia.gov/techlib-noauth/access-control.cgi/2013/138847.pdf>.
10. Josef Kallrath. *Gemischt-ganzzahlige Optimierung*. 2., überarbeitete und erweiterte Auflage. Wiesbaden: Springer, 2013. 381 S. ISBN: 978-3-658-00689-1. URL: <https://doi.org/10.1007/978-3-658-00690-7>.
11. Leena Suhl und Taïb Mellouli. *Optimierungssysteme*. 3., korrigierte und aktualisierte Aufl. Springer-Lehrbuch. Berlin: Springer Gabler, 2013. 306 S. ISBN: 978-3-642-38936-8. URL: <https://doi.org/10.1007/978-3-642-38937-5>.
12. Franz Xaver Stelz. „Saison-Trend-Zerlegung von Zeitreihen“. Bachelorarbeit. München: Ludwig-Maximilians-Universität München, 9. Jan. 2014. URL: https://epub.ub.uni-muenchen.de/21450/1/BA_Stelz.pdf.
13. Kenneth van den Bergh, Kenneth Bruninx und Erik Delarue. *A Mixed-Integer Linear Formulation of the Unit Commitment Problem*. Apr. 2014. URL: https://www.mech.kuleuven.be/en/tme/research/energy_environment/Pdf/wpen2014-07-vbis.pdf.
14. Wolfgang Domschke u. a. *Einführung in Operations Research*. 9., überarbeitete und verbesserte Auflage. Lehrbuch. Berlin; Heidelberg: Springer Gabler, 2015. XVI, 276 Seiten. ISBN: 3-662-48215-0.
15. Noah Daniel Pflugradt. „Modellierung von Wasser und Energieverbräuchen in Haushalten“. Chemnitz: Technische Universität Chemnitz, 2016. URL: <https://nbn-resolving.org/urn:nbn:de:bsz:ch1-qucosa-209036>.
16. William E. Hart u. a. *Pyomo - Optimization Modeling in Python*. 2nd ed. 2017. New York, NY: Springer, 7. Juni 2017. 295 S. ISBN: 978-3-319-58819-3.
17. Rob J. Hyndman und George Athanasopoulos. *Forecasting Principles and Practice*. 2. Auflage. Otexts, 2018. 382 Seiten. ISBN: 978-0-9875071-1-2.
18. *Standardlastprofile Strom* URL: <https://www.bdew.de/energie/standardlastprofile-strom/>



Kommunikation und Automation

5

Informationen werden in der Energiewirtschaft selbstverständlich nicht nur lokal verarbeitet. Es existieren unüberschaubar viele Anwendungsfälle, in denen Informationen unternehmensintern oder über Unternehmensgrenzen hinweg über Kommunikationsnetze übertragen werden müssen. In diesem Kapitel werden wir uns einigen dieser Anwendungsfälle widmen.

Für die Organisation des physikalisch-technischen Energieflusses spielt z. B. die Anlagenautomation eine große Rolle. Zustandsdaten und Schaltbefehle müssen dabei über große Distanzen zwischen Schaltanlagen, Kraftwerken, dezentralen Erzeugungseinheiten und Leitsystemen ausgetauscht werden. Für die Abbildung der Anlagendaten in Informationsmodellen und ihrer sicheren Übertragung über Kommunikationsnetze kommen entsprechende Kommunikationsstandards zum Einsatz. Einige dieser Standards werden in Abschn. 5.1 vorgestellt.

Da ein guter Teil des Endenergieverbrauchs innerhalb von Gebäuden anfällt, rückt die Automation der dort installierten Anlagen ebenfalls zunehmend in den Mittelpunkt des Interesses. Durch einen hohen Automationsgrad erhofft man sich dort nicht nur eine Reduktion des Energieverbrauchs, sondern auch einen steigenden Komfort und zunehmende Sicherheit. Mit einigen Standards der Gebäudeautomation werden wir uns daher in Abschn. 5.2 befassen.

Um Energieflüsse effizient organisieren zu können, spielen Verbrauchsdaten eine Schlüsselrolle. Damit solche Verbrauchsdaten aber z. B. für die Steuerung von Erzeugungseinheiten genutzt werden können, müssen sie in großer Menge automatisiert, zeitnah und sicher erhoben und über Kommunikationsnetze an die relevanten Stellen übertragen werden können. Technisch soll dies mithilfe sog. intelligenter Messsysteme umgesetzt werden. Abschn. 5.3 stellt dazu einige Details der in Deutschland geltenden Regelungen dar und führt in die technischen Möglichkeiten des Smart Meter Gateway ein.



Abb. 5.1 Die Höchstspannungsschaltanlage der Firma Amprion GmbH in Hattingen

In vielen Geschäftsprozessen der Energiewirtschaft ist durch die zunehmende Spezialisierung der Unternehmen eine unternehmensübergreifende Abstimmung notwendig. Ein Beispiel ist der Wechsel eines Endkunden von einem Energieversorgungsunternehmen (EVU) zu einem anderen. In verschiedenen Prozessgruppen wurde genau definiert, welche Informationen zu übertragen sind und wie dies technisch umzusetzen ist. Mit diesen sog. Marktprozessen werden wir uns in Abschn. 5.4 befassen.

5.1 Netzautomation

Schaltanlagen stellen wichtige Infrastrukturelemente des elektrischen Energiesystems dar. In den Schaltanlagen erfolgt die eigentliche Stromverteilung und das Zusammenfassen von Lasten bzw. Verbrauchern (siehe [45, S. 609 f.]). Schaltanlagen finden sich auf allen Spannungsebenen. Allein auf der Höchstspannungsebene werden in Deutschland mehrere Hundert solcher Anlagen betrieben. Abb. 5.1 zeigt die Höchstspannungsschaltanlage der Firma Amprion GmbH in Hattingen.

In Schaltanlagen werden **Betriebsmittel** der sog. **Primär- und Sekundärtechnik** betrieben. Als Primärtechnik werden alle energietechnischen Betriebsmittel, wie z. B.

Leistungstransformatoren, Leistungsschalter und Sammelschienen bezeichnet, welche die eigentliche Funktion der Schaltanlage umsetzen. Als Sekundärtechnik bezeichnet man hingegen diverse Einrichtungen zur Fernsteuerung, Überwachung oder Messwerterfassung.

Um Daten aus einer Schaltanlage an die Leitebene weiterleiten zu können, existieren besondere Elemente in der Feldebene, die **Remote Terminal Unit (RTU)**. RTUs haben die Aufgabe, zwischen unterschiedlichen Kommunikationsprotokollen der Fernwirktechnik und von Feldbusssystemen zu vermitteln. Sie stellen die Schnittstelle zu einem entfernten Supervisory-Control-and-Data-Aquisition-(SCADA-)System dar (siehe Abschn. 2.6). Wird z. B. Modbus als Kommunikationsstandard für die Feldebene eingesetzt, dient ein RTU als Modbus Master. Er hat dann die Aufgaben, die Daten der Slaves zyklisch abzufragen, ggf. zu historisieren und an das SCADA-System zu übermitteln. Die funktionale Trennung zwischen Sensoren, Aktoren, speicherprogrammierbarer Steuerung (SPS) und RTU verwischt in Schaltanlagen dabei zunehmend. Mittlerweile werden immer mehr Funktionen in einem einzelnen Gerät integriert, z. B. Messen, Steuern und Regeln. Es entstehen sog. Intelligent Electrical Device (IED), die auf der Feld- und Steuerungsebene mehrere Aufgaben übernehmen. RTUs und IEDs sind der Sekundärtechnik zuzurechnen.

Innerhalb der Schaltanlagen kommunizieren IEDs über Bussysteme miteinander. Man unterscheidet hier in letzter Zeit zwischen einem sog. Prozessbus (*Process Bus*) und einem Stationsbus (*Station Bus*). Der Prozessbus verbindet Sensoren und Aktoren mit den IEDs, über den Stationsbus kommunizieren IEDs untereinander. Die Schaltanlagen sind zudem meist mithilfe sog. Fernwirktechnik mit einer Leitwarte verbunden. Auf all diesen Kommunikationsverbindungen können unterschiedliche Kommunikationslösungen eingesetzt werden. Abb. 5.2 zeigt die Kommunikationsarchitektur in und mit Schaltanlagen und verteilten (regenerativen) Energieanlagen.

Bis in die frühen 1990er-Jahre wurden für die digitale Kommunikation in und mit den Schaltanlagen meist proprietäre Systeme einzelner Hersteller genutzt. Um die Interoperabilität der Systeme zu erhöhen und die Abhängigkeit einzelner Hersteller zu reduzieren, entstanden danach mehrere Kommunikationsstandards. In diesem Kontext sind vor allem die drei Standards DNP3, IEC 60870-5 und IEC 61850 nennenswert. Im Folgenden wollen wir alle drei Standards kurz beleuchten.

Das Distributed Network Protocol (DNP3) ist ein nachrichtenbasiertes Kommunikationsprotokoll, welches meist für die SCADA-Anbindung bzw. auf dem Stationsbus eingesetzt wird. DNP3 ist ein offener Standard, der als IEEE 1815 verabschiedet wurde und durch die DNP User Group verwaltet wird (siehe [17]). Er ist ähnlich, aber komplexer als Modbus. DNP3 ist speziell für die digitale Kommunikation zwischen RTUs, IEDs und SCADA-Systemen gedacht. Er wird hauptsächlich in Nord- und Südamerika, Südafrika, Asien und Australien in der Energiewirtschaft, aber auch in der Gas-, Wasser- und Abwasserversorgung sowie der Ölindustrie eingesetzt (siehe [1]). Die beiden Standards DNP3 und IEC 60870-5 haben gemeinsame Wurzeln. Anfang der 1990er-Jahre wurden durch den IEC die 870-5 Transmission Protocols definiert, die später zum IEC 60870-5

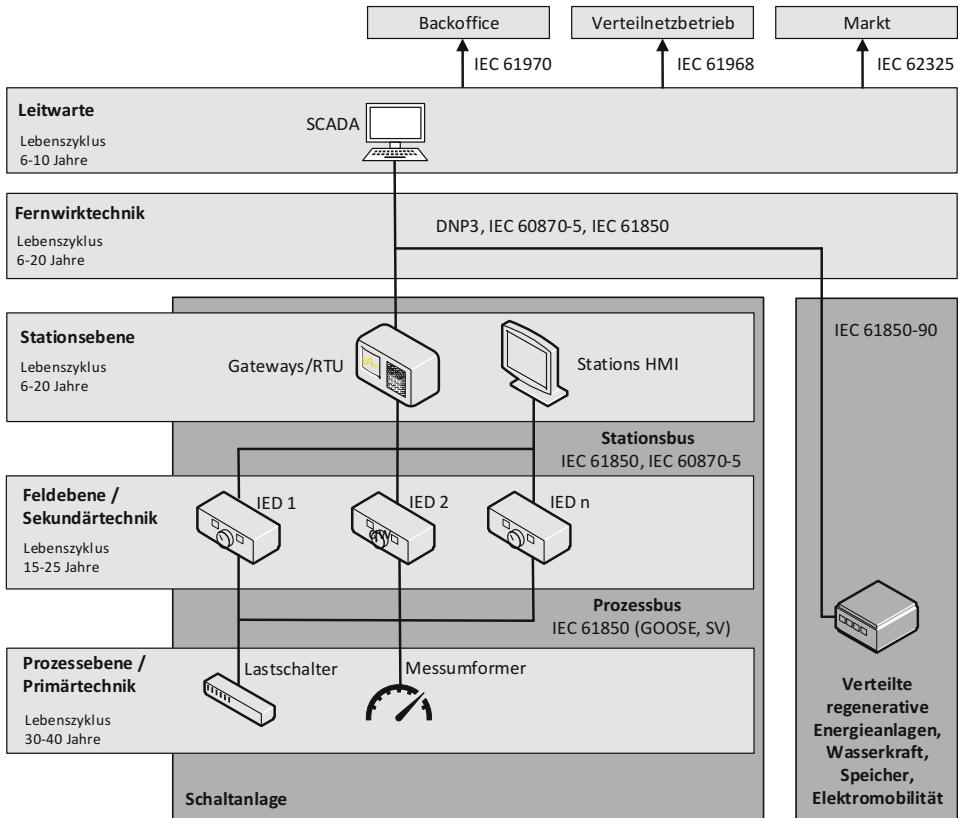


Abb. 5.2 Kommunikationsarchitektur in und mit Schaltanlagen

weiterentwickelt wurden. Teil der ursprünglichen Spezifikation waren die vier Nachrichtenformate FT1.1, FT1.2, FT2 und FT3. DNP3 nutzt das FT3-Nachrichtenformat.

Bei DNP3 werden ähnlich zu Modbus die Gerätetypen Masterstation und Outstation unterschieden. Eine Outstation (z. B. ein IED) besitzt Daten wie Messwerte oder Schaltzustände, die gelesen und/oder geschrieben werden können. DNP3 adressiert Stationen in einem flachen Adressraum mit numerischen Adressen zwischen 0 und 65519. Bestimmte Adressen sind für Broadcasts von Master- zu Outstations reserviert. DNP3 kann auf serieller Datenübertragung über RS-232 bzw. RS-485 oder auf einem TCP/IP-basierten Netzwerk aufsetzen.

Das Nachrichtenformat von DNP3 ist wesentlich komplexer als bei Modbus. Insbesondere besitzt DNP3 eine **Pseudotransportschicht**, die Teil der Anwendungsschicht ist (siehe [7]). Sie sorgt dafür, dass Daten der Anwendungsschicht auf mehrere Teile aufgeteilt und jeweils für sich übertragen werden. Anwendungsdaten werden auf sog. Application Protocol Data Units (APDUs) von maximal 2048 Bytes aufgeteilt. Diese können wiederum auf mehrere Transport Protocol Data Units (TPDUs) von max. 250 Bytes aufgeteilt

werden, bevor sie versendet werden. Im Fehlerfall, wie z. B. einem Time-out oder bei fehlerhaft übertragenen Daten, sorgt die Pseudotransportschicht für eine Wiederholung der Übertragung. Diese Schicht wird auch dann eingesetzt, wenn DNP3 nicht auf einer seriellen Verbindung, sondern auf TCP/IP aufsetzt.

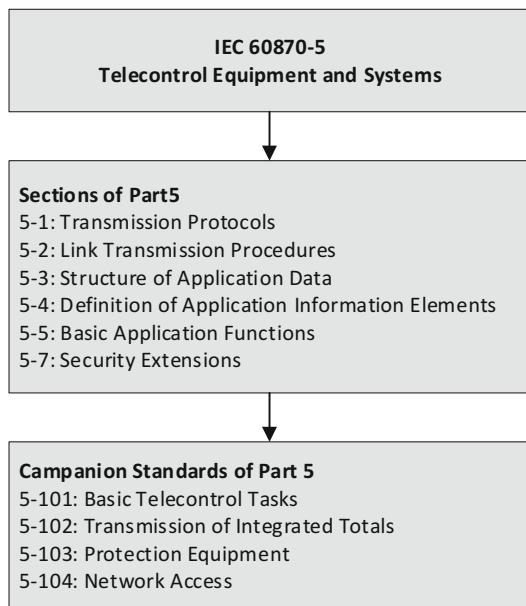
Das Informationsmodell einer DNP3-Outstation ist deutlich komplexer als das von Modbus, ergibt sich aber eher implizit aus dem Nachrichtenformat. Die Daten einer Outstation werden als *Points* bezeichnet. Ein Point kann man sich als Variable eines bestimmten Datentyps vorstellen. Ein Point kann z. B. den Messwert eines Sensors oder den aktuellen Zustand eines Lastschalters repräsentieren. Ein Point basiert dabei auf einem Datentyp, der durch eine *Object Group* und eine *Variation* bestimmt wird. Die Object Group definiert dabei die Art der Daten, z. B. analoge schreibbare Daten (Analog Output Object, Gruppennummer 40). Die Variation legt fest, wie viele Bits die Daten im Speicher einnehmen und wie diese Bits zu interpretieren sind, z. B. Variation 04 = *Long Floating Point Analog Status*. Ein Point wird durch seinen Object-Group-Index, den Variation-Index und den Point-Index eindeutig in einer Outstation identifiziert. Jeder Point kann einer von vier Prioritätsklassen (Class) zugewiesen werden. Werden Points der Prioritätsklasse 1-3 zugewiesen, werden alle Änderungen dieser Points mit Zeitstempel in einem *Event Buffer* vorgehalten. Diese Daten werden dann als *Event Data* bezeichnet. Für die Vergabe von Zeitstempeln stellt DNP3 Mechanismen bereit, um die Uhrzeit zwischen Master- und Outstation zu synchronisieren.

Der Datenaustausch zwischen Masterstation und Outstation kann auf verschiedene Arten erfolgen. Die Modi müssen dabei so gewählt werden, dass keine Nachrichtenkollisionen auftreten können, da DNP3 zur Behandlung solcher Kollisionen selbst keine Mechanismen vorhält. Beim *Polling* fragt der Master eine Outstation regelmäßig ab (*Unbalanced Communication*). Diese Variante muss gewählt werden, wenn ein Master mit vielen Outstations über einen gemeinsamen seriellen Bus verbunden ist. Dabei können entweder die aktuellen Werte bestimmter Points (*Static Poll*) oder alle Änderungen einer bestimmten Prioritätsklasse (*Class Poll*) abgefragt werden. Die Antwortnachricht kann Daten mehrerer Points, unterschiedlicher Object Groups beinhalten.

Eine Outstation kann auch ungefragt Nachrichten (*Unsolicited Messages*) an eine Masterstation senden. Dieses Verfahren wird als Report-by-Exception (RBE) bezeichnet. Dieser Modus ist nur bei Point-to-Point-Topologien oder TCP/IP-Netzen möglich, da es sonst zu Nachrichtenkollisionen kommen kann. Mithilfe von RBE meldet eine Outstation die (signifikante) Änderung von Daten. Dies kann das Nachrichtenvolumen gegenüber Polling deutlich reduzieren, da nur dann Nachrichten versendet werden, wenn relevante Zustandsänderungen auftreten. Masterstations können den Versand solcher Nachrichten aber auch deaktivieren. Polling und RBE können zu insgesamt vier verschiedenen Betriebsmodi kombiniert werden. DNP3 bietet dadurch eine große Flexibilität, um digitale Kommunikation im Rahmen der Automatisierung von Energieanlagen zu realisieren (siehe [1, Abschn. 6.5]).

Ein Kommunikationsprotokoll wie DNP3 zu implementieren, ist hoch komplex. Um technische Lösungen auf Basis von DNP3 umzusetzen, werden daher meist fertige

Abb. 5.3 Hierarchischer Aufbau der Standardfamilie IEC 60870-5



Programmbibliotheken eingesetzt. Hier existieren sowohl kommerzielle Lösungen (z. B. von Digital Microworks), aber auch quelloffene Alternativen. Mit **OpenDNP3** ist eine ausgereifte Open-Source-Lösung verfügbar. OpenDNP3 ist eine C++-basierte Bibliothek, es existieren aber zudem Anbindungen für C#, Java und Python. Die Bibliothek kann sowohl auf Mikrokontrollern zur Implementierung von IEDs oder auf vollwertigen PCs eingesetzt werden. OpenDNP3 steht unter der Apache-2.0-Lizenz und kann daher auch in kommerziellen Projekten Verwendung finden. Der Quellcode des Projekts findet sich bei Github (siehe [53]). OpenDNP3 verfügt zudem über eine aufgeschlossene und hilfsbereite Community, die sich über eine Google-Groups-Gruppe austauscht.

Mit dem **IEC 60870-5** existiert eine weitere Standardreihe, die sich mit der digitalen Kommunikation im Kontext industrieller Automation beschäftigt. Die Anfänge des Standards reichen bis ans Ende der 1980er-Jahre zurück und bilden auch die Basis für DNP3 (siehe [1, Abschn. 7]). Der Standard mit dem Titel *Telecontrol Equipment and Systems* zerfällt dabei hierarchisch in mehrere Teile. Teil 5 (IEC 60860-5) widmet sich den Kommunikationsprotokollen. Weitere Unterabschnitte (*Sections*) und Erweiterungen (*Companion Standards*) füllen bewusst offen gelassene Lücken um spezifische Details. Erst auf der Ebene der Companion Standards (Abschn. 5-101 bis 5-104) wird ein Detailgrad erreicht, dass von einem vollständigen Kommunikationsstandard gesprochen werden kann. Abb. 5.3 zeigt den hierarchischen Aufbau von IEC 60870-5.

In Europa findet die Standardreihe breite Anwendung in der Automatisierung von Energieanlagen. Der Standard stellt für diese Anwendungsfälle in den Abschnitten 5-102

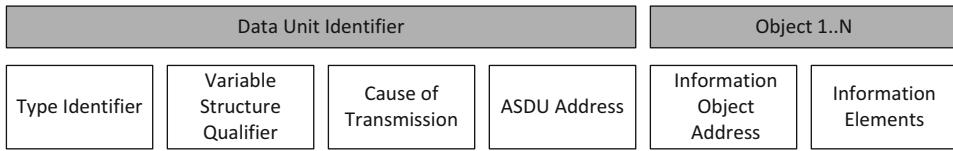


Abb. 5.4 Das ASDU-Format von IEC 60870-5

und 5-103 spezielle Datenobjekte bereit. IEC 60870-5 kommt dabei meist für die SCADA-Anbindung bzw. auf dem Stationsbus zur Anwendung.

Wie DNP3 ist auch IEC 60870-5 ein nachrichtenbasierter Kommunikationsstandard. Eine *Controlling Station* (wie eine Masterstation in DNP3) kommuniziert dabei mit mehreren *Controlled Stations* (wie eine Outstation in DNP3). Mit den Companion Standards 5-101 und 5-104 existieren dabei zwei sehr unterschiedliche Varianten.

IEC 60870-5-101 (als T101 bezeichnet) basiert auf einer seriellen Datenübertragung (*Non-networked Version*). Dabei werden Punkt-zu-Punkt- oder Multidrop-Topologien auf Basis von RS-232 oder RS-485 unterstützt. Bei Multidrop werden für die Stationsadresse max. 2 Byte vorgesehen, es können also max. 65.535 Stationen adressiert werden. T101 besitzt dabei wie DNP3 keine Mechanismen, um Nachrichtenkollisionen zu behandeln. Daher darf bei einer Multidrop-Topologie nur die Controlling Station die Kommunikation initiieren. Auf der Sicherungsschicht wird das FT1.2-Frameformat genutzt. IEC 60870-5-104 (als T104 bezeichnet) nutzt TCP/IP als Übertragungsweg (*Networked Version*). Alle Schichten unterhalb der Anwendungsschicht werden dann durch den TCP/IP-Protokollstapel ersetzt. Während DNP3 auch auf Basis von TCP/IP weiterhin den eigenen Pseudotransportlayer nutzt, verlässt sich T104 hier voll auf TCP/IP. Die Stationen werden dann auch über IP-Adressen adressiert.

T101 und T104 teilen sich dieselbe Anwendungsschicht. Diese wird hauptsächlich durch das Nachrichtenformat definiert, welches als **Application Service Data Unit (ASDU)** bezeichnet wird. Eine Nachricht beinhaltet dabei immer genau eine ASDU, welche vor dem Versand durch unterliegende Schichten in eine APDU verpackt wird. Eine ASDU besteht aus einem Nachrichtenkopf, dem *Data Unit Identifier*, und der eigentlichen Nutzlast der Nachricht. Abb. 5.4 zeigt den Aufbau einer ASDU.

Der Nachrichtenkopf einer ASDU beinhaltet mehrere wichtige Angaben:

- **Type Identification**

Die *Type Identification* legt die Struktur, den Datentyp und das Format der Nutzdaten in der ASDU fest. Es existieren verschiedene Gruppen, z. B. Status und Messwerte (1–40) oder Steuerkommandos (45–51 und 48–64). Da nur ein Type Identification pro ASDU gesetzt werden kann, müssen alle Nutzdaten der ASDU denselben Typ besitzen.

- **Variable Structure Qualifier**

Der *Variable Structure Qualifier* definiert, wie die Nutzdaten der ASDU strukturiert sind und wie viele Objekte enthalten sind.

- **Cause of Transmission (COT)**

Der *Cause of Transmission* legt die Art der Anfrage fest, ob es sich z. B. um eine Abfrage oder ein Kommando handelt. Dadurch wird klar, warum überhaupt Daten übermittelt werden sollen.

- **ASDU Address**

Die *ASDU Address* beinhaltet die Stationsadresse und ggf. noch eine logische Adresse innerhalb der Station, die *Section*. Diese Adresse ist für alle Daten, die in der ASDU übertragen werden, gleich.

Die Daten einer Station basieren im Wesentlichen auf **Information Objects**. Eine ASDU kann zwischen einem und 127 Information Objects beinhalten. Ein Information Object besteht selbst wieder aus mehreren Teilen. Die Information Object Address (IOA) stellt zusammen mit der Common Address der Station die eindeutige Adresse von Informationen innerhalb eines Systems dar. Ein Information Object kann ein oder mehrere Information Elements beinhalten, die aber alle vom selben Typ sein müssen. Ein Information Element kann man sich als Variable eines bestimmten Datentyps vorstellen. IEC 80670-5 definiert ca. 40 verschiedene Information Elements, z. B. *Single Point with Quality* oder *Short Floating Point Number*. Ein Information Element legt fest, wie sein Bitstring zu interpretieren ist. Je nach Typ kann ein Information Element zusätzlich einen *Quality Descriptor* umfassen. Dieser liefert zusätzliche Informationen zum Zustand der Daten, z. B. *Correct/Incorrect*, *Up to Date/Not Updated*. Ein Information Object kann zudem über einen Zeitstempel (*Time Tag*) verfügen.

IEC 60870-5 unterstützt auf der Anwendungsschicht unterschiedliche Kommunikationsarten zwischen Master- und Outstations. Diese werden im Wesentlichen durch den Cause of Transmission definiert. Einige Beispiele dafür sind:

- **Interrogation**

Die Interrogation bezeichnet das initiale Lesen aller oder einer bestimmten Gruppe von Daten durch eine Masterstation. Als Antwort erhält die Controlling Station mitunter mehrere Nachrichten.

- **Period, Cyclic oder Background Scans**

Durch Periodic, Cyclic oder Background Scans werden Daten einer Outstation regelmäßig an eine Masterstation gesendet, ohne dass eine Masterstation zuvor eine Anfrage gesendet hat. Die Daten werden auch dann gesendet, wenn sich keine Änderungen ergeben haben (*Static Data*).

- **Spontaneous**

Mithilfe von *Spontaneous*-Nachrichten sendet eine Outstation spontan, also ohne Zyklus oder vorherige Anfrage, Daten an einen Master. Dieser Modus ist vergleichbar mit Report-by-Exception in DNP3. Im Gegensatz zu DNP3 können diese Nachrichten aber nicht vom Master deaktiviert werden.

Kommunikationsarten, die von einer Outstation initiiert werden können (z. B. Period oder Spontaneous) setzen eine bestimmte Netzinfrastruktur voraus. Auf einem seriellen Bus darf ausschließlich die *Unbalanced Communication* (Polling) genutzt werden.

Auch für IEC 60870-5 existieren mehrere offene Programmbibliotheken, sodass Systeme umgesetzt werden können, die auf Basis dieses Standards miteinander kommunizieren. Die Firma MZ-Automation stellt beispielsweise zwei Bibliotheken bereit. Eine reine C-Lösung kann auf Mikrokontrollern eingesetzt werden, eine C#-Implementierung zielt auf leistungsstärkere Plattformen ab. Beide Bibliotheken unterstützen sowohl die T101 (Non-network Version) als auch T105 (Networked Version). Die Bibliotheken stehen unter der GNU General Public License 3.0, für den kommerziellen Einsatz wird aber eine kostenpflichtige Lizenz empfohlen. Der Quellcode beider Bibliotheken kann von Github bezogen werden.¹ Eine Alternative findet sich bei der OpenMUC Initiative. Diese wird durch das Fraunhofer Institut für Solare Energiesysteme ISE mit dem Ziel betrieben, technische Lösungen für das Smart Grid und die Integration erneuerbarer Energien bereitzustellen. Hier findet sich die java-basierte Bibliothek j60870, mit der T105-basierte Kommunikation umgesetzt werden kann (siehe [66]). Die Bibliothek steht ebenfalls unter der *GNU General License*.

Sowohl DNP3 als auch IEC 60870-5 sind Kommunikationsstandards, die sich in der Praxis bewährt haben. Obwohl beide Standards denselben Zweck verfolgen, unterscheiden sich DNP3 und 60870-5 in den Details mitunter deutlich. Dennoch ist keiner der beiden Standards dem anderen überlegen.

- Die Datenmodelle sind unterschiedlich.

Bei DNP3 werden Daten von Outstations in sog. Points organisiert, deren Datentyp durch eine Object Group und eine Variation bestimmt wird. Bei IEC 60870-5 wird von sog. Information Objects gesprochen, die typisierte Information Elements beinhalten.

- Der Nachrichtentransport basiert auf unterschiedlichen Konzepten.

Während DNP3 in allen Varianten den sog. Pseudotransportlayer nutzt, setzt die Anwendungsschicht von IEC 60870-5-104 auf den TCP/IP Stack für die Übertragung der Daten und verlässt sich dort komplett auf dessen Features.

- Die Anwendungsschicht stellt unterschiedliche Methoden bereit.

Bei DNP3 können Polling und Report-by-Exception zu mehren Betriebsmodi kombiniert werden. Bei IEC 60870-5 existieren mit den Cause-of-Transmissions viele unterschiedliche Kommunikationskonzepte, z. B. Interrogation, Period/Cyclic/Background Scans oder Spontaneous-Nachrichten.

Protokolle, wie Modbus, DNP3 und IEC 60870-5, sind im Sinne von IT-Standards bereits verhältnismäßig alt. Man kann sie als traditionelle Kommunikationsprotokolle bezeichnen. Sie stammen aus einer Zeit, in der die Bandbreite von Netzwerken (z. B. serielle

¹<https://github.com/mz-automation>.

Kommunikation) und die Performance von SPS nur sehr gering waren. Diese Situation hat sich mittlerweile aber grundlegend geändert. Die Bandbreite, die Verfügbarkeit von Kommunikationsstandards (Ethernet, TCP/IP) und die Hardwarefähigkeiten von SPS bzw. IEDs haben stark zugenommen. Bandbreite und Performance von SPS sind heute nicht mehr der eigentliche Kostentreiber. Dies führt dazu, dass die übertragenen Datenmengen größer werden können und mehr Intelligenz von der Automatisierungstechnik erwartet wird. Es rücken daher andere Aspekte zunehmend in den Vordergrund. Modbus, DNP3 und IEC 60870-5 ermöglichen den Entwicklern große Freiheiten in der Gestaltung der Daten innerhalb der IEDs. Die Bedeutung der Daten ist dabei nicht festgelegt. Die Datentypen in den Nachrichten legen meist lediglich die Codierung fest, z. B. 32 Bit Fließkommazahl. Ob dies aber als ein Spannungswert in Volt an einem Leistungsschalter interpretiert werden kann, wird oft erst aus der Dokumentation der Geräte klar. Dadurch können IEDs allerdings in bestehenden Konfigurationen kaum 1:1 ausgetauscht werden. Alle Datenempfänger, wie z. B. RTUs und SCADA-Systeme müssen neu konfiguriert werden, sobald ein neues IED eingesetzt wird. Dies ist aufwendig und daher teuer. Es entstand daher eine große Nachfrage nach neuen Standards mit geänderten Zielen:

- verteilte Datenhaltung erleichtern und Interoperabilität erhöhen,
- standardisiertes objektorientiertes Informationsmodell bereitstellen,
- Selbstbeschreibungsfähigkeit mithilfe von Metadaten ermöglichen.

IEC 61850 ist ein Standard, der aus den neuen Erwartungen an Kommunikationsprotokollen im Kontext der Schutz- und Leittechnik in elektrischen Schaltanlagen entwickelt wurde. Das Hauptziel des Standards ist die Kommunikationsinteroperabilität leittechnischer Geräte, die auf allen Ebenen der Automatisierungspyramide miteinander kommunizieren können. Zwei oder mehr intelligente Geräte sollen dabei Informationen austauschen und eindeutig interpretieren können. Der Standard wurde mehrfach erweitert, so stehen mittlerweile Erweiterungen bereit, um verteilte, regenerative Energieanlagen wie Wind- oder Wasserkraftanlagen zu steuern (siehe [68]).

IEC 61850 basiert auf mehreren Basiskonzepten. Im Kern wird zunächst ein objektorientiertes Informationsmodell eingeführt, mit dessen Hilfe alle Elemente einer Schaltanlage (i. d. R. IEDs) abgebildet werden können. Es werden dabei Standardklassen definiert, sodass bestimmte Arten von Geräten über immer gleiche Eigenschaften verfügen, wodurch die Austauschbarkeit erhöht wird. Es wird erwartet, dass dieser Teil des Standards zukünftig nur selten geändert werden muss (siehe [22]). Darauf aufbauend wurden mehrere Kommunikationsprotokolle definiert. Für unterschiedliche Kommunikationsszenarien kann dadurch ein herstellerunabhängiger Datenaustausch zwischen Geräten gewährleistet werden. Dabei ist das Informationsmodell von den Kommunikationsprotokollen abstrahiert, sodass zukünftig auch neue Protokolle eingeführt werden können. Dieser Teil des Standards wird sicher häufiger geändert, da sich Kommunikationstechnologien rasant entwickeln. Eine weitere Neuerung ist, dass auch der Entwicklungsprozess von Schaltanlagen unterstützt wird. Anlagen und ihre Bestandteile können mithilfe der sog.

Substation Configuration Language (SCL) in allen Phasen der Entwicklung dargestellt und in einem standardisierten Datenformat basierend auf Extensible Markup Language (XML) dargestellt werden (siehe [26]). Dadurch können Entwicklungswerzeuge unterschiedlicher Hersteller leichter integriert werden.

Bei DNP3 und IEC 60870-5 wird das Informationsmodell implizit durch die Nachrichtenformate festgelegt. Bei IEC 61850 hingegen existiert ein explizites objektorientiertes Informationsmodell. Ein reales IED stellt dabei einen Server dar, welcher über Schnittstellen von einem Client genutzt werden kann. Ein Server kann dabei aus einem oder mehreren logischen Geräten (*Logical Device*) bestehen. Diese basieren wiederum auf mehreren sog. Logical Nodes (LNs). Abb. 5.5 zeigt das Unified-Modelling-Language-(UML-)Klassendiagramm für einen Server als Ausschnitt aus dem Informationsmodell des IEC 68150 (siehe dazu auch [59], S. 28]).

Jedes *Logical Device* besitzt mindestens drei LNs. Das Logical-Node-Zero (LLNO) und das Logical-Node-Physical-Device (LPHD) bündeln gewisse Konfigurationseinstellungen und werden als Infrastruktur-Nodes bezeichnet. Ein Logical Device besitzt zudem mindestens ein weiteres domänenspezifisches LN, welches die eigentliche Funktionalität des Geräts implementiert.

Die LNs sind die zentralen Bausteine des IEC 61850, aus denen eine Schaltanlage logisch aufgebaut wird. Ein LN realisiert dabei eine klar abgrenzbare Funktion innerhalb einer Schaltanlage, z. B. einen Leistungsschalter. Ein LN stellt dafür Daten in Form von *Data Objects* bereit, die aus einem oder mehreren Attributen bestehen. Die Werte der Attribute repräsentieren letztendlich die relevanten Informationen, wie Schaltzustände oder Messwerte.

Im Gegensatz zu DNP3 und IEC 60870-5 sind Entwickler beim IEC 61850 aber nicht frei in ihrer Entscheidung, wie ein spezifisches IED aufzubauen ist. Um die Interoperabilität zu erhöhen, definiert IEC 61850-7-4 insgesamt 13 Gruppen domänenspezifischer LNs, die *Node Groups*. In den Gruppen sind insgesamt 91 unterschiedliche Arten von LNs vordefiniert. Bei diesen LNs wird z. B. festgelegt, welche Attribute mindestens vorhanden sein müssen. Ein Leistungsschalter muss z. B. mindestens das Attribut „Pos“, also die Leistungsschalterposition auf Basis des Datentyps Controllable Double Point (CDP), be-

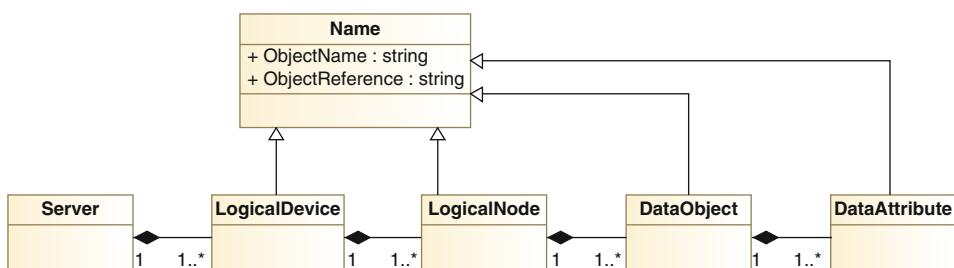
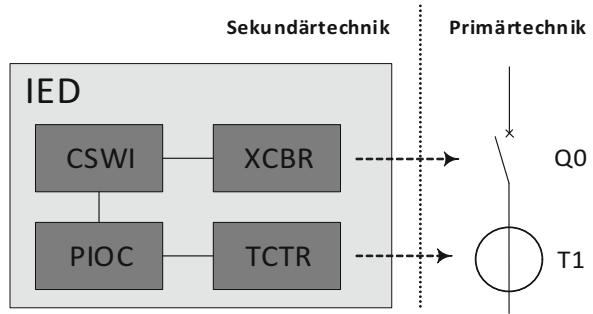


Abb. 5.5 Elementares Klassenmodell des Abstract Communication Service Interface des IEC 61850

Abb. 5.6 Interner, logischer Aufbau eines Leistungsschalters nach IEC 61850 und der Zusammenhang mit der Sekundärtechnik. In Anlehnung an [2]



sitzen. Abb. 5.6 zeigt den logischen Aufbau eines beispielhaften Leistungsschalters. Dieser ist intern aus mehreren LNs aufgebaut, z. B. einem Stromwandler (TCTR), einem Melder für Überströme (PIOC), einer Schaltersteuerung (CSWI) und einem Lasttrennschalter (XCBR).

