Institut für Kartographie und Geoinformation

Untersuchungen zur Beschreibung, Verwaltung und Auswertung von Benutzer- und Berechtigungsprofilen für objektorientierte Geoinformationssysteme

Inaugural-Dissertation
zur
Erlangung des Grades
Doktor-Ingenieur
(Dr.-Ing.)
der
Hohen Landwirtschaftlichen Fakultät
der
Rheinischen Friedrich-Wilhelms-Universität
zu Bonn

vorgelegt am 9. Mai 2003
von
Dipl.-Inform. René Thiele
aus Bonn

Referent:	Professor Dr.-Ing. D. Morgenstern
Korreferent:	Professor Dr. rer. nat. L. Plümer
Tag der mündlichen Prüfung:	23. Juli 2003

Abriss

Untersuchungen zur Beschreibung, Verwaltung und Auswertung von Benutzer- und Berechtigungsprofilen für objektorientierte Geoinformationssysteme

Zur Beschreibung komplexer, raumbezogener Phänomene sind objektorientierte Datenmodelle besonders gut geeignet. Die natürlichen Strukturen sowie die Zusammenhänge von Realwelt-Objekten können dabei homogen und semantiknah auf entsprechende Modellstrukturen abgebildet werden. Darüber hinaus wird bei der Modellierung von Objekten die Beschreibung von Eigenschaften und Methoden als Einheit betrachtet. Dadurch können die für Geo-Objekte typischen Prozesse und Transaktionen direkt in deren Modellbeschreibung integriert und damit konsistenzerhaltende Schnittstellen definiert werden.
Bei aktuellen Entwicklungen im GIS-Bereich entsteht ein wachsender Bedarf nach gezielter Beschränkung der Zugriffsbefugnisse von Anwendern auf die modellierten Geodaten, entsprechend der jeweiligen Kompetenzen, Informationsbedürfnisse und Vertrauenswürdigkeiten. Dieser Bedarf entsteht dadurch, dass im Geodaten-Bereich immer größere Datenmengen mit völlig unterschiedlichen fachlichen Ausprägungen und Zusammenhängen über einheitliche Anwendungsschnittstellen einer Vielzahl von Benutzern zur Verfügung gestellt werden sollen. Dadurch kann die Effizienz der Modelle im Sinne von Konsistenz, Verfügbarkeit und Aktualität erheblich gesteigert werden. Gleichzeitig entsteht aber die Gefahr, dass Benutzer bewusst oder unbewusst Eigenschaften von Geo-Objekten verändern oder einsehen können, die nicht in Ihren Zuständigkeitsbereich oder zu den ihnen zugestandenen Wissensbedürfnissen gehören. Dies stellt insbesondere dann ein Problem dar, wenn durch ein GIS rechtlich oder wirtschaftlich relevante Informationen verarbeitet werden.
Die gängigen Zugriffs- und Berechtigungskonzepte für Datenbanken sind im allgemeinen auf die Eigenschaften relationaler Datenmodelle zugeschnitten. Dabei nimmt die Beschreibung von Sichten eine zentrale Position innerhalb dieser Konzepte ein. Auf der Basis von Anfragen werden aus vorhandenen Relationen neue Relationen abgeleitet und für ausgewählte Benutzer sichtbar bzw. zugreifbar gemacht.
Für objektorientierte Modelle raumbezogener Informationen muss die Sicht als Mittel der Zugriffsbeschränkung neu interpretiert werden. Insbesondere das Konzept der Spezialisierung durch Vererbung, der Aufbau komplexer Strukturen durch Aggregationen sowie die Eigenschaft der Objektidentität implizieren eine veränderte Semantik von Modellsichten und stellen besondere Anforderungen an Berechtigungsstrukturen. In Verbindung mit den typischen Eigenschaften des Raumbezugs entstehen Zugriffskriterien, die durch klassische Sichten-

und Berechtigungskonzepte nicht hinreichend abgedeckt werden. Die Ergebnisse von Anfragen an objektorientierte Modelle sind nicht Mengen neu strukturierter Tupel, sondern Geo-Objekte innerhalb eines Semantikraums, die in einem räumlichen und fachlichen Kontext erscheinen.
In der vorliegenden Arbeit wird ein Berechtigungskonzept für Geodaten entworfen, mit dem aus modellkonformen Beschreibungen von Zugriffsrechten konsistente und abgeschlossene Teilmodelle abgeleitet werden können. Die Zusammenhänge und Abhängigkeiten von Objekten oder Teilobjekten werden aus der Modellbeschreibung erkannt und aufgedeckt.
Benutzer werden in diesem Entwurf als Instanzen entsprechender Benutzerklassen definiert, die durch Vererbung Zugriffs-Privilegien von Basisklassen erhalten und diese zu Profilen aggregieren. Dadurch entsteht eine Berechtigungshierarchie, in die jede Benutzerklasse eingeordnet werden kann. Die möglichen Formen der Objektzugriffe werden im Sinne der Datenkapselung durch die öffentlichen Klassenmethoden der entsprechenden Teilschemata festgelegt. Mithilfe einer geeigneten räumlichen Anfragesprache können objektorientierte Sichten innerhalb eines gegebenen Semantikraums ermittelt und dargestellt werden. Aus dem entworfenen Berechtigungskonzept wird eine generische Zugriffsarchitektur abgeleitet, die als Implementierungsbasis einer Nutzer- und Rechteverwaltung dient.

Abstract

Studies of description, administration and evaluation of user- and authorisation-profiles for object-oriented geographical information systems.

Object-oriented data models are particularly suitable for the description of complex structured spatial phenomenons. Natural structures as well as relationships between realworld-objects can be mapped homogeneously and according to semantics to modelstructures. Moreover properties and methods of objects are defined and handled similarly. Thus consistent interfaces can be defined by integrating characteristic processes and transactions of spatial objects into the model-definitions.
In current GIS developments selective restriction of user rights concerning the access to spatial data, according to his competence, information demand and trustability is a growing demand. This requirement is based on the growing amount of GIS-relevant data with quite different thematic characteristics and relationships, which should be accessible for a multiplicity of different users by using unified interfaces. Thus the efficiency of data models can be improved in terms of consistency, availability and timeliness. Contrariwise exists the risk of consciously or unconsciously exceeding of competence by reading or modifying properties of confidential spatial data. In particular when a concerned GIS handles geographic information with relevance in terms of law or economics this becomes a problem.
Popular authorisation concepts for databases are generally arranged in terms of relational data models. Thereby the definition of views is a central aspect of authorisation. Based on relational queries, new relations are derived from existing ones and become accessible for selected users.
The definition of views as a main concept for the declaration of access rights must be reconsidered in terms of object-oriented spatial data models. In particular the paradigm of specialisation by inheritance, the modeling of complex structures by the use of aggregation as well as the paradigm of object identities imply different semantics of views and special demands for appropriate authorisation concepts. Considering the characteristics of spatial data, the required criteria of authorisation are handled insufficiently by any distinguished authorisation model. The results of queries to object-oriented models are not sets of restructured relational tuples but geo-objects in a semantic space, that is embedded in a spatial and technical context.
In the present treatise an authorisation concept for geodata is designed, that enables the derivation of consistent and closed submodels by description of access rights compliant to the model. Relationships and dependencies between objects or parts of objects are recognized and recovered by the use of model

description.
In this draft users are defined as instances of according user-classes. User-classes derive access-rights from their base-classes and aggregate them to profiles. Thereby user-classes are ordered in the resulting authorisation-hierarchy. The different kinds of access to object-data are specified by public class-methods of the accessible subscheme, in terms of information hiding and encapsulating. Object-oriented views of a semantic space can be derived and presented by the use of an appropriate spatial query language.
A generic access architecture is derived from the proposed authorisation concept and is the basis for an implementation of a user- and authorisation-structure.

Inhaltsverzeichnis

Kapitel 1

Einleitung

In kaum einem anderen Anwendungsbereich der Datenbank- und Informationstechnologie werden derart komplex strukturierte Daten in vergleichbar großen Mengen gespeichert, wie bei der Verarbeitung raumbezogener Informationen mittels Geoinformationssystemen (GIS). Dieser hoch spezialisierte Typ von Informationssystemen dient im allgemeinen der Erfassung, Verwaltung, Auswertung und Bereitstellung von Informationen über Objekte der realen Welt, die in einem gegebenen thematischen Kontext auf der Grundlage eines gewählten oder vorgegebenen Datenmodells beschrieben und interpretiert werden. Bei diesem Vorgang werden die Realweltobjekte in geeigneter Weise abstrahiert und in die Strukturen des jeweiligen GIS abgebildet. Dabei entstehen Modellobjekte, die im allgemeinen sowohl alphanumerische, als auch räumlich-geographische Eigenschaften besitzen und darüber hinaus vielfältige fachliche und topologische Beziehungen untereinander aufweisen. Diese Modellobjekte werden aufgrund ihres Raumbezugs auch als Geo-Objekte bezeichnet. Neben den Eigenschaften raumbezogener Objekte können mit geeigneten Modellen auch zeitabhängige Vorgänge und Prozesse der Erdoberfläche dargestellt werden. Geo-Objekte charakterisieren damit ausgewählter Aspekte bestimmter raumbezogener Phänomene unter Berücksichtigung der jeweiligen Aufgaben und Interessen der Anwender eines GIS. Die Menge aller Abstraktionen eines Informationssystems definiert ein Modell des gewählten Realwelt-Ausschnitts. Es ist durchaus möglich und auch beabsichtigt, dass identische Phänomene der realen Welt in zwei Geoinformationssystemen mit unterschiedlichen Eigenschaften und Funktionen beschrieben werden und somit von verschiedenen Modellen repräsentiert werden. Neben dem Ziel einer effizienten und zielgerichteten Modellierung ausschließlich fachlich relevanter Phänomene, kann die Ausblendung von Objektdetails oder ganzer Objekte in einem Datenmodell auch aus der Absicht heraus motiviert sein, spezielle Informationen nur be-

stimmten Benutzern unter bestimmten Rahmenbedingungen zur Verfügung zu stellen und nur bei Erfüllung weiterer einschränkender Kriterien, diese auch zur Bearbeitung den Benutzern anzubieten. Eine wesentliche Voraussetzung zur Realisierung von benutzerabhängigen Zugriffsbeschränkungen für raumbezogene Daten ist die Existenz eines geeigneten Berechtigungskonzeptes, das auf die Besonderheiten raumbezogener Datenmodelle abgestimmt ist.
Der steigende Bedarf an derartigen Berechtigungskonzepten für GIS wird durch die aktuellen Bestrebungen und Entwicklungen im GIS-Umfeld verdeutlicht. Zukünftig sollen raumbezogene Informationen vermehrt über vereinheitlichte Schnittstellen möglichst großen Kreisen von Anwendern und Interessenten angeboten werden. Dabei sollen Daten und Dienste über sogenannte Portale außerhalb einer Systemumgebung verfügbar gemacht werden. Diese Portale unterstützen standardisierte Austausch- und Kommunikationsprotokolle und eröffnen Anwendern somit die Möglichkeit, verschiedene und (weltweit) verteilte Datenquellen sowie Dienste verschiedener Anbieter zu nutzen, ohne dabei die Systemarchitektur hinter den Portalen kennen und unterstützen zu müssen. Das Interesse hinter diesen Konzepten ist im wesentlichen auf die erfolgreiche Vermarktung von Geodaten ausgerichtet. Aber auch effizientere und kürzere Kommunikationswege und Arbeitszyklen bei verteilten GIS-Projekten sowie konsistente und redundanzfreie Datenhaltung werden mit der Etablierung *Web*-basierter Technologien im GIS-Bereich angestrebt. In zunehmendem Maße wird der Wert von Geoinformationen und den darauf abgestimmten Systemen auch für kommerzielle Einsatzgebiete erkannt und gefördert. So ergeben sich Anwendungsmöglichkeiten, unter Anderem in den Bereichen *Facilities Management*, Logistik, Funknetzplanung und *Location Based Services*. Dazu werden Daten unterschiedlicher räumlicher und fachlicher Herkunft zusammengeführt und nach thematischen Aspekten ausgewertet. Dies stellt nicht zuletzt hohe Ansprüche an die Aktualität der verwendeten Basisdaten, was durch die Ansätze *Web*-basierter Lösungen mit *Online*-Zugriffen unterstützt wird. Mit den Möglichkeiten des Zugangs zu Informationen bzw. zu Informationssystemen, steigt grundsätzlich auch die Gefahr des Missbrauchs der zugänglichen Daten oder Dienste. Dies gilt auch - oder insbesondere - für Geoinformationen. Dabei können Informationen ausspioniert oder in unzulässiger Weise verfälscht und zerstört werden. Planungs- und kostenrelevante Grenzverläufe, Informationen über die räumliche Lage bestimmter Objekte oder die Eigentümernachweise von Liegenschaften können bei raumbezogenen Informationssystemen von Missbrauch und Manipulation betroffen sein. Insbesondere die personenbezogenen Daten öffentlicher Stellen, die in großen Mengen in der Kataster- und Vermessungsverwaltung, der Stadtplanung und Raumordnung oder auch im Umwelt- und Verkehrswesen verarbeitet werden, unterliegen besonderen Schutzansprüchen, die nicht zuletzt verfassungs- und datenschutzrechtlich motiviert sind. Im allgemeinen können die schadhaften Auswirkungen

von Angriffen gegen IT-Systeme nach drei Aspekten unterschieden werden:

- **Verletzung von Persönlichkeitsrechten** - Durch unzulässige Kenntnis, Vervielfältigung oder Weitergabe personenbezogener Daten werden Persönlichkeitsrechte, wie sie bereits im Grundgesetz verankert sind, massiv verletzt. Das Recht auf „informationelle Selbstbestimmung" ist in den Datenschutzgesetzen des Bundes und der Länder festgehalten. Mit dem „Volkszählungsurteil" von 1983 fand das Prinzip der *Zweckgebundenheit der Daten* Einzug in datenschutzrechtliche Bestimmungen.
- **Wirtschaftlicher Schaden** - Durch unbefugten Zugang zu wertvollen Informationen - wie zum Beispiel Marktanalysen, Kundenprofile, Konstuktionspläne - verschaffen sich Angreifer Wettbewerbsvorteile. Gleichzeitig entstehen dem rechtmäßigen Eigentümer der betroffenen Daten wirtschaftliche Nachteile, wenn er am Zugang zu den Daten gehindert wird oder diese durch einen Angreifer verfälscht wurden.
- **Personen- oder Sachschaden** - Angriffe auf besonders sensible Daten können unmittelbare Auswirkungen auf betroffene Realweltobjekte oder Personen haben. Insbesondere die Behinderung der Verfügbarkeit von Informationen und die Verfälschung von Daten stellen hohe Risikopotenziale dar. Beispielsweise die Gefährdung des Flugverkehrs und der Sicherheit von Passagieren durch fehlende, fehlerhafte oder verzögerte Information über die Positionen von Flugzeugen in den Informationssytemen der Flugsicherung. Eine vergleichbare Sicherheitsrelevanz hat der rechtzeitige Zugang zu fehlerfreien und vollständigen Wetterdaten für die Hochseeschifffahrt.