Die LNs einer Gruppe werden nach einem bestimmten Schema benannt. Der erste Buchstabe identifiziert dabei die Funktionsgruppe, z. B. „X“ für Schaltgerät oder „M“ für Messwesen. Neben den LNs besitzen auch alle anderen Modellelemente, also Logical Devices, Data Objects und Attributes, einen eindeutigen Namen. Über die Hierarchie der Elemente kann ein Element dann global eindeutig adressiert werden. Die Position des Leistungsschalters Nr.1 in der Schaltanlage Atlanta5 könnte z. B. durch den Pfad Atlanta_HV5/XCBR1\$Pos\$stVal adressiert werden. In IEC 61850 existiert somit kein flacher, sondern ein hierarchischer Adressraum (siehe auch [39]).

Im Gegensatz zum Common Information Model (CIM) wurde bei der Definition der Node Groups nicht durchgängig UML eingesetzt, um das Modell zu definieren. IEC 61850-7-4 führt die unterschiedlichen Gruppen und die Attribute in vielen einzelnen Tabellen auf. Dadurch kommt es mitunter zu logischen Brüchen und auch Widersprüchen (siehe [2]).

Interoperabilität setzt nicht nur ein standardisiertes Informationsmodell voraus, sondern auch eine klare Definition, wie auf dieses zugegriffen werden kann (siehe [8, S. 9]). Auf dem zuvor dargestellten Informationsmodell werden daher im Teil IEC 61850-7-2 abstrakte Kommunikationsdienste eingeführt, die als **Abstract Communication Service Interfaces (ACSI)** bezeichnet werden. Die ACSI definiert eine Vielzahl von Funktionen, z. B. für die client/server-basierte Kommunikation:

- das Lesen und Schreiben der Werte einzelner oder Gruppen von Attributen,
- die Steuerung von Geräten (z. B. *Select Before Operate* oder *Direct Operate*),
- der eventbasierte Versand von Werten nach signifikanten Änderungen (*Reporting*),
- das lokale Zwischenspeichern von Werten mit Zeitstempeln (*Logging*),
- die Abfrage des Datenmodells für die Selbstauskunft über Geräte sowie
- die Übertragung von Dateien zu Konfigurationszwecken.

Auch für die zeitkritische peer-to-peer-basierte Übertragung von Daten, die hauptsächlich auf dem Prozessbus stattfindet, werden Funktionen definiert. Die ACSI legen dabei zunächst nicht fest, wie die Daten technisch übertragen werden sollen. Es wird lediglich definiert, welche Funktionen auf den Daten nutzbar sind. Im praktischen Einsatz wird daher die Abbildung auf konkrete Kommunikationsprotokolle benötigt. Diese Entkopplung sorgt dafür, dass neue Kommunikationsprotokolle leicht angebunden werden können. Zurzeit sind mehrere solcher Abbildungen definiert. Abb. 5.7 zeigt die Abbildung der ACSI auf die aktuell im IEC 61850 nutzbaren Kommunikationsprotokolle.

Für die client/server-basierte Kommunikation auf dem Stationsbus oder mit externen Einheiten, wie z. B. einem SCADA oder verteilten Energieanlagen, wird die Manufacturing Messaging Specification (MMS) verwendet (siehe [31]). MMS ist ein als ISO 9506 standardisiertes Anwendungsprotokoll, welches unabhängig von der unterliegenden Übertragungstechnik genutzt werden kann. Im Kontext von IEC 61850 kommt aber meist eine TCP/IP-basierte Datenübertragung zum Einsatz. MMS führt ein eigenes Informationsmodell ein. Ein MMS-Server wird als Virtual Manufacturing Device (VMD) bezeichnet, welches Objekte besitzt, die wiederum Daten in Form von Attributen bereitstellen. Auf diesen Elementen werden über 80 verschiedene Dienste angeboten, um Daten zu lesen, zu schreiben, Abonnements auf Zustandsänderungen einzurichten, Metadaten von Geräten auszulesen usw.

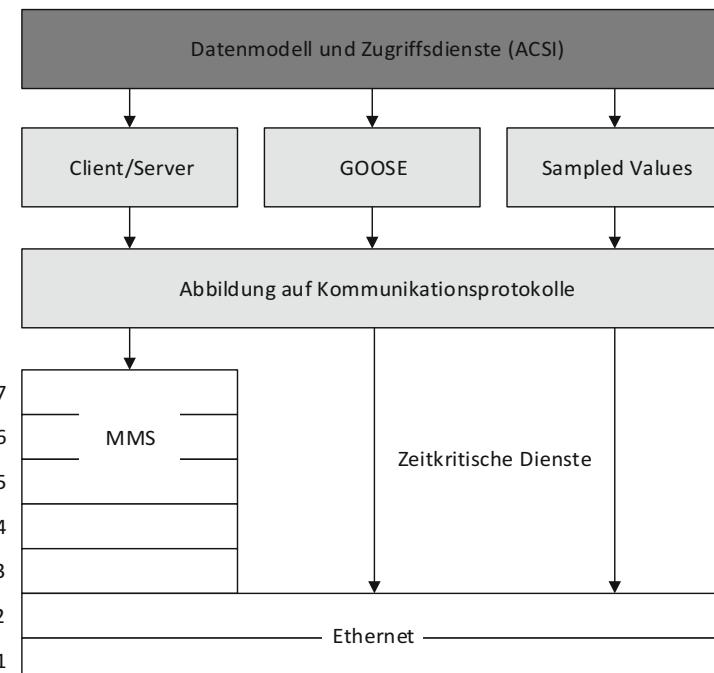


Abb. 5.7 Abbildung des Datenmodells und der Zugriffsdiene auf Kommunikationsprotokolle in IEC 61850

Für den zeitkritischen Datenaustausch auf dem Prozessbus existieren zwei Abbildungen des ACSI auf konkrete Protokollrealisierungen, die auf dem Publish/Subscribe-Verfahren basieren. Mithilfe der sog. Generic Object Oriented Substation Events (GOOSE) können spontane Zustandsänderungen behandelt werden, die Sampled Values (SV) dienen dazu, Messwerte in einem genau definierten Zeitintervall auf dem Prozessbus zu übertragen. Beide Protokolle werden technisch durch spezielle Ethernettelegramme umgesetzt.

Auch für IEC 61850 stehen Programmzbibliotheken bereit, um kompatible Geräte zu entwickeln bzw. die Kommunikation mit solchen Geräten abilden zu können. Mit **OpenIEC61850** existiert z. B. eine rein java-basierte Lösung. Sie stammt ursprünglich aus dem Kontext der OpenMUC Initiative, wird aber mittlerweile durch die Firma Beanit GmbH aus Berlin weiterentwickelt. Die Bibliothek steht unter der Apache-2.0-Lizenz. Der Quelltext findet sich bei Github (siehe [67]). Auch die MZ-Automation GmbH stellt für IEC 61850 eine Programmzbibliothek bereit. Die Bibliothek namens „libIEC61850“ basiert auf der Programmiersprache C und kann auch auf Mikrokontrollern mit geringem Ressourcenverbrauch eingesetzt werden. Die Bibliothek steht unter der GNU General Public License, für eine kommerzielle Nutzung wird aber erneut eine entsprechende Lizenz empfohlen. Der Quelltext findet sich ebenfalls bei Github (siehe [66]).

Um IT-Anwendungen in der Energiewirtschaft zu integrieren, kann CIM als gemeinsamer Standard eingesetzt werden. Mithilfe von CIM können u. a. Netzmodelle, Asset-Daten oder kaufmännische Informationen dargestellt und über Unternehmensgrenzen hinweg ausgetauscht werden. CIM wird dabei eher auf einer technisch höheren Abstraktionsebene des Energiesystems eingesetzt, z. B. für die Marktkommunikation und auf Unternehmensebene. Auf einer technisch niedrigeren Abstraktionsebene hat sich mit IEC 61850 ein entsprechender Standard etabliert. Auf Basis des sog. Smart Grid Architecture Model (SGAM) (siehe Abschn. 2.7) werden CIM und IEC 61850 als Kernstandards für die Umsetzung des Smart Grid angesehen (siehe [14, S. 193 ff.]).

5.2 Gebäudeautomation

Von Gebäudeautomation (GA) spricht man, wenn in Gebäuden bspw. Komponenten aus den Gewerken Heizung, Klima, Lüftung, Beleuchtung, Beschattung, Feuerschutz und Sicherheitstechnik überwacht, gesteuert und automatisiert geregelt werden. Es existieren verschiedene gute Gründe, um GA zu betreiben. Durch eine intelligente Heizung, Belüftung und Klimatisierung können die Wirtschaftlichkeit, die Energieeffizienz und die Umweltverträglichkeit von Gebäuden erhöht werden. In Deutschland finden sich aktuell ca. 22 Millionen Gebäude, darunter ca. 19 Millionen Wohngebäude. In den Gebäuden werden aktuell ca. 35 % des gesamten Endenergieverbrauchs in Deutschland aufgewendet (siehe [54]). Der größte Teil entfällt dabei auf die Wärme- und Warmwasserversorgung. Abb. 5.8 zeigt die Entwicklung des Endenergieverbrauchs innerhalb von Gebäuden in Deutschland zwischen den Jahren 2008 und 2018. Das Energiekonzept der Bundesre-

Gebäuderelevanter Endenergieverbrauch für Raumwärme, Raumkühlung, Warmwasser und Beleuchtung*

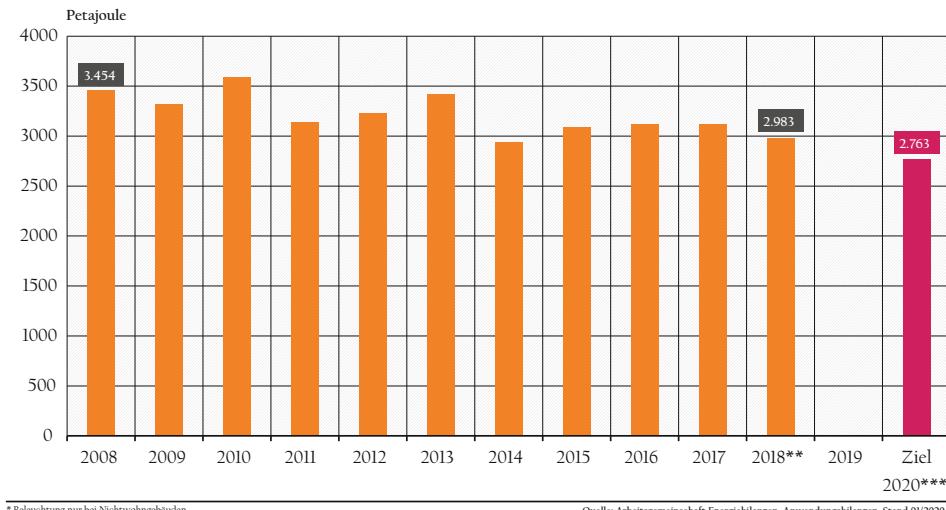


Abb. 5.8 Gebäuderelevanter Endenergieverbrauch für Raumwärme, Raumkühlung, Warmwasser und Beleuchtung. Bildquelle: Umweltbundesamt. <https://www.umweltbundesamt.de/indikator-energieverbrauch-fuer-gebaeude>

gierung zielt auf eine Energieeinsparung von 20 % gegenüber dem Jahr 2008 ab (siehe Abschn. 2.2).

In mehreren Studien wurde gezeigt, dass ein hoher Automatisierungsgrad den Energie- und Wasserverbrauch in Gebäuden deutlich senken kann (siehe [10] und [12]). Auch die Gebäude Sicherheit kann durch eine zentrale Zutrittssteuerung und eine Integration von Brand- und Einbruchsmeldung steigen. Durch die Integration mehrerer Gewerke, durch eine höhere Funktionalität und bessere Bedienbarkeit wird zudem der Komfort des Gesamtsystems gesteigert. Dieser Aspekt spielt insbesondere im Umfeld von privaten Wohngebäuden eine herausragende Rolle.

In kommerziellen Gebäuden, wie Hotels, Verwaltungsgebäuden oder Krankenhäusern, ist der Einsatz von Automatisierungstechnik heute bereits Stand der Technik (siehe [34], S. 247). Im Wohnungsbau sind entsprechende Systeme ebenfalls auf dem Vormarsch. Durch den Einsatz von intelligenten Systemen der Gebäudeautomation können Gebäude zu **Smarthomes** (private Nutzung) bzw. Smart Buildings (gewerbliche Nutzung) werden (siehe [28]). Dafür müssen Haus-, Energie- und Kommunikationstechnologie interdisziplinär zusammenwirken, was auch als Domotronik bezeichnet wird.

Es existieren mehrere Normen, die sich im Allgemeinen mit der Gebäudeautomation beschäftigen. In der ISO/IEC 16484 finden sich Aussagen zu den Grundbegriffen der Gebäudeautomation, der eingesetzten Hardware, den Funktionen, ihren Anwendungen und dem BACnet-Protokoll (siehe z. B. [15]). Durch die Standardreihe VDI 3814 wird

dieser Standard in Deutschland um nationale Belange ergänzt (aktuell in Überarbeitung). Die Informations- und Kommunikationsstandards der Gebäudeautomation werden in einigen weiteren Standards behandelt, auf die wir im Folgenden eingehen werden.

In großen Systemen der Gebäudeautomation finden sich ähnliche Strukturen wie in der industriellen Prozessautomatisierung. Auch die Systeme der Gebäudeautomation sind meist hierarchisch in der Feld-, Automations- und Managementebene organisiert (siehe Abschn. 2.6). Basiert ein System ausschließlich auf Komponenten der Feld- und Automationsebene, spricht man auch von **Raumautomation** oder **Gebäudesystemtechnik**. Von einer echten Gebäudeautomation spricht man meist erst dann, wenn höherwertige Funktionen auf der Managementebene hinzukommen, z. B. das Energiemanagement von Gebäuden, das sog. Predictive Maintenance oder die Enterprise Integration von z. B. ERP-Systemen. In der Raum- bzw. der Gebäudeautomation spielt dabei die Echtzeitfähigkeit der Systeme eine eher untergeordnete Rolle. Es wird aber eine höhere Flexibilität und Rekonfigurierbarkeit der Systeme erwartet.

Auch in der Raum- und Gebäudeautomation werden für die Kommunikation zwischen Feld-, Automations- und Managementebene Kommunikationssysteme benötigt. Auf niedrigeren OSI-Ebenen organisieren sie den Transport von Nachrichten in einem Kommunikationsnetz. Auf den höheren Schichten werden Informationsmodelle, Dienste und Schnittstellen definiert, um damit Anwendungen der Gebäudeautomation zu realisieren. In diesem Bereich sind mittlerweile unüberschaubar viele Systeme, mehr oder weniger offene Plattformen, Technologien und Standards verfügbar. Im Umfeld von privaten Gebäuden werden Systeme, wie z. B. Loxone, HomeMatic, Xiaomi Home, Philipps Hue, Z-Wave, EnOcean, DECT ULE uvm. angeboten. In **Zweckbauten** werden überwiegend KNX, LonWorks und BACnet eingesetzt. Auch die großen US-Konzerne Google, Amazon und Apple haben den Markt für sich entdeckt und bieten entsprechende Lösungen an. Die Systeme unterscheiden sich dabei stark in ihren Zielen, Ansprüchen und Möglichkeiten. Im Folgenden werden wir uns ausschließlich mit denjenigen Standards auseinandersetzen, die oft in Zweckbauten eingesetzt werden.

Die Kommunikationsstandards werden auf den Automationsebenen nicht gleichberechtigt eingesetzt. KNX und LonWorks decken meist die Feld- und Automationsebene ab (Raumautomation). BACnet und OPC/UA verfügen darüber hinausgehend über Möglichkeiten, auch auf der Managementebene eingesetzt zu werden. Abb. 5.9 zeigt den Einsatz der Kommunikationsstandards auf den Hierarchieebenen der Gebäudeautomation.

KNX ist ein Kommunikationsstandard der Raumautomation. Der ursprüngliche Name des Standards war Europäischer Installationsbus (EIB), der neue Name wurde durch Zusammenschluss verschiedener Organisationen im Jahr 1999 geschaffen. Die Grundidee von KNX ist Dezentralität, die Intelligenz des Systems ist auf viele Teilnehmer verteilt. Durch Konfiguration einer virtuellen Leitung sorgt die Zustandsänderung eines Lichttasters direkt dafür, dass eine Lampe ein- oder ausgeschaltet wird. Eine zentrale Steuerung ist dafür nicht notwendig. Dies hat den Vorteil, dass beim Ausfall einzelner Komponenten das Restsystem i. d. R. weiter arbeiten kann. Abb. 5.10 zeigt einen Versuchsaufbau

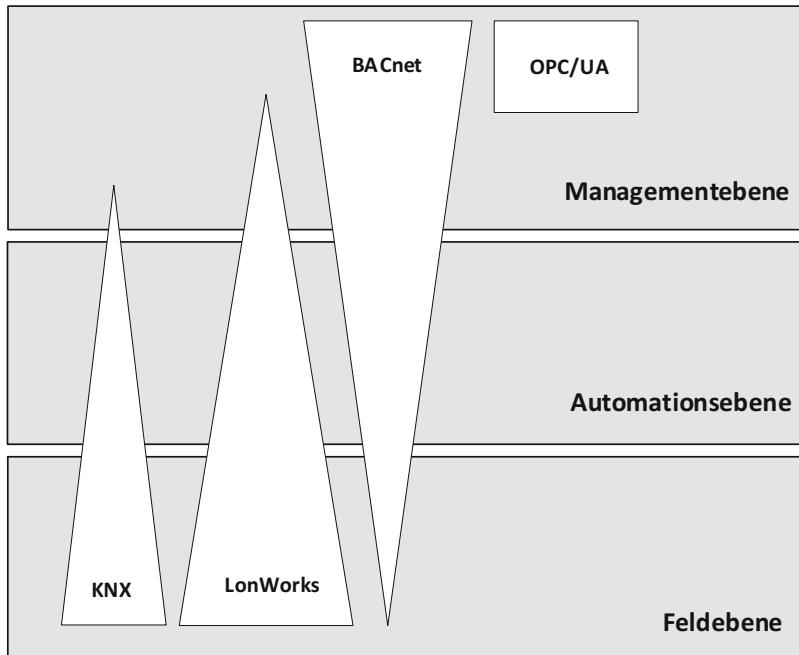


Abb. 5.9 Kommunikationsstandards auf den Ebenen der Gebäudeautomation. In Anlehnung an [23]

mit Komponenten von KNX und BACnet im Gebäudeautomationslabor der Hochschule Hamm-Lippstadt.

KNX ist gut für Schaltbefehle und kleine Mengen von Sensordaten geeignet, bietet aber von Haus aus keine Möglichkeiten an, höherwertige Funktionen auf der Managementebene zu realisieren oder zu integrieren. Wegen der eher geringen Übertragungsrate und kleinen Framegrößen eignet sich KNX nicht für größere Datenmengen. Abb. 5.11 zeigt die Zuordnung der KNX-Funktionalität zu den sieben Schichten des ISO/OSI-Modells.

Auf der Bitübertragungsebene bietet KNX unterschiedliche Alternativen an. KNX TP ist wohl das am häufigsten eingesetzte Übertragungsmedium. Dabei wird eine verdrillte Zweidrahtleitung als gemeinsamer Bus genutzt (z. B. YCYM 2x2x0,8), der den Teilnehmern sowohl als Datenleitung als auch als Spannungsversorgung dient. Die Spannungsversorgungen speisen 30 V in das System ein, Busteilnehmer arbeiten bei Spannungen zwischen 21 V und 30 V fehlerfrei.

Bits werden ähnlich zu RS-485 anhand einer Spannungsdifferenz codiert (siehe Abschn. 2.5). Eine „1“ wird durch den Ruhezustand auf dem Bus repräsentiert, für eine „0“ wird die Spannung um 5 V abgesenkt. Als Zugriffsverfahren wird CSMA/CA genutzt, dabei ist die „0“ das dominierende Bit. Durch Verwendung der „0“ im Kopf einer Nachricht kann ihre Priorität aktiv beeinflusst werden. Daten werden bei KNX TP mit



Abb. 5.10 Versuchsaufbau mit KNX und BACnet im Gebäudeautomationslabor an der Hochschule Hamm-Lippstadt

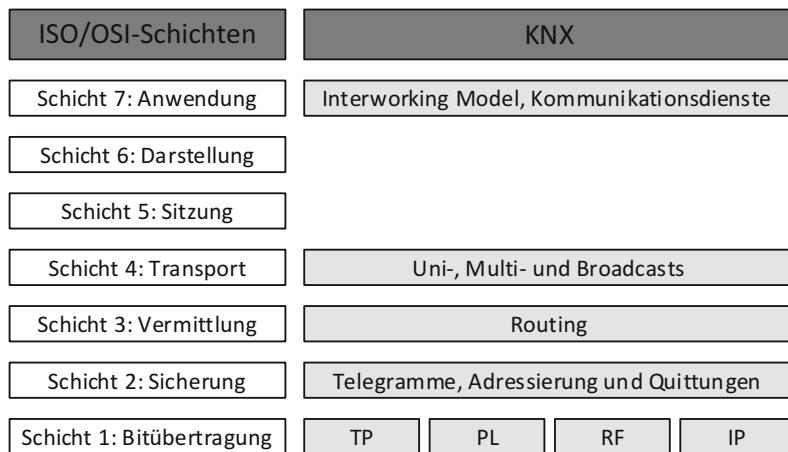


Abb. 5.11 KNX und die sieben Schichten des ISO/OSI-Modells

ca. 9600 Bits/s asynchron übertragen. Neben KNX TP existieren noch weitere alternative Übertragungsmedien:

- Bei **KNX PL** werden die Signale über Powerline auf die Wechselspannung aufmoduliert.
- KNX RF** überträgt die Daten per Funk auf der freien Trägerfrequenz 868,3 MHz.
- Bei **KNX IP** werden die KNX-Telegramme in IP-Pakete verpackt, sodass herkömmliche LAN-Netzwerke genutzt werden können. Diese Möglichkeit wird meist dann eingesetzt, wenn mehrere KNX-Linien mithilfe von Kopplern über ein gemeinsames IP-basiertes Backbone mit höherer Bandbreite und Übertragungsgeschwindigkeit miteinander verbunden werden sollen. Auch können so mehrere Gebäudestandorte mithilfe von VPN über das Internet miteinander integriert werden.

Auf der Sicherungsschicht werden **Telegramme** eingeführt, die aus einzelnen Bytes bestehen. Bei KNX TP wird jedes Byte eines Telegramms zeichenweise durch einen Universal Asynchronous Receiver Transmitter (UART) versendet. Jedes Zeichen wird durch ein Startbit eingeleitet, die Nutzdaten bestehen aus acht Datenbits. Anschließend werden ein Paritätsbit und ein Endbit gesendet. Zwischen den so einzeln übertragenen Zeichen und nach jedem Telegramm müssen bestimmte Übertragungspausen eingehalten werden. KNX definiert drei unterschiedliche Arten von Telegrammen. Datentelegramme, Abfragetelegramme und Bestätigungstelegramme. Der grundsätzliche Aufbau eines Datentelegramms wird in Abb. 5.12 gezeigt.

Die ersten beiden Bits des Kontrollfelds definieren die Art und wegen des Buszugriffsverfahrens auch die Priorität des Telegramms. Die höchste Priorität haben Systemtelegramme, danach folgen Alarmfunktionen, normale Telegramme mit hoher und normale Telegramme mit niedriger Priorität.

Als Reaktion auf ein Datentelegramm werden auf der Sicherungsschicht (außer bei RF) Bestätigungstelegramme verschickt. Eine Bestätigung kann positiv (ACK) oder negativ (NAK, BUSY) ausfallen. Alle Empfänger einer Nachricht senden das Bestätigungstelegramm gleichzeitig. Wegen des Buszugriffsverfahrens setzen sich negative Bestätigungstelegramme gegenüber positiven durch. Im Falle keiner oder einer negativen Quittung wird die Nachricht bis zu dreimal wiederholt.

Die Sicherungsschicht führt auch zwei jeweils 16 Bit breite Adressarten ein, die Geräteadresse (physische Adresse) und die Gruppenadresse.

Die **Geräteadresse** ist eine systemweit eindeutige, logische Adresse, die für Unicast-Nachrichten genutzt werden kann. Die Quelladresse eines Datentelegramms ist immer eine Geräteadresse. Für die Geräteadresse stehen 16 Bits zur Verfügung, diese sind in drei

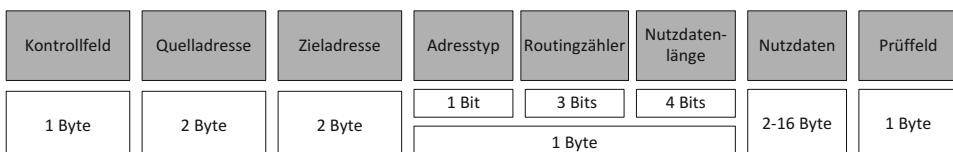


Abb. 5.12 Aufbau eines Datentelegramms in KNX

Hierarchiestufen aufgeteilt (z. B. Bereich=4 Bit, Linie=4 Bit, Teilnehmer je Linie=8 Bit). Die Geräteadresse reflektiert meist den physischen Einbauort im Gebäude, z. B. „1.2.2“ für Bereich 1, Linie 2, Gerät 2. Es wird empfohlen, die Geräteadresse sichtbar am Gerät zu hinterlegen.

Gruppenadressen dienen der Multicast-Adressierung. Die 16 Bits der Gruppenadresse können während der Projektierung frei in mehrere Hierarchiestufen aufgeteilt werden, z. B. in die Haupt-, Mittel- und Untergruppe. Jeder Gruppe wird dann eine bestimmte Bedeutung zugeordnet, z. B. Gewerke, Funktionsbereiche oder Räume. Die Gruppenadresse mit dem Wert null wird für Broadcasts verwendet.

Bei KNX TP können bis zu 256 Geräte an einen Bus (genannt **Linie**) angeschlossen werden. Eine solche Linie bildet dann ein einzelnes **Netzsegment**. Mithilfe von Kopplern können mehrere Segmente zu komplexen, galvanisch getrennten Baumstrukturen verbunden werden. Ringtopologien sind dabei allerdings nicht gestattet. Damit Nachrichten zwischen Segmenten ausgetauscht werden können, führt KNX auf der Vermittlungsschicht Mechanismen für das Routing ein. Koppler werten dazu den Routingzähler eines Datentelegramms aus, um zu entscheiden, ob eine Nachricht an das nächste Segment weitergereicht werden soll. Sogenannte KNX/IP-Router können dazu genutzt werden, um mehrere Segmente über ein gemeinsames IP-basiertes Backbone miteinander zu verbinden.

Auf der Transportschicht führt KNX mehrere Kommunikationsarten ein, um mithilfe der beiden Adressarten Nachrichten an einen oder mehrere Empfänger zuzustellen:

- Bei **verbindungslosen Multicasts** werden Nachrichten auf Basis der Gruppenadressen an mehrere Empfänger gesendet. Eine logische Verbindung wird dazu nicht aufgebaut.
- Bei **verbindungslosen Broadcasts** werden Nachrichten an alle Teilnehmer in einem Netzsegment zugestellt. Auch bei dieser Nachrichtenart muss keine logische Verbindung aufgebaut werden. Diese Kommunikationsart wird z. B. im Rahmen der Erstkonfiguration für die Vergabe der Geräteadressen genutzt.
- KNX stellt sowohl verbindungslose als auch verbindungsorientierte **Unicasts** zur Verfügung. Dabei werden Nachrichten an einzelne Geräte übertragen, was meist im Rahmen der Erstkonfiguration des Systems genutzt wird.

KNX implementiert keine Sitzungsschicht.

Die Anwendungsschicht definiert Kommunikationsdienste auf Objekten des Informationsmodells von KNX, dem **Interworking Model**. Kernelement des Informationsmodells sind die **Data Points**, über die ein KNX-Gerät Zustandsdaten und Konfigurationsparameter verwaltet.

Die Data Points eines KNX-Geräts können unterschiedliche Rollen besitzen. Inputs und Outputs repräsentieren operative Daten eines Geräts, z. B. den Schaltzustand einer Leuchte oder die Vorlauftemperatur eines Heizkreislaufs. Im Gegensatz dazu stellen Parameter die Konfigurationseinstellungen eines Geräts dar. Data Points werden durch ihren Data Point Type (DPT) weiter spezifiziert. Der Haupttyp legt Format und die Bitcodierung fest,

z. B. Ganzzahl, Wahrheitswert oder Datum. Ein Subtyp kann die anwendungsspezifische Semantik der Daten definieren, z. B. absolute Temperatur, Leistung in kW oder RGB-Wert einer Beleuchtung.

Um eine Anwendung der Gebäudesystemtechnik auf Basis von KNX zu realisieren, wird ein Output eines Geräts mit beliebig vielen Inputs eines anderen Geräts verbunden. Für die Bindung müssen Output und Inputs auf demselben Datentyp basieren. Die **Bindung** wird anhand einer gemeinsamen Gruppenadresse hergestellt, die somit eine virtuelle Leitung zwischen verteilten Sensoren und Aktoren repräsentiert. Ist die Bindung hergestellt, führt die Änderung eines Outputs in einem KNX-Gerät automatisch dazu, dass ein oder mehrere Inputs anderer Geräte den geänderten Wert annehmen. Anstelle einer festen Verkabelung zwischen einem Schalter und einer Leuchte wird die Bindung zwischen Output und Inputs bei der Einrichtung des Systems konfiguriert und kann jederzeit geändert werden. Die lokale Anwendung im Gerät muss sich dabei nicht mit den Kommunikationsdetails befassen. Ein Data Point stellt auf der Anwendungsschicht somit die Schnittstelle zu den unterliegenden Kommunikationsschichten dar und abstrahiert von den Details der Datenübertragung über den Datenbus.

Die Data Points eines KNX-Geräts können in zwei verschiedenen Formen auftreten. Ein Gerät kann eine flache Liste von **Group Objects** besitzen oder Eigenschaften (*Properties*) als Teil eines **Interface Object** organisieren. Interface Objects dienen meist dazu, Konfigurationsparameter eines Geräts zu bündeln. Sowohl Group Objects als auch Properties besitzen zusätzliche Möglichkeiten, um Metadaten zu verwalten.

Die Dokumente des KNX-Standards definieren das Informationsmodell von KNX lediglich verbal bzw. mithilfe eigener Abbildungsformen. Abb. 5.13 zeigt einen Versuch, einen Ausschnitt des KNX Interworking Model in Form eines Klassendiagramms der UML darzustellen.

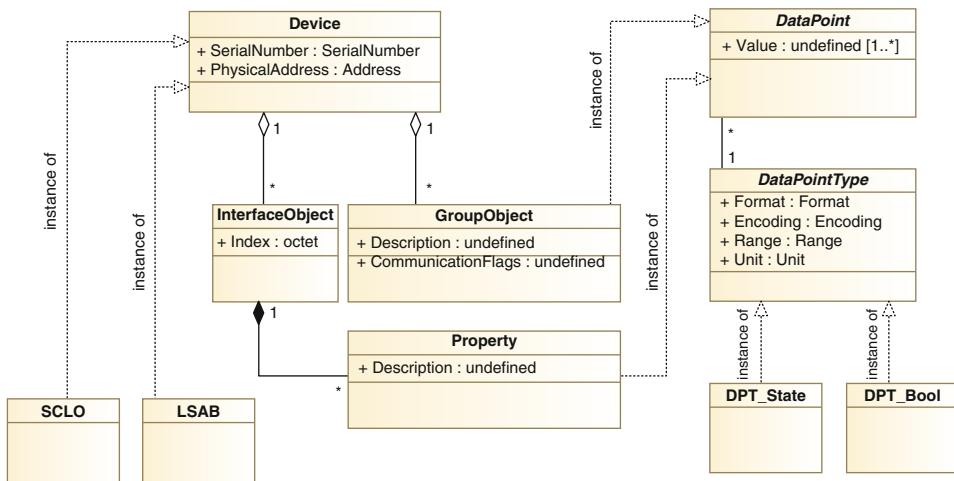


Abb. 5.13 Ausschnitt aus dem KNX Interworking Model in Form eines Klassendiagramms der UML

In Abschnitt 7 des KNX-Standards wird eine ganze Reihe von standardisierten Funktionsblöcken eingeführt. Darin wird festgelegt, welche Inputs, Outputs und Parameter ein Gerät besitzen muss, um z. B. als Lichtschalter (*Light Switching Actuator Basic*, LSAB) oder als Systemuhr (*System Clock*, SCLO) zu gelten. Ein Gerätehersteller muss sich an diese Vorgaben halten, damit Geräte durch die KNX-Assoziation zertifiziert werden können.

Nachdem die Hardwareseite eines KNX-Systems aufgebaut wurde, müssen Geräte konfiguriert und die Bindung hergestellt werden. Dazu wird ein entsprechendes Inbetriebnahmewerkzeug benötigt. Die KNX-Organisation stellt dazu die kostenpflichtige Engineering-Tool-Software (ETS) für das Betriebssystem Windows bereit. ETS kann über einen USB-Koppler bzw. einen KNX/IP-Router direkt auf eine KNX-Installation zugreifen, um mithilfe von Unicast-Nachrichten die Konfigurationseinstellungen an die Geräte zu senden. Abb. 5.14 zeigt ein Bildschirmfoto von ETS.

Die Konfiguration eines KNX-Netzwerks mithilfe von ETS besteht dabei i. d. R. aus den folgenden Schritten:

- der Import der Produktkataloge der genutzten KNX-Geräte,
- das Erstellen einer virtuellen Gebäudestruktur und die darauf basierende Vergabe der physikalischen Adressen an die Geräte,
- die Erstellung von Gruppenadressen auf Basis einer Gewerkestruktur.

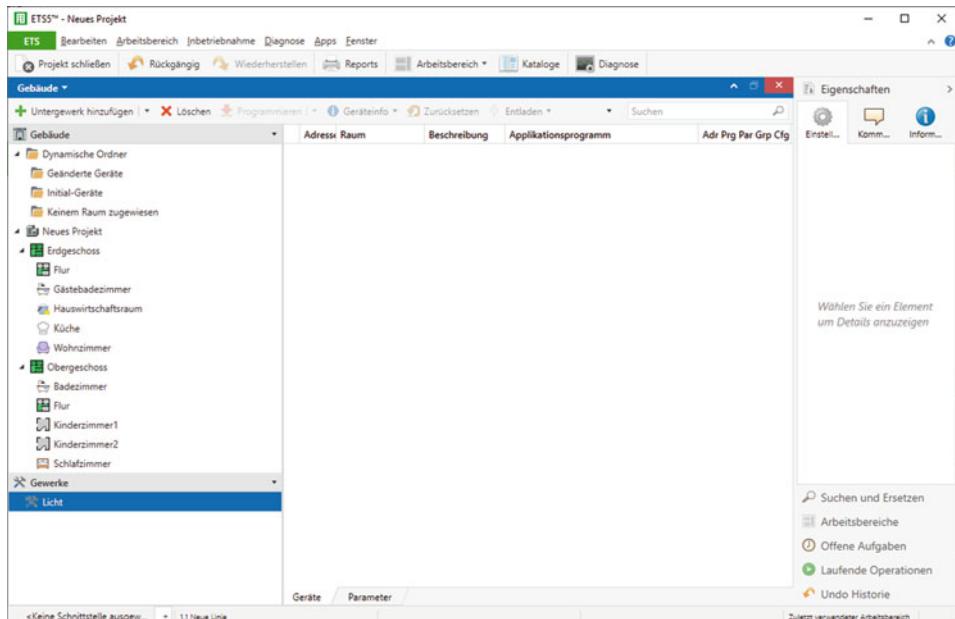


Abb. 5.14 Bildschirmfoto der ETS-Software. Copyright durch KNX Association

Um KNX-kompatible Geräte zu realisieren, kann z. B. ein Arduino-Mikrokontroller als Basis eingesetzt werden. Der Arduino wird dazu über seine serielle Schnittstelle mit einem Buskoppler, auch als Bus Coupling Unit (BCU) bezeichnet, verbunden, der die Aufgabe übernimmt, die Bitübertragungsschicht für KNX TP zu realisieren. Für die Erzeugung und Interpretation von KNX-konformen Nachrichten existieren für Arduino mehrere freie Programmabibliotheken. Im Internet finden sich mehrere Projekte, in denen auf dieser Basis KNX-kompatible Geräte aufgebaut wurden (siehe z. B. [56]).

Um KNX-Projekte auf PC-Architekturen umsetzen zu können, ist das sog. Calimero-Projekt hilfreich (siehe [4]). Die Java-Bibliothek bildet die ISO/OSI-Ebenen 2–7 des KNX-Systems ab und kann dazu genutzt werden, um mit KNX-Geräten zu kommunizieren oder KNX-Geräte vollständig in Software nachzubilden. Calimero kann auch dazu genutzt werden, um mithilfe einer preisgünstigen KNX/USB-Schnittstelle einen vollwertigen KNXnet/IP-Server zu realisieren (siehe [60]).

LonWorks ist ein zu KNX ähnlicher Kommunikationsstandard, der ebenfalls eher der Raumautomation zuzurechnen ist. LonWorks wird ebenfalls (aber nicht ausschließlich) in der Gebäudeautomation eingesetzt. LonWorks basiert auf der Idee, dass in einem sog. Local Operating Network (LON) mehrere Geräte über ein gemeinsames Kommunikationsprotokoll namens LonTalk Daten austauschen.