Zur Vermeidung der genannten Verletzungen bedarf es ausreichender Sicherheitsstrategien und Abwehrmechanismen. Ein zentrales Problem ist hierbei die Festlegung einer Zugriffs- und Berechtigungsstruktur, deren Beschreibung und Auswertung im Sinne der hinterlegten Berechtigungsprofile den Gegebenheiten moderner GIS-Architekturen gerecht werden muss. Die Beschreibung von Berechtigungsprofilen muss insbesondere die Semantik und den fachlichen Kontext der zu schützenden Daten berücksichtigen. Die im Datenmodell explizit und implizit formulierten Zusammenhänge und Abhängigkeiten zwischen Geo-Objekten müssen bei der Erteilung und Auswertung von Zugriffsrechten mit einbezogen werden, um somit die Durchsetzbarkeit von Zugriffsbeschränkungen überwachen und die Konsistenz der Daten gewährleisten zu können. Der Entwurf eines solchen Berechtigungskonzeptes ist der zentrale Gegenstand und die Zielsetzung der vorliegenden Arbeit.
Der Entwurf eines Berechtigungskonzeptes stellt für Geoinformationssysteme

eine besondere Herausforderung dar: Die Anwender eines GIS haben in der Regel unterschiedliche und sehr spezielle Aufgaben, Interessen, Qualifikationen und Zuständigkeiten und müssen mit entsprechenden Kompetenzen gegenüber dem System ausgestattet werden. Dabei muss ein Benutzer nicht selten auch auf Daten - direkt oder indirekt - zugreifen, die zwar jenseits seines Zuständigkeitsbereichs liegen, für seine weitere Arbeit aber benötigt werden. Im einfachsten Fall handelt es sich dabei um Basisdaten, die nur in bestimmten Ausschnitten und dabei ausschließlich lesend zugegriffen werden dürfen. Neben der Frage, welche Geo-Objekte von einem Benutzer verwendet werden dürfen, ist also auch entscheidend, was der Benutzer mit diesen Objekten machen darf. Bei klassischen Datenbankarchitekturen wird diese Fragestellung mit der Erteilung von *lesenden* oder *schreibenden* Zugriffsrechten hinreichend beantwortet. Im Fall von GIS muss dieses Problem jedoch deutlich differenzierter betrachtet werden. Aufgrund der Strukturen raumbezogener Daten, sind diese - und damit auch entsprechende Zugriffsrechte - in hohem Maße kontextabhängig. Dies zeigt sich beispielsweise dadurch, dass eine Veränderung punktförmiger Geo-Objekte unmittelbare Auswirkungen auf topologisch abhängige linienförmige Geo-Objekte hat. Diese können ihrerseits für flächenförmige Geo-Objekte bestimmend sein. Aus fachlicher Sicht hat ein verändernder Zugriff auf ein Flurstück mindestens Einfluss auf benachbarte Flurstücke. Darüber hinaus kann beispielsweise eine Flurstücksgrenze gleichzeitig Flurgrenze und bei weiterer Spezialisierung auch Gemarkungsgrenze sein. In diesem Fall muss das entsprechende Geo-Objekt deutlich stärkeren Restriktionen unterliegen, als dies für reine Flurstücksgrenzen erforderlich ist. Geo-Objekte werden in der Regel erst durch ihre fachliche Spezialisierung zu sensiblen Daten im Sinne der Sicherheit. So kann die Existenz und geometrische Ausprägung eines Gebäudes ein allgemein bekannter und visuell überprüfbarer Sachverhalt sein, während seine spezielle - beispielsweise militärische - Nutzung besonderen Geheimhaltungsforderungen unterliegt. Mit der bei GIS durchaus gängigen Technik der objektorientierten Modellierung wird die fachliche Spezialisierung von Geo-Objekten durch Vererbung zwischen den beschreibenden Klassen definiert. Dabei werden Attribute und Methoden von Basisklassen an die erbenden Klassen weitergegeben und gegebenfalls durch weitere Attribute oder Methoden, die auch überschreibenden Charakter haben können, ergänzt. Die Vererbung zwischen den beschreibenden Klassen von Geo-Objekten impliziert somit eine Hierarchie von Objekteigenschaften und Zugriffsmethoden. Der fachliche Zusammenhang zwischen Geo-Objekten wird im Datenmodell durch Beziehungen beschrieben, die die Instanzen eines vorgegebenen Beziehungstyps sind. Diese legen jeweils die beteiligten Klassen, die fachliche Interpretation und die zulässigen Kardinalitäten von Beziehungen dieses Typs fest. Mit den objektorientierten Modellierungstechniken werden semantische Datenmodelle beschrieben, mit denen die fachliche Struktur von Geo-Objekten, im

Sinne sogenannter *Business Objects* oder Fachobjekte, hervorgehoben und in gleicher Weise die verwendeten, internen Speicher- und Verwaltungsstrukturen gekapselt werden. Die Kommunikation mit den Daten eines GIS kann damit ausschließlich auf den Austausch von Fachobjekten eingeschränkt werden.
Um eine effiziente und durchgängige Definition und Auswertung von Benutzerrechten zu ermöglichen, muss sich die Beschreibung von Berechtigungsprofilen auf derartige Fachobjekte bzw. auf durch Vererbung implizierte Teilobjekte beziehen und dabei sowohl fachliche, als auch strukturelle Abhängigkeiten zwischen den vorhandenen Objekten berücksichtigen. Jede Benutzersicht soll somit die Semantik eines (Teil-)Datenmodells im Sinne der sichtbaren Klassenarchitektur besitzen. Sicherheitsstrategien, die auf gängigen zugriffsbeschränkenden Berechtigungsverfahren für Datenbanken basieren, erscheinen in diesem Sinne für die Definition von Zugriffsstrukturen auf Geodaten als ungeeignet oder zumindest unzureichend, da diese im wesentlichen eine zeilen- und spaltenweise Ausblendung von Tabelleninhalten realisieren. Grundlage für diese Form der Zugriffsbeschränkung ist die Definition von Sichten mittels geeigneter Anfragesprachen. Für Geoinformationen auf der Basis objektorientierter Datenmodelle stellt sich die Beschreibung von Sichten deutlich schwieriger dar. Insbesondere die Spezialisierung von Klassen und Objekten durch das Konzept der objektorientierten Vererbung sowie die Beschreibung komplexer Objekte durch Aggregationen und Assoziationen müssen sich in einer Sichtenbeschreibung für objektorientierte Datenmodelle wiederfinden und ihre Semantik entsprechend berücksichtigt werden. Die vorliegende Arbeit verfolgt das Ziel, einen alternativen Vorschlag eines Zugriffs- und Berechtigungskonzeptes für Geoinformationssysteme zu erarbeiten, der auf einer modellkonformen Beschreibung von Benutzer- und Berechtigungsprofilen basiert. Der benutzerabhängige Zugriff auf Geo-Objekte soll dabei nach drei Kriterien beschränkbar sein:

1. Der Objektart, definiert durch die Klassenzugehörigkeit eines Geo-Objektes.

2. Den Objektdetails, entsprechend den verfügbaren Zugriffsmethoden des sichtbaren Datenschemas.

3. Den Objekteigenschaften, beschrieben durch räumliche und fachliche Attribute und Beziehungen.

Mit dem Aufstellen einer objektorientierten Zugriffsarchitektur soll eine effiziente und konsistenzerhaltende Rechteverwaltung realisiert werden, die den besonderen Eigenschaften raumbezogener Datenstrukturen Rechnung trägt.
Die vorliegende Arbeit ist inhaltlich in folgende Abschnitte gegliedert:

Teil I: Motivation, Hintergründe und wissenschaftliche Grundlagen.

Kapitel 2: Geoinformationssysteme werden in typischen Anwendungsgebieten exemplarisch, nach Aspekten ihres Sicherheitsbedarfs dargestellt. Dabei wird der Einsatz von GIS im öffentlichen Sektor von dem im industriellen Bereich unterschieden. Aufgrund der besonderen Anforderungen hinsichtlich der Sicherheitsanforderungen werden netzbasierte Anwendungen für GIS gesondert untersucht.
Kapitel 3: Die Modellierung raumbezogener Informationen wird an dieser Stelle genauer untersucht. Dabei werden insbesondere die verschiedenen Modellierungstechniken für Geodaten dargestellt. Die Vorteile objektorientierter Beschreibungen für Geoinformationen werden dabei besonders hervorgehoben. Diese bilden in einer abstrahierten Form zugleich die Modellierungsbasis für die weiteren Untersuchungen.
Kapitel 4: Dieser Abschnitt soll dem Leser das Thema Informationssicherheit in einer allgemeingültigen Form näherbringen. Dabei werden konkrete Probleme, theoretische Lösungsansätze und standardisierte Bewertungskriterien bezüglich der Sicherheit von Daten und Informationssystemen erläutert.

Teil II: Entwicklung und Auswertung eines neuen Ansatzes.

Kapitel 5: Vor der Erarbeitung einer Zugriffsarchitektur für Geodaten, sollen in diesem Abschnitt die konzeptionellen Besonderheiten objektorientierter raumbezogener Datenmodelle, hinsichtlich der Beschreibung von Berechtigungsstrukturen, herausgestellt werden. Dazu werden die objektorientierten Modellierungstechniken im Einzelnen charakterisiert und die damit verbundenen Abhängigkeiten zwischen modellierten Geo-Objekten näher untersucht.
Kapitel 6: Die Erläuterung der grundlegenden Ideen und Ansätze einer objektorientierten, modellkonformen Zugriffs- und Berechtigungsstrategie steht im Mittelpunkt dieses Abschnitts. Dabei werden die Eigenschaften objektorientierter Geodatenmodelle, insbesondere unter Berücksichtigung von Beziehungs- und Vererbungsstrukturen, zur effizienten Beschreibung von Benutzer- und Berechtigungsprofilen verwendet. Die Untersuchungen führen zur formalen Beschreibung benutzerabhängiger Semantikräume und Teilmodelle.
Kapitel 7: Methoden und Strategien zur Filterung von Objektdetails werden mit dem Ziel der Definition einschränkender Sichten dargestellt. Dabei werden insbesondere die Auswirkungen der Beschränkung von Semantikräumen auf die Repräsentation von Geo-Objekten und die modellkonforme Darstellung ihres Raumbezugs veranschaulicht.
Kapitel 8: Mithilfe deskriptiver, objektorientierter Anfragesprachen unter Verwendung von benutzer- und thememabhängigen Teilschemata werden fachliche Sichten auf Geoinformationssysteme beschrieben, die die Zugriffsrechte

der anfragenden Benutzer berücksichtigen. Dabei wird die Definition und Auswertung fachlich motivierter Arbeitsbereiche für Benutzer mit eingeschränkten Zugriffsrechten untersucht.
Kapitel 9: Der Entwurf einer Zugriffsarchitektur beschreibt einen Ansatz zur Implementierung des entworfenen Berechtigungskonzeptes. Dabei wird ein Klassendesign vorgestellt, dessen externe Schnittstellen den Benutzern einheitliche Zugriffe auf die Daten und Dienste eines GIS ermöglichen und dabei Abgeschlossenheit der präsentierten Teilmodelle und die konsistenzerhaltenden Eigenschaften der möglichen Transaktionen gewährleisten.
Kapitel 10: Mit einer Zusammenfassung der gewonnenen Ergebnisse wird die Arbeit abgeschlossen.

Kapitel 2

GIS unter dem Aspekt der Sicherheit

Die zentralen Aufgaben von Geoinformationssystemen (**GIS**) liegen in der Erfassung, Aufbereitung, Verwaltung, Bereitstellung und Präsentation von Geodaten Diese setzen sich in der Regel aus fachlichen und räumlichen Informationen über Objekte zusammen. Die fachlichen Aspekte der abgebildeten Geo-Objekte dienen der alphanumerischen Beschreibungen von Objekteigenschaften vor dem Hintergrund eines thematischen Kontextes. Die räumlichen Aspekte repräsentieren die räumliche Ausprägung von identifizierten Objekten der Erdoberfläche durch die Verwendung geometrischer und graphischer Modelle in einem geeigneten Bezugssystem, das eine für die zu bewältigende Aufgabenstellung notwendige Metrik und Topologie besitzt. Das wachsende Interesse an Geodaten aus verschiedenen wirtschaftlichen, öffentlichen und privaten Bereichen, z.B. Versicherungen, Logistikunternehmen, Mobilfunkanbieter, Vermessungs- und Planungsämter, Feuerwehr, Polizei und Rettungsdienste, sowie die gleichzeitig zunehmende Verbreitung und Nutzung von Datenbank-, Intranet und Internet-Technologien führt zu verstärkten Bestrebungen der Besitzer solcher Daten, diese einem möglichst goßen Anwenderkreis, unter Nutzung dieser Technologien, zugänglich zu machen. Aktuelle Forschungen und Entwicklungen im GIS-Sektor befassen sich demzufolge intensiv mit Fragen der schnellen, zuverlässigen und gezielten Verfügbarkeit von Geodaten auf Anfrage (*on demand*) unter dem Anspruch von Konsistenz, Redundanzfreiheit und Aktualität. Gleichzeitig sollen aber nicht alle Benutzer über die gleichen Zugriffsrechte verfügen. Bestimmte Informationen sollen nur von bestimmten Benutzern einsehbar oder veränderbar sein.
Geoinformationssysteme, die potenziell Geodaten unterschiedlicher sensibilität verwalten, müssen in der Lage sein, sich gegen die Gefahren unbefugter Ein-

sichtnahme und die Veränderung sicherheitskritischer Daten zu schützen.
Die Gefahren für die Sicherheit von Geodaten sind insbesondere dann schwierig zu behandeln, wenn anonymisierte Benutzer über das Internet auf die Informationssysteme zugreifen. Aber auch interne Benutzer eines Informationssystems, zum Beispiel innerhalb einer Behörde oder eines Betriebs, unterscheiden sich in ihrer Kompetenz, ihren Zuständigkeiten, den Qualifikationen und der Vertrauenswürdigkeit. Daher müssen diese mit unterschiedlichen Privilegien hinsichtlich des Zugriffs auf interne Datenbestände ausgestattet werden.
Die Sicherheitsrelevanz von Informationen wird im wesentlichen von den Gefahren bestimmt, die von einer missbräuchlichen Behandlung der Daten ausgehen. Enthalten die Informationen ausschließlich Sachdaten, so ergeben sich mögliche sicherheitsrelevante Gefahren aus dem fachlichen Kontext, in dem die Daten interpretiert werden können. Dabei ist entscheidend, welche Vorteile ein Angreifer aus der Kenntnis oder durch die Manipulation von Daten erzielt, welche Nachteile dem Datenanbieter durch einen Angriff auf die Daten entstehen und welche direkten oder indirekten Schäden für Dritte auftreten können. Werden von einem Informationssystem darüber hinaus personenbezogene Daten verwaltet, so wird die Sicherheitsrelevanz der Daten noch um juristische Aspekte ergänzt: Verpflichtungen zum Schutz personenbezogener Daten vor unbefugten Zugriffen lassen sich unmittelbar aus den Persönlichkeitsrechten des Grundgesetzes ableiten. Dort sind in *Art.2(1)GG* die freie Entfaltung der Persönlichkeit und in *Art.1(1)GG* der Schutz der Menschenwürde festgeschrieben. Das „Volkszählungsurteil" [13] vom 15.12.1983 lieferte eine verfassungsrechtliche Interpretation dieser Artikel bezüglich der Erfassung und Speicherung personenbezogener Daten und führte das Prinzip der „Zweckgebundenheit personenbezogener Daten" ein. Danach obliegt jedem Einzelnen das grundsätzliche Recht auf informationelle Selbstbestimmung und damit die Befugnis, selbst über die Preigabe und den Verwendungszweck seiner persönlichen Daten bestimmen zu können. Darüber hinaus wird die Notwendigkeit von organisatorischen und verfahrenstechnischen Sicherungselementen betont. Gesetzliche Festlegungen zum Schutz personenbezogener Daten und der Einführung geeigneter organisatorischer und technischer Maßnahmen werden in Abhängigkeit vom Geltungsbereich im Bundesdatenschutzgesetz [12] oder den Datenschutzgesetzen der Länder - z.B.: [11] - getroffen. Bei GI-Systemen kommen drei Aspekte zusammen, die hinsichtlich Ihrer Sicherheitsrelevanz untersucht werden müssen. Zunächst enthalten Geoinformationen häufig komplex strukturierte Sachdaten, die in Abhängigkeit vom Verwendungszweck (z.B. Geomarketing) hohen subjektiven Wert haben. Diese werden nicht selten um sensible, personenbezogene Informationen ergänzt oder mit diesen verknüpft. Hinzu kommt bei Geoinformationen der sicherheitskritische Aspekt des Raumbezugs, durch den Sachdaten oder Personen mit Positionen, Lagebeziehungen, Ausdehnungen oder einer Geometrie assoziiert werden. Umgekehrt ermöglicht der Zugriff auf den

Raumbezug, Positionen oder Bereiche der Erdoberfläche mit sachlichen oder personenbezogenen Informationen in Verbindung zu bringen. Die Sicherheitsrelevanz des Raumbezugs in GIS kann an verschiedenen Beispielen verdeutlicht werden - dazu gehören die allgegenwärtige Präsenz von Überwachungskameras (Kaufhäuser, Banken, Bahnhöfe,...) und der verbreitete und zunehmend zivile Einsatz des *Global Positioning Systems* (GPS), durch die sich nahezu unbegrenzte Überwachungsmöglichkeiten von Personen eröffnen. Ebenso sicherheitskritisch sind die möglichen Konsequenzen aus unbefugten Zugriffen oder fehlerhaften Positionsangaben in den Sicherheits- und Informationssystemen des neugeordneten europäischen Luftraums. Auch und insbesondere für GIS muss also:

- der Sicherheitsbedarf und die Sicherheitsrisiken ermittelt und analysiert,
- Kriterien für den sicheren Einsatz des Systems festgelegt,
- Konzepte zur Vergabe und Verwaltung von Berechtigungen entworfen und
- geeignete technische und organisatorische Maßnahmen zur vertraulichen und integeren Behandlung der Daten ergriffen werden.