LonWorks wurde in den 1980er-Jahren von der Firma Echelon entwickelt. Die Firma wurde u. a. von Mike Markulla gegründet, einer der ersten CEOs und Investoren von Apple. Im Jahr 2018 wurde Echelon dann von Adesto übernommen. LonWorks wurde zwar in einem Abschnitt von ISO/IEC 14908 standardisiert, einige Teile des Systems existieren aber weiterhin ausschließlich in Form proprietärer Produkte von Echelon. LonWorks kann daher als *weniger offen* angesehen werden als z. B. KNX oder BACnet.

LonWorks und KNX verbindet eine ähnliche Idee, wie verteilte Automationsanwendungen erstellt werden können. Auch hier werden Anwendungen auf Basis gleichberechtigter, verteilter Geräte realisiert, deren Daten über eine virtuelle Verbindungen miteinander gekoppelt werden können. LonWorks implementiert dafür den gesamten ISO/OSI Stack. Die Plattform stellt dabei einige Möglichkeiten mehr zur Verfügung als z. B. KNX. Mit den LonWorks Network Services (LNS) bietet LonWorks z. B. eine zentrale Softwarekomponente für den Zugriff auf ein LON. Im Gegensatz zu KNX können höherwertige Automationsfunktionen so über eine Webschnittstelle zumindest angebunden werden. Abb. 5.15 zeigt die Abbildung von LonWorks auf den ISO/OSI Stack.

Ein LonWorks-Gerät wird als Knoten (*Node*) bezeichnet. Hardwareseitig basiert ein solcher Knoten meist auf dem **Neuron-Chip**, einem von Echelon speziell entwickelten Prozessorsystem, der von Toshiba und Cypress gefertigt wird. Die Namensgebung soll an die Neuronen eines menschlichen Gehirns erinnern. Auch im Gehirn existiert keine zentrale Steuerung. Jedes Neuron implementiert eine spezifische Funktionalität, die Gesamtintelligenz ist auf sehr viele solcher Neuronen verteilt. Fällt ein Neuron aus, funktioniert das restliche Gehirn aber weiter. Der Neuron-Chip realisiert die Schichten 2–6 des ISO/OSI Stack. In seinem Speicher kann auch die Anwendungslogik eines Geräts abgelegt werden. Diese wird in der Programmiersprache Neuron C entwickelt.

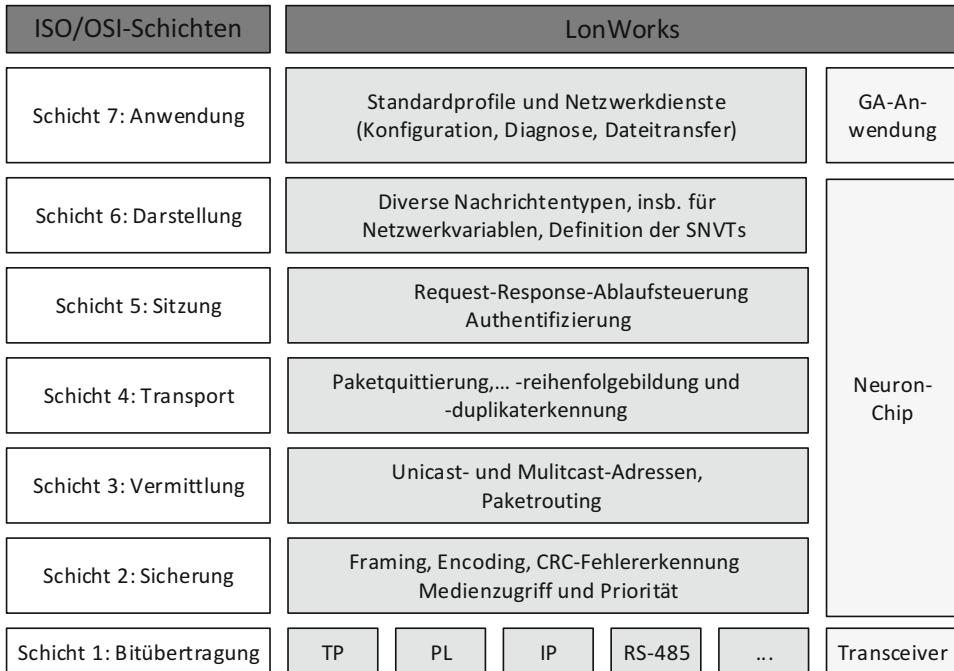


Abb. 5.15 LonWorks und das ISO/OSI-Modell

Die Bitübertragungsschicht von LonWorks wird meist durch ein eigenes Bauteil abgebildet, den sog. Transceiver. Dieser wird an den Neuron-Chip angeschlossen oder ist dort bereits integriert. Es existiert eine große Vielfalt von Transceivern für unterschiedlichste Übertragungsmedien, z. B. Twisted Pair, Powerline, Funkübertragung, Lichtwellenleiter, Infrarot, RS-485 und viele andere. Mithilfe des beliebten Transceivers TP/FT-10 können z. B. auf Basis von Twisted-Pair-Kabeln beliebige Topologien aufgebaut werden (auch Ringe). Die Daten werden dabei mit ca. 78 kBit/s übertragen, deutlich schneller als bei KNX.

LonTalk führt auf der Sicherungsschicht für die Nachrichtenübertragung ein eigenes Frameformat ein. Dieses besteht aus einer Präambel, dem Link-Layer-Header, den Nutzdaten, der CRC-Prüfsumme und dem sog. End-of-Packet-Indicator. Nachrichten werden asynchron übertragen, alle Geräte synchronisieren sich jedoch auf dem Bus auf einem gemeinsamen Takt über den End-of-Packet-Indicator einer vorhergehenden Nachricht.

LonTalk setzt zur Medienzugriffssteuerung eine spezielle Variante des CSMA mit dem Namen **Predictive P-persistent CSMA** ein (siehe z. B. [3]). Kollisionen werden behandelt, können aber nicht vermieden werden. Im Falle einer Kollision wartet ein Gerät eine zufällig gewählte Wartezeit, um den Buszugriff erneut zu prüfen. Die Wartezeit wird dabei dynamisch an die Netzauslastung angepasst. Je größer der Datenverkehr auf dem Bus, desto größer wird auch die Breite der Wartezeit gewählt. Dieses Verfahren sorgt aktiv für

eine niedrige Kollisionsrate, auch über dem Sättigungspunkt des Übertragungsmediums hinaus (siehe auch [5, S. 47 f.]).

Die Sicherungsschicht führt zudem Prioritäten für Nachrichten ein. Für hoch priorisierte Nachrichten müssen kürzere Wartezeiten nach der letzten Nachricht auf dem Bus eingehalten werden als für niedriger priorisierte. Innerhalb eines Geräts und in Routern werden hoch priorisierte Nachrichten zudem an den Anfang der Nachrichtenwarteschlange gestellt.

LonTalk erlaubt mithilfe von Routern oder Repeatern den Aufbau beliebig komplexer Topologien auf Basis gleicher oder unterschiedlicher Übertragungsmedien, sog. Channels. Damit Nachrichten in einer solchen Topologie erfolgreich zugestellt werden können, werden für das Routing von Nachrichten entsprechende Mechanismen eingeführt. LonTalk definiert dazu auf der Vermittlungsschicht ein zusätzliches Datagram-Format, welches als Nutzlast im Frameformat der Sicherungsschicht verpackt wird. Als Bestandteil des Datagram-Formats werden dabei unterschiedliche Arten von Zieladressen eingeführt (siehe [5, S. 50–59]).

Grundsätzlich besitzt jedes Gerät zunächst eine feste, global eindeutige Adresse, die **Neuron-ID**. Sie ist 6 Byte groß und kann z. B. für Punkt-zu-Punkt-Kommunikation während der Einrichtung genutzt werden. Ein Gerät erhält zudem während der Einrichtung eine zweite, logische Adresse. Diese ist, ähnlich wie die KNX-Geräteadresse, hierarchisch aufgebaut und besteht aus der Domäne (*Domain*), dem Subnetz (*Subnet*) und dem Knoten (*Node*). Die Domäne stellt eine logische Sammlung von Geräten dar, z. B. innerhalb eines Gebäudes, die über mehrere Channels verteilt sein können. Geräte können nur innerhalb einer Domäne miteinander kommunizieren. Innerhalb einer Domäne werden Geräte auf Subnetze aufgeteilt. Es können pro Domäne bis zu 255 Subnetze definiert werden, die wiederum bis zu 127 Geräte beinhalten können. Die dritte Art von Adressen ist die Gruppenadresse. Ein Gerät kann so konfiguriert werden, dass es innerhalb einer Domäne Teil von bis zu 15 Gruppen ist.

Auf Basis dieser Adressen werden unterschiedliche Kommunikationsarten abgebildet. Für Unicast-Nachrichten wird die Neuron-ID oder eine eindeutige Kombination aus der Domain-, Subnet- und Node-Adresse verwendet. Multicast-Nachrichten basieren auf der Gruppenadresse und Broadcasts können mithilfe der Domain- und/oder der Subnet-Adresse versandt werden.

Auf der Transport- und Sitzungsschicht führt LonTalk mehrere Nachrichtendienste ein. Die Dienste unterscheiden sich vor allem im Bezug zur Zuverlässigkeit der Nachrichtenübermittlung. Am verlässlichsten ist der bestätigte Nachrichtenversand (*Acknowledged*). Eine Uni- oder Multicast-Nachricht wird dabei von allen Empfängern quittiert. Fehlt die Quittierung von einem Empfänger, wird die Nachricht wiederholt. Das Time-out und die Anzahl der Wiederholungen sind dabei konfigurierbar. Beim wiederholten Nachrichtenversand (*Repeated*) wird die Nachricht an alle Empfänger von vornherein mehrere Male hintereinander versendet. Dieses Schema wird i. d. R. dann genutzt, wenn sichergestellt werden soll, dass eine Nachricht zwar von vielen Empfängern garantiert empfangen

werden soll, Quittierungsnachrichten aller Empfänger das Netz aber überlasten könnten. Am wenigsten zuverlässig ist der unbestätigte Nachrichtenversand (*Unacknowledged*). Dieser kann z. B. für den regelmäßigen Versand von unkritischen Messwerten genutzt werden. Die Sitzungsschicht stellt zudem den Anfrage/Antwort-Nachrichtendienst zur Verfügung. Dieser Dienst ähnelt dem bestätigten Nachrichtenversand, die Antwort eines Diensteanbieters kann aber auch Nutzdaten beinhalten. Ist die Zieladresse einer solchen Anfrage ein Broadcast, wird lediglich die erste Antwort verarbeitet, alle anderen Antwortnachrichten werden verworfen. Für die Anwendungsschicht sind diese Abläufe auf der Transport- und Sicherungsschicht nicht sichtbar.

Der wichtigste Beitrag der Darstellungsschicht ist die Einführung der sog. Netzwerkvariablen (*Network Variables*). Die Funktionsweise der Netzwerkvariablen entspricht mehr oder weniger den Data Points von KNX und bildet auch in LonWorks die Basis für verteilte Automationsanwendungen. Netzwerkvariablen werden durch eine Richtung und einen Datentyp charakterisiert. Die Richtung legt fest, ob die Daten gelesen und geschrieben (Input) bzw. nur gelesen (Output) werden dürfen. Der Datentyp einer Variable ist entweder ein einfacher skalarer Typ, wie z. B. *Signed Short* oder *Double Float*, oder ein daraus zusammengesetzter Typ. In einem Gerät wird eine Netzwerkvariable durch eine eigene 14 Bit breite Adresse, den **Network Variable Selector** identifiziert. So kann von außerhalb des Geräts eindeutig auf eine Netzwerkvariable zugegriffen werden.

Wie bei KNX wird die Bindung zwischen Netzwerkvariablen im Rahmen der Einrichtung des Systems konfiguriert. Eine Bindung kann genau dann hergestellt werden, wenn sich die Richtungen der zu bindenden Variablen unterscheiden, die Datentypen aber identisch sind. Im Betrieb sorgt dann der LonTalk-Netzwerk-Stack dafür, dass die Änderung einer Netzwerkvariablen in einem Gerät über entsprechende Nachrichten die Änderung von Netzwerkvariablen in anderen Geräten herbeiführt. Der Entwickler einer Gebäudeautomationsanwendung muss sich mit den Details der Datenübertragung nicht befassen. Bezuglich seiner Netzwerkvariablen ist ein LonWorks-Gerät selbstauskunfts-fähig. Die Netzwerkvariablen, ihre Datentypen und Beschreibungen können über das Kommunikationsnetz ausgelesen werden.

Aus Gründen der Interoperabilität wurden durch die **LonMark** Organisation verschiedene Standards verabschiedet. Dadurch soll die Kompatibilität zwischen Geräten unterschiedlicher Hersteller gewährleistet werden. Zu den wichtigsten Standards zählen die sog. Standard Network Variable Types (SNVT), die Standard Configuration Property Types (SCPT) und Functional Profiles. Die SNVT und SCPT legen die Semantik von Netzwerkvariablen fest. SNVT führen Datentypen für operative Daten eines Geräts ein, z. B. den Schaltzustand eines Tasters, die SCPT hingegen definieren Konfigurationseinstellungen. Alle diese Typen basieren auf einfachen Datentypen, ergänzen diese aber um weitere Angaben zur Bedeutung der Daten. Es werden z. B. die physikalische Einheit (z. B. Celsius), der erlaubte Wertebereich und meist auch ein bestimmter Wert festgelegt, der einen ungültigen Wert reflektiert (*Invalid*). Über den **Scaling Factor** kann der Wertebereich der Variable angepasst werden. Aktuell existieren über 200 SNVT, z. B. `SNVT_Switch`, `SNVT_Power` oder `SNVT_freq_hz`, und über 400 SCPT,

z. B. SCPTwindSpeedLimit, SCPTsceneColor oder SCPTminRnge. Mithilfe von *Functional Profiles* werden Funktionsblöcke für Geräte definiert. Es wird dabei festgelegt, welche SNVT und welche SCPT zu einem Funktionsblock gehören müssen, damit dieser als ein bestimmtes Gerät angesehen werden kann, z. B. als SFPTlampActuator.

Teil der LonWorks-Architektur sind die sog. **LNS**, eine datenbankbasierte Softwarekomponente, die einen zentralen Zugriff auf ein LON anbietet. Die LNS verwalten alle Metadaten zu den Geräten und Netzwerkvariablen eines LON und stellen u. a. eine einheitliche Client/Server-Schnittstelle für Einrichtungs- und Administrationswerkzeuge bereit. Echelon bietet darauf aufbauend weitere Produkte, wie z. B. den sog. i.LON Smartserver, an, um auch auf operative Daten der LonWorks-Geräte über eine Webschnittstelle zugreifen zu können. Auf dieser Basis können auch höherwertige Managementfunktionen oder SCADA-Systeme angebunden werden.

Das sog. IzoT Commissioning Tool (ehemals LonMaker) der Echelon Corporation nutzt ebenfalls die LNS. Dieses grafische Einrichtungswerkzeug für das Betriebssystem Windows hilft bei der Konfiguration der Geräte und der Administration und Überwachung eines LON-Systems. Es bietet eine CAD-ähnliche Umgebung zur Erstellung von LonWorks-Netzen und nutzt dabei Microsoft Visio zur Visualisierung der Netzstruktur. Für die Einrichtung der Geräte werden allerdings Lizenzgebühren der Echelon Corporation fällig, sog. Credits. Eine Alternative zu diesem Werkzeug sind die Werkzeuge der Firma Newron Systems. Diese setzen ebenfalls auf den LNS auf, können allerdings kostenfrei eingesetzt werden.²

Auch im LonWorks-Umfeld existieren Möglichkeiten, neue Geräte zu entwickeln bzw. mit Geräten im Netz zu kommunizieren. LonWorks-Geräte basieren i. d. R. auf dem Neuron-Chip und entsprechenden Transceivern. Für die Entwicklung solcher Geräte bietet Echelon eine ganze Reihe von Entwicklungskits und Werkzeugen an. Ähnlich wie bei KNX kann aber auch auf Basis von z. B. Arduino eine Kommunikation mit LonWorks realisiert werden, zumindest dann, wenn ein LonWorks-Transceiver genutzt wird (siehe z. B. [36]).

Building Automation and Control Networks (BACnet) ist ein weiterer Kommunikationsstandard speziell für den Bereich der Gebäudeautomation. Er wurde von der amerikanischen ASHRAE entwickelt, der *American Society of Heating, Refrigeration, and Air Conditioning Engineers Inc.*³ Die aktuelle Version des Standards ist ANSI/ASHRAE Standard 135-2016, in Europa ist er als ISO 16484-5 verabschiedet. Die Idee hinter BACnet ähnelt ein wenig den Zielen hinter dem IEC 61850 (siehe Abschn. 5.1). Anstelle den Fokus auf die technischen Details der Datenübertragung zu legen, stehen eher die Interoperabilität der Geräte und ein entsprechendes Informationsmodell im Vordergrund. Dieses Informationsmodell soll auch dann weiter gültig bleiben, wenn in Zukunft neue Technologien für die Datenübertragung entstehen und in den Standard integriert werden.

²<http://www.newron-system.com/spip.php?article22>.

³<http://www.ashrae.org>.



Abb. 5.16 BacNet-Versuchsstand zur Simulation einer Heizungsanlage im Gebäudeautomationslabor der Hochschule Hamm-Lippstadt

Abb. 5.16 zeigt den BACnet-Versuchsstand zur Simulation einer Heizungsanlage im Gebäudeautomationslabor der Hochschule Hamm-Lippstadt.

BACnet definiert auf dem Informationsmodell eine Vielzahl von Diensten. Die Funktionalität von BACnet geht dabei weit über das hinaus, was KNX und LonWorks anbieten. Neben der Basisfunktionalität eines Kommunikationsprotokolls der Gebäudeautomation, dem Lesen und Schreiben verteilter Gerätedaten, werden auch hochkomplexe Abläufe und Geräte auf der Managementebene unterstützt. BACnet orientiert sich dabei am Client/Server-Modell. Ein Server stellt auf Basis seiner Objekte, den Instanzen des Informationsmodells, den Clients Dienste über das Kommunikationsnetz zur Verfügung. Der Client kann sich vom Server auch über Datenänderungen und Alarne informieren lassen. Diese Kommunikationsart entspricht dann dem Publish/Subscribe-Modell. Wir werden hier den Standard nur oberflächlich betrachten. Weitergehende Informationen zum Standard sind z. B. in [20] und [40] zu finden.

BACnet basiert auf einem reduzierten Protokollstapel und implementiert lediglich die Bitübertragungs-, die Sicherungs-, Vermittlungs- und Anwendungsschicht. Abb. 5.17 zeigt die Zuordnung des BACnet-Protokollstapels zum ISO/OSI-Modell.

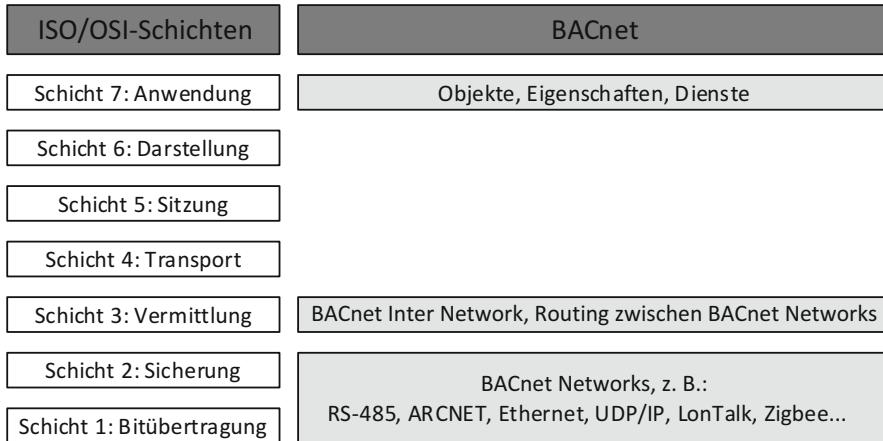


Abb. 5.17 Die vier Schichten des BACnet-Kommunikationsstapels

Auf der Bitübertragungs- und Sicherungsschicht versucht BACnet, das Rad nicht grundsätzlich neu zu erfinden. Größtenteils werden bereits bestehende Verfahren aufgegriffen und ggf. erweitert, die Vermittlungs- und Anwendungsschicht ist dabei für alle Übertragungsmedien identisch. Aktuell stehen als Übertragungsmedien RS-485, ARCNET, Ethernet, UDP/IP, LonTalk und Zigbee zur Verfügung. Bei der Datenübertragung über RS-485 mussten noch relativ viele BACnet-eigene Festlegungen getroffen werden. Für die Kollisionsvermeidung wird z. B. ein Verfahren namens Master-Slave/Token-Passing genutzt, welches vom Prinzip her dem Verfahren von Profibus sehr ähnlich ist (siehe 2.6). Bei der Nutzung von Ethernet und LonTalk werden die Nachrichten lediglich in Frames der jeweiligen Standards verpackt. BACnet-Geräte, die über LonTalk kommunizieren, sind dabei allerdings nicht mit LonWorks-Geräten kompatibel, da unterschiedliche Anwendungsschichten genutzt werden.

Bei Verwendung von UDP über IP-Netze und Zigbee führt BACnet jeweils einen zusätzlichen **Virtual Link Layer** ein. Bei UDP/IP wird dieser als BVLL bezeichnet, bei Zigbee als BZLL. Dadurch werden aus der Sicht von BACnet Funktionalitätslücken von IP-Netzen und Zigbee ausgeglichen. Bei IP-Netzen muss BACnet z. B. ein spezielles Gerät einführen, das BACnet Broadcast Management Device (BBMD), um damit Broadcast-Nachrichten über Subnetzgrenzen hinweg zu ermöglichen. Dafür ist aber die Vergabe von statischen IP-Adressen erforderlich, was die Konfiguration und den Betrieb (z. B. im Bezug mit NAT) mitunter schwierig macht.

Ein einzelnes Netzwerk auf Basis einer dieser Übertragungsmedien wird im Kontext von BACnet auch als BACnet Network bezeichnet. Jeder der Standards verwendet dabei sein eigenes Adressschema, um Endpunkte zu identifizieren, z. B. eine IP-Adresse bzw. eine Ethernet-MAC-Adresse.

Auf der Vermittlungsschicht realisiert BACnet ein Routing zwischen mehreren unterschiedlichen BACnet Networks. Jedes dieser Netzwerke besitzt dazu eine eindeutige

Netzwerknummer. Router können dann mehrere solcher Netzwerke zu einem großen BACnet *Internetwork* verbinden. BACnet-Geräte können dadurch medien- und netzwerkübergreifend miteinander kommunizieren.

Ein einzelnes Gerät in einem solchen Internetwork wird über die Kombination der Netzwerknummer und der MAC-Adresse des BACnet-Netzwerks identifiziert. Um das Routing einfach zu halten, wird dabei nur ein einzelner Kommunikationspfad zwischen zwei BACnet-Netzwerken erlaubt.

Ein BACnet Internetwork ist in der Lage, mehrere unterschiedliche Automationsstandards über ein gemeinsames Backend miteinander zu integrieren. Dazu können entweder native BACnet-Geräte in den unterschiedlichen Netzwerken über Router miteinander kommunizieren oder mehrere Automationsstandards über Gateways integriert werden. Dies ist einer der Hauptgründe, warum BACnet oft auf der Managementebene eingesetzt wird.

Auf der Anwendungsschicht werden für die jeweiligen Anwendungsfälle insgesamt acht unterschiedliche Nachrichtentypen eingeführt. Diese Nachrichtentypen unterscheiden sich z. B. auch aufgrund der gewählten Adressierungsart (Unicast, Multicast, Broadcast). Jeder Nachrichtentyp wird auf Basis von ASN.1 auf APDUs abgebildet. Die Anwendungsschicht von BACnet übernimmt dabei zusätzlich Aufgaben, die der Pseudotransportschicht von DNP3 ähneln (siehe Abschn. 5.1). Nachrichten werden bei Bedarf unabhängig von den unterliegenden Kommunikationssystemen in mehrere TPDUs segmentiert und wieder korrekt zusammengesetzt. Eine Aufgabe, die normalerweise von der Transportschicht übernommen wird.

Kernelement der Anwendungsschicht ist aber das objektorientierte Informationsmodell. Obwohl bei BACnet das Informationsmodell eine im Vergleich zu KNX oder LonWorks herausragende Stellung einnimmt, wurde aber in den Standarddokumenten dennoch keine Modellierungssprache wie UML eingesetzt, um dieses zu spezifizieren. Abb. 5.18 stellt dennoch einen Versuch dar, einen kleinen Ausschnitt des BACnet-Informationsmodells in Form eines Klassendiagramms der UML darzustellen.

Geräte verfügen wie bei KNX oder LonWorks über Daten, die über das Netzwerk gelesen oder geschrieben werden sollen, z. B. Messwerte oder berechnete Werte. Das Informationsmodell definiert dabei die Semantik der in den Nachrichten verpackten Daten.

Ein BACnet-Gerät organisiert alle seine Daten in Form von **Objekten**. Diese Objekte stellen die Schnittstelle dar, die für die Realisierung von Anwendungen der Gebäudeautomation genutzt werden können. Ein Objekt besteht aus einer Menge von Eigenschaften (*Properties*), welche die eigentlichen Träger der Daten sind. Objekte sind somit Container der eigentlichen Daten.

Eine einzelne Eigenschaft wird durch mehrere Attribute charakterisiert, z. B. einen eindeutigen Bezeichner, einen Datentyp und den *Conformance Code*. Der Datentyp kann ein einfacher skalarer Typ, wie Real oder CharacterString, oder ein zusammengesetzter Typ wie die BACnetAddress sein. Der *Conformance Code* kann die Werte „R“, „W“ oder „O“ annehmen. Dabei steht „R“ für nur lesbar und verpflichtend vorhanden, „W“ für schreibbar und verpflichtend vorhanden und „O“ für optional.

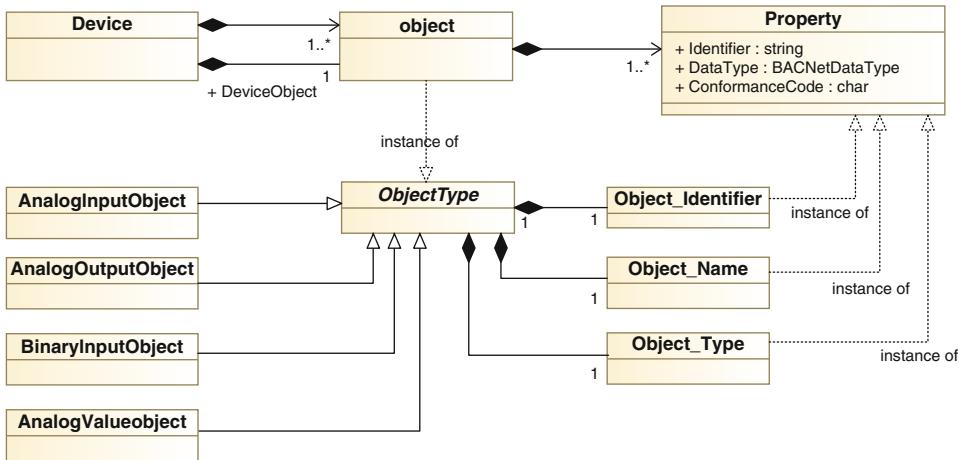


Abb. 5.18 Ausschnitt aus dem BACnet-Informationsmodell als Klassendiagramm der UML

Ein Objekt repräsentiert oft einen bestimmten Funktionsblock eines BACnet-Geräts, z. B. einen Sensor, Aktor. Ein solches Objekt kann aber auch komplexe Konzepte, wie z. B. ein Trendprotokoll oder einen Ereignisbeobachter, realisieren. Eigenschaften von Objekten können sowohl Instanz- als auch Metadaten repräsentieren.

Ein Objekt basiert immer auf einem **Objekttyp**. Ein solcher Objekttyp legt die Eigenschaften fest, die ein Objekt besitzen muss, um eine bestimmte Art eines Funktionsblocks zu realisieren. Beispiele für solche Objekttypen sind **AnalogInput**, **BinaryInput**, **Schedule**, **Calendar** oder **File**.

Objekte, die Instanzen eines Objekttyps sind, besitzen verpflichtend einige weitere Eigenschaften, z. B. **Object_Identifier**, **Object_Name**, **Object_Type** und **Property_List**. Diese bieten die Möglichkeit der Selbstauskunft über ein BACnet-Objekt. Die meisten Objekte verfügen über eine Eigenschaft **Present_Value**, die den aktuellen Wert der Eigenschaft beinhaltet. Viele Objekte bieten zudem verschiedene weitere Eigenschaften, welche die Validität der Daten anzeigen oder über die physikalische Einheit des Wertes Auskunft erteilen.

Neben den Objekten, die Funktionsblöcke repräsentieren, besitzt ein BACnet-Gerät mindestens ein weiteres Objekt, das *Device-Objekt*, welches die Selbstauskunftsfähigkeit des Geräts ermöglicht. Über mehrere Eigenschaften stellt dieses Objekt verschiedene Metadaten zum Gerät zur Verfügung, z. B. den Herstellernamen und die Liste der Objekttypen und Dienste, die das Gerät unterstützt.

Dienste beschreiben Wege, wie BACnet-Geräte miteinander auf Basis der Objekte und deren Eigenschaften interagieren. Die verfügbaren Dienste reichen von recht elementaren Operationen, wie z. B. dem Lesen einer einzelnen Eigenschaft, bis zu hochkomplexen Abläufen, wie dem Einrichten einer Alarmierung. Derartige Abläufe setzen dann die Existenz ganz bestimmter Objekte und Eigenschaften in den Geräten voraus.

Die BACnet-Dienste können in mehrere Klassen eingeteilt werden:

- Die **Object Access Services** dienen dazu, die Eigenschaften von Objekten über das Netzwerk lesen und schreiben zu können. Beispiele für diese Dienste sind `ReadProperty` oder `WriteProperty`.
- Die **Alarm und Event Services** helfen dabei, Abonnements auf Zustandsänderungen von Eigenschaften einzurichten bzw. über Fehler automatisiert informiert zu werden. Beispiele für diese Dienste sind `SubscribeCOV` oder `ConfirmedCOVNotification`.
- Die **File Access Services** werden für die Übertragung von (Konfigurations-)Dateien genutzt. Ein Beispiel ist der Dienst `AtomicReadFile`.
- Die **Remote Device Management Services** dienen dazu, Geräte und Objekte im Netzwerk ausfindig zu machen. Beispiele sind `Who-Is`, `Who-Has` oder `I-Am`.
- Die letzte Gruppe von Diensten sind die **Virtual Terminal Services**, die dazu genutzt werden können, eine bidirektionale Verbindung zu einem Programm innerhalb eines Geräts zu eröffnen. Diese Dienste werden heute nicht mehr allzu häufig genutzt. Ein Beispiel ist `VT-Open`.

In komplexen Fällen kann ein Service einen Zustand über eine Zustandsmaschine verwalten. In diesem Fall werden der erlaubte Ablauf und Inhalt von Nachrichten festgelegt. In jedem Fall werden für jeden Service die Priorisierung und die Segmentierung von Nachrichten definiert und es wird festgelegt, wann Abbrüche, Zeitüberschreitungen und Wiederholungen stattfinden sollen. Alle BACnet-Dienste können dabei mit lediglich acht unterschiedlichen Nachrichtentypen realisiert werden, die sich z. B. aufgrund der Adressierungsart (Unicast, Multicast, Broadcast) unterscheiden. Jeder Nachrichtentyp wird auf Basis von ASN.1 auf Bitströme abgebildet.

Die elf Dienste aus der Dienstklasse Alarne und Ereignisse erweisen sich gerade im Bereich der Gebäudeautomation als besonders nützlich. Die in dieser Klasse organisierten Dienste befassen sich insgesamt mit der Frage, wie Zustände in einer oder mehreren Objekteigenschaften automatisch überwacht, bewertet und gemeldet werden können. Die Nutzung dieser Dienste vermeidet die Notwendigkeit, Eigenschaften zyklisch abfragen zu müssen (Polling), was den Netzwerkverkehr reduziert und die Sicherheit der Anlage erhöht. Die Möglichkeiten, die BACnet durch die Dienstklasse Alarne und Ereignisse anbietet, sind extrem mächtig und gehen weit über das hinaus, was durch KNX und LonWorks unterstützt wird.

Das einfachste Verfahren wird als **ereignisorientierte Datenübertragung** (Change of Value Reporting (COV)) bezeichnet. Dazu melden sich Abonnenten mithilfe des Diensts `SubscribeCOV` oder `SubscribeCOVProperty` an einem Objekt an, über dessen Wertveränderung sie informiert werden wollen. Welche Eigenschaften dabei genau überwacht werden, hängt vom Objekttyp ab, meist handelt es sich aber um die Eigenschaften `Present_Value` bzw. `Status_Flags`. Die aktiven Abonnements verwaltet das meldende Objekt in seiner Eigenschaft `Active_COV_Subscriptions`,

die Größe des Schwellwerts wird in der Eigenschaft COV_Increment abgebildet. Bei einer Wertänderung, die über dem Schwellwert liegt, werden die Abonnenten automatisch über den neuen Wert informiert.

Im Gegensatz zur ereignisorientierten Datenübertragung dienen Ereignisse und Alarme vor allem der Meldung fehlerhafter oder kritischer Zustände. BACnet unterscheidet dabei zwei Verfahren, die als **objektinternes Melden** (*Intrinsic Reporting*) bzw. **regelbasiertes Melden** (*Algorithmic Reporting*) bezeichnet werden. Der einzige Unterschied zwischen beiden Mechanismen liegt darin, ob ein Objekt die eigenen Eigenschaftswerte (*Intrinsic Reporting*) oder Eigenschaftswerte anderer Objekte (*Algorithmic Reporting*) überwacht (siehe [11, S. 13]). Um beide Verfahren abzubilden, führt BACnet eine Reihe zusätzlicher Objekttypen ein, z. B. Event Enrollment, Notification Class, Notification Forwarder und Alert Enrollment (siehe [20]).

Bei beiden Mechanismen können die gleichen vordefinierten Bewertungsregeln (*Algorithms*) genutzt werden, um Eigenschaftswerte zu überwachen. Diese Bewertungsregeln werden zwischen *Event* und *Fault Algorithms* unterschieden. Beispiele für Eventalgorithmen sind CHANGE_OF_VALUE, FLOATING_LIMIT oder OUT_OF_RANGE. Beim objektinternen Melden werden die Bewertungsregeln durch das Objekt selbst ausgeführt, beim regelbasierten Melden findet dies in einem eigenen Objekt, dem Event Enrollment Objekt statt.

Sowohl beim objektinternen Melden als auch beim regelbasierten Melden verwaltet der Überwachungsmechanismus eine Zustandsmaschine. Die Bewertungsregeln lassen deren Zustand zwischen normal, nicht normal und fehlerhaft wechseln. Jeder Zustandswechsel entspricht einem Ereignis, welches gemeldet werden kann. Der aktuelle Zustand ist in der Objekteigenschaft Event_State zugänglich. Die Erzeugung und der Verteilung der Ereignismeldungen übernimmt ein eigenes Objekt, das sog. Meldungsklassenobjekt (*Notification Class Object*). Anhand dieses Objekts kann feingranular festgelegt werden, welche Benachrichtigungen erzeugt werden, welche Priorität diese besitzen, ob die Meldungen quittiert werden müssen und wer diese erhält (mitunter auch zeitabhängig). Die Verteilung der Benachrichtigungen kann auch durch ein eigenes Gerät zentral unterstützt werden. Dieses Gerät stellt dazu ein *Notification-Forwarder-Objekt* zur Verfügung.

Um auf der Managementebene höherwertige Funktionen umsetzen zu können, bedarf es einer Anwendungsschnittstelle zum Gebäudeautomationssystem. Meist wird dafür heute ein zentraler Gateway-Server eingesetzt. Dieser abstrahiert von den unterliegenden Protokollen und bietet einen standardisierten und meist aggregierten Zugriff auf die Informationen. Die Kommunikation mit so einem Gateway wird heute technisch meist auf Basis von Webdiensten realisiert, die auf dem Simple Object Access Protocol (SOAP) oder RESTful-Standard basieren.

KNX bietet hierzu keine intrinsischen Möglichkeiten an. Die Spezifikation einer allgemeinen Webserviceschnittstelle wurde zwar begonnen, wurde am Ende aber verworfen. Entsprechende Anbindungen können bei KNX daher bislang nur auf Basis proprietärer Produkte realisiert werden. Bei LonWorks wird für ein solches Szenario ein Produkt durch

die Firma Echelon angeboten. Der sog. i.LON Smartserver dient auf Basis des LNS als Webservice-Gateway zu einem LON.

Bei BACnet wird hier eine andere Strategie verfolgt. Der Standard wurde in mehreren Schritten um Details zur Ausgestaltung einer standardisierten Webschnittstelle erweitert, die als **BACnet/WS** bezeichnet wird (siehe [33]). Frühere Versionen nutzen SOAP und XML, aktuell werden RESTful-Services auf Basis von JSON-Daten präferiert. BACnet/WS führt ein eigenes Informationsmodell ein. Dieses neue Informationsmodell wurde deshalb notwendig, da die Schnittstelle auch Szenarien abbilden sollte, die auf Basis des BACnet-Informationsmodells so nicht einfach umzusetzen waren, z. B. Auskunft über die physische und logische Struktur des Systems zu liefern. Ein Versuch, dieses Modell in Form von UML darzustellen, findet sich in [32].

Um komplexe Anwendungen der Gebäudeautomation zu realisieren, müssen Geräte unterschiedlicher Hersteller miteinander kombiniert werden. Damit die Geräte zusammenarbeiten können, müssen diese jeweils *passende* Dienste anbieten. Ein einzelnes BACnet-Gerät realisiert dabei in der Regel nur einen bestimmten Ausschnitt des Standards, also nur eine Teilmenge der Dienste. Die sog. BACnet Interoperability Building Blocks (BIBBs) dokumentieren, was ein bestimmtes BACnet-Gerät *kann* (siehe [40, S. 89]). Aktuell sind 102 solcher BIBBs definiert. BIBBs werden anhand einer bestimmten Codierung ausgedrückt. Diese besteht aus drei Teilen, z. B. „DS-RP-B“. Der erste Teil des Codes drückt den sog. Interoperabilitätsbereich aus. Die fünf Interoperabilitätsbereiche sind:

- gemeinsame Datennutzung (DS),
- Alarm und Ereignismanagement (AE),
- Zeitplan (SCHED),
- Trendaufzeichnung (T),
- Device und Netzwerkmanagement (DM).