Die Daten eines Informationssystems, insbesondere eines GIS, können als sicherheitskritisch eingestuft werden, wenn deren missbräuchliche Behandlung in Form von unbefugter Einsicht oder Weitergabe, unzulässiger Modifikation oder unsachgemäßer, zweckentfremdeter Behandlung

- zur Verletzung von Persönlichkeitsrechten,
- zu materiellem, finanzellem oder wirtschaftlichem Schaden oder
- zur Gefährdung der Gesundheit oder des Lebens von Personen

führen kann. Die Anwendungsbereiche in denen GIS zum Einsatz kommen, finden eine ständige Erweiterung und lassen sich kaum noch zusammenfassend darstellen. Daher sollen im Folgenden nur einige typische Anwendungsgebiete von GIS nach Aspekten der Sicherheitsrelevanz der verarbeiteten Daten charakterisiert werden. Dabei werden die Anwendungsbereiche öffentliche Verwaltung, Privatwirtschaft bzw. Industrie und Internet-basierte GIS unterschieden.

2.1 GIS in der öffentlichen Verwaltung

Die bei weitem größten Anbieter und Nutzer von Geodaten sind öffentliche Einrichtungen von Bund, Ländern und Gemeinden. Diese stellen Geobasisdaten zur Verfügen, die z.B. Informationen über die Topographie der Landschaft (ATKIS®) oder bezüglich der Liegenschaften eines räumlichen Ausschnitts (ALK, ALB) enthalten. Darüber hinaus produzieren die entsprechenden Stellen öffentlicher Verwaltungen aber auch raumbezogene Fachdaten wie z. B. Flächennutzungspläne, Bebauungspläne, Wander- oder Straßenkarten. Unter Aspekten der Sicherheit gebührt Geodaten, die unter anderem von Katasterämtern, Landesvermessungsämtern oder Amtsgerichten erfasst, verwaltet und ausgewertet werden, besondere Beachtung, da diese

- als öffentliche Register grundsätzlich der Öffentlichkeit zugänglich sein müssen,
- zur Erfüllung öffentlicher Aufgaben und in Verantwortung gegenüber dem Bürger eine hohe Verfügbarkeit, Zuverlässigkeit und Integrität besitzen müssen und
- in Verknüpfung mit personenbezogenen Daten Risiken der Verletzung von Persönlichkeitsrechten bergen.

Die sich daraus ergebenden Probleme und Fragestellungen hinsichtlich der Datenhaltung werden durch die Diskussionen im Rahmen der Liegenschaftsverwaltung verdeutlicht. Der hohe Bedarf der Liegenschaftsverwaltung an strukturierten Zugriffsbeschränkungen und einem geeigneten Management von Benutzern und Berechtigungen soll hier kurz begründet werden:

Das Grundbuch wird von den Amtsgerichten entsprechend den Vorgaben der Grundbuchordnung geführt. In ihm werden die rechtlich erforderlichen Angaben über Grundstücke und deren Eigentümern gespeichert. Außerdem verwaltet das Grundbuch Rechte an den Grundstücken - wie Hypotheken und Grundschulden - sowie deren Inhaber. Es ist hierbei durchaus vorstellbar, dass „Unbefugte" über die Grundbucheinträge direkt oder indirekt - z.B. durch Schlussfolgerungen aus verfügbaren Informationen - Einblick in die Vermögensverhältnisse gespeicherter Personen erhalten.
Das Liegenschaftskataster ist ein von den Vermessungs- und Katasterverwaltungen geführtes öffentliches Register zum landesweiten Nachweis, zur Darstellung und zur Beschreibung der Liegenschaften (Flurstücke und Gebäude).

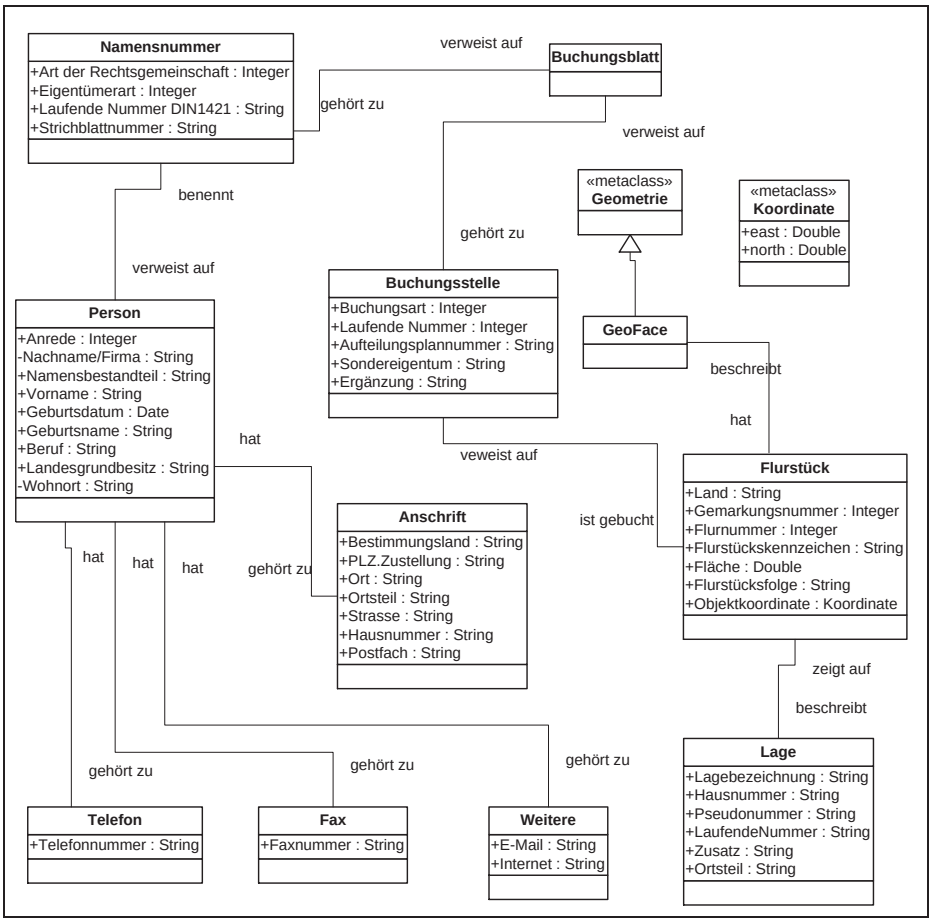

Abbildung 2.1: *Ausschnitt aus dem ALKIS® - Fachdatenschema.*

Darüber hinaus soll es den Anforderungen des Rechtsverkehrs, der Verwaltung und der Wirtschaft gerecht werden. Diese Anforderungen kommen insbesondere aus den Bereichen Landesplanung, Bauleitplanung, Bodenordnung, Umweltschutz und Naturschutz. Das Liegenschaftskataster umfasst die Informationen der Liegenschaftskarte und des Liegenschaftsbuches, die in der Regel in digitaler Form als „Automatisierte Liegenschaftskarte" (ALK) und als „Automatisiertes Liegenschaftsbuch" (ALB) geführt werden. Im Liegenschaftskataster muss zunächst zwischen den reinen Sachdaten und den sicherheitsktitischen personenbezogenen Daten unterschieden werden. Zu den Sachdaten der Liegenschaften gehören die Daten der Grundriss- und der Punktdatei, in denen unter anderem Informationen über Lage, Nutzung und Größe der beschriebenen Objekte enthalten sind. Die sicherheitskritischen personenbezogenen Daten tragen Informationen über Eigentümer und Erbauberechtig-

te wie: Name, Geburtsdatum, Anschriften und Anteils- und Gemeinschaftsverhältnisse. Die Einsicht in die Daten des Liegenschaftskatasters wird nach den Vorgaben des Vermessungs- und Katastergesetzes des jeweiligen Landes geregelt - z.B. nach dem VermKatG NW für Nordrhein-Westfalen. Danach haben zunächst die in dem Gesetz genannten Dienststellen Einsicht in die Daten zur Erfüllung ihrer Aufgaben. Eigentümer, Erbbauberechtigte und Notare haben ein Recht auf Auskunft über die sie betreffenden Liegenschaften. Darüber hinaus können andere Antragsteller durch „Glaubhaftmachen" eines berechtigten Interesses an bestimmten Daten Einsicht in diese erhalten. Dabei unterliegt die Anerkennung eines berechtigten Interesses grundsätzlich einer subjektiven Einschätzung des verantwortlichen Sachbearbeiters. Auch wenn das „Glaubhaftmachen" der Berechtigung des Interesses stärker einzustufen ist und höhere Anforderungen an den Interessenten stellt, als die bloße Darlegung desselben, bleibt dem datenführenden Amt ein großer Ermessensspielraum bei der Erteilung von Einsichten in und Auskünften über die betroffenen Daten. Grundbuch und Liegenschaftskataster verdeutlichen den bei der Speicherung personenbezogener Daten in öffentlichen Registern auftretenden Konflikt zwischen dem öffentlichem Interesse und der gleichzeiten Forderung nach Schutz von Persönlichkeitsrechten. Nach §10 VermKatG NW wird die Schutzwürdigkeit personenbezogener Daten sogar begrenzt, wenn das öffentliche Interesse an den Daten höher zu gewichten ist, als die Rechte eines Einzelnen oder wenn der Schutz der Daten das Prinzip der Verhältnismäßigkeit beim Einrichten technischer und organisatorischer Mechanismen verletzen würde. Ausreichender Datenschutz ist also im derzeitigen Zustand von Grundbuch und Liegenschaftskataster nur bedingt gegeben. Die Konzepte, Methoden und Daten von ALB und ALK sollen zukünftig im ALKIS®-Datenmodell (ALKIS = Amtliches Liegenschaftskataster-Informationssystem) zusammengeführt werden. Abbildung 2.1 zeigt einen Ausschnitt des ALKIS®-Fachdatenschemas. Dabei entstehen erneut besondere Anforderungen an geeignete Benutzer- und Berechtigungsstrukturen. Neben der fachlichen Verschmelzung der Datenmodelle, die weitere direkte Referenzen zwischen sicherheitskritischen und sicherheitsunkritischen Informationen herstellt, umfasst die ALKIS®-Konzeption die Modellierung einer Reihe sicherheitsrelevanter Geschäftsprozesse, mit denen auf die ALKIS®-Daten zugegriffen werden soll. Berechtigungsstrukturen für ALKIS® müssen also in der Lage sein, neben den Zugriffsrechten für die reinen Daten eines GIS auch Zugriffsrechte für Geschäftsprozesse oder deren Teilprozesse zu beschreiben. Geschäftsprozesse können im allgemeinen als Dienste im Rahmen eines Workflowsystems realisiert und damit auf Klassenmethoden zurückgeführt werden. Insofern können Geschäftsprozesse im Sinne objektorientierter Modelle beschrieben werden. Dabei wird insbesondere die aus klassischen Modellen bekannte, strikte Trennung zwischen Daten und Methoden aufgelöst. Bei der Erteilung von Benutzerrechten muss folglich nicht nur der Zugriff auf

spezifische Daten beschränkt, sondern auch die Verfügbarkeit und Ausführung von Methoden, im Sinne von Diensten, eingeschränkt und überwacht werden. Die Entwicklungen und Bestrebungen beim Thema ALKIS® können als repräsentativ für eine ganze Reihe von GIS im öffentlichen Sektor angesehen werden. Dabei wird von all diesen Systemen das für die Sicherheit relevante Ziel verfolgt, durch den Einsatz von Web-Diensten und -Technologien verteilte GIS-Architekturen zu realisieren und somit Arbeitsabläufe zu optimieren und die Informationsbasis durch Kaskadierung der Systeme zu verbessern. Die implementierten Dienste sollen neben Auskunftsfunktionen für Bürger und Interessierte, insbesondere auch Schnittstellen für externe Dienstleister (zum Beispiel ÖbVI) anbieten. Entsprechend den vielseitigen Nutzungsmöglichkeiten dieser Technologie müssen auch Berechtigungskonzepte für verteilte, webbasierte Systeme komplexen Anforderungen genügen.

2.2 Kommerzielle Nutzung von GIS

Auch im industriellen und privatwirtschaftlichen Bereich wird zunehmend der Wert raumbezogener Daten erkannt und genutzt. Geodaten haben insbesondere in Branchen wie Energieversorgung, Versicherungswesen, Logistik, Bauwesen, Mobilfunk und Immobilienverwaltung ihren Nutzen unter Beweis gestellt und werden zu Zwecken der räumlichen Überwachung, der Lage- und Routenplanung, der Auswertung von Standortparametern, der Ortung von Objekten oder der Simulation von Ereignissen eingesetzt. Dabei werden in der Regel, verfügbare Geobasisdaten - z.B. der deutschen Grundkarte 1:5000 (DGK5) - mit erfassten, ermittelten oder erworbenen thematischen Daten überlagert und von hoch spezialisierten Fachapplikationen interpretiert. Betrachtet man diese Anwendungsfelder unter Aspekten der Sicherheit der verwendeten raumbezogenen Informationen, so ist von Belang,

1. zu welchem Zweck die raumbezogenen Informationen verwendet werden,
2. welche Benutzer Zugang zu den Informationen haben und
3. wie sicherheitskritisch die Daten selber einzustufen sind.