Der zweite Teil des Codes repräsentiert den Dienst, den das Gerät unterstützt, z. B. RP für Read Property. Der dritte Teil des Codes zeigt an, ob das Gerät den Dienst als Client nutzen kann (A) oder als Server anbietet (B). Der Code „DS-RP-B“ drückt daher aus, dass ein Gerät die gemeinsame Datennutzung über den Dienst „Read Property“ als Server anbietet. Ein solches Gerät kann von einem anderen Gerät genutzt werden, wenn es den BIBB „DS-RP-A“ implementiert.

BACnet-Geräte werden vom Hersteller mit einer Protokollunterstützungsbestätigung (Protocol Implementation Conformance Statement (PICS)) ausgestattet. Darin werden alle BIBBs angeführt, die das Gerät unterstützt. Ein Planer kann auf dieser Basis die Geräte auswählen, die er für bestimmte Anwendungsfälle benötigt. BACnet führt zudem eine Reihe von sog. Gerätetypen (*Device Profiles*) ein, die für jeden Gerätetyp festlegen, welche BIBBs unterstützt werden müssen. Neben Profilen für einfache Aktoren (B-SA) und Sensoren (B-SS) sind auch komplexe Geräte auf der Managementebene definiert, z. B. BACnet Advanced Workstation (B-AWS). Der Arbeitskreis Maschinen-

und Elektrotechnik staatlicher und kommunaler Verwaltungen (AMEV) hat darüber hinaus einige weitere sog. Gerätetypen eingeführt (siehe [55]).

Für die Inbetriebnahme, Konfiguration und Diagnose eines Gebäudeautomationssystems werden auch bei BACnet entsprechende Werkzeuge benötigt. Im Gegensatz zu KNX oder LonWorks existiert hier aber kein Standardwerkzeug eines einzelnen Herstellers oder einer Interessensgruppe. Entsprechende Lösungen werden von mehreren Geräteherstellern und offenen Projekten angeboten. Auf den Webseiten des BACnet-Standards existiert eine Übersicht zu kommerziellen und freien Werkzeugen.⁴

Ein Beispiel für ein kommerzielles Diagnosewerkzeug ist der sog. BACnet Quick Test der Firma Polarsoft.⁵ Mithilfe dieser Client-Anwendung für das Betriebssystem Windows können die Geräte in einer bestehenden BACnet-Installation ausfindig gemacht und deren Objekte und Dienste untersucht werden. Für die Nutzung der Anwendung muss eine Lizenz erworben werden.

Die Visual Test Shell for BACnet ist ein weiteres solches Werkzeug.⁶ Auch damit können bestehende BACnet-Installationen untersucht, aber auch konfiguriert werden, z. B. COV-Abonnements. Das Werkzeug erlaubt zudem die feingranulare Analyse der BACnet-Nachrichten auf den Übertragungsmedien sowie automatisierte Tests von Geräten mithilfe einer Scriptsprache. Bei der Visual Test Shell handelt es sich um kostenfrei nutzbare Public-Domain-Software.

Um zu BACnet kompatible Anwendungen zu entwickeln, existiert eine Vielzahl von freien oder kommerziellen Bibliotheken für diverse Programmiersprachen, wie Python, Java oder C#. Für die Entwicklung eigener BACnet-Geräte existiert auch eine Bibliothek in der Programmiersprache C. In der aktuellen Version wird bereits eine Vielzahl von Objekten und Diensten unterstützt. Der Programmcode ist auf diverse Plattformen portierbar (z. B. Windows, Linux, RTOS und diverse Mikrocontroller) und steht unter der GNU General Public License. Der Quellcode des Projekts kann von der Sourceforge-Plattform heruntergeladen werden.⁷ Mit dem Open-Source-Projekt esp8266-BACnet wurde diese Bibliothek auch auf den Mikrocontroller ESP8266 portiert. Dieser 32-Bit-Mikrocontroller kostet oft weniger als 3 € und verfügt bereits über integriertes WLAN. Einfache Sensoren oder Aktoren können auf dieser Basis sehr preisgünstig entwickelt werden.

Die **Open Platform Communications United Architecture (OPC UA)** ist ein allgemeiner Datenaustauschstandard für die Maschine-zu-Maschine-Kommunikation. OPC UA findet im Umfeld der sog. Industrie 4.0 Anwendung und verspricht dort, die herstellerunabhängige Kommunikation von der Feld- bis zur Managementebene zu ermöglichen. OPC UA wird in sehr vielen Bereichen eingesetzt, u. a. auch in der Prozessautomatisierung der Energiebranche oder für die Anbindung von Human Machine Interface (HMI) in der

⁴<http://www.bacnet.org/Developer/>.

⁵<https://polarsoft.com/bqt.asp>.

⁶<https://sourceforge.net/projects/vts/>.

⁷<http://bacnet.sourceforge.net/>.

Gebäudeautomatisierung. OPC UA wird auch gerne als gewerkeübergreifender Standard genutzt, wenn es um die sog. Enterprise Integration von Automationsanwendungen geht. OPC UA ist der Nachfolger von OPC, die auf der proprietären Technologie COM/DCOM von Microsoft basierte. OPC UA dagegen ist plattformunabhängig.

Im OSI-Modell macht OPC UA lediglich Vorgaben für die Sitzungs-, Darstellungs- und Anwendungsschicht. Für die Datenübertragung auf niedrigeren OSI-Schichten werden Standardprotokolle, wie TCP/IP, UDP, HTTP oder MQTT genutzt. OPC UA ist als IEC 62541 standardisiert (siehe [35]). Der Kern des Standards besteht derzeit aus 14 Teilen und kann von den Webseiten der OPC Foundation kostenlos heruntergeladen werden.⁸ OPC UA wurde durch das Reference Architecture Model for Industry 4.0 (RAMI 4.0) als einzige Empfehlung für die Umsetzung der Kommunikationsschicht gelistet (siehe [49]).

Wie auch BACnet trennt auch OPC UA die Informationsmodellierung von den Transportmechanismen der Daten. Der Kern des Standards ist ein sehr mächtiges Informationsmodell. Dieses basiert auf einer eigenen Modellierungssprache, samt einer grafischen Notation, die nicht direkt zur UML kompatibel ist. Objektorientierte Konzepte, wie Vererbung und Instanziierung, werden darin in Form eigener Beziehungstypen umgesetzt. Im Anhang von Teil 3 des Standards wird aber auch eine Übersetzung der Elemente in UML vorgeführt (siehe [42]). Abb. 5.19 zeigt einen Ausschnitt dieses Modells. Es ist zu beachten, dass die Assoziationen, wie z. B. HasComponent, eine OPC-UA-eigene Beziehung realisieren und auf einem spezifischen Beziehungstyp (*Reference Type*) basieren.

Die Grundelemente des Informationsmodells sind Knoten (*Nodes*) und Beziehungen (*References*; siehe [6, S. 22 ff.]). Sie sind Elemente des Metamodells und werden in den Teilen 3 und 5 des Standards definiert. OPC UA modelliert auf Basis dieser Elemente mehrere sog. Basisinformationsmodelle, z. B. um aktuelle Geräteinformationen, historische Daten bzw. Ereignisse darzustellen. Die Knoten und Beziehungen, die ein OPC UA Server bereitstellt, werden als sein Adressraum (*Address Space*) bezeichnet. Abb. 5.20 zeigt zwei beispielhafte Adressräume von OPC-UA-Servern, die aus Knoten und Beziehungen aufgebaut sind.

Ein **Knoten** besitzt Attribute, die wiederum durch weitere Eigenschaften, wie z. B. einen Namen, beschrieben werden. Jeder Knoten besitzt darüber hinaus die Fähigkeit, Beziehungen zu anderen Knoten aufzubauen. Solche Beziehungen können ganz unterschiedliche Bedeutungen haben, z. B. dass ein Objekt ein Container für Daten ist, eine Methode Parameter besitzt oder eine Variable auf einen Datentyp verweist. Knoten treten grundsätzlich in Form verschiedener Knotentypen auf, die ihnen eine Bedeutung geben. Durch den Knotentyp wird festgelegt, welche Attribute und Beziehungen vorhanden sein müssen. Die wichtigsten Knotentypen sind Objekt, Variable und Methode.

Ein Knoten kann **Beziehungen** zu anderen Knoten aufbauen, was man sich wie einen gerichteten Zeiger vorstellen kann. Dem Zielknoten muss die Beziehung nicht bekannt

⁸<https://opcfoundation.org/>.

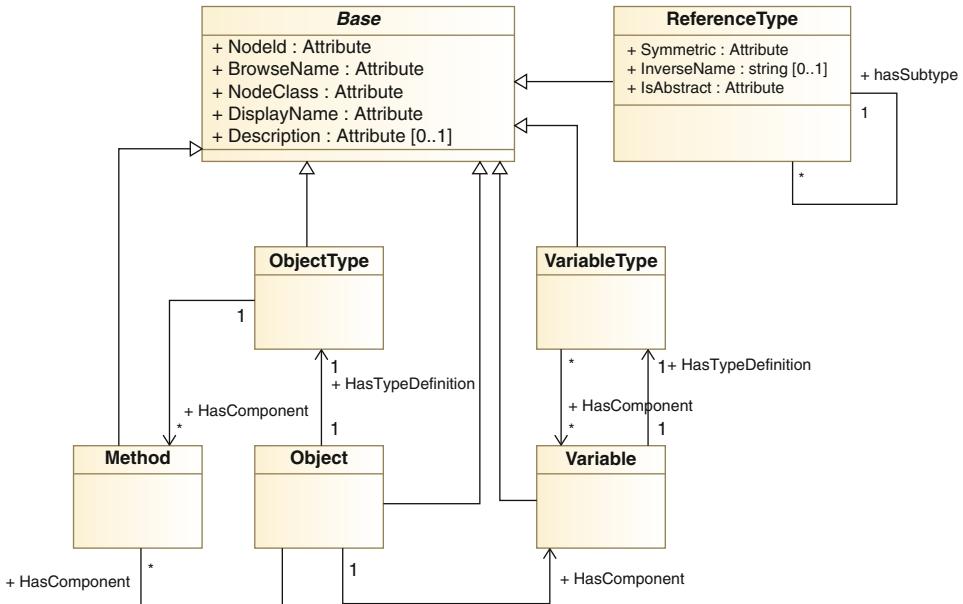
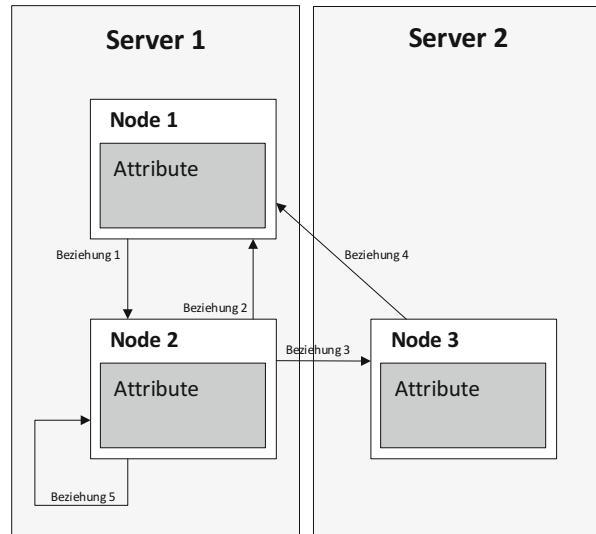


Abb. 5.19 Ausschnitt aus dem UML-Klassendiagramm des OPC-UA-Informationsmodells

Abb. 5.20 Beispielhafte Adressräume von zwei OPC-UA-Servern mit Knoten und Beziehungen



sein und er muss auch nicht zum selben Adressraum gehören. Eine Beziehung besitzt selbst keine eigenen Attribute. Die Bedeutung einer Beziehung wird aber durch einen Beziehungstyp beschrieben, der selbst wieder ein Subtyp eines Knotens ist. Ein Beziehungstyp besitzt z. B. die Attribute `IsAbstract`, `Symmetric` und `InverseName`. OPC UA führt eine ganze Reihe von **Beziehungstypen** ein. Diese sind in einer Hierarchie,

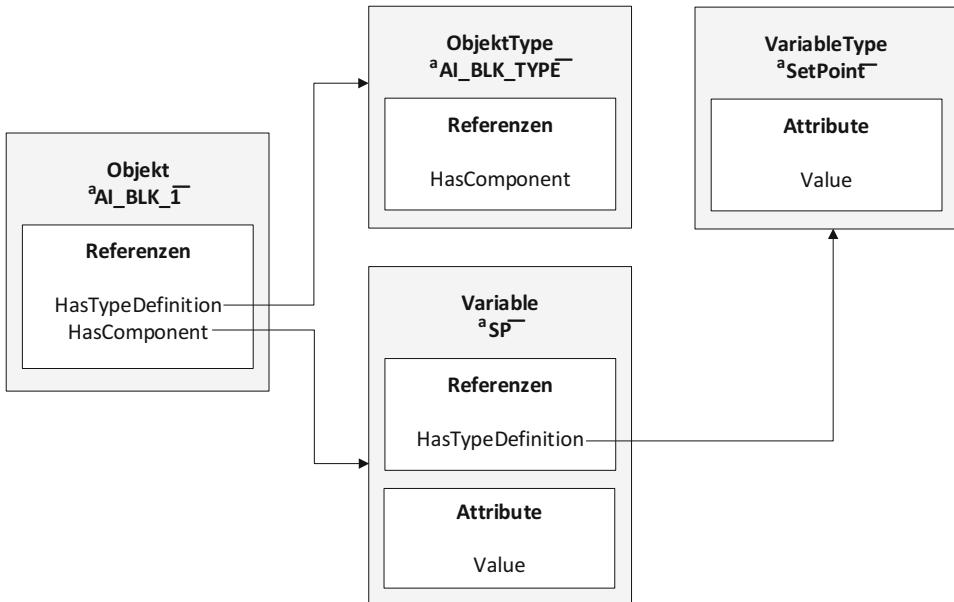


Abb. 5.21 Beispiel für ein Beziehungsgeflecht zwischen Objekten, Objekttypen, Variablen und Variablenarten in OPC UA. In Anlehnung an [42, S. 10]

ähnlich einer Vererbungshierarchie angeordnet. Beispiele für solche Beziehungstypen sind **HasComponent**, **HasProperty**, **HasTypeDefinition** oder **HasSubtype**. Neben den vordefinierten Beziehungstypen können auch neue Typen eingeführt werden.

Objekte referenzieren ihre Variablen über den Beziehungstyp **hasComponent**. Über den Beziehungstyp **hasTypeDefinition** wird zudem auf einen Objekttyp verwiesen, der dadurch festlegt, welche Attribute und Referenzen (z. B. Variablen) das Objekt besitzen muss. Für Objekttypen existiert wie bei den Referenztypen eine vordefinierte Hierarchie. Ein Beispiel eines vordefinierten Objekttyps ist z. B. **FolderType**, mit dessen Hilfe der Adressbereich eines Servers in einer baumartigen Hierarchie von Knoten organisiert werden kann. Ein Gerätehersteller kann auch eigene, neue Objekttypen einführen. Abb. 5.21 zeigt ein beispielhaftes Beziehungsgeflecht zwischen unterschiedlichen Knotentypen.

Abschnitt 4 des OPC-UA-Standards befasst sich mit Diensten auf dem Informationsmodell (siehe [43]). Dienste stellen abstrakte und von der Datenübertragung unabhängige Methoden bereit, damit die Daten eines OPC-UA-Servers gefunden, gelesen, abonniert und manipuliert werden können. Dabei kann sowohl auf die Bewegungs- als auch auf Metadaten, wie z. B. die Typinformationen, über diese Dienste zugegriffen werden. OPC-UA-Server sind daher auch selbstauskunftsfähig. Die OPC-UA-Dienste sind nicht zustandslos, wie z. B. bei Hypertext Transfer Protocol (HTTP). Auf mehreren Abstraktionsebenen existieren Kommunikationskontakte, z. B. eine Session. Die Mehrheit der OPC-UA-Dienste befasst sich mit der Verwaltung solcher Kontexte. Auch ein Abon-

nement von Benachrichtigungen wird durch so einen Kontext dargestellt. Ähnlich zu BACnet bietet OPC UA hier umfangreiche Möglichkeiten an. Die Benachrichtigung über die Änderungen von Variablenwerten und von Objektereignissen basieren dabei auf dem Publish/Subscribe-Modell. Viele der restlichen Dienste setzen auf das Client/Server-Prinzip, z. B.:

- das Auffinden von Servern und Knoten,
- das Lesen und Schreiben von Knotendaten,
- das Aufrufen von Methoden und Programmen,
- die Manipulation des Adressraums eines Servers.

Für die Datenübertragung wird auch bei OPC UA eine Abbildung des Informationsmodells auf Datenmodelle benötigt. Kap. 6 des OPC-UA-Standards führt mehrere solcher Abbildung ein, z. B. ein Binärformat, XML und JavaScript Object Notation (JSON). Ähnlich zu BACnet werden sämtliche Anfragen und Antworten auf nur wenige Nachrichtentypen abgebildet. Für den Nachrichtentransport erlaubt OPC UA prinzipiell alle Technologien, die eine Vollduplexkommunikation erlauben. Da OPC UA sowohl das Client/Server- als auch das Publish/Subscribe-Modell nutzt, sind verschiedene Protokolle mehr oder weniger gut geeignet. Für das Client/Server-Modell bieten sich z. B. Kommunikationsprotokolle, TCP/IP, HTTP/HTTPS und WebSockets an. Der Standard beinhaltet hierzu entsprechende Hinweise. Das Publish/Subscribe-Prinzip kann leichter mithilfe von User Datagram Protocol (UDP) oder Messaging-Middleware, wie z. B. MQTT, realisiert werden.

Die Modellierung des Adressraums eines OPC-UA-Servers ist mitunter recht arbeitsintensiv. Um ein reales Gerät abzubilden, müssen entsprechende Objekttypen definiert, Objekte abgeleitet und mit Variablen ausgestattet werden. Auch die Abbildung der Ereignisverarbeitung und der Darstellung historischer Daten bedarf eigener Informationsmodelle. Die Modellierung geschieht dabei meist mithilfe spezieller Werkzeuge. Ein Beispiel ist der OPC UA Address Space Model Designer der Firma CommServer.⁹ Das Werkzeug steht unter der MIT-Lizenz und kann kostenlos von Github bezogen werden.¹⁰ Die Serialisierung des Informationsmodells geschieht in einem eigenen XML-Format, welches im Anhang des 6. Teils des OPC-UA-Standards definiert wird (siehe [44]). Dadurch wird der herstellerunabhängige Austausch von Informationsmodellen zwischen Modellierungswerkzeugen ermöglicht. Einen ähnlichen Ansatz verfolgt im Kontext des IEC 61850 auch die SCL (siehe Abschn. 5.1).

⁹ <http://www.commsvr.com/Products/OPCUA/UAModelDesigner.aspx>.

¹⁰ <https://github.com/mpostol/ASMD>.

Beispiel 5.2.1 Das folgende XML-Dokument stellt einen Adressraum eines OPC-UA-Servers dar. Das Modell besteht aus einem Objekt, welches über nur eine Variable verfügt.

```

1 <?xml version='1.0' encoding='utf-8'?>
2 <UANodeSet xmlns="...">
3   <Aliases>
4     <Alias Alias="Double">i=11</Alias>
5     <Alias Alias="Organizes">i=35</Alias>
6     <Alias Alias="HasTypeDefinition">i=40</Alias>
7     <Alias Alias="HasComponent">i=47</Alias>
8   </Aliases>
9   <UAObject BrowseName="1:MyObject" NodeId="ns=1;i=1" ParentNodeId
10    = "i=85">
11     <DisplayName>MyObject</DisplayName>
12     <Description>MyObject</Description>
13     <References>
14       <Reference IsForward="false" ReferenceType="Organizes">i
15         =85</Reference>
16       <Reference ReferenceType="HasTypeDefinition">i=58</Reference
17         >
18       <Reference ReferenceType="HasComponent">ns=1;i=2</Reference>
19     </References>
20   </UAObject>
21   <UAVariable AccessLevel="3" BrowseName="1:MyVariable"
22     DataType="Double" NodeId="ns=1;i=2"
23     ParentNodeId="ns=1;i=1" UserAccessLevel="3">
24     <DisplayName>MyVariable</DisplayName>
25     <Description>MyVariable</Description>
26     <References>
27       <Reference IsForward="false" ReferenceType="HasComponent">ns
28         =1;i=1</Reference>
29       <Reference ReferenceType="HasTypeDefinition">i=63</Reference
30         >
31     </References>
32     <Value>
33       <uax:Double>6.7</uax:Double>
34     </Value>
35   </UAVariable>
36 </UANodeSet>
```

Um eigene OPC-UA-Server und -Clients zu realisieren, existiert eine Vielzahl von Programmzbibliotheken. Auf der Github-Plattform stellt die OPC Foundation selbst einige Referenzimplementierungen unter der GPL 2.0 bereit, z. B. für C# und Java.¹¹ Es wird auch ein sog. Model Compiler angeboten, der ein in XML serialisiertes Informations-

¹¹<https://github.com/OPCFoundation>.

modell in eine C#-Implementierung für einen OPC-UA-Server übersetzen kann. Für eingebettete Systeme bietet sich das Projekt Open62541 an. In diesem Projekt arbeiten diverse Forschungseinrichtungen und Unternehmen gemeinschaftlich an einer schlanken C-basierten Programmzbibliothek. Ein OPC-UA-Server kann mithilfe dieser Bibliothek in weniger als 100 kB realisiert werden. Das Projekt steht unter der Mozilla Public License v2.0 und kann kostenlos von Github heruntergeladen und genutzt werden.¹² Mit dem Projekt FreeOpcUa existiert auch für die Programmiersprache Python eine Programmzbibliothek. Diese steht unter der LGPL 3.0 und kann ebenfalls von Github bezogen werden.¹³

Beispiel 5.2.2 Der folgende Python-Programmcode zeigt die Realisierung eines minimalen OPC-UA-Servers auf Basis der FreeOpcUa-Programmzbibliothek. Darin wird ein Server gestartet und dem Adressraum ein neues Objekt mit einer einzelnen Variablen hinzugefügt. Der Server empfängt eingehende Anfragen über TCP/IP auf dem Port 4840.

```
1 import time
2 from opcua import ua, Server
3
4 endpoint = "opc.tcp://0.0.0.0:4840/freeopcua/server/"
5 uri = "http://examples.freeopcua.github.io"
6
7 server = Server()
8 server.set_endpoint(endpoint)
9 idx = server.register_namespace(uri)
10 objects = server.get_objects_node()
11
12 myobj = objects.add_object(idx, "MyObject")
13 myvar = myobj.add_variable(idx, "MyVariable", 6.7)
14 myvar.set_writable()
15
16 # starting:
17 server.start()
18
19 try:
20     while True:
21         time.sleep(1)
22 finally:
23     server.stop()
```

¹²<https://github.com/open62541/open62541>.

¹³<https://github.com/FreeOpcUa/python-opcua>.

Dieser Server kann mit dem folgenden Beispiel-Script abgefragt werden:

```
1 from opcua import Client
2 from opcua import ua
3
4 endpoint = "opc.tcp://localhost:4840/freeopcua/server/"
5 client = Client(endpoint)
6
7 client.connect()
8 root = client.get_root_node()
9 myvar = root.get_child(["0:Objects", "2:MyObject", "2:MyVariable
   "])
10
11 print("Value: ", myvar.get_value())
12
13 client.disconnect()
```

Das Metamodell von OPC UA ist so mächtig, dass damit domänenübergreifend Konzepte der Prozessautomatisierung beschrieben werden können. Durch die generische Struktur lassen sich auch andere Informationsmodelle, wie z. B. BACnet, CIM oder IEC 61850, durch das Metamodell der OPC UA abbilden. Für BACnet wurde eine solche Abbildung in Kooperation zwischen der BACnet Interest Group Europa und der OPC UA Foundation geschaffen. Dabei werden die BACnet-Objekte und -Eigenschaften auf eigene Objekttypen in der OPC UA abgebildet. Durch diese Abbildung können standardisierte Gateways geschaffen werden, die ein BACnet Network an ein OPC-UA-System anbinden können (siehe [37]). Wie eine Abbildung des CIM auf OPC UA durchgeführt werden kann, wird in [9] vorgeführt. Dabei werden die CIM-Klassen ebenfalls hauptsächlich auf Objekttypen des OPC UA abgebildet. Dadurch könnten CIM-basierte Daten über die Kommunikationsinfrastruktur von OPC UA ausgetauscht werden. Eine Abbildung des Informationsmodells des IEC 61850 auf OPC UA wird in [13] beschrieben. Eine solche Abbildung hilft aufgrund der Selbststarkunftsähnlichkeit des OPC-UA-Systems dabei, eine generische Anbindung von Schalt- und verteilten Erzeugungsanlagen an SCADA-Systemen zu ermöglichen.

KNX und LonWorks sind typische Kommunikationsstandards, die sich in vielen Belangen recht ähnlich sind. Ihr Hauptfokus liegt auf dem sicheren Datenaustausch auf der Feld- und Automationsebene im Kontext der Gebäudeautomation. Beide Systeme können eine Vielzahl von Übertragungsmedien nutzen. LonWorks bietet sowohl bei den Übertragungsmedien als auch bei den Kommunikationsarten ein wenig mehr Flexibilität als KNX. Beide Systeme setzen auf Dezentralität. Auf der Basis von gemeinsamen Adressgruppen wird eine virtuelle Verbindung zwischen Sensoren und Aktoren realisiert. Beide Systeme definieren dazu jeweils ein eigenes, recht einfaches Informationsmodell. Typisch für reine Kommunikationsstandards ergeben sich diese Informationsmodelle dabei eher implizit aus den Nachrichtenformaten. Im Vergleich zu LonWorks ist KNX tendenziell der *etwas offener* Standard. Bei LonWorks basieren die Geräte meist auf

dem Neuron-Chip und sog. Transceivern für die Anbindung an ein Übertragungsmedium. KNX-Geräte können leichter mithilfe günstiger Mikrocontroller und offener Bibliotheken realisiert werden. Im Kontext von LonWorks existiert zwar die LonMark Organisation, die hauptsächlich die Gerätenormung vorantreibt, man kommt aber kaum umhin, proprietäre Produkte der Firma Echelon einzusetzen. Im Gegensatz zu KNX existieren bei LonWorks verschiedene Schnittstellen, um auf ein Gerätenetz zuzugreifen, sodass höherwertige Automationsaufgaben zumindest ermöglicht werden. Bei KNX muss für die Systemeinrichtung das kostenpflichtige ETS genutzt werden, bei LonWorks stehen alternative Ersteinrichtungswerkzeuge zur Verfügung.

Im Vergleich zu KNX und LonWorks sind BACnet und insbesondere OPC UA keine reinen Kommunikationsstandards mehr, sondern Informationsstandards. Der Kern beider Standards ist ein komplexes Informationsmodell, auf dem eine Vielzahl von Diensten basiert, die im Rahmen des Client/Server-Modells über ein Kommunikationsnetz genutzt werden können. Das Informationsmodell von BACnet orientiert sich dabei an den Notwendigkeiten der Gebäudeautomation, das Informationsmodell von OPC UA geht darüber weit hinaus und kann basierend auf einem eigenen Metamodell auch andere Domänen abbilden. Beide Standards sind offen und unabhängig von einzelnen Herstellern. BACnet greift für die Datenübertragung auf existierende Standards, wie z. B. UDP/IP oder LonTalk, zurück. OPC UA ist dahingehend noch offener und setzt lediglich ein Übertragungsverfahren voraus, welches eine Fullduplex-Kommunikation ermöglicht. Die virtuelle Bindung von Sensoren und Aktoren, wie sie bei KNX und LonWorks umgesetzt ist, wird bei BACnet und OPC UA durch eine Serie von Diensten ersetzt, die ein Abonnement von Zustandsänderungen, Alarmen und Ereignissen ermöglicht. Sowohl BACnet als auch OPC UA bieten dabei einen wesentlich größeren Funktionsumfang als KNX oder LonWorks. Insbesondere sind auch Geräte auf der Managementebene möglich, die höherwertige Automationsaufgaben übernehmen können. BACnet wird gerne als Integrationsplattform genutzt, um verschiedene Gebäudeautomationsstandards, z. B. KNX und LonWorks, in einem gemeinsamen Backend zu integrieren. OPC UA geht sogar einen Schritt weiter und kann als Integrationsplattform für unterschiedliche Automationsdomänen dienen.

5.3 Smart Meter

Die Energiewende sorgt für eine Transformation der Energiesysteme. Anstelle von einigen großen Erzeugungsanlagen, die in die Höchstspannungsnetze einspeisen, kommen mehr und mehr kleine und dezentrale Erzeugungseinheiten zum Einsatz. Zudem kann es dazu kommen, dass Kunden, die zuvor preisfixiert Energietarife genossen haben, zukünftig zu sog. Prosumenten werden, die ihre Energie auf Basis preisflexibler Energietarife beziehen. Ein solches Energiesystem ist nur noch schwierig zentral zu steuern und zu regeln. Die Energieflüsse sollen sich daher möglichst selbstständig, dezentral und intelligent selbst regeln. Ein solches Energiesystem wird häufig auch als Smart Grid bezeichnet. Das Smart Grid stellt enorme Anforderungen an die Energieinformatik. Um Energieflüsse regeln zu

können, werden Daten, wie Last- und Netzzustandsdaten, möglichst in Echtzeit benötigt. Diese müssen erhoben, übertragen und verarbeitet werden. Die flächendeckende Erhebung solcher Daten ist ein Eckpfeiler der Energiewende.

Nach einer längeren Entwicklung wurde im Jahr 2016 das **Gesetz zur Digitalisierung der Energiewende** verabschiedet. Das Ziel ist, die technische Infrastruktur für die Digitalisierung der Energiewende voranzutreiben. Der zentrale Bestandteil des Gesetzes ist das Messstellenbetriebsgesetz (MsbG). Dieses definiert den Begriff des intelligenten Messsystems. Es legt u. a. Rollen, Pflichten, technische Standards und Regularien zum Datenschutz fest. Es gilt hauptsächlich für Strom, in reduzierter Form aber auch für Erdgas.

Das MsbG führt mit dem Messstellenbetrieb auch eine neue Rolle in der Energiewirtschaft ein. Der **Messstellenbetreiber** ist verantwortlich für den Einbau, die Wartung und den Betrieb eines intelligenten Messsystems und erhält die dort erhobenen Daten. Im Standardfall ist zunächst der Verteilnetzbetreiber der sog. grundständige Messstellenbetreiber (gMSB), der Letztkunde kann aber auch einen anderen Messstellenbetreiber frei wählen.

Das Gesetz definiert ein **intelligentes Messsystem** als eine moderne Messeinrichtung die mit einer Kommunikationseinrichtung verbunden ist. Die moderne Messeinrichtung wird dabei als Smart Meter (SM) bezeichnet, die Kommunikationseinrichtung als Smart Meter Gateway (SMG). Gemäß EU-Energiebinnenmarktrichtlinie müssen die Mitgliedstaaten 80 % der Verbraucher mit Smart-Metern ausstatten, wenn dieses wirtschaftlich ist. Die Wirtschaftlichkeit wurde im Auftrag des BMWi durch eine Studie gezeigt (siehe [18]). Nach MsbG besteht bis spätestens zum Jahr 2032 eine Einbaupflicht für Smart Meter. Dabei gelten bestimmte Staffelungen. In Neubauten müssen ab sofort Smart Meter eingebaut werden, bis zum Jahr 2032 auch bei Bestandsbauten. Dabei wird eine Kommunikationseinrichtung in Form eines Smart Meter Gateway nur unter bestimmten Bedingungen Pflicht, z. B. wenn der Energieverbrauch größer als 6000 kWh pro Jahr beträgt oder z. B. eine regenerative Erzeugungsanlage installiert ist. Als Voraussetzung wurde zudem festgelegt, dass die Installation eines intelligenten Messsystems möglich und technisch vertretbar sein muss. Dazu gehört auch, dass SMGs von mindestens drei Herstellern verfügbar sind, die durch das Bundesamt für Sicherheit in der Informationstechnik (BSI) zertifiziert wurden. Die dritte Zertifizierung wurde am 19.12.2019 ausgesprochen. Damit sind nun Geräte der Hersteller Sagencom Dr. Neuhaus, PPC und EMH Metering zertifiziert. Das BSI hat die positive Markterklärung folglich im Januar 2020 veröffentlicht.

Die Zertifizierung eines SMGs geschieht dabei auf Basis der technischen Richtlinien TR-03109 des BSI (siehe [21]). Die Richtlinien wurde stark von der Open Metering System Specification (OMSS) beeinflusst (siehe [24] und [25]). Ein Smart Meter Gateway besitzt dabei mindestens drei Schnittstellen:

- Local Metrological Network (LMN),
- Home Area Network (HAN),
- Wide Area Network (WAN).

Es ergeben sich daraus drei voneinander isolierte Kommunikationsbereiche, zwischen denen der SMG vermittelt. In jedem Kommunikationsbereich werden eigene Anforderungen bzgl. Kommunikationsstandards, Protokolle, Sicherheit und Datenschutz gestellt. Dabei werden auf allen Ebenen des ISO/OSI-Modells Pflichtvorgaben gemacht, aber auch einige Freiheitsgrade eingeräumt.

Im LMN kommuniziert der Smart Meter Gateway mit einem oder mehreren modernen Messeinrichtungen, die sich i. d. R. im selben Gebäude befinden. Die Messdaten werden vom SMG z. B. periodisch abgerufen, mit einem Zeitstempel versehen, tarifiert und gespeichert. Für die Zeitstempel wird ein externer Zeitserver eingesetzt, wofür das Network Time Protocol (NTP) genutzt wird. Der Smart Meter selbst verhält sich dabei meist wie ein Server, er wird durch den SMG über eine Anfrage ausgelesen, der Smart Meter sendet dann eine entsprechende Antwort zurück. Damit dies funktioniert, müssen SMG und SM einige gemeinsame Voraussetzung erfüllen:

- Es muss eine Kommunikationsverbindung aufgebaut worden sein,
- SMG und SM müssen eine gemeinsame Anfragesprache sprechen und
- sie müssen ein gemeinsames Informationsmodell teilen, sodass die Bedeutung der Daten klar ist.

Im LMN können SMG und die SMs drahtlos oder kabelbasiert miteinander kommunizieren. Für alle Schichten des OSI-Modells werden dafür genaue Vorgaben gemacht. Für eine Zertifizierung des SMG ist die Implementierung aller dieser Vorgaben zwingende Voraussetzung. Abb. 5.22 zeigt eine Übersicht zu den Kommunikationsvorgaben des BSI zum LMN eines Smart Meter Gateway.

Für die drahtlose Kommunikation im LMN wird der Wireless M-Bus eingesetzt. Dabei werden die Daten im lizenzenfreien ISM-Band im Bereich von 868 MHz übertragen (siehe [52]). Der Wireless M-Bus setzt dabei auf den allgemeinen Vorgaben des M-Bus-Standards auf und nutzt die dort definierte Anwendungsschicht (siehe [46]). Der M-Bus ist dabei ein typischer Feldbusstandard, der bei den Geräten zwischen Master und Slave unterscheidet. Lediglich der Master darf von sich aus Anfragen an Slaves in Form von Telegrammen versenden, Slaves antworten lediglich auf Anfragen eines Masters. Die grundsätzliche Struktur eines Telegramms und die Syntax der Nutzdaten wird durch den M-Bus-Standard vorgegeben. Ein M-Bus-Telegram ist maximal 255 Byte lang und besteht aus dem Datensatzkopf und den eigentlichen Nutzdaten. Im Datensatzkopf werden durch den Dateninformationsblock (DIB) die Länge, der Typ und die Codierung der Nutzdaten festgelegt. Der Wertinformationsblock definiert die physikalische Einheit und einen Multiplikator für die Nutzdaten. Die erlaubten Datentypen werden in der Feinspezifikation der drahtlosen LMN-Schnittstelle, die auf der OMSS basiert, weiter eingeschränkt. Abb. 5.23 zeigt den strukturellen Aufbau eines M-Bus-Telegramms.