Mit dem Verwendungszweck und den damit verbundenen Auswertungsmethoden kann sich der Sicherheitsbedarf gespeicherter Daten erheblich verändern. Relativ unkritische Einzelinformationen können im Kontext der Auswertung des Gesamtsystems hohe Sicherheitsrelevanz erhalten. Ein Beispiel hierfür liefern die Verfahren des sogenannten Geomarketings, die zu Zwecken der Umsatz- und Gewinnsteigerung von Unternehmen Anwendung finden. Dabei werden

raumbezogene Daten, die für die Marketingstrategien des Unternehmens von besonderem Interesse sind, erfasst und gesammelt. Darunter fallen Daten über die regionale Verteilung der Kaufkraft, das Konsumverhalten oder die Erschlossenheit von Regionen im Vertriebsnetz. Die Auswertung der gespeicherten Informationen liefert im allgemeinen sehr präzise und für das Unternehmen wertvolle Kundenprofile und Standortbewertungen, die dem Unternehmen die Einleitung gezielter Maßnahmen ermöglicht. Neben der Möglichkeit, dass die erhaltenen Ergebnisse auch für Mitbewerber bzw. Konkurrenten von hohem Wert und Interesse sein könnten und damit Anreize für potenzielle Angreifer eines solchen Systems geschaffen werden, muss hinterfragt werden, ob die gezielte Auswertung derart unfangreicher, raumbezogener Konsum- und Kundendaten Rückschlüsse auf das Verhalten einzelner Personen ermöglicht und damit eine datenschutzrechtliche Verletzung darstellt.
Im Zusammenhang mit den Benutzern eines GIS in Unternehmen, ist für die Sicherheit der Daten von Bedeutung, welche Personen Zugang zu den Daten haben. Grundsätzlich kann danach unterschieden werden, ob das Unternehmen selber Endanwender der Daten ist und die Zugriffsrechte auf das GIS ausschließlich innerhalb der eigenen Unternehmensstruktur erteilt werden, oder ob die Firma gegenüber Dritten als Anbieter von Daten oder Diensten auftritt und diese - beispielsweise über das Internet - einer goßen Anzahl von Benutzern zugänglich macht. Die Brisanz raumbezogener Informationen wird durch die Diskussion um die Möglichkeit und Zulässigkeit der Ortung von Mobiltelefonen (*Pinpointing*) verdeutlicht. Aufgrund des engen Rasters von Funkstationen, über die sich ein Kunde mit seinen Mobiltelefon bei einem Mobilfunknetz einloggen muss, wird ein Telefon in ausreichend abgedeckten Gebieten von mehreren Stationen gleichzeitig empfangen. Dadurch wird es dem Mobilfunkbetreiber - oder demjenigen der Zugang zu diesen Informationen hat - möglich, das Mobiltelefon und damit dessen Besitzer sehr genau zu lokalisieren (je nach der Größenordnung des Rasters handelt es sich um wenige Meter). Auch wenn diese Technik eigentlich genutzt werden soll, um Mobilfunk-Kunden ortsbezogene Dienstleistungen (*Location Based Services*) anzubieten oder Menschen in Notsituationen schneller helfen zu können, so stellt die Möglichkeit der Ortung von über 30 Mio. Mobilfunkanwendern allein in Deutschland doch einen erheblichen Eingriff in die Privatsphäre dieser Personen dar. Ein vor diesem Hintergrund entstandener Vorschlag für eine EU-Richtlinie ([23]) sieht vor, ausschließlich dem Kunden die Kontrolle über die Lokalisierbarkeit seines Telefons zu übertragen. Allerdings sieht dieser Richtlinienvorschlag auch eine Reihe fragwürdiger Ausnahmen vor: Beispielsweise sollen Rettungsdienste und staatliche Organe wie Polizei, Bundesgrenzschutz und Geheimdienste auf Anforderung Zugriff auf die Daten erhalten können.
Ähnlich sicherheitskritische Möglichkeiten der Nutzung raumbezogener Daten bietet der Einsatz des *Global Positioning Systems* (GPS). Mit geeigneten Sen-

deeinrichtungen ist es technisch möglich, die über GPS ermittelten Daten (Position, Geschwindigkeit, Haltepunkte, Haltezeiten, Fahrtrichtung) an eine weitere Empfangsstation zu übermitteln. Damit sind grundsätzlich Möglichkeiten gegeben, GPS-Empfänger bzw. Fahrzeuge oder Personen, die mit diesen ausgestattet sind, ohne Sichtkontakt zu lokalisieren und zu verfolgen. So können zum Beispiel Speditionen und Logistikunternehmen ihre Fahrzeuge und deren Fahrer überwachen. Autovermietungen und Reiseveranstalter werden in die Lage versetzt, das Verhalten ihrer Kunden hinsichtlich bevorzugter Ziele und Routen, besuchter Restaurants und Hotels oder der Teilnahme an Veranstaltungen analysieren und diese Informationen gewinnbringend - z.B. durch gezielte Werbemaßnahmen - einsetzen zu können. Gleichzeitig kann dabei aber auch eine erhebliche Beschränkung der Persönlichkeitsrechte stattfinden.
Des Weiteren müssen Informationsysteme von Banken, Versicherungen oder Kreditkartenunternehmen erwähnt werden, deren gezielte Auswertungen sicherheitskritische, raumbezogene Informationen über Kunden liefern können. So lässt die Verwendung von Geldautomaten oder Kreditkarten Rückschlüsse auf den Aufenthaltsort und das Konsumverhaltenverhalten des Karteninhabers zu. Trotz der hier gewählten Beispiele beschränken sich die Bedrohungen für privat-wirtschaftlich genutzte raumbezogene Daten keineswegs nur auf mögliche Verletzungen von Persönlichkeitsrechten - diese sind lediglich die offensichtlichsten Gefahren. Die Geodaten enthalten mitunter ausgesprochen sensible Informationen über Lage, Ausdehnung und Nutzung von Geo-Objekten, die nicht jedem Mitarbeiter eines Unternehmens oder Anwender eines GIS zugänglich sein sollen.
Unternehmen, die sich am Markt behaupten müssen, sind in erster Linie an dem wirtschaftlichen Nutzen verfügbarer Daten interessiert. Daten gegen Missbrauch zu Schützen beschränkt im allgemeinen deren Verwendbarkeit und damit auch deren Wert für das Unternehmen. So verwundert es nicht, dass von den Firmen relativ geringer Aufwand betrieben wird, geeignete Sicherheitsmaßnahmen zum Schutz von Informationen zu entwickeln. Wenn Sicherheitsstrategien existieren, dann aus der Motivation heraus, die eigenen Daten und Systeme vor den Angriffen und der Einsichtnahme von Mitbewerbern zu schützen, um die eigenen Wettbewerbsvorteile zu erhalten.

2.3 GIS im Internet

Seit Mitte der 1990er Jahre wurde die Entwicklung der IT-Landschaften maßgeblich vom Internet-Boom gepägt. Mit den ständig wachsenden Anwenderzahlen in den Industrienationen ist das Internet zu einem Breitenmedium geworden, das nicht mehr ausschließlich einem kleinen, elitären Kreis von Fachleuten und Wissenschaftlern zugänglich ist. Es war demnach auch naheliegend,

die bequeme Erreichbarkeit breiter Bevölkerungsschichten durch gezielte Angebote von Informationen und Dienstleitungen über das Internet zu nutzen. Insbesondere in der Geodatenverarbeitung existieren seit einigen Jahren massive Bestrebungen, Daten und webbasierte Applikationen im Internet bereit zu stellen und damit Internet-Lösungen für GIS zu etablieren. Die gängigen Internet-Techniken bieten dem Anbieter von Daten oder Dienstleistungen den Vorteil, Anwender und Interessierte mit relativ geringem Aufwand und Kosten erreichen und mit Informationen versorgen zu können. Auf *Client* Seite (Anwender) reicht dabei meistens ein gängiger *Web-Browser* (z.B. MS Internet Explorer oder Netscape Communicator), um Applikationen über das Internet ausführen oder Daten übertragen zu können. Im Geodatenbereich hat das Internet neben der Nutzung zum schnellen Austausch von Daten zwischen den informationsverarbeitenden Stellen auch für den Aufbau von bürger- oder kundennahen Informationssystemen einen hohen Stellenwert erreicht. Die allgemeine und häufig verwendete Bezeichnung „Internet-GIS" enthält zunächst nur die Information, dass Daten oder Dienste mit Raumbezug Internetfunktionalität nutzen oder bereit stellen. Im Einzelfall kann ein Internet-basierter Geodaten-Dienst durch Zuordnung zu einer oder mehrerer Kategorien [20] spezifiziert werden. Dabei können grundsätzlich fünf Typen von Internet-GIS unterschieden werden:

1. **Geodaten-Server** stellen raumbezogene Daten für den Offline-Betrieb in (de-facto-) Standard Formaten bereit. Die gewünschten Daten bzw. Dateien können vom Anwender mit Methoden des Servers gesucht, ausgewählt und basierend auf Übertragungsprotokollen wie FTP zum Anwender übertragen werden. Die Daten sind für den Anwender in einem statischen Zustand, da sie i.d.R. nicht zur Anfragezeit aus einer aktuellen Datenbank abgeleitet werden, sondern in Dateiform zum Abruf bereit liegen.

2. **Map-Server** bieten Karten zur Visualisierung auf einem *Client*-Rechner an. Diese sind entweder fertig konfiguriert und liegen somit statisch, z.B. als JPEG, TIFF oder integriert in HTML-Seiten auf dem Server bereit oder sie werden vom Anwender in Farben, Ausschnitt oder anderen Darstellungsparametern mit der Anfrage konfiguriert und dann dynamisch vom Server abgeleitet.

3. **Online Spatial-Information-Server** liefern nicht nur Karteninformationen zur Visualisierung auf dem *Client*-Rechner, sondern bieten auch Anfragefunktionalität zur Auswertung fachlicher Attribute und Beziehungen. Die Anfrage erfolgt dabei beispielsweise über *XML-Queries*, die über einen *WebFeatureServer* (WFS) ausgewertet werden oder durch die

direkte Eingabe eines *SQL-Statements* in einem *Java-Applet*, das direkt mit der Datenbank kommuniziert.

4. Die Bezeichnung **Online-GIS** kann verwendet werden, wenn der Internetauftritt vollständige GIS Funktionalität besitzt. Bei der Verwendung des Begriffs ist allerdings zu berücksichtigen, dass der Funktionsumfang eines GIS nicht eindeutig und verbindlich festgelegt ist. Es kann aber angenommen werden, dass zumindest die Datenauswertung und -abfrage nach fachlichen und räumlichen Kriterien möglich ist. Es sollte Präsentationsmöglichkeiten für die Geometrie des Raumbezugs sowie für die alphanumerischen Sachdaten geben. Darüber hinaus sollte ein *Online-GIS* auch eine echtzeitliche Datenerfassung und Fortführung über den Browser eines autorisierten *Clients* ermöglichen.

5. Bei **GIS-Funktions-Servern** stehen nicht Daten, sondern Methoden für Daten im Mittelpunkt der angebotenen Dienstleistung. Diese werden in Form von Funktionsbibliotheken den Anwendern zur Verfügung gestellt. Im Rahmen der Entwicklungen werden zwei grundsätzliche Wege angedacht, um Anwenderdaten zu bearbeiten und auszuwerten:

   (a) Die auszuwertenden Daten werden vom *Client* an den Server übertragen und dort von den entsprechenden Funktionen bearbeitet. Die Funktionsergebnisse werden über ein vereinbartes Protokoll wieder an den *Client* zurückgeschickt.

   (b) Die gewünschten Funktionen werden auf Anfrage vom Server zum *Client* übertragen und können dort lokal ausgeführt werden. Die Funktionen sind dabei in Komponenten verpackt und definieren einheitliche externe Schnittstellen.

Bei der Übertragung und Bereitstellung von Daten im Internet ist es naheliegend, dass der Sicherheit und der vertraulichen Behandlung der Informationen sowie der Zuverlässigkeit der Systeme eine zentrale Rolle zukommt und bei der Konzeption und Entwicklung besondere Beachtung findet oder zumindest finden sollte. Dabei sollte der bestmögliche Schutz aller beteiligten Komponenten erreicht werden: Der Schutz der Benutzer von Daten und Diensten (*Client*), der Schutz des Daten- und Diensteanbieters (*Server*) sowie der Schutz der Daten selber und der Realweltobjekte bzw. -personen, die direkt oder indirekt von den Inhalten der Daten betroffen sind.
Angebote von Geodaten und -diensten im Internet können neben statischen Webseiten, die auf dem jeweiligen Web-Server als fertige HTML-Datei abgelegt sind, auch aktive oder dynamische Inhalte enthalten.
Als aktive Inhalte bezeichnet man Programme, die auf dem *Server* des Anbieters gespeichert sind und mit dem Öffnen einer entsprechenden Internetseite

im Web-Browser auf dem Client-Rechner ausgeführt werden. Die Programme basieren in der Regel auf (insbesondere für das Internet entwickelten) Technologien wie Java, Javascript oder ActiveX und bergen demzufolge die Sicherheitsrisiken, die mit den jeweiligen Technologien verbunden sind. O.Kyas und M.Campo haben die Technologien aktiver Inhalte unter Sicherheitsaspekten näher untersucht und in [21] zusammengestellt.
Internetseiten mit dynamischen Inhalten werden bei jeder Anfrage durch einen *Client* neu erstellt. Im Gegensatz zu aktiven Webinhalten laufen die Programme zur Generierung der dynamischen Internetseiten auf dem *Server* ab. Eine Benutzeranfrage wird dabei in einem festgelegten Format als Parameter an ein Script oder Programm übergeben, das auf dem Server gespeichert ist. Dieses führt eine Anfrage auf einer Datenbank aus und bereitet das Ergebnis als Webinhalt - z.B. als HTML-Datei - auf. Dynamische Internetseiten werden beispielsweise - obwohl die Bezeichnung auf aktive Webinhalte hindeutet - mit der Microsoft Methode ASP (*Active Server Pages*) oder mit JSP (*Java Server Pages*- der entsprechenden Technologie von Sun Microsystems) entwickelt.
In der Konzeption von Internetauftritten - insbesondere wenn diese nicht nur allgemeinen Präsentations- und Informations-Charakter haben sollen, sondern unmittelbar Waren, Dienstleistungen oder spezielle Informationen anbieten - müssen immer auch Aspekte der Sicherheit untersucht und in der Systemarchitektur berücksichtigt werden. Häufig erscheint es, als könne Sicherheit für die Daten nur durch Abwertung des Angebots erreicht werden: Umfang und Informationstiefe der Daten müssen ebenso reduziert werden, wie die Funktionalität. Im Gegenzug müssen Maßnahmen zur Überprüfung und Kontrolle der Benutzer ergriffen werden, die dem potenziellen Benutzer den Zugang zu den Daten und Diensten erschweren. Das System insgesamt wird starr und verliert an Attraktivität. Das Bundesamt für Sicherheit in der Informationstechnik (BSI) hat in einer Studie zum „sicheren Internetauftritt im e-Government" [10] dieses Problem für Internetangebote von Bundes-, Landes- und kommunalen Behörden untersucht und verschiedene Architekturen vorgeschlagen und nach Sicherheitskriterien charakterisiert.
Zunächst wird die Architektur von Internetauftritten nach der Verbindung des WWW-Servers zum internen Netz der Behörde oder des Datenanbieters unterschieden.

2.3.1 WWW-Server ohne Verbindung zum internen Netz

Der sicherste Schutz interner Daten und Ressourcen vor Angriffen über den Server des Internetangebots besteht in einer physikalischen Trennung des WWW-Servers vom internen Netz der Behörde.

Für den Fall, dass der WWW-Server von der Behörde selber betrieben wird, ist

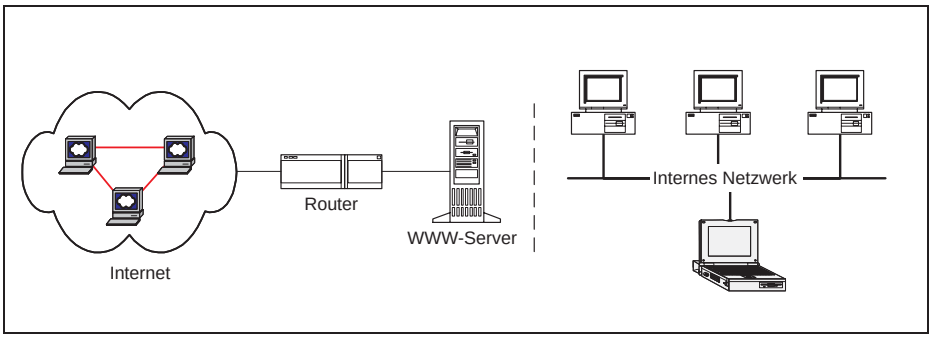

Abbildung 2.2: *Inhouse Betrieb ohne Verbindung zum internen Netz.*

in Abbildung 2.2 schematisch eine solche Architektur dargestellt. Die Daten des Internetauftritts liegen auf dem WWW-Server und werden bei Bedarf im Offline-Betrieb aktualisiert und gepflegt. Anwender, die das Internetangebot der Behörde nutzen wollen, erhalten über einen *Router* ausschließlich Zugriff auf den WWW-Server der Behörde. In der Regel erlaubt der Server für Benutzer aus dem Internet einen ausschließlich lesenden Zugriff. Der Schutz der Daten vor unbefugter Manipulation kann noch gesteigert werden, wenn diese auf einem *Read-Only*-Medium - zum Beispiel einer CD-ROM - gespeichert sind.