Für die drahtgebundene Übertragung wird RS-485 als Übertragungsmedium eingesetzt (siehe Abschn. 2.5). Als Vermittlungs- und Sicherungsprotokoll kommt dann High-Level Data Link Control (HDLC) zum Einsatz. In der Anwendungsschicht wird mit Companion

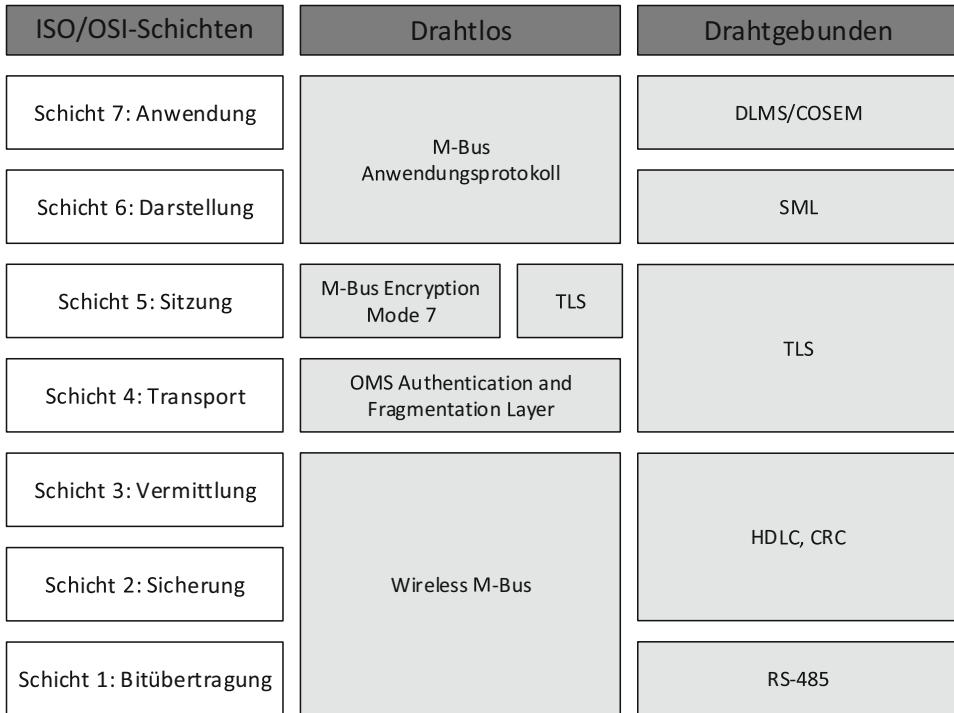


Abb. 5.22 Kommunikationsvorgaben des BSI zum LMN eines SMG. Bildquelle: [21, S. 51]

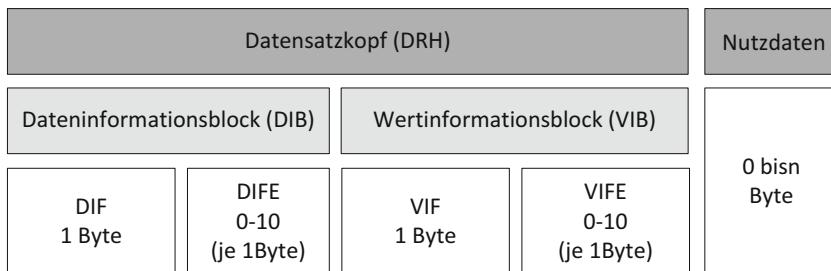


Abb. 5.23 Aufbau eines M-Bus-Telegramms

Specification for Energy Metering (COSEM) ein objektorientiertes Informationsmodell eingeführt (siehe [64]). Ein Smart Meter wird in COSEM als *Physical Device* bezeichnet und besteht aus mehreren logischen Einheiten (*Logical Device*). Jede logische Einheit besitzt einen eindeutigen Namen und bildet eine bestimmte Fähigkeit des Smart-Meters ab, z. B. einen Stromzähler. Eine solche Fähigkeit wird durch ein COSEM Object repräsentiert. Ein COSEM Object besitzt Attribute, z. B. Messwerte, und stellt Methoden bereit, z. B. *reset* oder *start*. Einem Attribut ist genau eine Bedeutung zugeordnet, es besitzt einen Datentyp, einen Wertebereich und Zugriffsrechte (siehe [16] und [41]). Es

existiert mindestens eine logische Einheit, das *Management Logical Device*, welches dazu genutzt werden kann, um eine Liste der logischen Einheiten abzufragen. Dadurch wird ein Smart Meter selbstauskunftsfähig. Abb. 5.24 zeigt den Aufbau eines *Physical Device* nach COSEM.

Das erste Attribut eines Objekts ist immer der *Logical Name*, der die Bedeutung des Attributs festlegt. Für diese Auszeichnung werden Kennzahlen nach dem sog. Object Identification System (OBIS) verwendet. OBIS ist ein internationaler Standard und wird ebenfalls als Teil von IEC 62056 definiert (siehe [64]). Eine OBIS-Kennzahl ist eine Zeichenkette und besteht aus bis zu 6 sog. Wertgruppen. Jede Wertgruppe wird durch ein bestimmtes Trennzeichen von der nächsten Wertgruppe getrennt. Die OBIS-Kennzahl „1.0.1.8.1.255“ steht z. B. für die elektrische Wirkleistung. Tab. 5.1 zeigt die Wertgruppen von OBIS und ihre Bedeutung.

OBIS-Kennzahlen werden noch an einigen anderen Stellen im Energiesystem genutzt, z. B. im Rahmen der Marktkommunikation (siehe Abschn. 5.4). Für die mit dem M-Bus-Protokoll übertragenen Daten existiert zudem eine Abbildung der Datentypen auf die entsprechende OBIS-Bedeutung.

Um die Daten eines SM auf Basis von COSEM abfragen zu können, müssen die COSEM-Objekte auf der Darstellungsschicht in Nachrichten verpackt werden können. Bei der drahtgebundenen Kommunikation zwischen SM und SMG wird dies mithilfe der

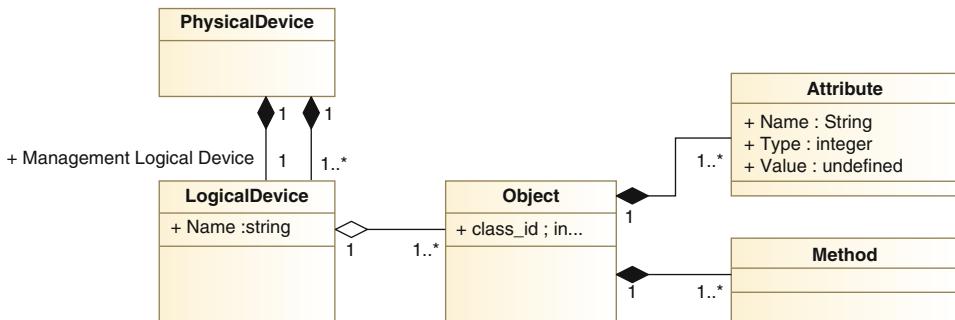


Abb. 5.24 Aufbau eines Physical Device nach COSEM

Tab. 5.1 Die Wertgruppen von OBIS und ihre Bedeutung

Wertgruppe	Stellen	Bedeutung
A	1	Medium, z. B. Elektrizität=1, Gas=7, Wasser=8
B	1–2	Kanal, Separation mehrerer Messkanäle (z. B. Phasen) in einem Gerät
C	1–2	Physikalische Bedeutung, z. B. Wirk-, Blind- oder Scheinleistung, Strom oder Spannung
D	1–2	Messart, z. B. Maximum oder Zeitintegral.
E	1–2	Tarifstufe. Wenn kein Tarif, dann 0
F	1–2	Vorwertzählerstand

Smart Message Language (SML) umgesetzt, die in IEC 62056-5-3 spezifiziert wird (siehe [38]). Nachrichtenformate und Codierungen werden darin genau festgelegt.

Der zweite Kommunikationsbereich des SMG ist das **HAN**. Ein Anwendungsfall für das HAN ist die Bereitstellung von Messdaten für Letztverbraucher und Servicetechniker. Die Messdaten werden dabei für jeden Zähler in aggregierter Form bereitgestellt, z. B. die letzten 5 Tage, die letzte Woche, das letzte Jahr. Das Ziel ist, Transparenz über Verbräuche herzustellen, wobei die Daten dann z. B. in lokalen Dashboards genutzt werden können.

Der zweite Anwendungsfall für das HAN ist weitaus komplexer. Im HAN können an der SMG verschiedene Akteure angeschlossen sein, z. B. regelbare Erzeuger wie Photovoltaikanlagen, Blockheizkraftwerke, oder unterbrechbare Verbrauchseinrichtungen wie Ladesäule oder Nachspeicherheizungen. Damit derartige Akteure von außen geregelt werden können, bietet der SMG im HAN die sog. Controllable-Local-System-(CLS-)Schnittstelle an. Dabei dient der SMG als Proxyserver zwischen den lokalen Akteuren und einem externen Marktteilnehmer, der diese Akteure steuern darf. Für das Steuerungsprotokoll kommt dafür SOCKSv5 zum Einsatz. Dieser Anwendungsfall ist im erheblichen Maße für die Komplexität des SMG verantwortlich.

Für das HAN werden vergleichsweise wenige technische Vorgaben gemacht. Die HAN-Schnittstelle wird auf Basis von Ethernet mit mind. 10 Mbit ausgeführt, das SMG muss dabei IPv4 unterstützen. Die IP-Adresse des SMG kann von einem DHCP-Server empfangen werden. Ansonsten sind keine Vorgaben zu Protokollen gemacht, was mitunter kritisiert wird (siehe [27]).

Der letzte Kommunikationsbereich des SMG ist das **WAN**. Über die WAN-Schnittstelle werden die Messwerte an den Smart Meter Gateway Administrator (SMGA) oder einen externen Marktteilnehmer übertragen, der Zugriff auf die CLS-Schnittstelle wird bereitgestellt und die externe Konfiguration des SMG wird ermöglicht. Auch hier wurden einige verbindliche Standards durch das BSI festgelegt. Abb. 5.25 zeigt die Kommunikationsvorgaben des BSI für das WAN eines Smart Meter Gateway.

In der technischen Richtlinie TR03109-1 werden für den SMG insgesamt 13 sog. Tarifanwendungsfälle (TAF 1 bis TAF 13) definiert. Diese Tarifanwendungsfälle legen fest, wie Messwerte durch den SMG zu verarbeiten sind und wie häufig und aus welchem Grund die Daten übertragen werden sollen. SMGs der ersten Generation unterstützen lediglich die Tarifanwendungsfälle TAF 1, 2, 6 und 7. Die ersten Anwendungsfälle gehören zur Gruppe der Tarifierung und Bilanzierung. Der einfachste Fall ist der sog. datensparsame Tarif (TAF 1), bei dem einmal monatlich Verbrauchswerte an den SMGA übertragen werden. Darüber hinaus existiert auch die Möglichkeit, zeit-, last-, verbrauchs- oder ereignisvariable Tarife im SMG zu hinterlegen. Durch Tarifanwendungsfall 9 (TAF 9) kann zudem die Isteinspeisung einer Erzeugungsanlage übertragen werden. TAF 10 bis TAF 13 sind der Erhebung von Netzzustandsdaten zugeordnet.

Für die Datenübertragung im WAN wird ein eigenes XML-Format (XML for COSEM) genutzt. Die Nachrichten werden dann mittels HTTP-Anfragen an einen RESTful-Service übertragen, der vom SMGA betrieben wird. Welche Übertragungsmedien für die Kommunikation im WAN genutzt werden sollen, ist nicht festgelegt. Hier kommt

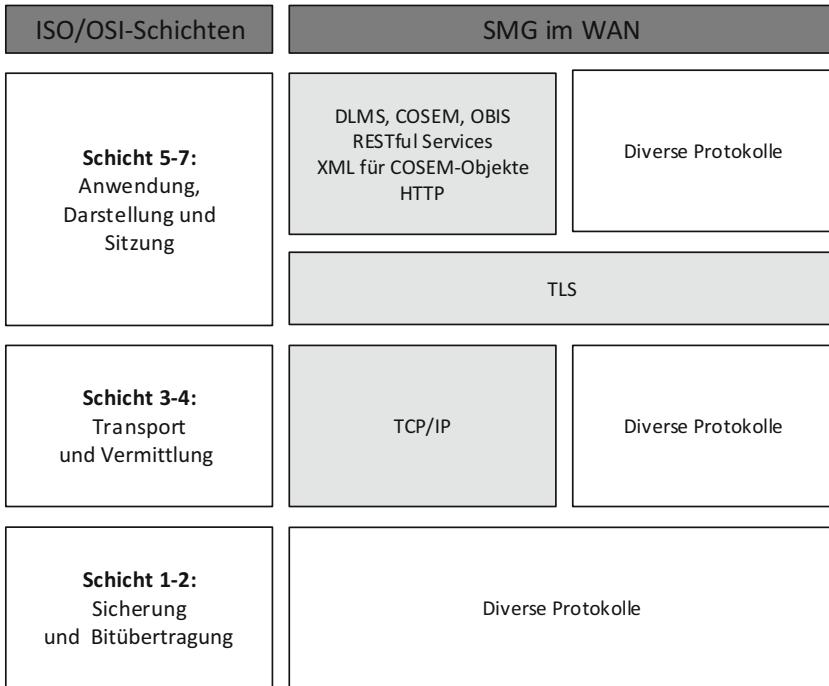


Abb. 5.25 Kommunikationsvorgaben für das WAN eines SMG. Bildquelle: [21, S. 31].

z. B. das öffentliche Mobilfunknetz mit General Packet Radio Service (GPRS) oder Long Term Evolution (LTE) in Betracht, was aufseiten der SMGA aber zu hohen Betriebskosten führen kann. Meist wird auch eine Ethernetschnittstelle angeboten, um bestehende Übertragungskanäle, wie z. B. den lokalen Internetrouter, nutzen zu können. Darüber hinaus existieren viele weitere Möglichkeiten. In der Praxis kommen z. B. Powerline oder LTE Band 31 zum Einsatz.

Powerline ist ein Standard für die Datenkommunikation, bei dem als Übertragungsmedium die Stromleitungen eines Nieder- bis Mittelspannungsnetzes genutzt werden. Für die Datenübertragung wird das Datensignal hochfrequent auf die 50 Hz-Schwingung des elektrischen Stroms aufmoduliert. Der Empfänger demoduliert das Signal wieder, um die Daten zu extrahieren. Für Powerline existieren mehrere, zueinander inkompatible Standards. Im Kontext von SMGs kommt meist IEEE-1901-FFT zum Einsatz, bei dem das Datensignal mittels sog. Orthogonal Frequency-Division Multiplexing (ODFM) moduliert wird. Für den Roll-out von SMGs kann Powerline nützlich sein, da evtl. kein eigenes Telekommunikationsnetz aufgebaut oder genutzt werden muss. Die Reichweite der Übertragung ist mit wenigen Hundert Metern allerdings sehr begrenzt. Der Messstellenbetreiber muss die Daten daher recht früh wieder mit entsprechenden Modems von der Stromleitung entkoppeln.

Im Jahr 2000 wurde der Betrieb des analogen C-Netzes in Deutschland eingestellt. Dadurch wurde die Funkfrequenz um 450 MHz frei, die aktuell befristet bis Ende 2020 an die Deutsche Telekom AG und die 450 Connect GmbH zugeteilt wurde. Die Unternehmen betreiben in diesem Frequenzbereich aktuell Übertragungsnetze auf Basis von LTE. Auf dieser Frequenz wird LTE als Band 31 bezeichnet (siehe [30]). Die Datenübertragung auf 450 MHz hat den Vorteil, mit bis zu 20 km hohe Reichweiten zu erlauben, wodurch im Vergleich zu höheren Frequenzen weniger Funkmasten notwendig werden. Zudem wird auch eine gute Gebäudeausleuchtung ermöglicht, um z. B. SMGs anbinden zu können. Branchenverbände fordern seit einiger Zeit, dass diese Frequenz exklusiv der Energiewirtschaft zur Verfügung gestellt werden soll. Dieser Forderung hat sich nun auch der Beirat der Bundesnetzagentur angeschlossen (siehe [50]). Mit hoher Wahrscheinlichkeit entsteht somit in den nächsten Jahren ein eigenes Telekommunikationsnetz für die Energiewirtschaft in Deutschland. Ein solches Netz kann dann nicht nur für die Zählerfernauslese genutzt werden. Als Voraussetzung für ein Smart Grid kann es dazu genutzt werden, kritische Infrastrukturen (z. B. Schalt- und Erzeugungsanlagen) datentechnisch anzubinden. Die 450 Connect GmbH plant bereits ein deutschlandweites Netz, welches schwarzfallfeste und sichere Datenübertragung bietet.

Die Kommunikation zwischen Smart Meter und Smart Meter Gateways ist relativ kompliziert. Es müssen viele Protokolle, Nachrichtenformate und Standards beachtet werden. Um entsprechende Lösungen zu konstruieren, welche z. B. in der Lage sind, die Daten eines Smart-Meters auszulesen, müssen viele technische Details beachtet und realisiert werden. Ein entsprechendes Projekt wurde an der Ostfalia Hochschule für angewandte Wissenschaften realisiert. Es wurde gezeigt, dass ein LMN-Interface eines SMG softwareseitig so implementiert werden kann, dass es die Daten eines SM zyklisch abfragt und in einer Weboberfläche darstellt (siehe [27]).

Das Fraunhofer-Institut für Solare Energiesysteme aus Freiburg stellt für die Implementierung der Kommunikation zwischen SM und SMG im Rahmen ihrer **OpenMUC**-Initiative verschiedene Bibliotheken zur Verfügung (siehe [57]). Es finden sich mehrere Java-Projekte, um die Protokolle des M-BUS, Wireless M-Bus und DLMS/COSEM abbilden zu können.

Ein weiteres interessantes Projekt wird als **Java-based Open-Source Smart Meter Gateway Experimentation Framework (jOSEF)** bezeichnet (siehe [29]). Das Projekt simuliert dabei das Verhalten eines realen SMG und implementiert die entsprechenden Funktionen standardgemäß. So wird z. B. in regelmäßigen Abständen über das LMN eine Verbindung zu den angeschlossenen Zählern aufgebaut, um mithilfe von COSEM/DLSM die Messwerte abzufragen und über das WAN weiterzuleiten. jOSEF ist ein interessantes Tool, um softwarebasiert ein Netz aus Zählern und einem SMG umzusetzen, wodurch neue Verfahren leicht getestet und realisiert werden können.

5.4 Marktkommunikation

Die zunehmende Spezialisierung der Energieversorgungsunternehmen in der Wertschöpfungskette der Energieversorgung sorgt für einen steigenden unternehmensübergreifenden Abstimmungsbedarf (siehe auch Abschn. 2.1). In vielen Fällen müssen Unternehmen der Energiewirtschaft ein gemeinsames Anliegen bearbeiten. Beispiele dafür sind das Bilanzkreismanagement bzw. die Abläufe des Wechsels von einem Endkunden von einem Energieversorger zu einem anderen. In vielen Situationen müssen in genau festgelegten Prozessen Informationen zwischen unterschiedlichen Akteuren ausgetauscht werden. Der Austausch von Informationen zwischen den einzelnen Rollen des Energiemarktes fällt unter den Begriff der **Marktkommunikation**. Um die Marktkommunikation in Deutschland zu standardisieren, existieren im Endkundenbereich bislang vier Prozessgruppen:

- die Geschäftsprozesse für die Kundenbelieferung mit Elektrizität (GPKE),
- die Marktregeln für die Durchführung der Bilanzkreisabrechnung Strom (MABIS),
- der Geschäftsprozess Lieferantenwechsel Gas (GeLi Gas),
- die Wechselprozesse im Messwesen (WIM).

Jede dieser Prozessgruppen besteht wiederum aus einer Vielzahl von Anwendungsfällen. Die wichtigsten Anwendungsfälle des GPKE sind z. B.:

- Kündigung, Lieferbeginn, Lieferende,
- Ersatz- und Grundversorgung,
- Zählerstand- und Zählerwertübermittlung,
- Netznutzungsabrechnung,
- Stammdatenaustausch.

Die Prozesse, die zur Umsetzung dieser Anwendungsfälle notwendig sind, werden durch die Bundesnetzagentur in Zusammenarbeit mit diversen Verbänden erarbeitet. Der Bundesverband der Energie- und Wasserwirtschaft e. V. (BDEW) spielt hier eine herausragende Rolle. In diesem Verband sind alle gesetzlichen Markttrollen organisiert. In den Prozessen werden nicht nur Verantwortlichkeiten, Fristen und Abläufe festgelegt, es werden insbesondere auch technische Vorgaben gemacht. Dabei werden Datenmodelle und Übertragungswege festgelegt. Abb. 5.26 zeigt den Anwendungsfall des Lieferbeginns aus den GPKE in Form eines sog. Sequenzdiagramms der UML.

Im Beispiel müssen sich die drei Rollen Altlieferant (LFA), Netzbetreiber (NB) und Neulieferant (LFN) in einer wohlgeordneten Abfolge von Nachrichten untereinander abstimmen. Damit dies möglich wird, braucht es klare Rollen und Verantwortlichkeiten. Diese Verantwortlichkeiten wurden in einem **Rollenmodell** für die Marktkommunikation Strom und Gas definiert. Das deutsche Rollenmodell ist dabei kompatibel mit dem entso-e/ebIX/efet Harmonised Electricity Role Model. Einige der wichtigsten Markttrollen und Objekte sind:

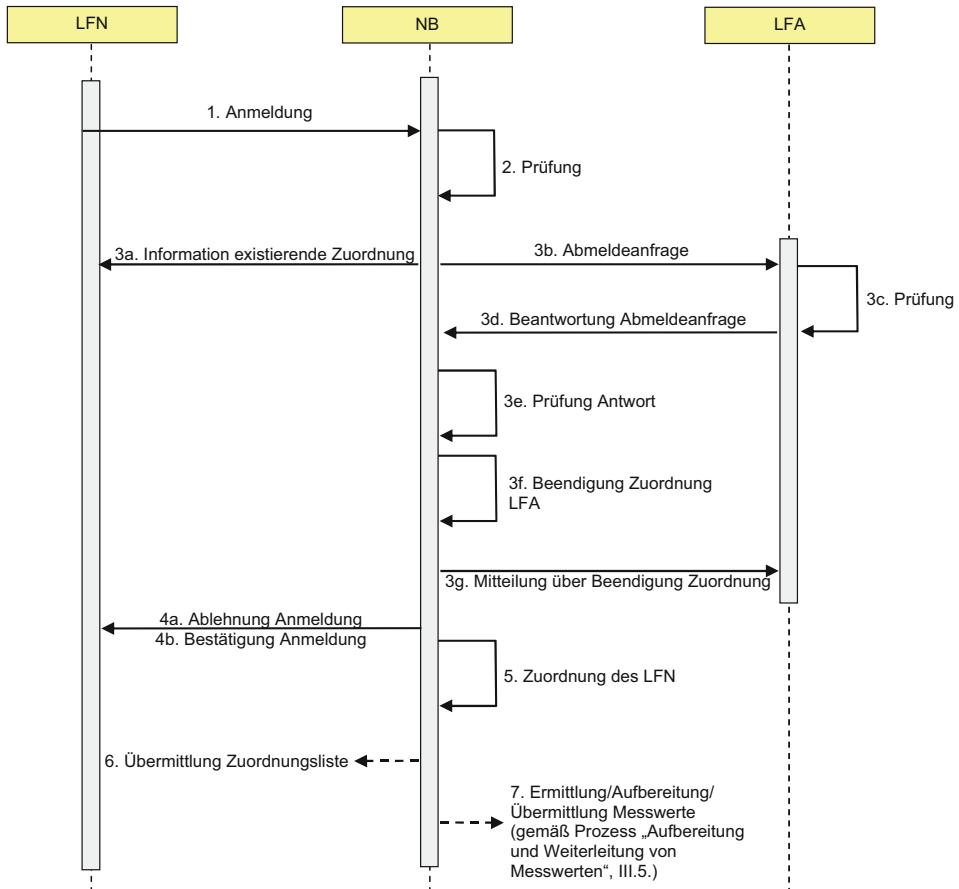


Abb. 5.26 Ablauf des Anwendungsfalls Lieferbeginn aus dem GPKE. Bildquelle: Bundesnetzagentur (BNetzA): Anlage zum Beschluss BK6-06-009, S. 33

- Bilanzkreiskoordinator (BIKO) und Bilanzkreisverantwortliche (BKV),
- Lieferant (LF),
- Netzbetreiber (NB),
- Messstellenbetreiber (MSB),
- Übertragungsnetzbetreiber (ÜNB),
- Bilanzkreis (BK), Marktlokation und Messlokation.

Um diese Prozesse umzusetzen, kommunizieren die Marktteilnehmer direkt miteinander. Für das Beispiel des Lieferbeginns aus Abb. 5.26 sendet z. B. der Altlieferant eine Nachricht an den zuständigen Netzbetreiber. Jedes Unternehmen, welches im deutschen Energiesystem eine Marktrolle einnimmt, muss daher theoretisch mit jedem anderen

kommunizieren können. Vor dem Hintergrund, dass aktuell über 1000 Lieferanten für elektrische Energie in Deutschland existieren, entsteht hier ein großer technischer Aufwand.

Der Austausch der Nachrichten hat nach einem bestimmten technischen Verfahren stattzufinden, welches in Deutschland durch die Arbeitsgruppe **EDI@Energy** des BDEW definiert wird (siehe [63]). Als Nachrichtenformat wird für die gesamte Marktkommunikation ein Datenformat basierend auf dem Electronic Data Interchange for Administration, Commerce and Transport (EDIFACT) eingesetzt. EDIFACT-Nachrichten werden in vielen Branchen, z. B. der Chemie-, Transport- oder Versicherungswirtschaft dazu eingesetzt, um elektronische Daten unternehmensübergreifend auszutauschen. EDIFACT-Nachrichten sind textbasiert. Durch EDI@Energy wurde das sog. UNOC Encoding festgelegt, d. h., es wird der ISO-8859-1-Zeichensatz (Latin-1) benutzt. Für jeden Anwendungsfall existiert ein eigener Nachrichtentyp, der durch einen 6-stelligen Code identifiziert wird. Bei GPKE sind dies z. B.:

- **MSCONS** für die Übermittlung von Lastgängen/Zählerständen,
- **UTILMD** für die Übermittlung von Kundenstammdaten,
- **ORDERS/ORDRSP** für die Übermittlung von Mehr-/Mindermengen.

EDIFACT-Nachrichten werden durch den Empfänger bestätigt. Hierzu existieren besondere Nachrichtentypen, z. B.:

- **CONTROL** für die Ankunftsmeldung und Syntaxprüfung,
- **APERAK** für eine fachliche Fehlermeldung und Quittierung der Meldung.

EDIFACT-Nachrichten bestehen aus sog. Segmenten. Ein Segment ist ein Abschnitt einer Nachricht und stellt eine bestimmte Information dar, z. B. „QTY+1:100“. Für jeden Nachrichtentyp werden verpflichtende und optionale Segmente und ihre Reihenfolge festgelegt. Segmente beginnen mit einer Segmentkennung, z. B. DTM für Datum oder QTY für Menge. Das Ende eines Segments wird i. d. R. durch das Hochkomma angezeigt. Segmente können einen oder mehrere Werte transportieren. Die Daten werden durch die beiden Trennzeichen „;“ und „+“ voneinander getrennt. Segmentkennungen, die mit UN beginnen, haben allgemeine Bedeutungen und kommen in jeder Nachricht vor. Einige Beispiele für solche Segmente sind:

- **UNB** = Anfang der Nachricht, genau einmal pro Nachricht,
- **UNZ** = Ende der Nachricht, genau einmal pro Nachricht,
- **UNH** = Header der Nachricht, genau einmal pro Nachricht.

Der folgende Text zeigt einen Ausschnitt aus einer hypothetischen Rechnung, die als EDIFACT-Nachricht umgesetzt wurde. Für die bessere Lesbarkeit wurden nach den Segmenten Zeilenumbrüche eingefügt. In einer realen EDIFACT-Nachricht wären diese nicht vorhanden.

```

1 UNA:+.?
2 UNB+UNOC:3+990365400007:500+:14+180927:1129+NJMKD9521
3 UNH+ENJMKD9521+INVOIC:D:06A:UN:2.6e
4 BGM+Selbst ausgestellte Rechnung (engl.?: "Self- billed invoice")
   +1+9
5 IMD++ABR
6 RFF+Z13:31006
7 NAD+MS+990365400007::293++Stadtwerke XYZ:::::Z02+Str:::4+Stadt
   ++59065+DE
8 CTA+IC+:Meier, Helmut
9 COM+?+49 2381 2744321:TE
10 COM+n.meier@stadtwerke-hamm.de:EM
11 NAD+MR+++:::::Z02
12 NAD+DP
13 CUX+2:EUR:4
14 LIN+1++9990001000053:Z01
15 QTY+47:5000:KWH
16 DTM+155:20170901:102
17 DTM+156:20180930:102
18 MOA+203:1000.00
19 PRI+CAL:0.20:::::ANN
20 TAX+7+VAT+++:19+S
21 UNS+S
22 MOA+77:1190.0
23 MOA+9:1190.0
24 TAX+7+VAT+++:19+S
25 MOA+125:1000.0
26 MOA+161:190.0
27 UNT+25+ENJMKD9521
28 UNZ+1+NJMKD9521

```

Im Kontext der Marktprozesse müssen in EDIFACT-Nachrichten einige Objekte wie Regelzonen, Unternehmen oder Marktlokationen eindeutig benannt werden. Hierzu existieren bestimmte Codes bzw. Kennungen. Eine Anlaufstelle für die Vergabe und Verwaltung solcher Codes in Deutschland ist die Energie Code & Services GmbH (siehe [62]).

- **Energy Identification Code (EIC)**

Der EIC ist europaweit gültig und wird eingesetzt, um einzelne Anlagen und Marktpartner eindeutig zu kennzeichnen. Die Regelzone Amprion wird z. B. durch den Code „10YDERWENETI“ identifiziert, das Bilanzierungsgebiet der Energie und Wasserversorgung Hamm GmbH besitzt den Code „11YR00000002492A“. Eine Übersicht aller Codes findet sich z. B. beim entso-e (siehe [61]).

- **Marktpartneridentifikationsnummer (MP-ID)**

Anhand der MP-ID kann nach den Festlegungen von EDI@Energy jeder Marktteilnehmer und seine jeweilige Rolle im deutschen Energiemarkt eindeutig identifiziert werden. Die Energie und Wasserversorgung Hamm GmbH besitzt z. B. die MP-ID „990026200007“.

- **Marktlokationsidentifikationsnummer (MaLo-ID)**

Die MaLo-ID ist eine 11-stellige Ziffer (mit Prüfziffer), um Entnahme-, Einspeise- und Messstellen in Deutschland eindeutig zu identifizieren. Sie wird bei elektrischem Strom durch den BDEW, bei Erdgas durch den Deutschen Verein des Gas- und Wasserfaches (DVGW) vergeben. Sie ist seit dem 01.02.2018 verpflichtend für die Marktkommunikation.

- **Stromnetzbetreibernummer (SNB)**

Die SNB identifiziert einen Verteilnetzbetreiber eindeutig. Als Beispiel steht die 262 für die Energie- und Wasserversorgung Hamm GmbH.

Für den Versand von EDIFACT-Nachrichten können unterschiedliche Übertragungswege genutzt werden. In der Praxis wird überwiegend Simple Mail Transfer Protocol (SMTP), also E-Mail, für den Versand der Nachrichten genutzt. Eine EDIFACT-Datei wird dann als Textdatei serialisiert und als Dateianhang an eine E-Mail angehängt. Dabei darf die E-Mail im Nachrichtenkörper keinen Inhalt besitzen und der Betreff der Nachricht muss mit dem Dateinamen der EDIFACT-Datei übereinstimmen. Für den Empfang der Nachrichten besitzen die Marktteilnehmer eine eigene E-Mail-Adresse, z. B. „edifact-strom@ewv-hamm-netz.de“.

Die Nachrichten sind mithilfe von Secure/Multipurpose Internet Mail Extensions (S/MIME) zu signieren und zu verschlüsseln. In Zukunft soll vermehrt AS2/AS4 als Übertragungsweg genutzt werden. Bei diesem Verfahren werden die EDIFACT-Dateien per HTTP übertragen, die EDIFACT-Dateien werden zusätzlich verschlüsselt. Der Vorteil von AS2/AS4 liegt darin, dass der Empfang der Nachrichten quittiert werden kann, was beim Versand der Nachrichten über SMTP nicht der Fall ist.

Das Regelwerk, um gültige EDIFACT-Nachrichten im Rahmen der Marktprozesse zu erzeugen, ist leider recht komplex und zudem regelmäßigen Änderungen unterworfen. In den Energieunternehmen werden daher entsprechende Werkzeuge benötigt, um die Abwicklung der Marktprozesse zu unterstützen, korrekte EDIFACT-Nachrichten zu erstellen, zu versenden und zu interpretieren. Leider existiert aktuell kein freies Werkzeug, welches zur marktkonformen Erzeugung oder Interpretation von EDIFACT-Nachrichten genutzt werden kann. Bis zum Sommer 2019 wurde von der Westnetz GmbH mit den EDIFACT Utilities eine java-basierte Referenzimplementierung bereitgestellt. Die Pflege des Systems wurde aber eingestellt und kann nicht länger genutzt werden.

Auch im Kontext des Bilanzkreismanagements müssen insbesondere der Verteilnetzbetreiber (VNB), der BIKO und der BKV Informationen austauschen. Zur Standardisierung dieser Prozesse wurde MABIS eingeführt. Auch bei MABIS werden dafür EDIFACT-Nachrichten genutzt. Bei der Bilanzkreisabrechnung oder für die Übertragung von Lastprofilen werden MSCONS-Nachrichten genutzt, die insbesondere Zeitreihendaten beinhalten.

Für die Abbildung der Marktkommunikation zwischen den Markttrollen existieren leider keine europaweiten und schon gar keine weltweiten Standards. In ansatzweise jedem Land werden hier eigene Prozesse, Standards und technische Verfahren etabliert.

Wie wir gesehen haben, wird in Deutschland die Marktkommunikation mithilfe von EDIFACT-Nachrichten abgewickelt, die zwischen den Marktrollen direkt per E-Mail ausgetauscht werden. Dieses Verfahren ist komplex und technisch nicht auf der Höhe der Zeit, da es die Energieunternehmen zwingt, eine 1:1-Kommunikation zu allen anderen Unternehmen abilden zu können. Allein die Verwaltung und Pflege der Kontaktdaten jedes Unternehmens erzeugen bereits einen großen Aufwand. In [19] wird daher als eine Verbesserungsmöglichkeit des deutschen Systems u. a. die Einführung eines Zentralregisters vorgeschlagen, in dem die E-Mail-Adressen, Ansprechpartner und kryptografische Schlüssel der Marktpartner verwaltet werden sollen. Der Versand der EDIFACT-Nachrichten per E-Mail ist allerdings an sich zu hinterfragen. E-Mails besitzen mitunter lange Laufzeiten und verfügen über keinen intrinsischen Mechanismus zur Eingangsbestätigung oder zur sicheren Übertragung. Ein Blick nach Österreich zeigt, dass durchaus auch andere, technisch versiertere Verfahren möglich sind. Auch dort existiert zunächst die gleiche Ausgangssituation. Marktpartner müssen auf Basis von Prozessen Informationen austauschen, z. B. wenn Endkunden ihren Energieversorger wechseln.

In Österreich wird für die Marktkommunikation die Kommunikationsplattform für den **energiewirtschaftlichen Datenaustausch (EDA)** betrieben. Alle Informationen zu Prozessen, Datenformaten und Technologien finden sich auf den Webseiten der ebUtilities (siehe [65]). Die EDA stellt eine zentrale Kommunikationsinfrastruktur für alle Marktteilnehmer am Energiemarkt in Österreich dar. Nachrichten werden dafür in einem speziellen XML-Format namens Electronix Business for XML (exXML) dargestellt. Für die Übertragung der Nachrichten wird ein Messagingsystem namens PONTON X/P genutzt, welches im Gegensatz zu E-Mail eine sichere und vertrauliche Nachrichtenübermittlung garantiert. Für PONTON X/P sind zudem eine ganze Reihe von Adapters verfügbar, sodass Anwendungssoftware in die EDA leicht integriert werden kann (z. B. für SAP). Mit ENERGYLink wird zudem speziell für das Wechselwesen eine webbasierte Plattform betrieben. Auf Basis des dort integrierten Self-Storage-Dienstes können Energieversorger sämtliche Prozesse vollständig über diese Plattform abwickeln und müssen dann theoretisch keine weiteren IT-Systeme entwickeln und betreiben. Technisch gesehen, ist die österreichische EDA der deutschen Systematik deutlich überlegen.

Eine weitere technische Alternative zur Umsetzung der Marktkommunikation wäre der Einsatz der **Blockchain-Technologie**. Eine Blockchain ist dabei zunächst nichts anderes als eine verteilte Datenbank. Jeder Teilnehmer verfügt dabei über seine eigene Kopie der Daten, wodurch die Daten für alle transparent werden. Die Daten, z. B. Transaktionen, werden dabei in miteinander verketteten Blöcken abgelegt. Die Verkettung findet über kryptografische Hash-Werte statt (siehe auch Abschn. 6.1). Der Hash-Wert eines vorherigen Blocks geht in die Daten des nachfolgenden Blocks ein, wodurch die nachträgliche Manipulation der Daten unmöglich wird. Um neue Blöcke mit Nutzdaten an die Blockchain anzuhängen, muss ein verteilter Konsensmechanismus existieren. Dabei einigen sich alle Beteiligten darauf, welche Daten in einen Block aufgenommen werden sollen. Der bekannteste Mechanismus wird als **Proof-of-Work** bezeichnet, der z. B. bei

der digitalen Währung Bitcoin eingesetzt wird. Dieses Verfahren ist sehr rechenintensiv, es kann aber durch andere Verfahren, wie z. B. **Proof-of-Stake** ersetzt werden.

Eine Blockchain kann prinzipiell für alle Prozesse genutzt werden, in denen einer zentralen Instanz Vertrauen für die Integritätsbestätigung einer Transaktion entgegengebracht werden muss. Beispiele dafür gibt es in der Energiewirtschaft ausreichend, z. B. die Abrechnung von Energieflüssen (siehe [47]). Dass die Blockchain-Technologie für die Energiewirtschaft Relevanz hat, zeigt sich auch an der sog. Blockchain-Initiative, die unter der Leitung des Bundesverbands Energiemarkt & Kommunikation e. V. (edna) bearbeitet wird. Aus dieser Blockchain-Initiative ist z. B. das Projekt **MaKoChain** hervorgegangen (siehe [58]). Darin wurde eine geschlossene Ethereum-Blockchain genutzt, um eine zentrale Plattform für die Marktkommunikation zu realisieren. Das Projekt ist aktuell im Status einer Machbarkeitsstudie.