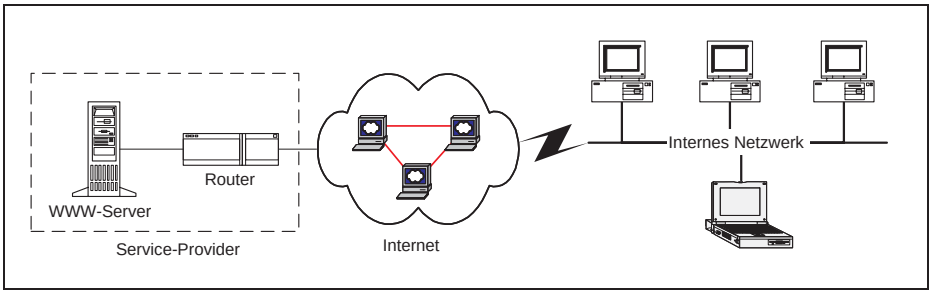

Abbildung 2.3: *Verwendung des WWW-Servers eines externen Dienstleisters über Web-Hosting.*

Eine Alternative zum *Inhouse*-Betrieb bietet das sogenannte *Web-Hosting*. Dabei wird die Infrastruktur eines vertrauenswürdigen Internet-Dienstleisters verwendet, um die Daten im Internet anzubieten. Der Begriff der Vertrauenswürdigkeit wird dabei durch vertragliche Rahmen untermauert. Das *Web-Hosting* bietet sich insbesondere dann als Lösung an, wenn der anbietenden Behörde die entsprechenden technischen oder auch personellen Ressourcen fehlen, um einen Internetauftritt im eigenen Haus einzurichten, und die Daten keine ständigen Aktualisierungen verlangen. Unter Sicherheitsaspekten ist zu beachten, dass die Daten der Kontrolle und den technischen Sicherheitskonzepten des Dienstleisters unterliegen. Da die Wartung der Daten und der Zugriff

auf den WWW-Server nicht der direkten Kontrolle der Behörde unterstehen, sind Verfügbarkeit und Integrität einem Risiko ausgesetzt. Darüber hinaus wird bei der Übertragung der Daten aufgrund der regionalen Distanz zwischen der Behörde und dem Standort des WWW-Servers häufig auf Methoden des Internet - wie E-Mail oder FTP - zurückgegriffen. Dadurch ergeben sich weitere Sicherheitsrisiken. Aus diesen Gründen wurde das Konzept des *Watchdog* eingeführt. Es handelt sich dabei um einen zusätzlichen Rechner mit Internetzugang, der die Aufgabe hat, sich über automatisierte Verfahren bei dem WWW-Server anzumelden und die Daten des Internetangebots mit den Originaldaten der Behörde zu vergleichen.
Um einen einheitlichen und vertrauenswürdigen Sicherheitsstandard für Internetangebote oberster Bundesbehörden durchzusetzen, stellt der dem Bundesministerium des Inneren unterstellte Informationsverbund Bonn-Berlin (IVBB) diesen Behörden WWW-Server zur Ablage ihrer Internetseiten bereit. Die Daten liegen dabei physikalisch entweder auf einem Server im *IP-Backbone* des IVBB oder auf einen Server der Behörde selber. Das *IP-Backbone* ist eine Ansammlung von Netzkomponenten, die unter anderem Server enthalten, die nach außen als eigenständige (virtuelle) WWW-Server erscheinen. Die Behörden, die dieses Angebot nutzen, sind ausschließlich über das IVBB an das Internet angeschlossen und durch ein *Firewall*- und Verschlüsselungskonzept geschützt.

2.3.2 WWW-Server mit Verbindung zum internen Netz

Besteht für die angebotenen Daten ein hoher Bedarf an Pflege und Aktualität und zusätzlich die Notwendigkeit aktiver Inhalte und der Anbindung an das interne Netz der Behörde (zum Beispiel aufgrund sich ständig ändernder interner Datenbanken, die Bestandteil des Internetangebotes sind), so muss die Internetanbindung der Behörde und die Verbindung zwischen dem internen Netz und dem WWW-Server mit besonderen Sicherheitsmaßnahmen versehen werden.

Eine übliche Methode ist der Einsatz von sogenannten *Applikation Gateways*, wie in Abbildung 2.4 skizziert. Ein *Applikation Gateway* ist ebenso wie der *Paketfilter* Bestandteil einer *Firewall*-Architektur, die die Aufgabe hat, eine Sicherheitsschwelle zwischen zwei Netzen zu schaffen, über die jede Kommunikation dieser beiden Netze ablaufen muss. In diesem Rahmen dient der *Applikation Gateway* dazu, auf Applikationsschicht eintreffende Datenströme von einem Netz in das andere weiter zu leiten und dabei die Verbindung derart logisch aufzutrennen, dass das *Applikation Gateway* die Aufgabe eines Vermittlers wahrnimmt: Der Client kommuniziert mit dem *Applikation Gateway* und das *Applikation Gateway* mit dem Server. Die IP-Adressen des internen Netzes sind außerhalb der *Firewall* nicht sichtbar. *Paketfilter* verfügen über

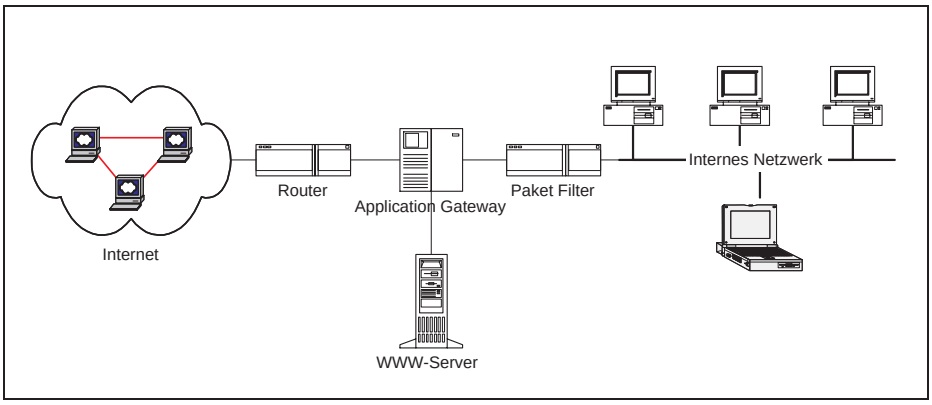

Abbildung 2.4: *Direkte Verbindung des WWW-Servers mit dem internen Netz über ein Application Gateway.*

eine spezielle Software, die die Datenströme zwischen Internet und WWW-Server überprüft. Aufgrund definierter Filterregeln entscheidet die Software, ob ein Datenstrom den *Paketfilter* passieren darf oder nicht. Bei geeigneter Konfiguration schaffen *Firewall*-Konzepte mit *Applikation Gateways*in Verbindung mit *Paketfiltern* ein hohes Maß an Sicherheit für das interne Netz. Dabei steigt mit dem Sicherheitsanspruch auch die Komplexität der *Firewall* im Aufbau ihrer Komponenten.
Welche Techniken für einen konkreten Internetauftritt zur Anwendung kommen, muss nach dem Sicherheitsbedarf der Daten, der Aufgabe des Internetaufftritts und einer Abwägung des technischen und organisatorischen Aufwands entschieden werden. Zur Zeit wird von System- und Datenanbietern mit Nachdruck an geeigneten Konzepten gearbeitet, Geodaten im Internet anzubieten und zum Teil in Verbindung mit *e-Commerce* Komponenten über das Internet zu vermarkten. Bis jedoch der Sicherheitsbedarf der betroffenen Daten endgültig ermittelt ist und einheitliche Sicherheitskonzepte gefunden sind, die das Vertrauen von Datenanbietern und Bürgern, die sowohl potenzielle Kunden als auch mögliche Objekte mit personenbezogenen Informationen in den Datenbeständen sind, in vollem Maße gewinnen können, wird wohl noch einige Zeit vergehen.

2.3.3 *OpenGIS Consortium* und GDI NRW

Das *OpenGIS Consortium* (OGC) ist ein internationaler Zusammenschluss von über 200 Mitgliedern aus Industrie, Verwaltung und Forschung. Das erklärte Ziel des Konsortiums liegt in der Vereinheitlichung und Standardisierung von Austauschformaten und Schnittstellen digitaler, raumbezogener Vorgänge. Die

Hauptmotivation für die Gründung der OGC im Jahr 1994, war der Wunsch und Bedarf nach Interoperabilität zwischen den Komponenten der bis dahin weitestgehend proprietären Entwicklungen im GIS Bereich. Das OGC sieht seine Aufgabe darin, durch die Festlegung von Spezifikationen die Voraussetzungen und Rahmenbedingungen für nationale und internationale Geodaten-Infrastrukturen zu schaffen und offene, interoperable GIS-Lösungen zu ermöglichen. Das Hauptaugenmerk der OGC liegt dabei auf der Definition netzbasierter Dienste (*Web-Services*) und Kommunikationsprotokolle (z.B. GML = *Geography Markup Language*).
Ein aktuelles Projekt, dessen Referenzmodell auf den Spezifikationen und den Erfahrungen der OGC basiert, ist die „Geodaten-Infrastruktur NRW" (GDI NRW). GDI NRW ist eine Initiative des Landes Nordrhein-Westfalen, die im Januar 2000 mit dem Ziel gestartet wurde, system- und plattformunabhängige, webbasierte Geodatendienste und somit einen Geoinformationsmarkt für Nordrhein-Westfalen aufzubauen. An GDI NRW beteiligen sich in Nordrhein-Westfalen ansässige Firmen, Institutionen und Forschungseinrichtungen aus dem GI-Umfeld, die geimsam den Bedarf des GI-Marktes an Web-Diensten ermitteln und diese spezifizieren.
Hintergrund für die Initiative ist die Erkenntnis, dass raumbezogene Informationen aus unterschiedlichen Anwendungsgebieten zwar in großem Umfang vorhanden sind, diese aber weitestgehend in systemabhängigen, proprietären Formaten vorliegen. Übergreifende Auswertungen von Geoinformationen nach unterschiedlichen fachlichen Aspekten werden dadurch deutlich erschwert. Durch einheitlich nutzbare Daten und Dienste soll das Marktpotenzial und der Marktwert von Geodaten gesteigert werden.
Der zentrale Ansatz zur Realisierung von Geodaten-Infrastrukturen basiert auf dem Entwurf einer *Web-Service*-Architektur, wie sie in Abbildung 2.5 dargestellt ist. Anbieter von Geodaten oder spezifischen Diensten für Geodaten stellen über ein einheitliches Portal *Web-Services* zur Verfügung, die sich jeweils genau einer von vier spezifizierten *Service*-Kategorien zuordnen lassen:

1. *Catalog Services* dienen dem Auffinden von Geoinformationsprodukten im Internet. Anhand von Metadaten werden Suchaufträge von Klienten umgesetzt und so der Zugriff auf Daten und Dienste gesteuert.

2. *Web Feature Services* implementieren die Benutzerschnittstellen zu *Feature*-Daten. Diese beschreiben Geo-Objekte mit ihren geometrischen (*Simple-Features*) und fachlichen Attributen.

3. *Web Mapping Services* ermöglichen den Zugriff auf digitale Kartenobjekte, die in geeigneten Grafikformaten, wie PNG[1] oder GeoTiff[2], beim

[1] Portable Network Graphics
[2] Geometry Tag Image File Format

Server hinterlegt sind. Neben den reinen Daten beinhalten *Web Mapping Services* auch entsprechende Portrayal-Dienste zur Darstellung der ausgewählten Kartenobjekte.

4. *Web Coverage Services* liefern Geodaten mit der Möglichkeit des clientseitigen Renderings. Dazu werden *Mapping Services* um Dienste zur Ausgabe von geographischen Werten und Merkmalen - sogenannten *Coverages* - erweitert.

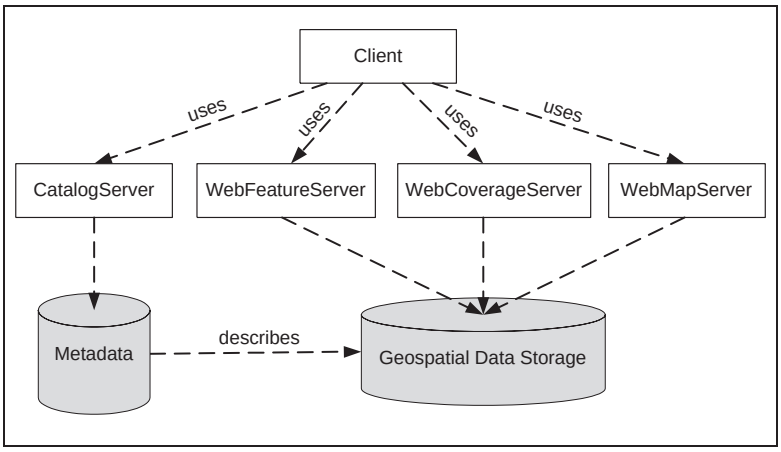

Abbildung 2.5: *Architekturentwurf der GDI für* Web-Services.

Durch einen hohen Abstraktionsgrad der Schnittstellenimplementierung unterstützen *Web-Services* die Plattformunabhängigkeit und Interoperabilität von GIS Komponenten. Anfrage- und Austauschformate müssen dazu allgemeingültigen und herstellerneutralen Spezifikationen genügen, die von Organisationen wie dem *World Wide Web Consortium* (W3C) oder dem OGC gefördert und verabschiedet werden.

Die Kommunikation zwischen einem OGC/GDI Client und einem OGC/GDI *Web Feature Server* basiert auf GDI-spezifizierten XML-Formaten, deren Kommunikationsschema in Abbildung 2.6 dargestellt ist. Der Client sendet kodierte Anfragen an einen *Web Feature Server*. Die Kodierung muss dabei die Spezifikationen des *OGC Filter-Encoding* erfüllen. Das *OGC Filter-Encoding* ist eine für Geodaten angepasste Form von *XML-Query*. Unterhalb der Schnittstelle werden die empfangenen Anfragen entsprechend den Anforderungen der Serverkomponente umgewandelt und zur Auswertung an die Datenhaltungskomponente weitergeleitet. Nach erfolgreicher Auswertung werden die Ergebnisobjekte in der Syntax der *Geography Markup Language* (GML) ausgedrückt und an den Client zurück geschickt.

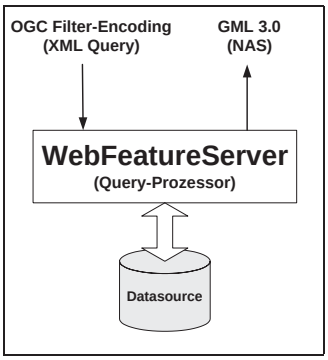

Abbildung 2.6: *Kommunikation mit* Web-Services *am Beispiel eines* Web Feature Servers.

Aus der Idee, Geodaten-*Services* über das öffentlich zugängliche Internet anzubieten, ergibt sich unweigerlich der Bedarf nach geeigneten Methoden, mit denen diese Dienste vor unbefugten Zugriffen geschützt werden können. Grundsätzlich haben sowohl die verwendeten Geodaten als auch die Dienste selbst einen wirtschaftlichen Wert, so dass der Anbieter von *Services* ein berechtigtes Interesse hat, dass diese ausschließlich von autorisierten Personen genutzt werden können und diese auch nur auf die Dienste zugreifen können, für die sie bezahlt haben. Darüber hinaus verarbeiten die verfügbaren Dienste zum Teil hochsensible Daten, die nur von einem eingeschränkten Benutzerkreis unter bestimmten Rahmenbedingungen, zweckgebunden eingesehen und verarbeitet werden dürfen. Im Rahmen von GDI NRW ist die Spezifikation zusätzlicher AAA-Dienste (AAA = Authentifizierung, Autorisierung und Access) zur Kontrolle und Steuerung der Zugriffe von Benutzern auf Geodaten-*Services* geplant. Mit den Konzepten aus den Kapiteln 6, 7 und 8 wird neben der allgemeingültigen Anwendbarkeit für GIS implizit ein Vorschlag zur Definition von Berechtigsstrukturen im Kontext von AAA-*Services* unterbreitet.