Ein wichtiger Teil des Bilanzkreismanagements ist das **Fahrplan-** bzw. **Nominierungsmanagement**. Jeder Bilanzkreisverantwortliche muss für seine Bilanzkreise täglich bis spätestens 14:30 Uhr Netzfahrpläne an die Übertragungsnetzbetreiber versenden, die diese dann über alle Bilanzkreise prüfen. Am Ende muss die Summe aller Ein- und Ausspeisungen über alle Fahrpläne der Bilanzkreise „0“ ergeben. Für Fahrpläne existieren verschiedene Geschäftsvorfälle, für die jeweils täglich getrennte Fahrpläne erstellt und versendet werden müssen (siehe [48]):

- Mittels einer **1:1-Nominierung** werden Handelsgeschäfte zwischen Bilanzkreisen angemeldet, die in unterschiedlichen Regelzonen beheimatet sind. Dabei darf nur ein bestimmter, vorher benannter Bilanzkreis in der anderen Regelzone beliefert werden.
- Die **1:N-Nominierung** wird dazu genutzt, Handelsgeschäfte zwischen Bilanzkreisen abzuwickeln, die keine 1:1-Nominierung benötigen (meist regelzoneninterne Handelsgeschäfte).
- Durch die **Erzeugungsprognose** wird dem ÜNB die Summenzeitreihe aller Einspeisungen mitgeteilt.
- Mithilfe der **Verbrauchsprognose** wird die Summe aller Verbräuche innerhalb eines Bilanzkreises angemeldet.

Die Notwendigkeit, solche Fahrpläne versenden zu müssen, existiert bereits länger, als die Marktprozesse MABIS und GPKE definiert sind. Wegen dieser Historie haben sich hier eigene Prozesse und Datenformate etabliert. Da Handelsgeschäfte auch regelzonen- und länderübergreifend stattfinden können, müssen diese Prozesse und Datenformate auch paneuropäisch genormt sein. In Europa werden die Prozesse und Datenformate zum Fahrplanmanagement durch entso-e festgelegt. Dabei können heute zwei unterschiedliche Informations- und Datenmodelle genutzt werden, das entso-e Scheduling System (ESS) und CIM. Aus historischen Gründen wird von den meisten Bilanzkreisverantwortlichen aktuell ESS eingesetzt. Abb. 5.27 zeigt einen Ausschnitt des von entso-e definierten Informationsmodells eines Fahrplans.

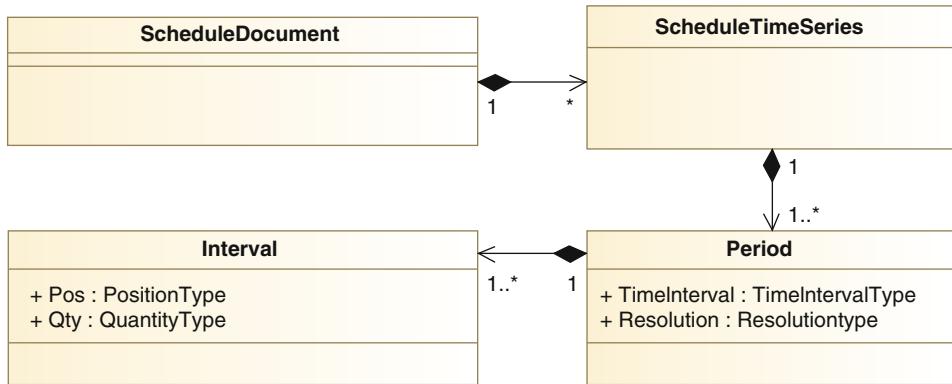


Abb. 5.27 Informationsmodell eines ESS-Fahrplans als Klassendiagramm der UML

Ein Fahrplan auf Basis von **ESS** besteht aus mindestens einer Zeitreihe als Objekt der Klasse **ScheduleTimeSeries**, die wiederum aus mehreren Perioden (Klasse **Period**) bestehen kann. Zeitreihenwerte werden durch Objekte der Klasse **Interval** abgebildet. Für die Übertragung eines Fahrplans müssen die Objekte dieses Informationsmodells in ein spezielles XML-Format serialisiert werden. Die ESS-Spezifikation definiert dazu eine entsprechende Document Type Definition (DTD). Das folgende (gekürzte) XML-Dokument realisiert einen solchen ESS-Fahrplan, der einige Zeitreihenwerte übermittelt.

```

1 <ScheduleDocument>
2   <ScheduleTimeSeries>
3     <Period>
4       <Interval v="2001-06-02T22:00Z/2001-06-02T23:00Z">
5         <Resolution v="PT15M"/>
6         <Interval>
7           <Pos v="1"/>
8           <Qty v="45"/>
9         </Interval>
10        <Interval>
11          <Pos v="2"/>
12          <Qty v="40"/>
13        </Interval>
14        <Interval>
15          <Pos v="3"/>
16          <Qty v="45"/>
17        </Interval>
18        <Interval>
19          <Pos v="4"/>
20          <Qty v="45"/>
21        </Interval>
22      </Period>
23    </ScheduleTimeSeries>
24  </ScheduleMessage>
  
```

ESS-Dateien werden meist per SMTP an die Übertragungsnetzbetreiber versendet. Dabei muss wie bei den EDIFACT-Nachrichten der Betreff der Nachricht mit dem Dateinamen der ESS-Datei übereinstimmen. Mitunter ist eine Übertragung der ESS-Dateien auch per File Transfer Protocol (FTP) möglich, wofür die Übertragungsnetzbetreiber eigene Server bereitstellen. Im Gegensatz zu SMTP hat FTP dabei den Vorteil, dass die erfolgreiche Übertragung der Nachricht direkt transparent wird. Der Empfang der Fahrpläne wird über spezielle ESS-Nachrichten bestätigt. Ein Beispiel sind positive oder negative Acknowledge-Nachrichten, welche den Empfang und die formale Korrektheit der Nachrichten signalisieren. Im weiteren Verlauf erhält der BKV dann vom BIKO entweder *Anomaly Reports* oder *Confirmation Reports*. Ein Anomaly Report deutet auf eine Abweichung der Fahrplananmeldungen zwischen zwei Bilanzkreisen hin und muss durch die Abstimmung unter den betroffenen BKVn behoben werden. Der Confirmation Report hingegen zeigt an, dass die übermittelten Fahrpläne insgesamt in Ordnung sind.

Wie für die Marktkommunikation mit EDIFACT existieren auch für ESS nur wenige freie Bibliotheken und Werkzeuge. Wahrscheinlich ist das Thema zu speziell, da insbesondere die Vorgaben von Edi@Energy ausschließlich den deutschen Markt betreffen. Mit dem **Communication Client for Traders (ComCT)** findet sich für den Fahrplanversand ein Werkzeug der Firma SOPTIM AG. ComCT ist ausschließlich für Windows verfügbar und kann standardkonforme ESS-Nachrichten erstellen und versenden. ComCT ist bis zur Version 1.3.1 frei nutzbar, da die Entwicklung von den deutschen Übertragungsnetzbetreibern übernommen wurde, und kann von den Webseiten der SOPTIM AG kostenlos heruntergeladen werden (siehe [69]). Abb. 5.28 zeigt ein Bildschirmfoto von ComCT.

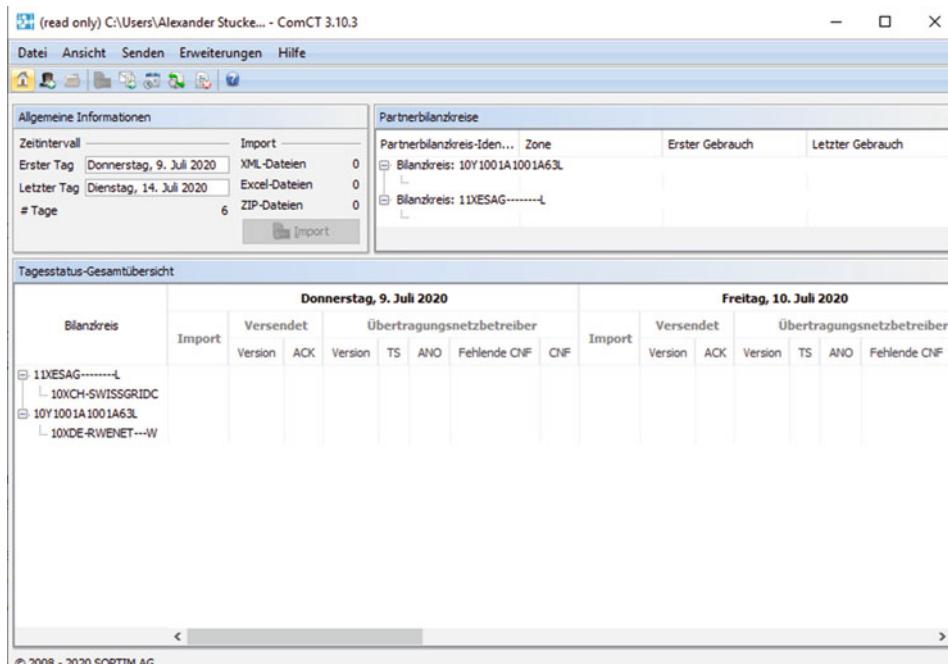


Abb. 5.28 Bildschirmfoto von ComCT

Unternehmen der Energiebranche sind an vielen weiteren Stellen durch Gesetze oder Regularien daran gebunden, Informationen zu übermitteln. Für diese Kommunikationspflichten existieren dann wiederum andere Prozesse, Informations- und Datenmodelle sowie Übertragungswege. Beispiele dafür sind:

- Veröffentlichungspflichten auf Basis von §17 StromNVZ,
- Regulation on Wholesale Energy Market Integrity and Transparency: REMIT,
- Novelle der EU-Finanzmarktrichtlinie: MiFID II/MiFIR,
- Herkunftsachweisregister nach §42 EnWg.

Wir werden an dieser Stelle nicht weiter auf diese Anwendungsfälle eingehen.

5.5 Zusammenfassung

In diesem Kapitel wurden Verfahren behandelt, die dazu dienen, Daten in der Energiewirtschaft über Kommunikationsnetze hinweg auszutauschen. Wie wir gesehen haben, existiert dazu eine ganze Reihe von Anwendungsfällen.

Ein Beispiel sind die Überwachung und der Betrieb von Energieanlagen, die eine Kernaufgabe der Energieversorgung ausmachen. Dabei müssen Komponenten der Feld-, Automations- und Managementebene teilweise in Echtzeit miteinander kommunizieren können. Die Energieinfrastruktur ist dabei meist über große Flächen verteilt, warum der Aspekt der digitalen Kommunikation zwischen Anlagen und den Führungssystemen besonders wichtig ist. Kommunikationsstandards, wie DNP3 oder IEC 60870-5, bilden diese Anforderungen ab. In den letzten Jahren hat sich allerdings der Fokus der eingesetzten Kommunikationsstandards verschoben. Das Problem der bandbreitenschonenden und verlässlichen Datenübertragung kann heute als gelöst angesehen werden. In den Vordergrund rücken aber zusehends die Semantik der Daten und die herstellerunabhängige Interoperabilität der eingesetzten Komponenten. Standards, wie z. B. der IEC 68150, bilden diese geänderten Anforderungen ab, indem objektorientierte Informationsmodelle eingeführt werden, an die auch in Zukunft neuere Kommunikationsprotokolle angebunden werden können.

Die Anlagenautomation spielt heute aber nicht mehr ausschließlich nur im großindustriellen Kontext von Kraftwerken und Schaltanlagen eine wichtige Rolle. Zunehmend werden Automationssysteme auch in Wohngebäuden und Zweckbauten eingesetzt, um z. B. den Energieverbrauch zu reduzieren oder den Komfort und die Betriebssicherheit zu erhöhen. Auch in diesem Kontext bedarf es Kommunikationssysteme und Standards, um entsprechende Komponenten abzubilden und ihre Bewegungs- und Metadaten über ein Kommunikationsnetz übertragen zu können. Entsprechend wurden in diesem Kapitel Standards, wie z. B. KNX, LonWorks, BACnet und OPC UA vorgestellt.

Ein weiterer Anwendungsfall für die Datenübertragung in der Energiewirtschaft ergibt sich aus dem Wunsch, Last und Erzeugung möglichst bereits auf kleiner Ebene

miteinander in Einklang zu bringen. Dazu müssen aber Last- und Netzzustandsdaten bei sehr vielen Endverbrauchern möglichst in Echtzeit automatisiert erfasst und übertragen werden können. Wie wir gesehen haben, dient dabei der sog. Smart Meter Gateway als zentrale Datendrehscheibe. Ein Smart Meter Gateway stellt mehrere, voneinander unabhängige Kommunikationsbereiche zur Verfügung, in denen unterschiedliche Übertragungsmedien, Protokolle und Sicherheitsvorgaben genutzt werden. Über einen speziellen Kommunikationsbereich, dem Home Area Network, kann der Smart Meter Gateway externen Marktteilnehmern sogar den Zugriff auf regelbare Erzeugungseinheiten oder Lasten gewähren.

Der letzte Abschnitt dieses Kapitels befasste sich mit der sog. Marktkommunikation. Unter diesen Begriff fallen dabei alle Kommunikationspflichten der Energieunternehmen, die durch die Umsetzung der Marktprozesse, wie z. B. GPKE oder MABIS, entstehen. Dabei tauschen die Marktteilnehmer konform ihrer jeweiligen Markttrolle digitale Informationen unternehmensübergreifend aus. Die Abwicklung dieser Prozesse ist jeweils national geregelt, einheitliche europäische Standards existieren dazu bislang nicht. Wie wir gesehen haben, werden in Deutschland für den Informationsaustausch zwischen Marktteilnehmern überwiegend sog. EDIFACT-Nachrichten genutzt, die im Anhang von E-Mails zwischen den Unternehmen ausgetauscht werden. Für das Nominierungsmanagement zur Anmeldung von Energieflüssen zwischen Bilanzkreisen oder Regelzonen existiert mit dem entso-e Scheduling System ein weiteres Datenmodell, um Netzfahrpläne darzustellen. In der Praxis werden diese Daten auf Basis von XML-Dateien ebenfalls meist als Anhang einer E-Mail an die Übertragungsnetzbetreiber versandt.

5.6 Kontrollfragen

- Welche Aufgaben besitzen Schaltanlagen und auf welchen Spannungsebenen sind diese Assets zu finden?
- Welche Betriebsmittel sind in Schaltanlagen zu finden?
- Wofür stehen die Abkürzungen RTU und IED? Was ist die Aufgabe dieser Elemente?
- Beschreiben Sie die Kommunikationsarchitektur in und mit Schaltanlagen.
- Wo in der Kommunikationsarchitektur wird der Standard DNP3 eingesetzt?
- Beschreiben Sie das Informationsmodell von DNP3.
- Welche Mechanismen hält DNP3 vor, um auf einem Bus Nachrichtenkollisionen zu erkennen bzw. zu beheben?
- Welche Kommunikationsmethode wird bei DNP3 mit der Abkürzung RBE bezeichnet?
- Wie wird das Nachrichtenformat von IEC 60870-5 bezeichnet?
- Was wird im Nachrichtenkopf des IEC 60870-5 durch die *Cause of Transmission* definiert?
- Was sind Gründe, dass mit dem IEC 61850 ein neuer Kommunikationsstandard im Kontext der Netzautomation entstanden ist?

- Mithilfe welcher Typen werden Geräte im Informationsmodell des IEC 61850 aufgebaut?
- Wozu dienen die sog. ACSI des IEC 61850? Nennen Sie Beispiele für Funktionen.
- Mithilfe welches Kommunikationsprotokolls wird die client/server-basierte Kommunikation des IEC 61850 meist umgesetzt?
- Welche Gründe kann es für die Gebäudeautomation geben?
- Was ist der Unterschied zwischen Raum- und Gebäudeautomation?
- Welche Übertragungsmedien werden durch KNX derzeit unterstützt? Welches wird am häufigsten eingesetzt?
- Welches Buszugriffsverfahren nutzt KNX TP? Was bedeutet es, wenn von dem „dominierenden Bit“ gesprochen wird? Wie wird diese Systematik bei Bestätigungstelegrammen genutzt?
- Was sind Geräte-, was Gruppenadressen in KNX? Welcher logische Aufbau wird bei beiden Adressarten meist gewählt? Wozu dient die Gruppenadresse hauptsächlich?
- Können bei KNX auch Ringtopologien aufgebaut werden?
- Wie wird das Informationsmodell von KNX bezeichnet?
- Warum kann KNX eher als Standard für die Raumautomation gesehen werden?
- Wie heißt das Kommunikationsprotokoll des LonWorks-Systems.
- Aus welchen Hardwarekomponenten besteht ein LonWorks-Gerät meist?
- Wie heißt die Medienzugriffssteuerung des LonWorks-Systems und wie funktioniert sie?
- Welche Adressarten existieren im LonWorks-System und welche Kommunikationsarten werden damit realisiert?
- Wozu dienen die sog. Netzwerkvariablen im LonWorks-System?
- Welche Aufgabe erfüllt die LonMark-Organisation?
- Welche Idee verbindet BACnet mit dem IEC 61850?
- Wozu müssen bei der Nutzung von BACnet über IP-Netze in jedem Subnetz spezielle Geräte betrieben werden? Wie heißen diese Geräte?
- Verwendet BACnet für jedes Übertragungsmedium dasselbe Adressschema? Wie wird ein BACnet-Gerät im gesamten Netz eindeutig identifiziert?
- Was ist das wichtigste Element des Informationsmodells von BACnet? Was repräsentiert es?
- Wozu dient das sog. Device Object von BACnet?
- Welche Verfahren stellt BACnet für die ereignisbasiert Meldung von Zustandsänderungen zur Verfügung?
- Was ist der Unterschied zwischen Intrinsic Reporting und Algorithmic Reporting?
- Nennen Sie Beispiele für Eventalgorithmen von BACnet.
- Welche Möglichkeiten existieren bei BACnet, externe Systeme anzubinden? Existieren solche Möglichkeiten auch bei KNX oder LonWorks?
- Über welche Mechanismen wird die Kompatibilität von BACnet-Geräten unterschiedlicher Hersteller gewährleistet? Existieren ähnliche Mechanismen auch bei KNX oder LonWorks? Wie sehen sie aus?

- Was sind die wichtigsten Elemente des Metamodells des OPC UA?
- Können im Informationsmodell des OPC UA ausschließlich Geräte der Gebäudeautomation abgebildet werden?
- Welche Übertragungsverfahren werden durch OPC UA unterstützt?
- Warum ist eine flächendeckende und zeitnahe Erhebung von Last- und Netzqualitätsdaten wünschenswert?
- Was wird als intelligentes Messsystem bezeichnet? In welchem Gesetz wird dieser Begriff eingeführt?
- Bis wann gilt in Deutschland eine Einbaupflicht für Smart Meter?
- Welche Kommunikationsbereiche besitzt ein Smart Meter Gateway? Welche Aufgaben fallen in jedem Bereich an?
- Was wird durch COSEM spezifiziert? Wie ist COSEM aufgebaut?
- Was ist eine OBIS-Kennzahl? Wie ist sie aufgebaut?
- Was wird bei einer OBIS-Kennzahl durch die erste Wertgruppe definiert und wofür steht dort die Ziffer 1?
- Wozu dient die sog. CLS-Schnittstelle eines Smart Meter Gateway?
- Was legen die sog. Tarifanwendungsfälle beim Smart Meter Gateway fest? Was macht der sog. datensparsame Tarif?
- Wie überträgt der Smart Meter Gateway seine Daten technisch an den Messstellenbetreiber? Wie werden die Daten codiert?
- Welche Übertragungsmedien können für die Datenübertragung zwischen Smart Meter Gateway und Messstellenbetreiber genutzt werden?
- Ist die Marktkommunikation europäisch einheitlich standardisiert?
- Welche Prozessgruppen wurden im Rahmen der Marktkommunikation in Deutschland bislang spezifiziert?
- Welches Datenformat wird für die Datenübertragung im Rahmen der Marktprozesse in Deutschland eingesetzt? Wie werden diese Nachrichten übertragen?
- Welche Art von Daten werden durch MSCONS-Nachrichten übermittelt?
- Wäre es auch möglich, ein blockchain-basiertes System für die Organisation der Marktkommunikation einzusetzen? Gibt es hier Beispiele?
- Ist das Fahrplanmanagement europaweit standardisiert?
- Welches Datenformat wird für die Übertragung von Netzfahrplänen genutzt? Worauf basiert es?

Literatur

1. Gordon Clarke. *Practical Modern SCADA Protocols: DNP3, 60870.5 and Related Systems*. 1. Aufl, Newnes, 20. Jan. 2004. 544 S.
2. Tatjana Kostic, Otto Preiss und Christian Frei. „Understanding and using the IEC 61850: a case for meta-modelling“. In: *Computer Standards & Interfaces* 27 (Juni 2005), S. 679–695. issn: 09205489. <https://doi.org/10.1016/j.csi.2004.09.008>. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0920548904001023> (besucht am 30.06.2019).

3. Marek Miśkowicz. „Prediction Efficiency in Predictive p-CSMA/CD“. In: *Emerging Directions in Embedded and Ubiquitous Computing*. Hrsg. von Xiaobo Zhou u. a. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2006, S. 203–212. ISBN: 978-3-540-36851-9. https://doi.org/10.1007/11807964_21.
4. Boris Malinowsky, Georg Neugschwandtner und Wolfgang Kastner. „Calimero: Next generation“. In: *Proceedings konnex scientific conference*. Dusiburg, 2007.
5. *Introduction to the LonWorks Platform*. Revision 2. Echelon Corporation, 2009. URL: https://www.echelon.com/assets/blt893a8b319e8ec8c7/078-0183-01B_Intro_to_LonWorks_Rev_2.pdf (besucht am 04.03.2020).
6. Wolfgang Mahnke, Stefan-Helmut Leitner und Matthias Damm. *OPC Unified Architecture*. 2009. Aufl. Springer, 5. Apr. 2009. 351 S.
7. S. Mohagheghi, J. Stoupis und Z. Wang. „Communication protocols and networks for power systems-current status and future trends“. In: *2009 IEEE/PES Power Systems Conference and Exposition*. 2009 IEEE/PES Power Systems Conference and Exposition. März 2009, S. 1–9. <https://doi.org/10.1109/PSCE.2009.4840174>.
8. Klaus-Peter Brandt und Wolfgang Wimmer. „The concept of IEC 61850“. In: *Special Report IEC 61850*. ABB Review. ABB Group R&D and Technology, Aug. 2010, S. 7–12. URL: https://library.e.abb.com/public/a56430e1e7c06fdfc12577a00043ab8b/3BSE063756_en_ABB_Review_Special_Report_IEC_61850.pdf (besucht am 29.08.2019).
9. Sebastian Rohjans, Mathias Uslar und Hans-Jürgen Appelrath. „OPC UA and CIM: Semantics for the smart grid“. In: *ResearchGate*. Transmission and Distribution Conference and Exposition, IEEE PES. New Orleans, LA, USA, 19. Apr. 2010. <https://doi.org/10.1109/TDC.2010.5484299>. URL: https://www.researchgate.net/publication/224145454_OP_C_UA_and_CIM_Semantics_for_the_smart_grid (besucht am 22.04.2020).
10. Martin Becker und Peter Knoll. *Energieeffizienz durch Gebäudeautomation mit Bezug zur DIN V 18599 und DIN EN 15232*. Hochschule Biberach (HBC), 2011. URL: https://www.zvei.org/fileadmin/user_upload/Verband/Fachverbaende/Elektroinstallationssysteme/Studie_Energieeffizienz_durch_Gebaeudeautomation/Kurzfassung-ZVEI-Studie-Energieeffizienz-durch-Gebaeudeautomation.pdf.
11. Bernhard Isler. *BACnet Alarming Revised*. Nov. 2011. URL: <http://www.bacnet.org/Bibliography/BACnet-Today-11/Isler-2011.pdf> (besucht am 27.03.2020).
12. D. Trinchero u. a. „Integration of smart house sensors into a fully networked (web) environment“. In: *SENSORS, 2011 IEEE*. 2011, S. 1624–1627.
13. Mathias Uslar u. a. „IEC 61850 based OPC UA Communication“. In: *Proceedings of the 17th International Power Systems Computation Conference PSCC*. Stockholm, 2011.
14. Hans-Jürgen Appelrath, Hrsg. *Future Energy Grid*. Acatech-Studie. Berlin u. a.: Springer, 2012. 323 S. ISBN: 978-3-642-27863-1.
15. *DIN EN ISO 16484-5:2012-11 – Systeme der Gebäudeautomation – Teil 5: Datenkommunikationsprotokoll*. DIN – Deutsches Institut für Normung, 1. Nov. 2012.
16. Olivier Hersent, David Boswarthick und Omar Eloumi. *The Internet of Things*. 2., Auflage. New York, NY: Wiley, J, 2012. 376 S. ISBN: 1-119-99435-7.
17. „IEEE Standard for Electric Power Systems Communications-Distributed Network Protocol (DNP3) – Redline“. In: *IEEE Std 1815-2012 (Revision of IEEE Std 1815-2010) – Redline* (Okt. 2012), S. 1–821.
18. Helmut Edelmann und Thomas Kästner. *Kosten-Nutzen-Analyse für einen flächendeckenden Einsatz intelligenter Zähler*. 2013. URL: https://www.bmwi.de/Redaktion/DE/Publikationen/Studien/kosten-nutzen-analyse-fuer-flaechendeckenden-einsatz-intelligenterzaehler.pdf?__blob=publicationFile&v=5 (besucht am 25.06.2019).

19. Jacob Gottwald. „Optimierungsmöglichkeiten in der Marktkommunikation hinsichtlich des Lieferantenwechselprozess auf dem deutschen Energiemarkt“. Bachelorarbeit. Mittweida: Hochschule Mittweida, 2013. URL: https://monami.hs-mittweida.de/frontdoor/deliver/index/docId/3854/file/Bachelorarbeit_Jacob_Gottwald.pdf.
20. Michael Newman. *BACnet: The Global Standard for Building Automation and Control Networks*. Momentum Press, 2013.
21. *Technische Richtlinie BSI TR-03109-1*. Bonn: Bundesamt für Sicherheit in der Informations-technik (BSI), 28. März 2013. URL: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR03109_1.pdf?__blob=publicationFile&v=1.
22. Mathias Uslar und Robert Bleiker. „Automation for the Smart Grid: IEC 61850 – Substation Automation and DER Communication“. In: Mathias Uslar u. a. *Standardization in Smart Grids*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, S. 115–128. https://doi.org/10.1007/978-3-642-34916-4_7 (besucht am 21.08.2019).
23. Wolfgang Kastner u. a. „Building automation systems integration into the Internet of Things the IoT6 approach, its realization and validation“. In: *Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA)*. Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA). Sep. 2014, S. 1–9. <https://doi.org/10.1109/ETFA.2014.7005197>.
24. *Open Metering System Specification – Volume 1 – General Part*. 2.0.1. OMS Group, Okt. 2014. URL: https://oms-group.org/fileadmin/files/download4all/specification/Vol2/4.1.2/OMS-Spec_Vol1_General_v201_RELEASE.pdf (besucht am 25.06.2019).
25. *Open Metering System Specification – Volume 2 – Primary Communication*. 4.0.2. OMS Group, 27. Jan. 2014. URL: https://omsgroup.org/fileadmin/files/download4all/specification/Vol2/4.02/OMS-Spec_Vol2_Primary_v402.pdf (besucht am 03.06.2020).
26. Lin Zhu, Dongyuan Shi und Pengyuan Wang. „IEC 61850-Based Information Model and Configuration Description of Communication Network in Substation Automation“. In: *IEEE Transactions on Power Delivery* 29.1 (2014), S. 97–107. issn: 0885-8977. <https://doi.org/10.1109/TPWRD.2013.2269770>.
27. Rainer Bermbach. *Kommunikation im Local Metrological Network eines Smart Meter Gateway*. Forschungsbericht. Ostfalia Hochschule für angewandte Wissenschaften, 2015. URL: https://www.ostfalia.de/cms/de/pws/bermbach/.content/documents/Forschungsbericht-WF-4_WS15_Be.pdf (besucht am 28.02.2020).
28. *Deutsche Normungsroadmap Smart Home + Building Version 2.0*. Frankfurt: DKE - Deutsche Kommission Elektrotechnik Elektronik Informationstechnik, Aug. 2015. URL: <https://www.dke.de/resource/blob/778214/6ec4d037024b61a63d14544d181c638a/deutsche-normungs-roadmap-smart-home---building--version-2-0-data.pdf> (besucht am 09.06.2020).
29. Michael Hoefling u. a. „JOSEF: A Java-Based Open-Source Smart Meter Gateway Experimentation Framework Germany“. In: Energy Informatics – 4th D-A-CH Conference, EI 2015, Karlsruhe, Germany, November 12–13, 2015, Proceedings. Hrsg. von Sebastian Gottwalt, Lukas König und Hartmut Schmeck. Lecture Notes in Computer Science. Springer, 2015, S. 165–176. ISBN: 978-3-319-25875-1. URL: <https://doi.org/10.1007/978-3-319-25876-8%5C%14>
30. Bernd Holfeld u. a. „Smart Grid Communications: LTE Outdoor Field Trials at 450 MHz“. In: *2015 IEEE 81st Vehicular Technology Conference (VTC Spring)*. 2015 IEEE 81st Vehicular Technology Conference (VTC Spring). Glasgow, United Kingdom: IEEE, Mai 2015, S. 1–5. ISBN: 978-1-4799-8088-8. <https://doi.org/10.1109/VTCSpring.2015.7146091>. URL: <http://ieeexplore.ieee.org/document/7146091/> (besucht am 07. 11. 2019).
31. Karlheinz Schwarz. „The Standard Message Specification for Industrial Automation Systems ISO 9506 (MMS)“. In: *Industrial Communication Technology Handbook*. Second Edition. CRC Press, 2015.

32. Christoph Stampfel. „Building Automation Systems and RESTful BACnet/WS“. Bachelorarbeit. Wien: Technische Universität Wien, 21. Apr. 2015. URL: https://www.auto.tuwien.ac.at/bib/pdf_TR/TR0170.pdf (besucht am 31.03.2020).
33. *BACnet Data Communication Protocol for Building Automation and Control Networks - BSR/ASHRAE Addendum am to ANSI/ASHRAE Standard 135-2012*. ASHRAE, 2016. URL: <http://www.bacnet.org/Addenda/Add-135-2012am.pdf>.
34. Elmar Böllin u. a. *Regenerative Energien im Gebäude nutzen: Wärme- und Kälteversorgung, Automation, Ausgeführte Beispiele*. 2., überarb. Aufl. 2016. Wiesbaden: Springer Vieweg, 1. Apr. 2016. 296 S. ISBN: 978-3-658-12404-5.
35. *IEC TR 62541-1:2016 – OPC unified architecture – Part 1: Overview and concepts* International Electrotechnical Commission (IEC), Okt. 2016. URL: <https://www.vde-verlag.de/iecnormen/223905/iec-tr-62541-1-2016.html> (besucht am 07.05.2020).
36. Hijo Joy u. a. „Integration of Arduino as a slave system to LonWorks based System using I2C Interface“. In: International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering 5.7 (Juli 2016). issn: 2278 – 8875. URL: https://www.ijareee.com/upload/2016/july/36_Integration.pdf (besucht am 05.03.2020).
37. *OPC UA Information Model for BACnet*. 0.17PR. BACnet Interest Group Europe and OPC Foundation, 10. Mai 2016. URL: https://www.big-eu.org/fileadmin/downloads/BACnet_OPCT_UA_Information_Model.pdf.
38. *Electricity metering data exchange – The DLMS/COSEM suite – Part 5–3: DLMS/COSEM application layer*. International Electrotechnical Commission (IEC), 2017.
39. Stefan Feuerhahn. „Analysis and evaluation of the IEC 61850 communication standard for monitoring and control of distributed energy resources“. Dissertation. Aachen: Shaker Verlag, 2017.
40. Hans R. Kranz. *BACnet Gebäudeautomation 1.12: Grundlagen in deutscher Sprache*. Update auf 1.19. Karlsruhe: cci Dialog, 2017. 616 S. ISBN: 978-3-922420-25-5.
41. Petr Matoušek. „Analysis of DLMS Protocol“. Brno University of Technology, Dez. 2017. URL: <http://www.fit.vutbr.cz/research/pubs/tr.php.en?file=%2Fpub%2F11616%2FTR-DLMS.pdf&id=11616>.
42. *OPC 10000-3 – Part 3: Address Space Model*. Release 1.04. OPC Foundation, 22. Nov. 2017. URL: <https://opcfoundation.org/developer-tools/specifications-unified-architecture/part-3-address-space-model/> (besucht am 07.05.2020).
43. *OPC 10000-4 – Part 4: Services*. Release 1.04. OPC Foundation, 22. Nov. 2017. URL: <https://opcfoundation.org/developer-tools/specifications-unified-architecture/part-4-services/> (besucht am 07.05.2020).
44. *OPC 10000-6 – Part 6: Mappings*. Release 1.04. OPC Foundation, 22. Nov. 2017. URL: <https://opcfoundation.org/developer-tools/specifications-unified-architecture/part-6-mappings/> (besucht am 07.05.2020).
45. Adolf J. Schwab. *Elektroenergiesysteme*. 5th ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2017. 11120 S. ISBN: 978-3-662-55315-2. URL: <https://ebookcentral.proquest.com/lib/gbv/detail.action?docID=5041567>.
46. *DIN EN 13757-3:2018-06 – Kommunikationssysteme für Zähler – Teil 3: Anwendungsprotokolle* DIN – Deutsches Institut für Normung, Juni 2018. URL: <https://www.beuthde/de/norm/din-en-13757-3/276734223> (besucht am 03.06.2020).
47. Malte Stomberg, Alexander Stuckenholz und Torsten Cziesla. „SolChain“. In: *BWK – Das Energiefachmagazin* 11 (2018), S. 48–51.
48. *Prozessbeschreibung Fahrplanabwicklung in Deutschland*. 27. Feb. 2018. URL: https://www.netztransparenz.de/portals/1/Content/EU-Network-Codes/EB-Verordnung/20180227_Prozessbeschreibung_Fahrplanabwicklung_%20in_Deutschland_Version_4.0.pdf.

49. *RAMI 4.0 – Ein Orientierungsrahmen für die Digitalisierung*. Plattform Industrie 4.0, 22. Nov. 2018. URL: <https://www.plattform-i40.de/PI40/Redaktion/DE/Downloads/Publikation/rami40-einfuehrung-2018.html> (besucht am 07.04.2020).
50. *450 MHz Frequenznutzung für die Energiewirtschaft*. VDE Verband der Elektrotechnik Elektronik Informationstechnik e.V., 2. Okt. 2019. URL: <https://www.vde.com/de/presse/vde-fordert-450-mhz-frequenz-fuer-energiewirtschaft>.
51. Stefanie Dierks. *Verpflichtender Smart-Meter Rollout beginnt im Januar*. 18. Dez. 2019. URL: <https://www.energate-messenger.de/news/198422/verpflichtender-smart-meter-rollout-beginnt-im-januar> (besucht am 02.01.2020).
52. *DIN EN 13757-4:2019-09 – Communication systems for meters – Part 4: Wireless M-Bus communication*. DIN – Deutsches Institut für Normung, Sep. 2019. URL: <https://www.beuth.de/en/standard/din-en-13757-4/306311086> (besucht am 03.06.2020).
53. *DNP3 (IEEE 1815) protocol stack. Modern C++ with bindings for .NET and Java*. 2019. URL: <https://github.com/dnp3/opendnp3>.
54. *Gebäude Report Kompakt – Statistiken und Analysen zur Energieeffizienz im Gebäudebestand*. Deutsche Energie-Agentur (dena), 2019. URL: https://www.dena.de/fileadmin/dena/Publikationen/PDFs/2019/dena-GEBAEUDEREPORT_KOMPAKT_2019.pdf (besucht am 12.09.2019).
55. *Hinweise für Planung, Ausführung und Betrieb der Gebäudeautomation in öffentlichen Gebäuden*. 145. Berlin: Bundesministerium des Innern, für Bau und Heimat, 2019. URL: https://www.amev-online.de/AMEVInhalt/Planen/Gebaeudeautomation/GA%202019/AMEV_GA2019_2019-03-29.pdf (besucht am 27.01.2020).
56. Matthias Kleine. *KNX in Eigenregie mit dem Arduino*. 19. Mai 2019. URL: <https://hausautomatisierung.com/hardware/knx/2019/05/19/knx-undarduino-selbstbau.html> (besucht am 07.05.2020).
57. *OpenMUC – Software Solutions for Monitoring and Control Systems*. 2019. URL: <https://www.openmuc.org/>. (besucht am 20.02.2020).
58. Uwe Pagel. *MaKoChain: Lieferantenwechsel in Sekunden*. 2019. URL: <https://edna-bundesverband.de/news/makochain-lieferantenwechsel-in-sekunden/> (besucht am 25.06.2019).
59. Yubo Yuan und Yi Yang. *IEC 61850-Based Smart Substations: Principles, Testing, Operation and Maintenance*. 1. Aufl. Academic Press, 12. Juni 2019. 438 S.
60. *Calimero-Server* URL: <https://github.com/calimero-project/calimero-server> (besucht am 14.07.2020).
61. *EIC Approved Codes*. URL: <https://www.entsoe.eu/data/energy-identification-codes-eic/eic-approved-codes/> (besucht am 14.07.2020).
62. *Energie Codes & Services GmbH*. URL: <https://www.energicodes-services.de/> (besucht am 28.02.2020).
63. *Forum Datenformate des Edi@Energy*. URL: <https://www.edi-energy.de/> (besucht am 28.02.2020).
64. *IEC 62056-61:2002-02 – Electricity metering – Data exchange for meter reading, tariff and load control – Part 61: Object identification system (OBIS)*.
65. *Informationsplattform der österreichischen Energiewirtschaft zur Veröffentlichung branchenspezifischer Datenaustauschformate sowie zu den von der Energiewirtschaft angewendeten Kommunikationsplattform Energiewirtschaftlicher Datenaustausch (EDA)*. URL: <https://ebutilities.at> (besucht am 28.02.2020).
66. *MZ Automation GmbH*. URL: <https://github.com/mz-automation> (besucht am 27.02.2020).
67. *OpenIEC61850* URL: <https://github.com/beanit/openiec61850> (besucht am 27.02.2020).
68. Josef Papenfort und Pascal Dresselhaus. „Genormte Kommunikation nach IEC 61850 für die Schaltanlagenautomatisierung“. In: *atp magazin*. Hrsg. von Jürgen Franke. Bd. 12/2008. Essen: Vulkan-Verlag GmbH, S. 28–31. URL: ftp://ftp.beckhoff.de/press/2008/german/atp_122008.pdf
69. *SOPTIM ComCT: Communication Client for Traders*. URL: <https://www.soptim.de/de/vertrieb-beschaffung-handel/soptim-connect/ess-konvertierung/> (besucht am 14.07.2020).



IT-Sicherheit

6

In der Energiewirtschaft dient die Informations- und Kommunikationstechnik (IKT) bekanntlich der Sicherstellung einer verlässlichen Energieversorgung. Fallen Systeme aus oder werden bewusst manipuliert, kann es im schlimmsten Fall zu einem Schwarzfall (*Blackout*) kommen. Ein großflächiger und längerfristiger Schwarzfall würde massive Funktions- und Versorgungsstörungen, wirtschaftliche Schäden und eine erhebliche Gefährdung der öffentlichen Sicherheit und Ordnung zur Folge haben (siehe [5]). Durch die weite Verbreitung der IKT in der Energiewirtschaft werden Cyberangriffe zu einer ernsten Bedrohungslage (siehe z. B. [12]). Bei den registrierten Cyberangriffen auf kritische Infrastrukturen in Deutschland liegt die Energieversorgung bereits auf Platz zwei (siehe [23]). IT-Sicherheit ist daher ein wichtiges Querschnittsthema, das alle Komponenten des Energiesystems betrifft.

Der folgende Abschnitt dient daher als ein erster Überblick über den Bereich der IT-Sicherheit im Kontext der Energiewirtschaft. Zunächst werden die Grundwerte dargestellt, die es bei der IT-Sicherheit i. d. R. einzuhalten gilt. Danach werden einige typische Gefahren beschrieben und mithilfe welcher Abwehrstrategien diese meist behandelt werden. Danach werden wir uns mit dem geltenden Rechtsrahmen für die deutsche Energiewirtschaft beschäftigen. Die in Deutschland geltenden Regularien nehmen dabei häufig Bezug auf die von der Smart Grid Security Group (SGIS) dargestellte Standardreihe IEC 270xx und den für die Energiewirtschaft speziellen Teil IEC 27019. Wie wir sehen werden, sind Energieversorgungsunternehmen gesetzlich dazu verpflichtet, ein Managementsystem zur Sicherstellung und kontinuierlichen Verbesserung ihrer Informationssicherheit aufzubauen.

Konkrete technische Maßnahmen zur Absicherung der Netzautomation werden für mehrere dort genutzte Kommunikationsstandards gemeinsam in der Standardreihe IEC 62351 behandelt. Darauf werden wir ebenfalls eingehen. Für die Gebäudeautomation

existiert kein solcher gemeinsamer Sicherheitsstandard. In den letzten Jahren wurden in vielen Systemen aber punktuelle Maßnahmen nachgerüstet, die wir uns in Bezug zu den Standards, die in diesem Lehrbuch behandelt wurden, ansehen wollen.

IT-Sicherheit spielt auch im Kontext intelligenter Messsysteme eine wichtige Rolle. Die Gründe und die technischen Maßnahmen, die vor allem aus den Sicherheitsrichtlinien des Bundesamt für Sicherheit in der Informationstechnik (BSI) für Smart Meter Gateways in Deutschland erwachsen, werden wir ebenfalls darstellen. Das Kapitel wird durch einige Hinweise abgerundet, welche die IT-Sicherheit für der Abwicklung der Marktkommunikation in Deutschland betreffen.

6.1 Grundwerte

Im Allgemeinen bedeutet Sicherheit die Abwesenheit von Gefahren. IT-Sicherheit ist daher die Abwesenheit von Gefahren, welche die ordnungsgemäße Nutzung der eingesetzten IKT-Komponenten stören. Die Informationssicherheit geht noch einen Schritt weiter. Sie inkludiert auch alle nichttechnischen Systeme, z. B. Geschäftsprozesse oder die Aufbauorganisation eines Unternehmens im Kontext der Informationsverarbeitung. Auch physische Angriffe auf die Infrastruktur bzw. das Erschleichen von Informationen mithilfe von *Social Engineering* werden dabei berücksichtigt.

Dieser Unterschied schlägt sich auch im Schichtenmodell des IT-Grundschutz-Kompendiums des BSI nieder (siehe [31]). Die Informationssicherheit wird durch das gesamte Modell abgedeckt. Die IT-Sicherheit findet sich dagegen eher in den Schichten *System-Bausteine* sowie in Teilen in der *Detektion und Reaktion*. IT- und Informationssicherheit teilen sich allerdings gemeinsame Grundwerte. Diese stellen Forderungen auf, um ein Mindestmaß an Sicherheit zu gewährleisten. Diese Grundwerte werden im Folgenden vorgestellt. Der Fokus dieses Kapitels liegt aber auf dem Schutz der IKT-Systeme, also der IT-Sicherheit.

Die **Verfügbarkeit** ist das erste Schutzziel zur Darstellung der Informationssicherheit. Es zielt darauf ab, dass eingesetzte Informations- und Kommunikationssysteme für befugte Nutzer zugänglich und verlässlich nutzbar sind. Die Funktionalität kann dann durch die Nutzer in ausreichender Qualität abgerufen werden.

Die geforderte Verfügbarkeit eines IKT-Systems wird i. d. R. in Prozent angegeben. Die Verfügbarkeit V steht dann für das Verhältnis zwischen tatsächlicher Nutzbarkeit eines IKT-Systems und der ursprünglich gewünschten Servicezeit.

$$V = \frac{\text{Servicezeit} - \text{Ausfallzeit}}{\text{Servicezeit}} \quad (6.1)$$

Die Verfügbarkeit von IKT-Systemen wird durch eine Vielzahl von sehr unterschiedlichen Gefahren bedroht (siehe [24]):

- **höhere Gewalt:** Blitzschlag, Brand oder Wasserschaden,
- **Fahrlässigkeit:** Fehlbedienung oder unsachgemäße Behandlung,
- **technisches Versagen:** Stromausfall, Verschleiß und Alterung, fehlerhafte Installationen oder mangelhafte Kompatibilität,
- **Vorsatz:** Manipulation, Einbruch, Hacking, Sabotage usw.

Beispiel 6.1.1 Ein IKT-System, welches durchgängig verfügbar sein muss, hat eine vereinbarte Servicezeit von 8760 Stunden pro Jahr. Bei einer geforderten Verfügbarkeit von 99,9 % darf dieses nur 8,76 Stunden, also 525 Minuten, pro Jahr nicht verfügbar sein. In diesem engen Zeitfenster müssen dann auch Wartungsmaßnahmen durchgeführt werden, währenddessen ein System auch meist nicht nutzbar ist.

Unter **Vertraulichkeit** wird der Schutz von IT-Systemen und Daten vor unberechtigtem Zugriff verstanden. Insbesondere wenn Daten über Kommunikationsnetze übertragen werden bzw. Anwendungsschnittstellen über solche Netze erreichbar sind, steigt die Gefahr des unberechtigten Zugriffs enorm.

Es ist leicht einsichtig, dass das Leitsystem eines Kraftwerks oder eines Energienetzes nicht über eine offene Webschnittstelle zugänglich sein sollte. Energieunternehmen verfügen aber auch über Betriebsgeheimnisse und verarbeiten im großen Maß personenbezogene Daten, wie z. B. individuelle Verbrauchsdaten. Entsprechend müssen wirksame Schutzmaßnahmen getroffen werden, um Systeme und Daten vor unberechtigtem Zugriff zu schützen. Die Vertraulichkeit von Daten ist dabei im Gegensatz zu den anderen Schutzz Zielen rechtlich vergleichsweise gut abgesichert (siehe [2]).

Die Vertraulichkeit von Daten und Systemen wird ebenfalls durch eine Reihe von Gefahren bedroht. Bei einem *Man-in-the-Middle*-Angriff fängt z. B. ein unbefugter Dritter Nachrichten auf einem offenen Kanal ab. Wörterbücher können dazu genutzt werden, um den Passwortschutz von Anwendungssystemen zu brechen. Selbst wenn bei der Datenübertragung die Nutzdaten selbst verschlüsselt sind, können allein die Metadaten bereits vertrauliche Informationen beinhalten, z. B. wer kommuniziert wann mit wem?

Ein weiteres, elementares Schutzz Ziel der Informationssicherheit ist die **Integrität**. Integrität nimmt Bezug auf die Korrektheit von Informationen und die Nachvollziehbarkeit von Änderungen. Daten sollen nicht unbemerkt verändert werden können, da den enthaltenen Informationen sonst kein Vertrauen entgegengebracht werden kann. Dies schließt auch die Funktionsweise der IKT-Systeme mit ein, die in der Lage sein müssen, die Daten korrekt und vollständig zu übertragen, zu speichern und zu verarbeiten (siehe [25, S. 5]). Die Integrität von Systemen und Daten wird durch eine Reihe von Gefahren bedroht, z. B. Programmfehler (*Bugs*), Fehler bei der Übertragung von Daten, Schadsoftware wie Viren oder Trojaner oder Angriffe wie SQL-Injection oder die absichtliche Wiederholung von Nachrichten. Ebenso vielfältig wie die Bedrohungen der Integrität können auch die Gegenmaßnahmen sein (siehe auch [26]). Dazu zählen Mechanismen, wie die sichere Programmierung und Qualitätssicherung der Programme, die Konsistenzprüfung von Daten an allen Schnittstellen, die Integritätsprüfung von Updates, die Nutzung von Kommunikations-Gateways mit Prüffunktionen und der Schutz vor Schadsoftware.

Eng verwandt mit Vertraulichkeit und Integrität ist die **Authentizität**. Die Authentizität verlangt die Verbindlichkeit und Zurechenbarkeit von Informationen. Eine Nachricht gilt als authentisch, wenn nachweisbar klar ist, dass sie von einem bestimmten Absender stammt. Authentizität ist in vielen Situationen von hoher Bedeutung. In der Energiewirtschaft werden viele Geschäftsprozesse digital abgewickelt, in denen große Werte bewegt werden, z. B. Marktprozesse oder Energiehandel. Dabei muss sichergestellt werden, dass z. B. eine Order klar einem bestimmten Marktteilnehmer zugeordnet werden kann. Auch der Zugang zu Softwaresystemen darf erst dann gestattet werden, wenn die Identität eines Nutzers zweifelsfrei nachgewiesen werden kann. Authentizität ist ohne Integrität nicht sicherzustellen. Können Daten auf dem Übertragungsweg manipuliert werden, kann eine Nachricht nicht authentisch sein.

Wie wir gesehen haben, wird die IT-Sicherheit durch eine Vielzahl von Gefahren bedroht. Die Grundwerte können dabei entweder unbewusst, z. B. durch technische Unzulänglichkeiten, oder durch bewusste Angriffe zum Zweck der Sabotage, Spionage, des Betrugs oder Diebstahls verletzt werden. Um die Gefahren kontinuierlich zu erkennen, zu überwachen und zu behandeln, wird die Einführung eines Risikomanagementprozesses empfohlen. Den in der Energiewirtschaft geltenden Rechtsrahmen und die in diesem Kontext wichtigen Standards sehen wir uns im nächsten Abschnitt an. Ebenso vielfältig wie die Bedrohungen sind auch die Gegenmaßnahmen. Im Folgenden sollen daher einige der typischen Maßnahmen dargestellt werden.

Eine Maßnahme, um die Verfügbarkeit von IKT-Systemen sicherstellen zu können, ist, diese entsprechend ausfallsicher zu konzipieren. Ein wichtiges Prinzip, welches dabei genutzt wird, ist die **Redundanz**. Durch ein redundantes Systemdesign kann im besten Fall der Ausfall oder das Fehlverhalten einzelner Komponenten identifiziert und kompensiert werden (siehe auch [12]). Fällt z. B. ein Server aus, kann ein zweiter seine Aufgabe im besten Fall so übernehmen, dass der Nutzer keine Einschränkungen erfährt. Kritische Funktionen sollten daher gezielt redundant und fehlertolerant ausgelegt und implementiert werden. Ein *Single-Point-of-Failure* sollte vermieden werden.

Redundanz kann technisch auf vielfältige Arten herbeigeführt werden, z. B. durch eine unterbrechungsfreie Stromversorgung, höhere Konnektivität durch mehrere Kommunikationspfade, Rechner im Clusterbetrieb oder eine Datensicherung. Auch sollten bewusst unterschiedliche Technologien, z. B. von unterschiedlichen Herstellern, parallel eingesetzt werden. Mitunter sind aber auch organisatorische Maßnahmen zu treffen, also ein zum normalen Betriebsablauf abweichendes Vorgehen, wie *Workarounds* oder Notfallprozesse.

Kryptografische Methoden spielen bei der Sicherstellung von Vertraulichkeit, Integrität und Authentizität eine besondere Rolle. Insbesondere wenn Daten über tendenziell unsichere Kanäle (z. B. das öffentliche Internet) übertragen werden, können Integrität und Vertraulichkeit nur sichergestellt werden, wenn Verschlüsselungsverfahren und digitale Signaturen eingesetzt werden.

Im Allgemeinen dienen Verschlüsselungsverfahren dazu, den Klartext einer Nachricht mithilfe eines Schlüssels in einen Geheimtext zu überführen. Ohne die Kenntnis des Schlüssels können unbeteiligte Dritte den Inhalt der Nachricht dann nicht mehr erfassen

oder ihn manipulieren. Heute werden meist öffentlich bekannte und standardisierte Verschlüsselungsverfahren eingesetzt. Die Sicherheit des Systems hängt dann einzig von der Wahl und der Geheimhaltung des Schlüssels ab und nicht von der Unkenntnis der Verfahren (sog. Kerckhoffs' Prinzip). Dabei gilt insbesondere die Länge des Schlüssels (in Anzahl von Bits) als Gradmesser für die Sicherheit. Je länger der Schlüssel, desto mehr Versuche muss ein Angreifer unternehmen, um ihn mithilfe eines sog. Brute-Force-Angriffs erraten zu können. Die Anwendung der Kryptografie zerfällt dabei in die Teilbereiche symmetrischer und asymmetrischer Chiffren, den kryptografischen Protokollen und den Hashfunktionen (siehe [16]).

Bei den **symmetrischen Chiffren** dient ein gemeinsamer Schlüssel sowohl für die Ver- als auch für die Entschlüsselung der Nachricht. Diese Verfahren sind leichter zu implementieren und effizient anzuwenden. Sie haben aber den Nachteil, dass der sichere Schlüsselaustausch zwischen Sender und Empfänger schwierig wird. Da Sender und Empfänger zudem im Besitz desselben Schlüssels sind, kann eine Nachricht nicht zweifelsfrei einer einzelnen Identität zugeordnet werden. Ein aktueller Vertreter eines symmetrischen Verschlüsselungsverfahrens ist der sog. Advanced Encryption Standard (AES) (siehe [16, S. 103–141]).

Bei den **asymmetrischen Chiffren** existieren hingegen zwei Schlüsselteile. Ein öffentlicher Schlüssel (*Public Key*) wird für die Verschlüsselung einer Nachricht genutzt und kann gefahrlos auch über unsichere Kanäle verteilt werden. Ein privater Schlüssel (*Private Key*) wird für die Entschlüsselung einer Nachricht benutzt und muss daher besonders geschützt werden. Das RSA-Kryptosystem ist ein aktueller Vertreter asymmetrischer Chiffren. Die Abkürzung RSA steht dabei für die Anfangsbuchstaben der Erfinder, Rivest, Shamir und Adelmann (siehe [16, S. 199–233]).

Der private Schlüssel kann auch dazu eingesetzt werden, um eine **Signatur** einer Nachricht zu berechnen. Mithilfe des öffentlichen Schlüssels kann dann verifiziert werden, dass der Sender im Besitz des privaten Schlüssels sein muss. Dadurch kann die Nachricht zweifelsfrei einem Sender zugeordnet werden. Eine Signatur kann dabei leider nur auf einem recht kurzen Klartext berechnet werden. Um die Signatur einer beliebig langen Nachricht zu erstellen, muss der Klartext kollisionsfrei auf eine Bitfolge bestimmter Länge abgebildet werden. Diese Aufgabe wird durch **kryptografische Hashfunktionen** übernommen (siehe [16, S. 335–362]). Die Signatur wird dann auf diesem Fingerabdruck erzeugt. Aktuelle Vertreter solcher Hashfunktionen sind z. B. SHA bzw. SHA-3.

Asymmetrische Verfahren benötigen längere Schlüssel, um dasselbe Sicherheitsniveau wie symmetrische Verfahren zu erreichen. Dadurch sind sie in ihrer Anwendung meist um den Faktor 2–3 langsamer. Sie werden daher häufig für die Verschlüsselung kleiner Datenmengen genutzt, z. B. wenn symmetrische Schlüssel sicher ausgetauscht werden müssen oder wenn Signaturen erzeugt werden sollen.

Auf Basis asymmetrischer Chiffren kann eine sog. Public-Key-Infrastruktur (PKI) aufgebaut werden. Ein **Zertifikat** verbindet den öffentlichen Schlüssel eines asymmetrischen Verschlüsselungsverfahrens mit der Identität einer Person oder eines Unternehmens. Die Ausstellung und Verteilung solcher Zertifikate wird meist durch eine zentrale Zertifizie-

rungsstelle (Certification Authority (CA)) übernommen. Die Zertifizierungsstelle bestätigt die Identität eines Zertifikatinhabers, indem das Zertifikat mit dem privaten Schlüssel der Zertifizierungsstelle signiert wird. Dies setzt ein gewisses Vertrauensverhältnis gegenüber der Zertifizierungsstelle voraus. Anstelle einer einzelnen Zertifizierungsstelle vertrauen zu müssen, werden daher mitunter auch andere Vertrauensmodelle genutzt, z. B. Web-of-Trust. Ein Standardformat für die Darstellung digitaler Zertifikate ist X.509. Entsprechende Zertifikate werden z. B. bei der verschlüsselten Kommunikation im Kontext der Transport Layer Security (TLS) oder verschlüsselter E-Mails im Rahmen des S/MIME-Standards genutzt.

Beim Aufbau einer verschlüsselten Kommunikation müssen einige Festlegungen getroffen werden. Die Kommunikationspartner müssen ein gemeinsames Verschlüsselungsverfahren und die Schlüssellänge wählen und ggf. einen Schlüssel sicher austauschen. Diese Aufgaben übernehmen **kryptografische Protokolle**. Bekannte Vertreter solcher Protokolle sind Internet Protocol Security (IPsec) bzw. TLS. TLS wird meist in Kombination mit TCP/IP eingesetzt und ist dann Bestandteil der Anwendungsschicht. Bei TLS muss sich mindestens der Server gegenüber dem Client mithilfe eines X.509-Zertifikats ausweisen. In kritischen Szenarien, in denen die Identität beider Kommunikationspartner gesichert sein muss, besitzt auch der Client ein X.509-Zertifikat. Nachdem Client und evtl. der Server die Identität des Kommunikationspartners geprüft haben, wird der Schlüssel eines symmetrischen Verschlüsselungsverfahrens bestimmt und sicher ausgetauscht. Dazu wird entweder der öffentliche Schlüssel des Serverzertifikats genutzt oder das sog. Diffie-Hellman-Schlüsselaustauschverfahren (DHKE).

Für den praktischen Einsatz kryptografischer Verfahren finden sich in den Technischen Richtlinien TR-02102 des BSI eine ganze Reihe von Empfehlungen. Der erste Teil der Richtlinie führt geeignete Verschlüsselungsverfahren und Hashfunktionen auf und gibt Hinweise zu Authentisierungs- und Schlüsseleinigungsverfahren (siehe [28]). Bei den symmetrischen Verschlüsselungsverfahren wird dort z. B. der AES mit einer Mindestschlüssellänge von 256 Bit empfohlen. Bei den asymmetrischen Verfahren wird u. a. RSA als geeignet dargestellt. Für die Erzeugung kryptografischer Hashes wird der Einsatz von SHA bzw. SHA3 mit einer Mindestlänge des Hashwerts von 256 Bit empfohlen. Der zweite Teil der Richtlinie geht auf die Verwendung von TLS ein. Es wird der Einsatz von TLS in der Version 1.2 empfohlen. Zudem werden dort geeignete Kombinationen von Schlüsseleinigungsverfahren, Verschlüsselungsverfahren, Schlüssellängen und Hashfunktionen in sog. Cipher-Suiten dargestellt (siehe [29]).

Die beste Verschlüsselung während der Datenübertragung ist allerdings nutzlos, wenn IKT-Systeme nicht durch eine wirksame **Zugangskontrolle** geschützt werden. Diese muss in der Lage sein, berechtigte Nutzer zu identifizieren und unterschiedliche Nutzungsrechte auf den angebotenen Diensten durchzusetzen. Die Identifikation der Nutzer kann technisch auf unterschiedliche Weise erfolgen. Häufig werden Benutzerkennungen und Passwörter genutzt, das Grundschutzkompendium des BSI empfiehlt aber die Nutzung von Mehr-Faktor-Authentisierung bzw. den Einsatz von Zertifikaten (siehe [31]). Nach erfolgreicher

Authentisierung werden Nutzungsrechte meist rollenbasiert vergeben. *Access-Control-Listen* dienen dann der Zuordnung der Nutzungsrechte zu den Rollen und Diensten.

Neben dem Einsatz von Verschlüsselungsverfahren kann die Kommunikationssicherheit noch durch eine Reihe weiterer Maßnahmen erhöht werden. Netzwerke sollten z. B. für unterschiedliche Informationen und Dienste getrennt voneinander betrieben werden, z. B. Intra- und Extranet. Diese Trennung kann technisch auf vielfältige Weise erreicht werden, z. B. durch den Einsatz von Paketfiltern (*Firewalls*), dem Betrieb von sog. demilitarisierten Zonen oder der Abschottung logischer Teilnetze mithilfe sog. Virtual Local Area Networks (VLANs). Darüber hinaus können viele weitere Maßnahmen ergriffen werden, z. B.: die Definition von Regeln für zulässige Verbindungszeiten oder die kontinuierliche Überwachung der Dienste und Benutzeraktivitäten (*Intrusion Detection*).

6.2 Rechtsrahmen und Standards

In Deutschland unterliegen Unternehmen der Energiewirtschaft im Bereich der Informationssicherheit besonderen Regelungen. In §11 des EnWG werden dabei für Netzbetreiber und Anlagenbetreiber getrennte Aussagen getroffen. EnWG § 11, Abs. a richtet sich an Netzbetreiber:

Der Betrieb eines sicheren Energieversorgungsnetzes umfasst insbesondere auch einen angemessenen Schutz gegen Bedrohungen für Telekommunikations- und elektronische Datenverarbeitungssysteme, die für einen sicheren Netzbetrieb notwendig sind. . . . Die Einhaltung kann von der Regulierungsbehörde überprüft werden.

EnWG §11, Abs. b richtet sich an Betreiber von Energieanlagen:

Betreiber von Energieanlagen . . . haben . . . einen angemessenen Schutz gegen Bedrohungen für Telekommunikations- und elektronische Datenverarbeitungssysteme zu gewährleisten, die für einen sicheren Anlagenbetrieb notwendig sind.

Mit Energieanlagen sind insbesondere Erzeugungs- und Speicheranlagen (elektrische Energie und Gas) gemeint, die eine bestimmte Größen überschreiten (sog. KRITIS-Anlagen). Darüber hinaus existieren viele weitere Gesetze und Vorschriften für Unternehmen der Energiewirtschaft, z. B. im Abschnitt für die Marktkommunikation nach Edi@Energy oder die Vorgaben des Bundesamts für Informationssicherheit zu den intelligenten Messsystemen.

Als Ergänzung zum EnWG wurde sowohl für Netzbetreiber als auch für Betreiber von Energieanlagen ein eigener **IT-Sicherheitskatalog** definiert (siehe [9] und [25]). Die zentrale Forderung dieser Sicherheitskataloge ist die Zertifizierung der Netzbetreiber nach IEC 27019. Als Nachweis ist der Bundesnetzagentur (BNetzA) eine Kopie des Zertifikats zu übermitteln. Bei Netzbetreibern sind alle IKT-Systeme zu berücksichtigen, die für einen sicheren Netzbetrieb notwendig sind, die direkt Teil der Netzsteuerung sind und unmittelbar Einfluss auf die Fahrweise nehmen. Es muss

ein Netzstrukturplan erstellt werden, bei dem alle IKT-Systeme in eine der folgenden Technologien einzuteilen sind:

- Leitsysteme und Systembetrieb,
- Übertragungstechnik und Kommunikation,
- sekundäre Automatisierungs- und Fernwirktechnik.

Für Betreiber von Energieanlagen muss eine Zoneneinteilung der IKT-Systeme vorgenommen werden. Maßgeblich dabei ist, ob eine Manipulation von Daten oder Systemen Auswirkungen auf die Fahrweise der Anlage haben.

Die Vorgaben des IEC 27019 basieren auf den allgemeinen Aussagen der Standardreihe IEC 270xx, die aus über 30 Einzelstandards besteht. Im Hauptdokument IEC 27001 werden branchenübergreifende Aussagen zur Informationssicherheit in Unternehmen getroffen. Dabei wird ein prozessorientierter Ansatz verfolgt (siehe [19, S. 22]). Der Fokus der Norm liegt nicht darauf, technische Einzelmaßnahmen aufzuführen, sondern vielmehr zu beschreiben, was gesichert werden sollte. Die SGIS stuft die Standardreihe IEC 270xx daher auch als *Requirements Standard* ein (siehe Abschn. 2.7)

Das Hauptziel des IEC 270xx ist es, ein Managementsystem für die Informationssicherheit in einer Organisation zu etablieren. Ein solches System wird auch als Information Security Management System (ISMS) bezeichnet. Für ein ISMS legt die Norm Anforderungen für die Einrichtung, Umsetzung, Aufrechterhaltung und fortlaufende Verbesserung fest. Ziel eines ISMS ist es, Maßnahmen kontinuierlich auf Wirksamkeit zu überprüfen und im Bedarfsfall anzupassen. Das Sicherstellen von Informationssicherheit wird dabei nicht als einmaliges Projekt, sondern als fortwährender Prozess gesehen, der organisatorisch eingebunden werden muss. Ein ISMS umfasst mindestens die folgenden Elemente (siehe [14]):

- **Die Formulierung von Sicherheitszielen**

Dabei können die Grundwerte um weitere Ziele, z. B. Authentizität usw. erweitert werden.

- **Die Bestimmung der betroffenen Assets**

Es ist ein Katalog aller betroffener Komponenten zu erstellen. Dies umfasst nicht nur physische Komponenten, wie Server, Router und Gebäude, sondern auch immaterielle Dinge, wie Daten oder Geschäftsprozesse.

- **Die Risikobeurteilung**

Risiken sind diejenigen Ereignisse, die geeignet sind, die Informationssicherheit der Assets negativ zu beeinflussen. Risiken sind mitsamt der geschätzten Schadenshöhe und den Eintrittswahrscheinlichkeiten zu erfassen.

- **Die Risikobehandlung**

Risiken können anhand von Maßnahmen reduziert bzw. beseitigt werden, indem z. B. der risikobehaftete Geschäftsprozess geändert wird. Maßnahmen zur Risikoreduktion können aus sehr unterschiedlichen Bereichen stammen, z. B. rechtliche, organisa-

torische, personelle oder technische Maßnahmen. Maßnahmen sind zu erfassen, zu implementieren und auf Wirksamkeit zu überprüfen.

- **Die kontinuierliche Verbesserung**

Ein ISMS ist ein komplexes System, welches schrittweise realisiert wird. Im Laufe des Betriebs wird das ISMS weiter verbessert, indem weitere Risiken erfasst und Maßnahmen entsprechend ausgebaut werden.

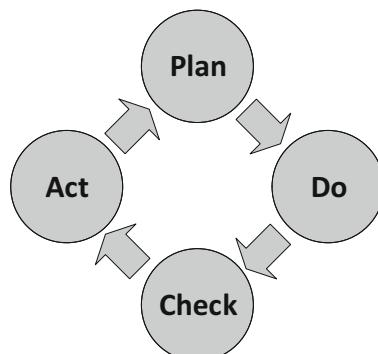
Um Informationssicherheit zu gewährleisten, reicht es nicht aus, die Aktivitäten einmal zu durchlaufen. Die kontinuierliche Verbesserung ist ein Kernprinzip des ISMS. Aus dem Qualitätsmanagement wurde daher die Systematik des **Deming-Kreislaufs** entliehen (siehe [4, S. 278]). Der Deming-Kreislauf besteht aus den vier Phasen Plan-Do-Check-Act (PDCA), die zyklisch in einer Endlosschleife durchlaufen werden. Abb. 6.1 zeigt die vier Phasen dieses Deming-Kreislaufs.

Im Anhang des IEC 27001 findet sich eine umfangreiche Liste von aktuell 114 Sicherheitsmaßnahmen, den sog. Controls. Die Controls sind in 14 thematische Zielkategorien eingeordnet. Für die Zertifizierung nach IEC 270xx müssen diese Controls für alle Komponenten bearbeitet, nicht aber unbedingt umgesetzt werden. Ist ein Control für eine Komponente nicht relevant, muss dies aber begründet werden. Einen tieferen Einblick in die Controls der IEC 27001 liefern z. B. [14] und [19]. Wir werden an dieser Stelle nicht weiter darauf eingehen.

Im IEC 27019 werden die allgemeinen Vorgaben zur Umsetzung eines ISMS auf die Prozesssteuerung und Automatisierungstechnik in der Energieversorgung übertragen. Darunter fallen nicht nur alle für die Prozesssteuerung und Automatisierung eingesetzten Komponenten, sondern auch intelligente Zähler und Energiemanagementsysteme. Nicht abgedeckt sind jedoch die Prozesssteuerungssysteme von Kernenergieanlagen, die im IEC 62645 behandelt werden.

Die meisten Controls des IEC 27001 werden im IEC 27019 um energieversorgungs-spezifische Maßnahmen ergänzt. Der Abschnitt A11 (Physische und umgebungsbezogene Sicherheit) wird z. B. um Hinweise für den Betrieb von Leitstellen erweitert. Im Abschnitt A12 (Betriebssicherheit) wird die Trennung von Entwicklungs-, Test- und

Abb. 6.1 Die vier Phasen des Deming-Kreislaufs



Betriebsumgebungen verlangt. Zudem werden besondere Maßnahmen für den Schutz gegen Schadsoftware dargestellt. Im Abschnitt A13 (Kommunikationssicherheit) wird die Absicherung der Prozessdatenkommunikation gefordert. Die dabei eingesetzten Protokolle, z. B. IEC 60870-5, DNP3 oder IEC 61850, sollen dabei durch die Aktivierung bereits unterstützter Sicherheitsmerkmale geschützt werden.

6.3 Kommunikationssicherheit

In nahezu allen internationalen Studien und Untersuchungen wird IEC 62351 als zentraler technischer Standard für die Informationssicherheit in Energienetzen gesehen (siehe [20, S. 21]). Auch der IEC 27019 nimmt an einigen Stellen Bezug auf IEC 62351. Das Hauptziel dieses Standards ist es, die Kommunikationssicherheit für Netzführungssysteme sicherzustellen. Im Fokus des Standards stehen daher alle IKT-Systeme, die im Rahmen der Kommunikationsarchitektur in und mit Schaltanlagen eingesetzt werden (siehe Abb. 5.2). Der Standard führt dazu eine ganze Reihe technischer Maßnahmen auf und zielt dabei insbesondere auf Kommunikationsprotokolle, wie IEC 60870-5, DNP3 und IEC 68150, ab. Die Standardreihe zerfällt in insgesamt 18 Abschnitte, die sich auf eine Reihe anderer Standards beziehen. Abb. 6.2 zeigt einen (unvollständigen) Überblick über die Struktur des IEC 62351.

Konkrete Maßnahmen für die Absicherung der Kommunikationsprotokolle finden sich in den Abschnitten 3 bis 6. IEC 62351-3 dient der Absicherung der Kommunikation über TCP/IP-basierte Kommunikationsnetze. Um Authentisierung, Vertraulichkeit und Integrität zu erreichen, wird der Einsatz von TLS näher ausgelegt. Optionale Bestandteile werden verbindlich vorgeschrieben und spezielle Anforderungen an die zu nutzenden Zertifikate gestellt. IEC 62351-4 zielt auf die Absicherung der Kommunikation über Manufacturing Messaging Specification (MMS) ab. Dabei werden mehrere sog. Transport-(OSI-Ebene 1–4) und Anwendungsprofile (OSI-Ebene 5–7) definiert, die den Einsatz von VPN und TLS beschreiben. IEC 62351-5 berücksichtigt die Besonderheiten der seriellen Kommunikation. Vorausgesetzt wird ein funktionierendes Zertifikatsmanagement, welches in IEC 62351-9 beschrieben wird. Nachrichten auf dem Bus sollen mithilfe eines sog. Hash-based Message Authentication Code (HMAC) signiert werden, damit ein Empfänger die Integrität der Anfrage prüfen kann. IEC 62351-6 behandelt die zeitkritische Kommunikation auf dem Prozessbus. Dabei ist zu beachten, dass unter harten Echtzeitanforderungen nur begrenzt Sicherheitsmaßnahmen definiert werden können. Neben der Abschottung logischer Teilnetze wird die Verwendung von Gruppenschlüsseln zur Berechnung eines HMAC bei Generic Object Oriented Substation Events (GOOSE) und Sampled Values (SV) empfohlen.

Der IEC 62351 geht auch auf andere Aspekte der Informationssicherheit mit Bezug zur Netzautomation ein. Der Fokus von IEC 62351-7 liegt z. B. auf dem Netzwerk- und Systemmanagement. Darin werden Datenobjekte definiert, die möglichst durch Standardprotokolle wie SNMP abgefragt werden können. Dadurch soll die Informationsinfrastruktur

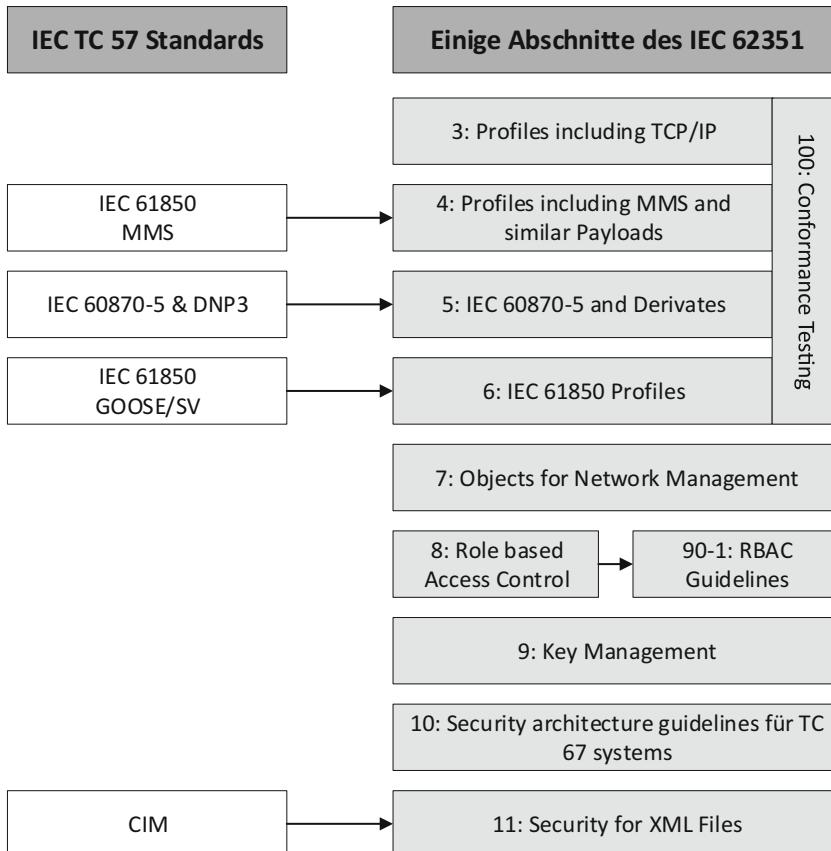


Abb. 6.2 Die Struktur des IEC 62351 und die Zuordnung der Abschnitte zu verwandten Standards

tur ebenso zuverlässig überwacht werden können wie die eigentliche Primärtechnik. Der IEC 62351-8 beschreibt Methoden zur Verwendung einer rollenbasierten Zugriffskontrolle für Energiemanagementsysteme. Dabei werden Rollen und Zuordnungen zu bestimmten Rechten sowie ein standardisiertes Verfahren der Integration des Verfahrens entlang der gesamten Energieversorgung definiert. Es werden zudem verschiedene Methoden beschrieben, wie die Rolleninformationen dargestellt und übertragen werden können. Im Abschnitt IEC 62351-11 werden Hinweise für die sichere Übertragung von Extensible Markup-Language-(XML-)Dateien gegeben. Dies ist z. B. für den Austausch von CIM-Daten oder die Nutzung der SCL in IEC 61850 hilfreich (siehe Abschn. 3.1 und 5.1).