2.4 Zusammenfassung

In den verschiedensten Bereichen von öffentlicher Verwaltung und Industrie werden raumbezogene Daten benötigt und genutzt. Mit der Anzahl der Benutzer und der Sensibilität der verarbeiteten Daten eines Systems tritt dabei der Aspekt der Sicherheit zunehmend in den Vordergrund. Insbesondere mit der Erschliessung des Internets als Kommunikations- und Anwendungsplattform müssen die verwendeten Systeme und Daten in geeigneter Weise vor unbefugten Zugriffen geschützt werden.

Kapitel 3

Datenmanagement in GIS

Die Objekte, die uns in der realen Welt umgeben, können in Abhängigkeit von der jeweiligen Interessenslage in unterschiedlicher Komplexität und mit beliebiger Detailtiefe wahrgenommen und beschrieben werden. Folglich unterliegt auch die Anzahl und Vielfalt der wahrnehmbaren Einzelobjekte prinzipiell keinen Einschränkungen. Darüber hinaus können diese Einzelobjekte in beliebig komplexen Beziehungen zueinander stehen. Demgegenüber stehen bei der Beschreibung und Abbildung von Phänomenen der realen Welt nur begrenzte formale, technische und personelle Ressourcen (Speicherplatz, Algorithmen, Zeit, Beschreibungswerkzeuge) zur Verfügung.
Um die wesentlichen und interessierenden Aspekte derartiger Realweltobjekte anderen mitteilen und in einer gemeinsamen Informationsquelle zur Verfügung stellen zu können, bedarf es einer im Sinne der Zielsetzung geeigneten Abstrahierung der abzubildenden Objekte. Die Objekte sollen dabei einer fachlichen Aufgabe entsprechend klassifiziert und attributiert werden, wodurch die in der Realität vorhandene Komplexität solange reduziert wird, bis die Objekte nur noch durch *die* Eigenschaften beschrieben und unterschieden werden, die zur Erfüllung der fachlichen Aufgabe benötigt oder gewünscht werden. Das Ergebnis der Abstrahierung ist ein **Modell** des gewählten Ausschnitts der realen Welt. Die Modelle, die im GIS-Kontext zur Anwendung kommen, haben dabei insbesondere die Aufgabe, die Phänomene der Erdoberfläche und bei hoch spezialisierten GIS die Eigenschaften der Erdkruste, des Erdmantels sowie der Erdatmosphäre zu beschreiben.

3.1 GIS-Architekturen

Die Auswahl oder Definition von Modellen zur Beschreibung der realen Welt bildet die Grundlage für den Entwurf von GIS-Architekturen. Die Objekte, die in GIS-Modellen beschrieben werden, bezeichnen wir mit den Begriffen „Geo-Objekte" oder „Spatial Data". Diese stehen für jene Art von Informationen, deren Besonderheit ihr Raumbezug oder eine räumliche Komponente ist. Anders als Objekte ohne Raumbezug besitzen Geo-Objekte also eine räumliche Ausprägung in einem ausgewählten räumlichen Bezugssystem. Diese Daten können, wie ihre Vorbilder in der realen Welt, beliebig komplexe Strukturen, Beziehungen und Abhängigkeiten aufweisen. Die Beziehungen und Abhängigkeiten sind in der Regel geometrischer, fachlicher oder topologischer Natur. Zur Beschreibung von Objekten und Phänomenen der realen Welt stehen grundsätzlich zwei Klassen von Modellen zur Auswahl:

1. Syntaktische Datenmodelle
    - hierarchische Datenmodelle
    - netzwerkartige Datenmodelle
    - relationale Datenmodelle
2. Semantische Datenmodelle
    - objektorientierte Datenmodelle

Bei der Abstrahierung raumbezogener Informationen setzen sich aufgrund der realitätsnahen und erweiterbaren Beschreibungsmöglichkeiten zunehmend semantische Datenmodelle in Form objektorientierter Strukturen durch. Dabei muss zwischen einem abstrakten Datenmodell und der logischen Speicherstruktur einer Datenbank oder einem Dateiformat, als Grundlage der permanenten Speicherung der beschriebenen Daten, streng unterschieden werden. Neben den Daten selbst sind die Methoden zur Bearbeitung dieser Daten ein wesentlicher Bestandteil eines Konzeptes zur Repräsentation von Realweltobjekten. Um die Daten in ihrer Gesamtheit für die verschiedenen Anwendungsgebiete nutzbar zu machen, müssen sie im Sinne einer geowissenschaftlichen Aufgabenstellung erfasst, verarbeitet, gespeichert und präsentiert werden können.
Geoinformationssysteme stellen heute eine Vielzahl der benötigten Methoden für Geodaten bereit. Eine einheitliche und allgemeingültige Definition des Begriffs „Geoinformationssystem" kann schon aufgrund der Vielseitigkeit der Anwendungsgebiete, des Bedarfs an hoch spezialisierten Lösungen für raumbezogene Fragestellungen und der daraus häufig folgenden proprietären Softwareentwicklungen nur bedingt formuliert werden. Die Bandbreite der Definitionen

in der Literatur reicht von den reinen Daten in analoger oder digitaler Form über Software- und Hardwarekomponenten bis hin zur Einbeziehung konkreter GIS-Projekte oder -Produkte (z.B. ATKIS® oder ALKIS®). Während klassische analoge Karten die Erde und die darauf befindlichen Objekte lediglich abstrahiert und generalisiert darstellen, werden in GIS diese räumlichen Informationen um fachliche Informationen ergänzt oder mit diesen verknüpft und mit einer dynamischen Fortführungslogik versehen. Durch die Möglichkeit der strukturierten und automatisierten Anfrage, Auswertung und Präsentation der Daten werden GIS zu leistungsfähigen Werkzeugen. Allgemein dienen Informationssysteme der Speicherung, Wiedergewinnung, Verknüpfung und Auswertung von Informationen. Ein Informationssystem besteht aus einer Datenverarbeitungsanlage, einem Datenbanksystem und den Auswertungsprogrammen. In diesem Sinne steht bei einem Geoinformationssystem die Speicherung, Auswertung und Bearbeitung von räumbezogenen Informationen im Mittelpunkt des Interesses.
Grundsätzlich können GIS, geprägt durch ihre Aufgaben, in präsentationsorientierte und informationsorientierte GIS unterteilt werden:

1. Bei *präsentationsorientierten Systemen* steht die digitale Bildschirmkarte im Mittelpunkt der Betrachtung. Diese soll mit Hilfe von CAD Werkzeugen (CAD = *Computer Aided Design*) nach kartographischen Aspekten erfasst und zur Darstellung gebracht werden. Die Semantik der dargestellten Objekte wird durch Symbole und Signaturen zum Ausdruck gebracht. Insofern dienen präsentationsorientierte GIS zunächst dazu, analoge Papierkarten in digitalen Datenformaten zu verarbeiten. Häufig ist bei solchen Systemen die Möglichkeit gegeben, mit Hilfe von Verknüpfungselementen (z.B. durch *SQL-Statements* via ODBC oder mit *URL-Encoding*) die Kartenobjekte mit Datensätzen einer externen Datenbank zu attributieren.

2. Umfangreichere Auswertungsmöglichkeiten bieten *informationsorientierte Systeme*. Die Objekte der realen Welt und deren Modelle stehen hier im Mittelpunkt des Interesses. Diese werden in eine Datenbankwelt mit der gewüschten Komplexität abgebildet und können mit den verfügbaren Methoden des Datenbanksystems ausgewertet werden. Die räumliche Ausprägung eines Objektes ist durch ein oder meherere Attribute eines Geometrietyps modelliert. Die Präsentation der Geometrie kann bei informationsorientierten GIS zum Zeitpunkt der Anfrage direkt aus der hinterlegten Datenbank abgeleitet und in einem Grafik-Client - entsprechend den Anforderungen des Benutzers - zur Darstellung gebracht werden.

Unter Aspekten der Zugriffsbeschränkung sind an die Daten präsentationsorientierter Systeme grundsätzlich die gleichen Anforderungen zu stellen, wie an analoge Papierkarten. Der Inhalt einer Karte wird in einer Datei oder in einer grafischen Ebene (*Layer*) gespeichert. Insofern kann eine effiziente Zugriffsbeschränkung für den grafischen Teil des GIS über Methoden des Betriebssystems und der grundsätzlichen Kontrolle des physischen Zugangs zu den DV-Anlagen erreicht werden. Die mit den Grafikelementen verknüpften Sachdaten können bei präsentationsorientierten GIS nur über die Grafikelemente selbst und damit bei entsprechender Zugriffsberechtigung für die Grafik erreicht werden. Die hinterlegten Sachdaten sind darüber hinaus (und in der Regel) durch Zugriffsmethoden des Datenbanksystems bzw. des angefragten *Web-Servers* gegen unerwünschte Zugriffe geschützt. Werden dagegen informationsorientierte GIS eingesetzt, so ergeben sich für eine effiziente Zugriffsstruktur wesentlich höhere und komplexere Anforderungen. Der Zugriff auf die Informationen erfolgt primär über die Objekte, die neben den Sachdaten mit unterschiedlichen Geheimhaltungs- und Sicherheitsanforderung sowie ihrem Raumbezug als Geometrieattribute auch Informationen über Beziehungen zu und Abhängigkeiten von anderen Objekten enthalten. Um nur Teilaspekte einzelner Objekte oder Objektklassen benutzerabhängig zugreifbar machen zu können, ohne dabei Informationskanäle zu öffnen oder die Integrität der Daten zu gefährden, bedarf es einer Zugriffsstruktur, die den fachlichen Kontext der Objekte sowie deren fachliche und räumliche Beziehungen mit erfasst. Mit der Definition einer GIS-Architektur werden auch die Methoden festgelegt, mit denen Phänomene der realen Welt in einem Datenmodell abstrahiert, über ein Datenbankmodell implementiert und schließlich in einer oder mehreren Anwendersichten präsentiert werden.

3.2 Speicherstrukturen für räumliche Informationen

Einen wesentlichen Bestandteil von GIS stellt die Erfassung, Speicherung und Darstellung der räumlichen Ausprägungen der modellierten Objekte dar. In Abhängigkeit von der Datenherkunft, der Erfassungstechnik, der Erfassungsabsicht und dem Verarbeitungszweck existieren grundsätzlich zwei Formen räumlicher Daten. Es sind dies Rasterdaten und Vektordaten.

3.2.1 Rasterdaten

Rasterdaten entstehen durch Scannen von Plänen oder Luftbildern bzw. durch unmittelbare digitale Fotografie der Erdoberfläche und der Verarbeitung von Satellitenaufnahmen. Rasterdaten werden durch eine Matrix definiert, die in

einer festgelegten Auflösung über das Abbild der Erdoberfläche gelegt wird und deren Felder einen Farb- oder Grauwert kodieren, der sich aus der abzubildenden Oberfläche ergibt. Jedes Feld der Matrix repräsentiert somit einen Bildpunkt des Rasters. Rasterdaten eignen sich zur Speicherung und Wiedergabe fotorealistischer Abbildungen der Erdoberfläche oder zur kontinuierlichen Beschreibung der Oberflächenbeschaffenheit (Höhen, Vegetation, geologische- oder meteorologische Eigenschaften).

3.2.2 Vektordaten

Für die Modellierung von Objekten mit komplexen räumlichen und topologischen Beziehungen sind Rasterdaten weniger geeignet. Vektordaten reduzieren den gewählten Ausschnitt auf die relevanten geometrischen Informationen, extrahieren die topologischen Beziehungen und bilden diese auf Modellobjekte geometrischer Primitive (Punkte, Linien, Polylinien, Polygone,...) ab. Auf diese Weise entsteht ein von Maßstab und Auflösung unabhängiges Abbild eines räumlichen Ausschnitts. Dabei ist die der Graphen-Theorie entstammende Knoten-Kanten-Maschen-Struktur eine typische Form der Beschreibung von Vektordaten.

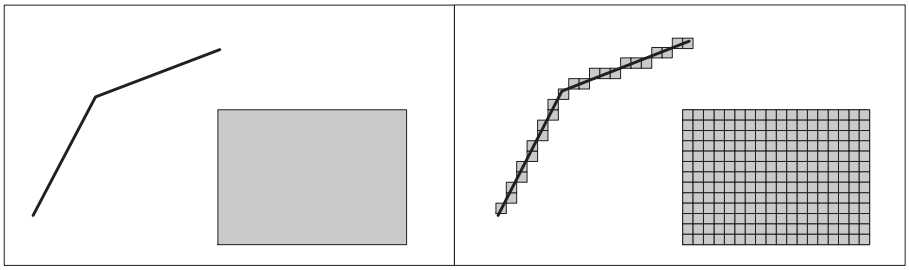

Abbildung 3.1: *Links:Geometrie im Vektordatenformat; Rechts: Der gleiche Ausschnitt im Rasterdatenformat.*

3.3 Effiziente Zugriffsstrukturen für räumliche Daten

Bei der Beschreibung räumlicher Datenmodelle durch eine Menge von Geo-Objekten, die mit räumlichen und nicht-räumlichen Attributen sowie fachlichen und geometrischen Beziehungen charakterisiert werden, entstehen schnell sehr große Datenmengen, die nach effizienten Verwaltungs- und Zugriffsstrukturen verlangen. Nur mithilfe solcher Strukturen ist es möglich, vorhandene

Rauminformationen bedarfsgerecht abfragen und zeitnah vom System erhalten zu können. Effizienten Verwaltungs- und Zugriffsstrukturen können für Rasterdaten durch *Quadtrees* und für Vektordaten durch *R-Trees* realisiert werden.

3.3.1 Geometrieverwaltung mit *Quadtrees*

Durch die organisatorische Zusammenfassung von Pixeln mit gleichen Werten, kann die Effizienz und Auflösung von Rastersystemen deutlich gesteigert werden. Einzelne Bereiche eines Bildes werden so zu thematischen und qualitativen Einheiten verschmolzen, wodurch die Summe der Einzelinformationen reduziert und gleichzeitig ein inhaltlicher Zusammenhang zwischen diesen Einzelwerten hergestellt wird. Eine dahingehende Optimierung von Rasterdaten ermöglicht die Einführung von *Quadtrees*. Die *Quadtree*-Zerlegung eines Bildbereichs basiert auf einer rekursiven Viertelung des fraglichen Ausschnitts. Diese wird solange durchgeführt, bis der jeweilige Bereich nur noch homogene Informationen trägt. Diese charakteristische Information wird nun - zusammen mit *einer* Koordinate - in einer baumartigen Struktur gespeichert. Der Aufbau der Speicherstruktur entspricht dabei dem Verlauf der Rekursionen bei der Zerlegung.

3.3.2 Geometrieverwaltung mit *R-Trees*

Zur Steigerung der Effizienz des Zugriffs auf vektorbasierte Geometriedaten eignen sich *R-Trees* in besonderer Weise, da ihre (konfigurierbaren) Baumeigenschaften schnellen, gezielten Zugriff auf die Objekte räumlicher Ausschnitte ermöglichen. Dabei sind in Abhängigkeit von spezifischen Algorithmen und Kriterien der Verwaltungsbäume, unterschiedliche Ausprägungen - wie *R*-Trees* oder *R+-Trees* - bekannt. An dieser Stelle soll nur das allgemeine Konzept von *R-Trees* erläutert werden. *R-Trees* organisieren räumliche Objekte nach geometrischen Kriterien hinsichtlich einer Optimierung des Zugriffs in einer Baumstruktur. Dazu werden die geometrischen Elemente hierarchisch in räumlichen Blöcken organisiert.

Ausgangspunkt für die Verwaltung von Vektordaten ist die *Bounding Box*. Die *Bounding Box* abstrahiert die räumliche Ausprägung eines Geometrieelements auf ein minimal überdeckendes (horizontal ausgerichtetes) Rechteck.