Auch in der Gebäudeautomation spielt IT-Sicherheit (mittlerweile) eine wichtige Rolle. Die meisten dort eingesetzten Systeme verfügten anfangs allerdings über keinerlei nennenswerte Mechanismen im Bezug zu IT-Sicherheit (siehe [8] und [3]). Die Systeme waren als isolierte Inselsysteme konzipiert. Die Kommunikation zwischen Sensoren,

Aktoren und übergeordneten Leitsystemen wurde meist unverschlüsselt und ohne Authentifizierung oder Integritätsprüfung abgewickelt. Erlangte ein Angreifer Zugang zum Übertragungsmedium, war es daher ein Leichtes, ein System komplett zu übernehmen (siehe z. B. [15]). Dies kann u. a. dazu genutzt werden, um sich physischen Zugang zu Gebäude(teilen) zu verschaffen, Sicherheitsmechanismen, wie z. B. Brandschutz, außer Kraft zu setzen oder das Nutzungsverhalten einzelner Bewohner auszuspähen. Viele Gebäudemagementsysteme sind heute aber längst keine isolierten Inselsysteme mehr. Oftmals sind sie mit dem öffentlichen Internet verbunden, um z. B. die Fernwartung zu ermöglichen. Die Gefahr für Angriffe steigt dadurch erheblich. Entsprechend existiert ein hoher Bedarf, die Kommunikationsstandards in diesem Bereich um Sicherheitsmechanismen zu erweitern. Im Gegensatz zur Netzautomation existiert für die Gebäudeautomation kein zentraler Standard vergleichbar dem IEC 62351, der Sicherheitsmaßnahmen für alle möglichen Protokolle bündelt. Die Kommunikationsstandards der Gebäudeautomation wurden in den letzten Jahren meist punktuell um entsprechende Maßnahmen erweitert. Um allerdings von diesen Erweiterungen profitieren zu können, müssen meist neue Geräte eingesetzt werden.

Der KNX-Standard führt einige Maßnahmen ein, um die IT-Sicherheit eines solchen Systems zu erhöhen. Von jeher konnte z. B. die ETS-Software zur Einrichtung und Konfiguration eines KNX-Systems mit einem Passwortschutz versehen werden. Dadurch konnten unbefugte Personen daran gehindert werden, eine KNX-Installation umzukonfigurieren. In neueren Versionen des Standards wurden mit **KNX/IP Secure** und **KNX/Data Secure** Mechanismen nachgerüstet, um Vertraulichkeit, Integrität und Authentizität auf verschiedenen Übertragungsmedien sicherzustellen. Beide Standards werden in der Praxis bislang eher selten genutzt.

Die Übertragung von KNX-Télégrammen über ein IP-basiertes Backend kann mit Hilfe von KNX IP Secure abgesichert werden (siehe [18]). Dazu müssen die KNX-Geräte, die z. B. Teil einer TP-Linie sind, selbst diese Erweiterung nicht unterstützen. Lediglich die KNX-Gateways, die mehrere Linien über ein IP-basiertes Netzwerk verbinden, müssen den neuen Standard implementieren. Der Gateway verpackt dabei ein KNX-Télégramm als Nutzdaten in einem IP-Paket und verschlüsselt es mithilfe von AES. Zusätzlich wird für die Integritätsprüfung ein HMAC angehängt. Bei Multicast-Kommunikation wird ein spezieller Gruppenschlüssel genutzt, bei Unicast-Kommunikation wird mithilfe einer Variante des Diffie-Hellman-Schlüsselaustauschverfahrens ein Session-Key ausgehandelt. In [7] wird gezeigt, dass die Verwendung von KNX/IP Secure allein weiterhin gewisse Sicherheitsschwächen aufweist.

Um auch auf den KNX-Linien Sicherheitsmaßnahmen einzuführen, wurde KNX/Data Secure entwickelt (siehe [22]). Dazu musste allerdings die Struktur der KNX-Télégramme erweitert werden, warum ein Mischbetrieb mit Geräten, die diesen Standard nicht unterstützen, auf einer Linie nicht möglich ist. Die neuen Télégramme besitzen z. B. einen HMAC für die Integritätsprüfung und eine *Sequence Number*, die verhindert, dass Télégramme wiederholt werden können. Die eigentlichen Nutzdaten der Télégramme werden mithilfe von AES verschlüsselt. Dazu existieren mehrere unterschiedliche Schlüssel. Für

die Einrichtung über die ETS-Software existiert der sog. Tool-Key, für die Laufzeitkommunikation zwischen den Geräten existiert für jede Gerät- oder Gruppenadresse ein eigener Schlüssel, den ein Gerät in der *Security Link Table* vorhält. Auf Basis dieser *Security Link Table* kann dann auch feingranular konfiguriert werden, welcher Client Zugriff auf welche Daten und Dienste eines Geräts bekommen soll und welche Dienste verschlüsselte Kommunikation zwingend erforderlich machen.

LonTalk scheint bislang kaum nennenswerte Sicherheitsmechanismen zu implementieren. Lediglich die Authentifizierung eines Geräts wird unterstützt. Dazu wird ein *Challenge-Response*-Mechanismus eingesetzt, bei dem ein einzelner systemweit gültiger 48 Bit langer Schlüssel benutzt wird, um einen Hashwert zur Authentifizierung zu berechnen. Es wurde allerdings gezeigt, dass dieses Verfahren leicht zu brechen ist (siehe [10]). Ein sicheres Verfahren, um den Schlüssel über das Netzwerk auszutauschen, existiert nicht. Auch können die Nutzdaten selbst nicht verschlüsselt werden (siehe auch [1]). LonTalk allein kann daher aktuell nicht als sicher angesehen werden.

Im Jahr 2004 wurde der BACnet-Standard im Rahmen eines Addendums um die *BACnet Security Architecture* erweitert. Die Erweiterungen machten die Verschlüsselung und Integritätsprüfung von Nachrichten möglich und führten einige zusätzliche Objekttypen ein. Es konnte dennoch gezeigt werden, dass die Netzwerksicherheit von BACnet durch verschiedene Angriffstypen gefährdet ist (siehe [11]). Ein Angriffsmöglichkeit bietet dabei z. B. das BACnet Broadcast Management Device (BBMD), welches im Kontext von IP-basierten Netzen eingesetzt werden muss. Vor allem aber wegen der mangelnden Industrieunterstützung wurde die BACnet Security Architecture im Jahr 2016 wieder aus dem Standard entfernt (siehe [27], S. 2). Stattdessen wurde im Jahr 2016 mit dem **BACnet Secure Connect (BACnet/SC)** ein anderer Ansatz vorgestellt (siehe [30]). BACnet/SC führt einen neuen *Virtual Link Layer* ein, der auf IP-basierten Netzen aufsetzt. Im Vergleich zu BACnet/IP wird die Kommunikationsarchitektur dabei stark verändert.

Zentrales Element von BACnet/SC ist der *Hub*, der auch die Notwendigkeit für ein BBMD ersetzt. Diese neue Komponente steuert den Datenverkehr in einem BACnet-Subnetz und regelt auch die Weitergabe von Nachrichten an mehrere Empfänger, wenn nötig. Die Notwendigkeit sowohl von netzweiten Broadcast-Nachrichten als auch von festen IP-Adressen entfällt. Da ein Hub eine zentrale Rolle in einem BACnet/SC-System übernimmt, kann diese Komponente auch redundant ausgelegt werden. Zudem ist es möglich, den Hub auch als cloudbasierten Service zu realisieren. Diese neue Architektur ähnelt der Idee sog. Message Oriented Middleware, wie z. B. MQTT. Nachrichten werden bei BACnet/SC per TCP/IP über Websockets übertragen. Die Kommunikation wird durch TLS verschlüsselt, Geräte nutzen dazu Zertifikate, warum eine PKI aufgesetzt werden muss. Der Vorteil von BACnet/SC liegt darin, dass bereits etablierte Sicherheitskonzepte genutzt werden, z. B. TLS. Durch die geänderte Kommunikationsarchitektur über den Hub können nun auch andere Maßnahmen leichter umgesetzt werden, z. B. der Betrieb von Firewalls oder die Abschottung von Netzbereichen durch VLANs. Geräte, die auf BACnet/SC basieren, können allerdings nicht ohne Weiteres mit regulären BACnet-

Geräten zusammenarbeiten. Bestehende Installationen können daher zunächst kaum von BACnet/SC profitieren, da die Geräte diesen Standard nativ unterstützen müssen.

OPC UA ist einer der wenigen Automationsstandards, die den Aspekt der IT-Sicherheit von Anfang an adressiert haben. In Teil 2 des Standards werden entsprechende Sicherheitsmechanismen dargestellt (siehe [13]). Der Nutzer von OPC UA kann dabei aus drei sog. Security Policies auswählen, um Schutzziele gezielt umzusetzen (None, Sign, SignAndEncrypt). Die Sicherheitsmechanismen werden auf Basis der Kommunikationskontexte von OPC UA umgesetzt (siehe [21, S. 32]). Auf der Transportebene kann zwischen Client und Server ein *Secure Channel* aufgebaut werden, bei dem Nachrichten verschlüsselt und signiert werden. In der Sitzungsschicht können Nutzer authentifiziert und autorisiert werden. Für die Identifikation von Nutzern können dabei unterschiedliche Mechanismen genutzt werden, z. B. X.509-Zertifikate. Das BSI hat in Zusammenarbeit mit dem TÜV dem OPC-UA-Standard einen hohen Grad an IT-Sicherheit bescheinigt (siehe [17]).

Intelligente Messsysteme stellen ein Bindeglied zwischen der Gebäudeautomation und dem Smart Grid dar. Aus mehreren Gründen müssen sie daher besonderen Sicherheitsanforderungen genügen. Es muss z. B. verhindert werden, dass über massenhaft kompromittierte Smart Meter die Fahrweise von Erzeugungseinheiten und Netzen beeinflusst werden kann. Auch der Zugriff auf steuerbare Lasten und Erzeuger in den Häusern muss abgesichert werden. Nicht zuletzt darf auch die betrügerische Manipulation der Lastdaten nicht einfach möglich sein.

Die intelligenten Messsysteme befinden sich dabei aber außerhalb des Sicherheitseinflusses der Netzbetreiber. Sie müssen daher resilient gegen alle Arten der Manipulation sein, auch gegen physische Attacken (siehe [12, Kap. 4]). Dass die Spezifikation sicherer Protokolle im Kontext intelligenter Messsysteme auch misslingen kann, zeigt das sog. Open Smart Grid Protocol (OSGP). Dieses wird weltweit (außer in Deutschland) für die Kommunikation mit intelligenten Messsystemen eingesetzt. Es konnte allerdings gezeigt werden, dass die in diesem Protokoll definierten Sicherheitsmechanismen nicht ausreichend sind und leicht gebrochen werden können (siehe [10]).

In Deutschland wurden im Kontext intelligenter Messsysteme durch das Messstellenbetriebsgesetz (MsbG) umfangreiche Regelungen zum Datenschutz und zur Datensicherheit gemacht. Insbesondere der Smart Meter Gateway (SMG) muss besonderen technischen Vorgaben entsprechen (siehe § 21–§ 24 MsbG). Dabei muss die technische Richtlinie BSI TR-03109 eingehalten werden (siehe [6]). Der Hersteller eines SMG und der Messstellenbetreiber sind zudem nach IEC 270xx zu zertifizieren. Die Messdaten müssen bei der Übertragung vor Abhören, Manipulation und Fälschung geschützt werden. Für alle Kommunikationsbereiche des SMG werden im BSI TR-03109 entsprechende kryptografische Vorgaben gemacht.

In allen Kommunikationsbereichen ist der Einsatz von TLS in Version 1.20 in Kombination mit einer PKI vorgeschrieben. Als Verschlüsselungsalgorithmus darf dabei ausschließlich AES mit mindestens 128 Bit Schlüssellänge in den Modi *Cipher Block Chaining* oder *Galois Counter Mode* genutzt werden. Auf nationaler Ebene existiert zudem eine CA. Endnutzerzertifikate werden durch untergeordnete CAs vergeben.

Im Local Metrological Network (LMN) darf eine TLS-Session nicht länger als 31 Tage aufrecht gehalten werden. Dabei dürfen auch nicht mehr als 5 MB an Daten übertragen werden. Im Home Area Network (HAN) darf eine TLS-Session nicht länger als 48 Stunden aufrecht gehalten werden. Besondere zusätzliche Richtlinien gelten für die Kommunikation im Wide Area Network (WAN). Der SMG darf z. B. keine Verbindungen entgegennehmen, die nicht vom Messstellenbetreiber ausgehen. Will ein externer Marktteilnehmer auf die Controllable-Local-System-(CLS-)Schnittstelle zugreifen, muss er zunächst den Smart Meter Gateway Administrator (SMGA) kontaktieren, der dann seinerseits den SMG anweist, eine Verbindung mit dem externen Marktteilnehmer aufzubauen.

In Deutschland wird die **Marktkommunikation** zwischen Akteuren direkt per E-Mail abgewickelt. Jeder Akteur muss dazu eine spezielle E-Mail-Adresse vorhalten und sie auf seiner Webseite veröffentlichen. Im Anhang der E-Mail werden die Nachrichten in Form maschinenlesbarer EDIFACT- oder ESS-Dateien versandt (siehe Abschn. 5.4). Der E-Mail-Standard (Simple Mail Transfer Protocol (SMTP)) definiert dabei ursprünglich keinerlei Sicherheitsmechanismen. Bei der Marktkommunikation wird Vertraulichkeit und Authentizität mithilfe von S/MIME sichergestellt, wobei die E-Mails mithilfe von X.509-Zertifikaten verschlüsselt und signiert werden. Die Verfügbarkeit des Postfachs muss dabei jeder Akteur selber sicherstellen. Dies hat den Vorteil, dass der Ausfall des Postfachs eines einzelnen Energieversorgungsunternehmen (EVU) andere Akteure kaum beeinflusst. Der große Nachteil von E-Mails ist allerdings, dass die erfolgreiche Zustellung rechtlich kaum nachgewiesen werden kann. Mittel- bis langfristig wäre es daher sicher sinnvoll, die Marktkommunikation in Deutschland auf technisch andere Füße zu stellen.

6.4 Zusammenfassung

In diesem letzten Kapitel des Lehrbuchs haben wir uns mit einem wichtigen Querschnittsthema der Energieinformatik befasst, der IT-Sicherheit. Um die Energieversorgung sicherzustellen, muss auch die zugehörige IKT-Infrastruktur vor Angriffen und Ausfällen geschützt werden. Dazu haben wir uns die entsprechenden Grundwerte Verfügbarkeit, Vertraulichkeit und Integrität angesehen, die es sicherzustellen gilt. Zudem wurde der Rechtsrahmen für die deutsche Energiewirtschaft vorgestellt, der sich im Energiewirtschaftsgesetz und diversen anderen Vorschriften niederschlägt. Insbesondere wird von EVU erwartet, dass sie sich konform zur IEC 270xx aufstellen und ein ISMS einführen und betreiben. Einige Maßnahmen dazu finden sich in den Controls des IEC 27091, des IEC 27019 und werden in weiteren Dokumenten des BSI oder des Bundesverbands der Energie- und Wasserwirtschaft e.V. (BWED) präzisiert. Im Kontext der Netzautomation spielt die Standardreihe IEC 62351 eine herausragende Rolle. Dort werden in insgesamt 18 Abschnitten Vorgaben gemacht, wie die Kommunikation abgesichert werden kann.

Viele Gebäudeautomationssysteme haben erst in den letzten Jahren damit begonnen, Sicherheitsmechanismen nachzurüsten. Für die in dieser Veranstaltung gezeigten Systeme

KNX, LonWorks, BACnet und OPC UA wurden entsprechende Maßnahmen dargestellt. Im Kontext intelligenter Messsysteme spielen in Deutschland die Maßgaben des BSI eine ausschlaggebende Rolle. Diese dienen dazu, die Übertragung und -verarbeitung der Messdaten im Kontext eines SMG abzusichern. Die Marktkommunikation zwischen den Akteuren des Energiesystems wird in Deutschland mithilfe von E-Mails abgebildet. Dabei wird S/MIME eingesetzt, um Vertraulichkeit, Integrität und Authentizität sicherzustellen.

6.5 Kontrollfragen

- Warum spielt IT-Sicherheit in der Energiewirtschaft heute eine immer größere Rolle?
- Was ist mit *Security Through Obscurity* gemeint und warum kann dieses Prinzip keinen ausreichenden Schutz entwickeln?
- Was ist der Unterschied zwischen Informations- und IT-Sicherheit?
- Welche Gefahren bedrohen die Verfügbarkeit eines IT-Systems?
- Wobei hilft Redundanz und wie stellt man diese her?
- Wie kann die Integrität bei der Übertragung von Daten sichergestellt werden?
- Wie gibt man die Länge eines Schlüssels an? Wofür ist die Länge ein Gradmesser? Warum?
- Was ist der Unterschied zwischen einer symmetrischen und einer asymmetrischen Chiffre? Nennen Sie Vertreter entsprechender Verfahren.
- Wieso werden nicht grundsätzlich ausschließlich asymmetrische Chiffren eingesetzt?
- Wofür werden kryptografische Hashverfahren benötigt?
- Wozu dient eine X.509-Signatur?
- Welche Aufgabe hat eine *Certification Authority*?
- Wie wird beim TLS-Protokoll der Schlüssel ausgetauscht?
- In welchem Gesetz wird von Netzbetreibern und Betreibern von Energieanlagen ein angemessener Schutz gegen Bedrohungen ihrer IT-Systeme gefordert? Wie und wem wird dieser Schutz nachgewiesen?
- Was ist die Kernforderung der Standardreihe IEC 270xx?
- Aus welchen Schritten besteht der sog. Deming-Kreislauf?
- Was für eine Kategorie von Standards ist der IEC 62351 laut der Einordnung der Smart Grid Security Group?
- Warum sind auf dem Prozessbus Sicherheitsanforderungen nur schwierig umsetzbar?
- Existiert ähnlich dem IEC 62351 auch ein gemeinsamer Sicherheitsstandard für die Gebäudeautomation?
- Was ist der Unterschied zwischen KNX/IP Secure und KNX/Data Secure?
- Welche Aufgabe hat bei BACnet/SC der sog. Hub? Können auch ältere Geräte von BACnet/SC profitieren?
- Darf ein Smart Meter Verbindungen aus dem WAN entgegennehmen? Warum?
- Wie wird die Vertraulichkeit im Rahmen der deutschen Marktkommunikation sicher gestellt?

- Welche Schwierigkeit kann sich bei der aktuellen technischen Umsetzung der Marktkommunikation in Deutschland ergeben?

Literatur

1. W. Granzer u. a. „Security in networked building automation systems“. In: 2006 IEEE International Workshop on Factory Communication Systems. Juni 2006, S. 283–292. <https://doi.org/10.1109/WFCS.2006.1704168>.
2. Mark Bedner und Tobias Ackermann. „Schutzziele der IT-Sicherheit“. In: *Datenschutz und Datensicherheit – DuD* 34.5 (1. Mai 2010), S. 323–328. ISSN: 1862–2607. URL: <https://doi.org/10.1007/s11623-010-0096-1> (besucht am 20.05.2020).
3. Wolfgang Granzer und Wolfgang Kastner. „Security Analysis of Open Building Automation Systems“. In: *Computer Safety, Reliability, and Security*. Hrsg. von Erwin Schoitsch. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2010, S. 303–316. ISBN: 978-3-642-15651-9. https://doi.org/10.1007/978-3-642-15651-9_23
4. Gerd F. Kamiske und Jörg-Peter Brauer. *Qualitätsmanagement von A-Z: Wichtige Begriffe des Qualitätsmanagements und ihre Bedeutung*. 7., aktualisierte und erweiterte. München Wien: Carl Hanser Verlag GmbH & Co. KG, 7. Apr. 2011. 404 S. ISBN: 978-3-446-42581-1.
5. Thomas Petermann u. a. *Was bei einem Blackout geschieht: Folgen eines langandauernden und großflächigen Stromausfalls*. 2. Auflage. Studien des Büros für Technikfolgen-Abschätzung beim Deutschen Bundestag 33. Berlin: edition sigma, 1. Jan. 2013. 259 S. ISBN: 978-3-8360-8133-7.
6. *Technische Richtlinie BSI TR-03109-1*. Bonn: Bundesamt für Sicherheit in der Informations-technik (BSI), 28. März 2013. URL: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR03109_1.pdf?__blob=publicationFile&v=1.
7. Aljosha Judmayer, Lukas Krammer und Wolfgang Kastner. „On the security of security extensions for IP-based KNX networks“. In: *2014 10th IEEE Workshop on Factory Communication Systems (WFCS 2014)*. 2014 10th IEEE Workshop on Factory Communication Systems (WFCS 2014). Mai 2014, S. 1–10. <https://doi.org/10.1109/WFCS.2014.6837593>.
8. Jörg Thoma. *Smarthomes, offen wie Scheunentore*. 15. Dez. 2014. URL: <https://www.golem.de/news/security-smarthomes-offen-wie-scheunentore-1412-111173.html> (besucht am 16.06.2020).
9. *IT-Sicherheitskatalog gemäß § 11 Absatz 1a Energiewirtschaftsgesetz*. Aug. 2015. https://www.bundesnetzagentur.de/SharedDocs/Downloads/DE/Sachgebiete/Energie/Unternehmen_Institutionen/Versorgungssicherheit/IT_Sicherheit/IT_Sicherheitskatalog_08-2015.pdf
10. Philipp Jovanovic und Samuel Neves. „Dumb Crypto in Smart Grids: Practical Cryptanalysis of the Open Smart Grid Protocol“. In: *IACR Cryptol. ePrint Arch.* (2015). https://doi.org/10.1007/978-3-662-48116-5_15.
11. Jaspreet Kaur u. a. „Securing BACnet’s Pitfalls“. In: *ICT Systems Security and Privacy Protection*. Hrsg. von Hannes Federrath und Dieter Gollmann. IFIP Advances in Information and Communication Technology. Cham: Springer International Publishing, 2015, S. 616–629. ISBN: 978-3-319-18467-8. https://doi.org/10.1007/978-3-3191-8467-8_41.
12. Florian Skopik und Paul Smith. *Smart Grid Security: Innovative Solutions for a Modernized Grid*. 1. Aufl. Syngress, 11. Aug. 2015. 317 S.
13. *IEC TR 62541-2:2016 – OPC unified architecture – Part 2: Security Model*. VDE Verband der Elektrotechnik Elektronik Informationstechnik e.V., Okt. 2016. URL: <https://www.vde-verlag.de/iec-normen/223904/iec-tr-625412-2-016.html> (besucht am 19.06.2020).

14. Heinrich Kersten u. a. *IT-Sicherheitsmanagement nach der neuen ISO 27001: ISMS, Risiken, Kennziffern, Controls*. 1. Aufl. 2016. Wiesbaden: Springer Vieweg, 23. Sep. 2016. 268 S. ISBN: 978-3-658-14693-1.
15. Thomas Mundt und Peter Wickboldt. „Security in building automation systems – a first analysis“. In: 2016 International Conference On Cyber Security And Protection Of Digital Services (Cyber Security). Juni 2016, S. 1–8. <https://doi.org/10.1109/CyberSecPODS.2016.7502336>.
16. Christoph Paar und Jan Pelzl. *Kryptografie verständlich: Ein Lehrbuch für Studierende und Anwender*. Springer Vieweg, 2016. 440 S. ISBN: 3-662-49296-2.
17. *Sicherheitsanalyse Open Platform Communications Unified Architecture (OPC UA)*. Bundesamt für Sicherheit in der Informationstechnik (BSI), 25. Apr. 2016. URL: https://www.bsi.bund.de/DE/Publikationen/Studien/OPCUA/OPCUA_node.html (besucht am 19.06.2020).
18. *Application Note 159/13 v06 - KNXnet/IP Security*. 27. Okt. 2017. URL: <https://knxcloud.org/index.php/s/RmWMJBoA7vCn2ch/download?path=%2F&files=AN159%20v06%20KNXnet-IP%20Secure%20AS.pdf> (besucht am 16.06.2020).
19. Michael Brenner u. a. *Praxisbuch ISO/IEC 27001: Management der Informationssicherheit und Vorbereitung auf die Zertifizierung. Zur Norm ISO/IEC 27001:2015*. 2., aktualisierte und erweiterte. München: Carl Hanser Verlag GmbH & Co. KG, 10. Apr. 2017. 254 S. ISBN: 978-3-446-45139-1.
20. *Deutsche Normungs-Roadmap IT-Sicherheit – Version 3.0*. DKE – Deutsche Kommission Elektrotechnik Informationstechnik, 21. Juni 2017. URL: <https://www.dke.de/resource/blob/778258/44c2d336c2702f285ba669ee5cd47b10/deutsche-normungs-roadmap-it-sicherheit-version-3-0-data.pdf> (besucht am 04. 06. 2020).
21. Miriam Schleipen, Hrsg. *Praxishandbuch OPC UA: Grundlagen – Implementierung – Nachrüstung – Praxisbeispiele*. 1. Aufl. Würzburg: Vogel Business Media, 1. Nov. 2017. 192 S. ISBN: 978-3-8343-3413-8.
22. *Application Note 158/13 v04 - KNX Data Security*. 5. Jan. 2018. URL: <https://knxcloud.org/index.php/s/RmWMJBoA7vCn2ch/download?path=%2F&files=AN158%20v04%20KNX%20Data%20Security%20AS.pdf> (besucht am 16.06.2020).
23. *Die Lage der IT-Sicherheit in Deutschland 2018*. Bonn: Bundesamt für Sicherheit in der Informationstechnik (BSI), 2018, S. 98. URL: <https://www.bmi.bund.de/SharedDocs/downloads/DE/publikationen/themen/it-digitalpolitik/bsi-lagebericht-2018.pdf> (besucht am 20.11.2019).
24. Claudia Eckert. *IT-Sicherheit: Konzepte – Verfahren – Protokolle*. 10th expanded and updated edition. München: De Gruyter Oldenbourg, 21. Aug. 2018. 1021 S. ISBN: 978-3-11-055158-7.
25. *IT-Sicherheitskatalog gemäß § 11 Absatz 1b Energiewirtschaftsgesetz*. Dez. 2018. URL: https://www.bundesnetzagentur.de/SharedDocs/Downloads/DE/Sachgebiete/Energie/Unternehmen_Institutionen/Versorgungssicherheit/IT_Sicherheit/IT_Sicherheitskatalog_2018.pdf.
26. Armin Selhofer, Kay Tidten und Stephan Beirer. *Anforderungen an sichere Steuerungs- und Telekommunikationssysteme*. Wien/Berlin: Österreichs E-Wirtschaft und BDEW Bundesverband der Energie- und Wasserwirtschaft e.V., 8. Mai 2018. URL: https://www.bdew.de/media/documents/Awh_20180507_OE-BDEW-Whitepaper-Secure-Systems.pdf (besucht am 18.11.2019).
27. *ANSI/ASHRAE Addendum by to ANSI/ASHRAE Standard 135-2016*. ASHRAE, 18. Nov. 2019. URL: <http://www.bacnet.org/Addenda/Add-135-2016by.pdf> (besucht am 18.06.2020).
28. *Technische Richtlinie TR-02102-1 - Kryptographische Verfahren: Empfehlungen und Schlüssellängen*. Bundesamt für Sicherheit in der Informationstechnik (BSI), 22. Feb. 2019. URL: <https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR02102/BSI-TR-02102.pdf> (besucht am 28.02.2020).
29. *Technische Richtlinie TR-02102-2 – Kryptographische Verfahren: Verwendung von Transport Layer Security (TLS)*. Bundesamt für Sicherheit in der Informationstechnik (BSI), 22. Feb. 2019. URL: <https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR02102/BSI-TR-02102-2.pdf> (besucht am 28.02.2020).

30. *ANSI/ASHRAE Addendum bj to ANSI/ASHRAE Standard 135-2016*. ASHRAE, 18. Nov. 2020.
URL: <http://www.bacnet.org/Addenda/Add-135-2016bj.pdf> (besucht am 18.06.2020).
31. *IT-Grundschutz Kompendium*. Bundesamt für Sicherheit in der Informationstechnik (BSI), 2020.
URL: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Grundschutz/Kompendium/IT_Grundschutz_Kompendium_Edition2020.pdf

Stichwortverzeichnis

A

ABB 59
Adressraum 188
Aggregation 34
Algorithmic Reporting 185
Ampere 11
Amprion 16
Anomaly Report 211
Anwendungsschicht 49
Apache HBASE 103
Apple 175
ArcGIS 93
Assoziation 34
Attribut 35
Auktion 21
Auszeichnung 39
Auszeichnungssprache 39
Authentizität 224
Autodesk 93
Automationsebene 54
Automatisierungstechnik 54

B

4B/5B-Codierung 45
BACnet 179
BACnet/WS 186
Bandbreitenreservierung 61
Baseload 21
Basisinformationsmodell 188
Basisklasse 35
Beanit GmbH 166
Best of Breed 97

Betriebsmittel 154
Bewegungsdaten 32
Beziehung 188
Beziehungstyp 189
Bezugssystem 90
Big Data 29
Bilanzkreis 16
Bindung 173
Bitcoin 29
Bitübertragungsschicht 45
Blindleistung 12
Blockchain 2, 29, 208
Branch and Bound 145
Branch and Cut 145
Broadcast 48
Brute-Force 225
Bus 46
Bustopologie 45

C

450 Connect GmbH 202
Cause of Transmission 160
Channel 177
CIMDesk 83
CIM RDF 81
CIMSpy 83
CIM User Group 76
CIM++ 85
Class Poll 157
Clean Dark Spread 148
Clean Spark Spread 148
Client/Server 48

Cloud-Computing 28
Coil 58
COM/DCOM 188
Commodity 21
ComServer 191
Confirmation Report 211
Control 229
Controlled Station 159
Controlling Station 159
CPLEX 136
Cypress 175

D

Darstellungsschicht 48
Data Package 108
Data Point 172
Data Science 42
Daten 31
 Bewegungs- 32
Datenbanksystem 40
Datenmodell 38
Datenqualität 32
Datensatz 40
Datentyp 35
Decision Support 148
Dekomposition 124
Deming-Kreislauf 229
Deserialisieren 38
Deutsche Telekom AG 202
digitale Revolution 27
Digitaler Schatten 66
Docker 102, 104
Domotronik 167
Drehstrom 12

E

ebUtilities 208
Echelon 175
Echtzeit 47, 56
E-DeMa 4
EEBus 4
EEG-Umlage 23
E-Energy 3
Einliniendiagramm 13
Einspeisestruktur 26
Elektrische Energie 11

Elektromobilität 26
Encoding Rules 49
Endenergieträger 11
Endkunde 15
Energie 10
Energiebörse 21
Energie Code & Service GmbH 206
Energiedaten 95
Energiedatenmanagementsystem 65
Energiegennzahlen 65
Energiehandel 21
Energieinformatik 1
Energiemanagement 65
Energiesektor 10
Energievertrieb 20
Energiewende 23
Energiewirtschaft 19
Energy 205
ENERGYLink 208
Enterprise Architect 37
entso-e 16
Erdgas 12
ESP8266 187
ESRI 93
eTelligence 4
EtherCAT 60
Ethereum 29
Ethernet 50
EtherNet/IP 60
Etikett 39
Event Buffer 157
Event Data 157
Exponentielles Glätten 127

F

Fahrlässigkeit 223
Fahrplanmanagement 209
Fehlermaß 123
Feldbus 56
Feldebene 54
Fenstergröße 124
Ferntransportnetz 15
Fernwirktechnik 56, 155
Frame 46
FreeOpcUa 193
Frequenz 12
Frequenzhaltung 16
funktionales Modell 78

G

Gebäudesystemtechnik 168
Generalisierung 36
Georg Dantzig 135
Geräteadresse 171
Gestehungskosten 17
GET 52
Gleichstrom 12
gleitender Mittelwert 124
Glättungsfaktor 127
GLPK 137
Grafana 104
Graphite 101
GRASS GIS 93
Group Object 173
Großhandelsmarkt 21
Gruppenadresse 172, 177
GUROBI 136

H

50Hertz 16
Hadoop 103
HBase 104
HDFS 103
höhere Gewalt 223
Holt-Winters-Verfahren 127
Hub 45

I

IEC 60870-5 158
IEC 61850 162
Inbetriebnahme 173
Industrial Ethernet 60
Industrie 4.0 187
Information Object 160
Informationsmodell 33
Integrität 223
intelligentes Messsystem 196
Interface Object 173
Interior-Point-Verfahren 136
Internetwork 182
Interrogation 160
Interworking Model 172
Intrinsic Reporting 185
IP-Adresse 51
Isogewinngerade 135
ISO/OSI-Modell 43

J

Jitter 61

K

Kanalcodierung 46
Kartesisches Koordinatensystem 89
kaufmännischer Energiefluss 18
Kerckhoffs' Prinzip 225
Klasse 34
Klimawandel 22
Knoten 188
KNX 168
Koeffizient 133
Kollision 46
Kollisionsbehandlung 47
Kollisionsvermeidung 46
Kommunikationsnetz 43
Kommunikationsstandard 43
Komposition 34
Koordinatenreferenzsystem 88
Koordinatensystem 86
Kraftwerkseinsatzplanung 147
Kryptografie 224
Kurventyp 98
Kyotoprotokoll 23

L

Lastdeckung 13
Lastflussberechnung 78
Leitungscodierung 45
Leitwarte 155
Level 123
libcimpp 85
Liberalisierung 19
lineare Funktion 133
lineare Optimierung 133
lineare Relaxation 145
Linie 172
LonMark 178
LonTalk 175
LonWorks 175

M

M/490 64
MaKoChain 209

Managementebene 55
Manchester-Verfahren 45
Map3D 93
Marktgebiet 13
Marktkommunikation 203
Maschine Learning 42
Master 57
Medienkontrolle 46
Merit Order 17
Messstellenbetreiber 196
Metadaten 32, 97
Metamodell 188
Metrik 101
Microgrid 2
Microsoft 188
MiFID 212
MiFIR 212
Mike Markulla 175
Modbus 57
Model Compiler 192
Modelio 37
Modellierungssprache 137
Modellparameter 142
Modicon 57
MQTT 191
Multicast 47
Multiplizität 35
MySQL 40
MZ-Automation 161

N

Network Variable Selector 178
Netzfrequenz 13
Netzhierarchie 15
Netzkopplungselement 45
Netzsegment 172
Netztopologie 45
Netzwerkvariable 178
Neuron C 175
Neuron-Chip 175
Neuron-ID 177
Newron Systems 179
Node Group 163
Nominierungsmanagement 209
Norm 61
Normungsroadmap 62
n-Polyeder 135

O
Object Group 157
Objekt 35
Objektorientierung 34, 78
Objekttyp 183
OFFIS 2
OPC UA 187
Open62541 193
OpenDNP3 158
OpenGridMap 112
OpenIEC61850 166
openKonsequenz 83
OpenMUC 161
OpenTSDB 103
Operations Research 133
Optimierungsmodell 133
Optimierungsproblem 133
Oracle DB 40
OSGeoLive 88
Overpass API 111
Overpass-Turbo 111

P
Paritätsbit 46
Peakload 21
Persona LLC 101
pgAdmin 88
Point 157
Polling 157
PONTON X/P 208
Portfoliomanagement 20
POST 52
PostGIS 87
PostgreSQL 40, 87
Powerline 201
Powerlink 60
Power-to-X 26
Predictive P-persistent CSMA 176
Primärenergieträger 10
Primärtechnik 154
Prioritätsklasse 157
Profibus 59
Profinet 60
Proof-of-Stake 209
Proof-of-Work 208
Prosument 195
Prozessbus 155
Pseudotransportschicht 156

- Publish/Subscribe 48
Punkt-zu-Punkt-Verbindung 45
PyCIM 85
Pyomo 137
Physikalisch-technischer Energiefluss 15
Python 42
- Q**
QGis 93
- R**
räumliche Datenbank 87
Rampe 145
Raumautomation 168
Redispatch 22
Redundanz 224
Regelleistung 16
Regelzone 13, 16
Register 58
relationales Datenbanksystem 40
REMIT 212
Repeater 45
Residuen 123
Restkomponente 123
Retention 102
Ringtopologie 45
RJ-45 50
Rohrleitung 12
Rollenmodell 203
Routing 47, 51
RS-232 50
RS-485 50
- S**
Saisonalität 123
Scaling Factor 178
Scan 160
Schaltanlage 154
Scheduling 61
Schwarzfall 221
Schwellwertproblem 145
Segment 205
Sektorenkopplung 26
Sekundärtechnik 154
Sequenzdiagramm 203
Sercos III 60
- Serialisieren 38
SHA 225
Sicherungsschicht 45
Siemens 59
Signalpegel 45
Simplex 135
Simulationsmodell 133
Single-Point-of-Failure 224
Sitzungsschicht 48
Slave 57
SMARD 109
Smart Building 167
Smart Grid 2, 27
Smarthome 167
Social Engineering 222
SOCKS 200
Solver 136
Spannung 11
Spannungshaltung 17
Sparx Systems 37
Spezialisierung 36
Sphärisches Koordinatensystem 89
Spiegelomotie 62
Spontaneous 160
SQL-Server 40
Stammdaten 32
Standard 61
Standardlastprofil 15, 121
Static Poll 157
Stationsbus 155
Sterntopologie 45
Stromstärke 11
Subklasse 35
Suchbaum 145
Switch 46
SYMPHONY 137
Synchronisation 61
Syntax 38
Systembaukasten 59
- T**
Tag 39, 101
Take or Pay 147
TCP/IP 51
technisches Versagen 223
Telegramm 171
Tennet 16
Tiefensuche 145

Timeout 48
Token-Passing 59
Topologie 13
Toshiba 175
Traifanwendungsfälle 200
Transceiver 176
Transnet BW 16
Transportschicht 48
Treibhauseffekt 22
Trend 123
Twisted Pair 50
Type Identification 159

Volt 11
Vorsatz 223
W
Web-of-Trust 226
Webservice 52
Wechselstrom 12
Wertgruppe 199
Westnetz GmbH 207
Wireless M-Bus 197
Wireshark 52

U

Unbalanced Communication 58
Unbundling 19
Unicast 47
Unsolicited Message 157

U

Übertragungsfehler 46
Übertragungsmedium 43
Übertragungsnetz 15

V

Variable Structure Qualifier 159
Variation 157
Verdichterstation 12
Vererbung 34, 35
Verfügbarkeit 222
Vermittlungsschicht 47
Verteilnetz 15, 16
Vertraulichkeit 223
Virtualisierung 28
Virtual Link Layer 181
Vollduplex 50

X

X.509 228

Z

20-20-20 Ziele 23
Zeitreihe 95, 97
Zeitreihenanalyse 123
Zeitreihendatenbank 101
Zeitreihenkomponente 123
Zertifikat 225
Zugriffskontrolle 46
zyklische Redundanzprüfung 46