Jeder geometrische Basistyp (Punkte, Linien, Flächen), dessen Instanzen als zusammengehörige Einheit von einem *R-Tree* verwaltet werden sollen, muss zu diesem Zweck eine Methode zur Berechnung der *Bounding Box* seiner Instanzen implementieren. Mit den Mitteln der objektorientierten Programmierung kann

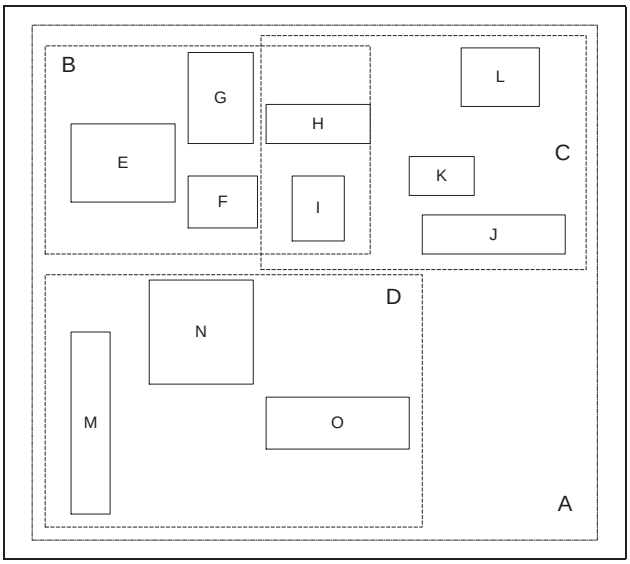

Abbildung 3.2: *Verteilung von* Bounding-Boxes *auf den verschiedenen Ebenen eines einfachen* R-Trees.

dies dadurch erreicht werden, dass alle Geometrieklassen von einer abstrakten Schnittstellenklasse (z.B. *RTreeElement*) abgeleitet sind. Diese definiert eine virtuelle Methode zur Berechnung der *Bounding Box*. Alle abgeleiteten Klassen sind nun gezwungen, die virtuelle Basismethode zu implementieren. *R-Trees* werden durch folgende Eigenschaften definiert:

- Der Root eines *R-Trees* hat mindestens zwei Nachfolgerknoten oder er ist selber Blattknoten.
- Für jeden *R-Tree* existieren zwei ganze Zahlen m und M mit $2 \leq m \leq M/2$, so dass jeder Nicht-Blattknoten (außer dem Root) zwischen m und M Nachfolgerknoten hat.
- Jeder Blattknoten (außer demn Root) besitzt zwischen m und M Einträge.
- Jeder Nicht-Blattknoten p enthält ein Rechteck R, so dass für alle Nachfolgerknoten k von p gilt: Das Rechteck Q von k wird vollständig von R überdeckt. Ist k Blattknoten, dann wird die *Bounding Box* aller Einträge von k durch das Rechteck Q von k vollständig überdeckt.
- Alle Blätter liegen auf der gleichen Baumebene.

Nach der Instanziierung neuer Geometrieelemente werden diese bei einem *R-Tree* registriert. Registrieren bei einem *R-Tree* ist dabei gleichzusetzen mit dem

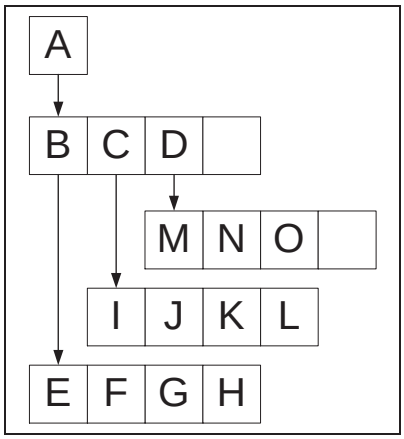

Abbildung 3.3: *Organisation der Geometrieelemente in den Knoten und Blättern eines* R-Trees.

Einfügen einer Referenz des neuen Objektes in einen Blattknoten der Baumstruktur. Zur Registrierung eines neuen Objektes müssen folgende Operationen ausgeführt werden:

- Suchen des richtigen Teilbaums im Sinne der Kriterien: *ChooseSubTree*.
- Eintragen der Objektreferenz: *RegisterElement*.
- Splitten überfüllter Teilbäume: *SplitSubTree* .

Werden Geometrieelemente gelöscht, müssen die entsprechenden Instanzen im *R-Tree* wieder deregistriert, ihre Referenz also aus der Baumstruktur entfernt werden. Dazu müssen folgende Operationen ausgeführt werden:

- Finden des Eintrags im Baum: *SearchElement*.
- Entfernen der Objektreferenz: *UnregisterElement*.
- Verschmelzung unterbesetzter Teilbäume: *JoinSubTree*.

Werden die geometrischen Eigenschaften eines Objektes verändert (Drehen, Verzerren, Verschieben), so wird das Objekt in der Baumstruktur zunächst deregistriert und anschliessend wieder registriert. Der *R-Tree* befindet sich zu jedem Zeitpunkt in einem ausbalancierten Zustand. Die Methoden des *R-Trees* sind hinsichtlich folgender Zielsetzungen entworfen und optimiert:

- Der Bereich, der durch das Rechteck eines Nicht-Blattknotens überdeckt wird, soll minimal sein. Die Geometrieelemente werden also derart unterhalb von Teilbäumen gruppiert, dass sie möglichst dicht beieinander liegen.
- Der Überlappungsbereich der Rechtecke zweier Knoten unterschiedlicher Teilbäume soll minimal sein. Dadurch wird bei der Suche nach bestimmten räumlichen Informationen die Anzahl der zu untersuchenden Teilbäume minimiert.
- Die Summe der Kantenlängen der begrenzenden Rechtecke jedes Knotens soll möglichst klein sein. Dadurch werden sich die *Bounding Boxes* der Baumknoten bei der Verteilung der Objekte der Form von Quadraten annähern.
- Der Speicherverbrauch soll minimiert werden. Die Tiefe des Baumes wird dadurch auf einem relativ niedrigen Niveau gehalten.

Abbildung 3.2 zeigt exemplarisch eine Verteilung von *Bounding Boxes*. Durch die gestrichelten Rechtecke, werden die Bereiche markiert, die zu einem Knoten gruppiert werden. Abbildung 3.3 zeigt die Organisation des gleichen Szenarios in einem *R-Tree*.

3.4 Datenbanken

Datenbanksysteme dienen der Verwaltung und Organisation endlicher Mengen atomarer Daten, der Zusammenfassung zu Datensätzen sowie der Beschreibung von Beziehungen und Abhängigkeiten zwischen Daten und Datensätzen. Die Interpretation der verwalteten Daten, Datensätze und Beziehungen liefert ein Datenmodell. Darüber hinaus stellen Datenbanksysteme Werkzeuge zur Definition, Manipulation und Abfrage von Daten sowie der Generierung neuer Informationen aus vorhandenen zur Verfügung.

3.4.1 Definitionen

In einer formalen Definition setzt sich ein Datenbanksystem (**DBS**) aus einem Datenbank-Managementsystem (**DBMS**), einer oder mehreren Datenbanken (**DB**), einer Menge von Benutzern (**U**) und Anwendungsprogrammen (**APP**) zusammen:

$$\mathbf{DBS} = \mathbf{DBMS} + \mathbf{DB}^* + \mathbf{U}^* + \mathbf{APP}^*$$

Das DBMS bildet die funktionale Schnittstelle zwischen dem Anwender und den Datenbanken. Das DBMS verfügt über eine *Data Definition Language* (**DDL**), eine *Data Manipulation Language* (**DML**) und eine Anfragesprache, über die der Benutzer gezielten Zugriff auf die gespeicherten Informationen der Datenbank erhält.
Die DDL unterstützt die Deklaration bzw. Definition von Objekten, die persistent in der Datenbank gespeichert werden sollen. Mit Hilfe der Methoden der DDL werden Schema-Informationen, also Informationen über die Struktur und die Eigenschaften der zu verwaltenden Objekte, in der Datenbank abgelegt. Die Menge aller Schema-Informationen einer Datenbank bildet das Datenbankschema.
Die DML eines Datenbanksystems ist eine Datenbanksprache zur Manipulation der Datenbank-Inhalte.
Mithilfe einer Anfragesprache können Objekte der Datenbank nach bestimmten Kriterien oder der algebraischen Kombination von Kriterien aus der Datenbank ausgelesen werden. Bei kommerziellen Datenbanksystemen sind diese Sprachkonzepte zur Definition, Manipulation und Anfrage von Daten durch den Sprachumfang einer SQL-Schnittstelle abgedeckt oder in gängige Hochsprachen wie C/C++, Java oder Delphi eingebettet und werden über eine Programmierschnittstelle (**API** = *Application-Programming-Interface*) für Anwendungsentwickler nutzbar. Das DBMS präsentiert sich also nach außen als Komponente zur Kapselung der internen Speicherstruktur von Datenbanken.
Die Menge der Benutzer einer Datenbank lässt sich in einem ersten Ansatz, in drei Gruppen unterteilen:

1. Die Datenbank-Administratoren haben weitestgehend uneingeschränkten Zugriff auf das Schema und die Daten der Datenbank sowie auf die Methoden des Datenbanksystems. Ihre Aufgaben bestehen in der Definition der Datenbankschemata, der Definition von Benutzern und der Zuweisung von Zugriffsrechten sowie der Wartung und Pflege des Datenbanksystems.

2. Anwendungsentwickler nutzen eine API des DBMS oder systemeigene Applikations-Entwicklungswerkzeuge, um Anwendungsprogramme zu erstellen. Diese Anwendungsprogramme dienen der Verarbeitung, Präsentation, Auswertung oder Erfassung bzw. Manipulation von Daten.

3. Die Endanwender einer Datenbank haben schließlich - über die verfügbaren Anwendungsprogramme - Zugriff auf die Datenbank und sind in ihren funktionalen Möglichkeiten auf die Methoden eingeschränkt, die diese Applikationen ihnen anbieten und durch die zugewiesenen Rechte des jeweiligen Benutzers abgedeckt sind.

Bevor Informationen in einer Datenbank verarbeitet werden können, obliegt es einem Administrator, das entworfene Datenmodell, als Abstrahierung eines Realweltausschnitts, auf die Verwaltungsstrukturen des Datenbanksystems abzubilden. Zu diesem Zweck muss das realweltorientierte Datenmodell in ein systemabhängiges, logisches Daten**bank**modell übersetzt werden. Die Formalisierung des Datenbankmodells erfolgt durch den Entwurf eines Datenbankschemas. Dieses beschreibt die Eigenschaften von zusammengehörigen Objektmengen durch Attribute, Datentypen und Beziehungstypen mit den Mitteln der DDL des Datenbanksystems. Das Datenbankschema steht somit in konkretisierender Beziehung zu einem abstrakten Datenschema. Die Beschreibung von Schemata lässt sich üblicherweise auf drei Ebenen projezieren, die im Folgenden dargestellt werden.

3.4.2 Die Drei-Ebenen-Architektur

Die *ANSI/SPARC Study Group on Database Management Systems* [3] schlug bereits 1975 eine konzeptionelle Datenbankarchitektur vor, die auch heute noch breite Anerkennung findet und in dieser Form die Grundlage der meissten kommerziellen und wissenschaftlichen Datenbanksysteme bildet. Diese *Drei-Ebeben-Architektur* unterscheidet bei der Modellierung in interne, externe und konzeptuelle Ebene einer Datenbank, die in direktem Sinnzusammenhang mit den drei entsprechenten Sichten auf die Daten stehen.

Abbildung 3.4 zeigt eine Skizze der Architektur mit ihren drei Ebenen und deren Wechselwirkung.

- Die **interne Ebene** liegt am dichtesten am physikalischen Speicher einer Datenbank. Das korrespondierende interne Schema beschreibt den Zugriff und die Organisation der Daten im logischen Adressraum einer Datenbank. Die Strukturen der internen Ebene haben wesentlichen Einfluss auf die Effizienz des gesamten Datenbanksystems.

- Oberhalb der internen Ebene liegt die **konzeptuelle Ebene**. Auf dieser Modellierungsebene ist die logische Gesamtsicht aller Daten des betrachteten Weltausschnitts sowie die Beziehungen dieser Daten untereinander abgelegt. Das korrespondierende konzeptuelle Schema abstrahiert sowohl die Sichten der einzelnen Anwendergruppen, als auch Aspekte der Implementierung.

- Die **externe Ebene** der Datenbank stellt die Schnittstelle zu den Anwendern und Applikationen dar. Die Sichten auf diese Ebene sind durch

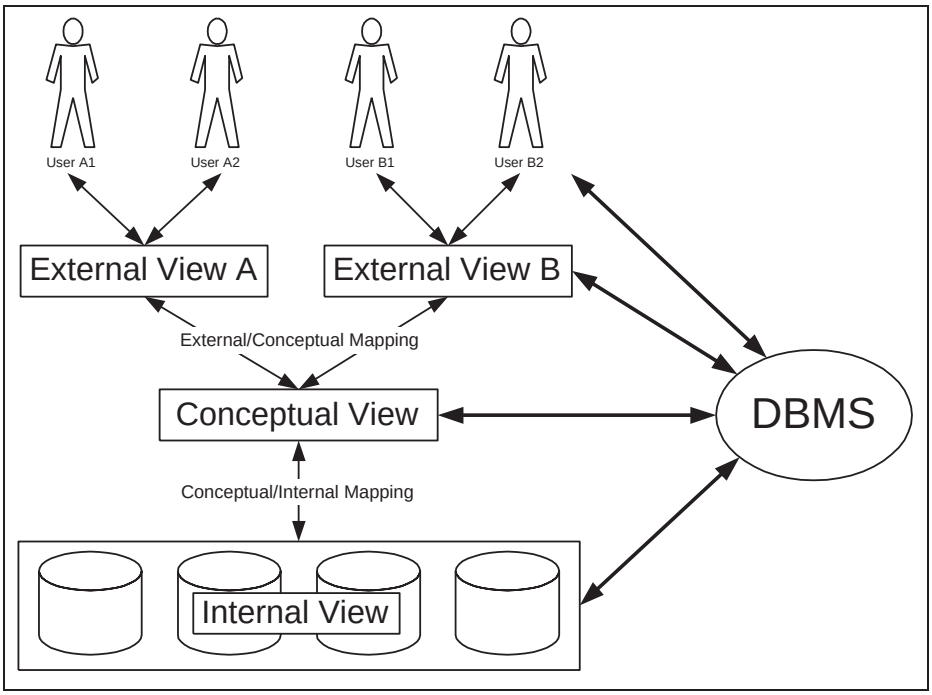

Abbildung 3.4: *Drei-Ebenen-Modell.*

individuelle und konfigurierbare Präsentationsformen gegeben. Die persistent gespeicherten Daten werden in Abhängigkeit vom Anwender und Anwendungsprogramm nach Kriterien wie Laufzeitverhalten, Berechtigungen, Zielsetzungen oder persönliche Vorzüge ermittelt, gefiltert, aufgearbeitet und präsentiert. Die externe Ebene findet ihre formale Repräsentation im externen Schema.

Die drei Ebenen stehen in ständiger Kommunikation miteinander. Bei jedem Zugriff eines Benutzers auf die Datenbank, bzw. auf eine externe Sicht der Datenbank, müssen die Abbildungen zwischen jeweils zwei übereinander liegenden Ebenen bis zum physikalischen Speicher und wieder zurück - bis zur externen Sicht - durchgeführt werden. Die drei Ebenen und deren gegenseitige Abbildungen werden vom DBMS organisiert.
Zum Entwurf konzeptueller und insbesondere semantischer Datenschemata stehen verschiedene formale Beschreibungstechniken zur Verfügung, die sich im allgemeinen grafischer und daher eingängiger Notationen bedienen. Die für die Softwareentwicklung bedeutendsten und als Standard etablierten sind *Entity-Relationship-Diagramme* (**ERD**), die *Object Modelling Technique* (**OMT**) und insbesondere deren Weiterentwicklung: die *Unified Modelling Language* (**UML**). Im Anhang A ist eine kurze Übersicht über die Notation der genannten Mo-

dellierungskonzepte aufgeführt.

3.5 Logische Datenbankmodelle

Die Übersetzung eines konzeptuellen Schemas in ein internes Schema wird geprägt und gegebenfalls beschränkt durch die logischen Speicherstrukturen, die verfügbaren elementaren Datentypen des Systems und insbesondere durch die Möglichkeiten der Abbildung von Beziehungen und Abhängigkeiten des gewählten Datenbanksystems. Von Bedeutung sind dabei Datenbanksysteme, die auf relationalen und solche, die auf objektorientierten Strukturen basieren. Zwischen diesen beiden Ansätzen bestehen grundlegende und insbesondere für GIS relevante Unterschiede. Bei der Wahl des Datenbankmodells ist zu berücksichtigen, dass die Übersetzung von einem abstrakten Datenmodell in ein logisches Datenbankmodell möglichst homogen verlaufen und den Anwendungszweck der Informationssystems unterstützen sollte. Unter Berücksichtigung kommerzieller Aspekte, wie der Nutzung gemeinsamer Ressourcen, Transparenz der Strukturen, Austausch von Datenbeständen, Wiederverwendbarkeit von Software und der Verwendung von "de Facto-„ Standardsystemen, haben sich relationale Datenbankmodelle (ORACLE, DB2, Informix) weitestgehend durchgesetzt. Auch wenn bei der Realweltmodellierung, insbesondere im GIS-Bereich, zunehmend auf objektorientierte Konzepte zurückgegriffen wird, so findet bei der Abbildung dieser Modelle in konkrete Datenbanksysteme überwiegend eine Übersetzung in die Tabellenstrukturen relationaler Datenbanken und damit ein Verzicht auf die Vorteile objektorientierter Speicherstrukturen statt. Im den nächsten Abschnitten sollen relationale und objektorientierte Datenbanksysteme im einzelnen charakterisiert und hinsichtlich ihrer Vor- und Nachteile für GIS-Anwendungen bewertet werden.

3.5.1 Relationale Datenbanksysteme

Das Relationenmodell basiert auf der mathematischen Definition einer Relation: Gegeben seien n nicht notwendigerweise unterschiedliche Domänen $D = D_1, ..., D_n$. Eine Relation R ist dann eine Teilmenge des kartesischen Produktes der Domänen.

$$R \subseteq D_1 \times ... \times D_n \tag{3.1}$$

Seien $A = A_1, ..., A_n$ Attribute und dom eine Funktion mit $dom : A \rightarrow D$. Wenn $\forall i_{1 \leq i \leq n} : dom(A_i) = D_i$, dann ist dom die Funktion, die jedes Attribut auf die Menge seiner möglichen Belegungen abbildet. Die Relation R ist dann definiert durch:

$$R \subseteq dom(A_1) \times ... \times dom(A_n) \tag{3.2}$$

Wir bezeichnen mit $schem(R) = (A_1 : D_1, ..., A_n : D_n)$ das Schema der Relation R und mit R eine Ausprägung, also eine Belegung der Relation zum aktuellen Zeitpunkt. Eine Menge von Relationen $R_1, ..., R_m$ bezeichnen wir als Relationale Datenbank RDB:

$$RDB = R_1 \cup ... \cup R_m \tag{3.3}$$

In Anlehnung an die allgemeine Verwaltungsstruktur können wir uns eine relationale Datenbank als eine Menge von Tabellen vorstellen, in denen die Daten der Relationen gespeichert werden. Dabei stellt eine Zeile jeweils ein Tupel oder einen Datensatz dar - also die Belegungen der Attribute einer Relation für genau ein Objekt. Als Beispiel sei eine einfache Relation *Point* zur Speicherung punktförmiger Geometrie genannt.

$$\begin{aligned} schem(Point) = (&id : LONG, x : DOUBLE, y : DOUBLE, \\ &z : DOUBLE, type : STRING) \end{aligned}$$

sei ein Relationenschema. Eine gültige Menge von Instanzen ist dann gegeben durch

$$\begin{aligned} Point = \{&(1, 10.1, 12.7, 0.0, TP), \\ &(2, 17.3, 27.74, 3.3, GP), \\ &(3, 34.3, 2.34, 1.2, TP)\} \end{aligned}$$

Die einzelnen Tupel einer Relation müssen unterscheidbar, eindeutig und identifizierbar sein, um Objekte der realen Welt repräsentieren zu können. In diesem Zusammenhang begegnen uns die Begriffe „Schlüssel" und „Primärschlüssel". Seien X und Y zwei nicht leere Teilmengen von A. Wir nennen Y *funktional abhängig* von X, wenn es keine zwei Tupel in R mit den selben Werten für X und verschiedenen Werten für Y geben kann - wenn also eine injektive Abbildung $f : X \rightarrow Y$ existiert. Die Existenz einer injektiven Abbildung wird dargestellt mit: $X \rightarrow Y$. Wenn nun für eine minimale Menge S mit $T = (A \backslash S)$ T *funktional abhängig* von S ist - also $S \rightarrow T$ gilt, dann heißt S Schlüssel. Ein ausgezeichnetes Attribut $PK \in S$ heißt Primärschlüssel. Komplexe Objekte, deren Struktur über unterschiedliche Beziehungstypen modelliert sind, werden im relationalen Modell über einen besonderen Schlüsseltyp, den sogenannten Fremdschlüsseln realisiert. Mit diesen werden Abhängigkeiten zwischen verschiedenen Relationen einer Datenbank definiert.
Sei R mit der Attributmenge A eine Relation einer relationalen Datenbank RDB. Eine nichtleere Menge $X \subseteq A$ heißt Fremdschlüssel in R, wenn es

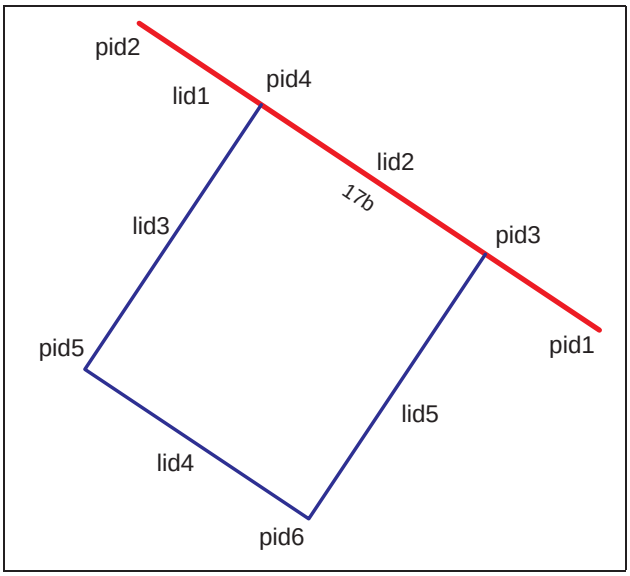

Abbildung 3.5: *Gebäude an einer Flurstücksgrenze im Punkt-Linien-Modell.*

in RDB eine Relation Q mit einer Attributmenge B gibt, so dass $Y \subseteq B$ Schlüssel in Q ist, $dom(X) = dom(Y)$ und für alle Tupel $r \in R, r = (a, b)$ mit $a \in dom(A \setminus X)$ und $b \in dom(X)$ genau ein Tupel $q \in Q, q = (b, c)$ mit $b \in dom(Y)$ und $c \in dom(B \setminus Y)$ existiert.
Eine Erweiterung unseres Beispiels zeigt, wie durch den Einsatz von Fremdschlüsseln geometrische Beziehung zwischen linienartigen Objekten und ihren Endpunkten modelliert werden können:

$$schem(Point) = (\underline{id : STRING}, x : DOUBLE, y : DOUBLE, z : DOUBLE, type : STRING)$$

$$schem(Curve) = (id : STRING, \underline{BeginPoint : LONG}, \underline{EndPoint : LONG}, type : STRING)$$

Im Schema der Relation *Curve* sind die Fremdschlüssel, im Schema der Relation *Point* der zugehörige Schlüssel unterstrichen dargestellt.

Relationen für Punkte und Linien nach dem beschriebenen Schema sind in Abbildung 3.5 dargestellt. Sie zeigen exemplarisch, wie mithilfe von Primär- und Fremdschlüsseln geometrische Objekte modelliert werden können.

id	x	y	z	type
pid1	2582631,54	5628150.26	0,0	10/119G
pid2	2582517.50	5628223.48	0,0	10/119G
pid3	2582588.07	5628178.17	0,0	12/151G
pid4	2582545.85	5628205.28	0,0	12/151G
pid5	2582511.43	5628151.66	0,0	12/151G
pid6	2582553.64	5628124.56	0,0	12/151G

Tabelle 3.1: *Relationale Punktmenge - Flurstücke und Gebäude.*

id	BeginPoint	EndPoint	type
lid1	pid1	pid2	10/233S
lid2	pid3	pid4	12/1013S
lid3	pid4	pid5	12/1013S
lid4	pid5	pid6	12/1013S
lid5	pid6	pid3	12/1013S

Tabelle 3.2: *Relationale Linienmenge - Flurstücke und Gebäude.*

Zur Informationsermittlung aus relationalen Datenbanken wurden drei Konzepte entworfen: Die Relationenalgebra, das Relationenkalkül und die *Structured Query Language* (**SQL**). Eine gemeinsame Eigenschaft der drei Konzepte ist es, aus vorhandenen Relationen neue Relationen abzuleiten und diese dem Benutzer als Ergebnis seiner Anfrage zu liefern.

- Formal ist eine Algebra ein Vektorraum V über einem Feld F, das mindestens eine Multiplikation enthält, mit der die Ring-Axiome erfüllt sind. Bei der Relationenalgebra ist der Vektorraum durch die Menge der zulässigen Relationen gegeben. Das Feld enthält die Operatoren Selektion σ, Projektion Π, Vereinigung $\cup$, Mengendifferenz $-$ und Kartesisches Produkt $\times$ und die daraus abgeleiteten Operatoren. Das Ergebnis einer Operation ist eine neue Relation.

- Der Relationenkalkül ist ein deklarativer Ansatz zur Abfrage von Informationen. Eine Anfrage wird mittels der Ausdrücke der Prädikatenlogik 1. Ordnung ($\forall$,$\exists$,$\wedge$,$\vee$,$\neg$) beschrieben. Die Ergebnismenge eines Kalküls sind Relationen, deren Tupel die Anfrage im Sinne der Prädikatenlogik erfüllen.

- Die *Structured Query Language* SQL ist Anfang der 70er Jahre von IBM im Rahmen der Prototypentwicklung *System R* zum ersten Mal unter dem Namen „*SEQUEL*" vorgestellt worden. Aus kommerziellen (relationalen) Datenbankprodukten ist SQL als Anfragesprache heute kaum

noch wegzudenken und stellt eine wichtige Benutzerschnittstelle zur Datenbank dar, über die es dem Anwender möglich ist, die Relationen der Datenbank über die Operatoren und Ausdrücke der Relationenalgebra und des Relationenkalküls zu bearbeiten, wobei die Konzepte in eine eingängige und verständliche Syntax verpackt sind. Die aktuellen Implementierungen von SQL gehen weit über die Anforderungen an eine Anfragesprache hinaus. Der Sprachumfang enthält Ausdrücke zum direkten Erzeugen, Löschen und Aktualisieren von Relationen und dem Einfügen und Schreiben in vorhandene Relationen. Insofern stellt SQL ein einheitliches Sprachkonzept für DDL, DML und Anfrage auf relationale Datenbanken dar. Eine ausführliche Beschreibung der Syntax und Semantik von SQL findet sich bei [38].

In kommerziellen und wissenschaftlichen Bereich haben sich bei relationalen Datenbanksystemen die Produkte *ORACLE*, *DB2* und *MS SQLServer* weitestgehend durchgesetzt.

3.5.2 Objektorientierte Datenbanksysteme

Objektorientierte Datenbanksysteme (**OODB**) entstanden aus der Idee, realweltnahe Modellstrukturen, wie sie beispielsweise mit UML-Diagrammen beschrieben werden können, in homogener Weise in die Speicherstruktur einer Datenbank abzubilden. Objektstrukturen der abstrakten Modellierung sollten sich also in den Speicherstrukturen des Datenbanksystems wiederfinden lassen. Die Grundlagen für die Realisierung persistenter Objektstrukturen liefern die Konzepte objektorientierter Programmiersprachen, wie *C++*, *Smalltalk*, *Delphi* oder *JAVA*. Bei der Konstruktion einer Klasseninstanz wird der benötigte Speicherplatz im Adressraum des Arbeitsspeichers reserviert und erst durch die bewusste Entfernung des Objektes wieder freigegeben. Während der Lebensdauer eines Objektes kann dieses über eine feste Adresse im Arbeitsspeicher angesprochen und referenziert werden. Bei objektorientierten Datenbanken wird die Generierung von Klasseninstanzen derart modifiziert, dass die Reservierung des Speicherplatzes in einem ausgezeichneten Bereich eines Sekundärspeichers erfolgt - dieser Bereich kann entweder durch eine Datei auf der Festplatte (*Databasefile*) oder durch eine Partition (*rawfs-partition*) spezifiziert sein. Objekte werden somit in einem festen Bereich eines Permanentspeichers konstruiert und können dauerhaft über ihre Adresse oder einen *Offset* referenziert werden. Objektorientierte Datenbanken unterstützen damit im vollen Umfang die Paradigmen objektorientierter Programmierung, die im Kern durch die folgenden Begriffe repräsentiert werden:

- **Objekt:** Das „Objekt" ist der zentrale Begriff der objektorientierten Paradigmen. Er beschreibt eine einzelne, unterscheidbare, existierende und eindeutig identifizierbare Entität eines Systems. Die Eindeutigkeit eines Objektes wird durch seine Identität oder einen Identifikator erreicht. Dies ist eine charakteristische Eigenschaft des Objektes, die unabhängig vom aktuellen Zustand oder Verhalten des Objektes ist. Objekte sind Träger ihrer Eigenschaften, die in Form einer Belegung der Objektattribute mit konkreten Werten den aktuellen Zustand eines Objektes beschreiben. Das Verhalten von Objekten wird über Methoden festgelegt, die in der zugehörigen Klasse (s.u.) beschrieben sind und vom Objekt selber ausgeführt werden. Objekte kommunizieren miteinander über Nachrichten. Diese fordern ein Objekt dazu auf, sich in einer bestimmten Weise zu verhalten, also eine Operation mit bestimmten Parametern auszuführen. Dabei übergibt der Sender eine Nachricht - identifizierbar durch einen eindeutigen Namen - zusammen mit einer Parameterliste an einen Empfänger.

- **Klasse:** Eine Klasse legt die Struktur und das Verhalten einer Menge von Objekten fest, die eine semantische Einheit bilden, die also gemeinsame stereotype Eigenschaften besitzen. Die Objekte, die zu einer bestimmten Klasse gehören, werden auch Instanzen oder Exemplare der Klasse genannt. Klassen sind die Träger von Metainformationen, die in Form von Attributen definiert werden. Attribute sind benannte Datenelemente definierter Datentypen, die in jeder Instanz der Klasse mit konkreten Werten belegt sind. Neben den Attributen definieren die Klassen auch das Verhalten ihrer Objekte durch die Implementierung von Methoden. Diese Methoden stellen in der Regel die Schnittstelle der Objekte zur Aussenwelt dar: Sie fangen Nachrichten anderer Objekte, setzten und lesen Attributwerte und verschicken Nachrichten. Attribute und Methoden einer Klasse werden in einen privaten- (*private*) und einen öffentlichen- (*public*-) Bereich aufgeteilt. Im *private*- Bereich einer Klasse befinden sich die Attribute und Methoden, auf die ausschließlich die Instanzen dieser Klasse Zugriff haben. Die externe Kommunikation mit Objekten ist ausschließlich über die öffentlichen Attribute und Methoden der zugehörigen Klasse möglich. Das Konzept der *private-*/*public*-Bereiche dient der Datenkapselung und Objektsicherheit: Der Austausch von Nachrichten soll ausschließlich über explizit definierte Schnittstellen stattfinden.

- **Vererbung:** Die Klassenvererbung ist ein Konzept zur Abstahierung bzw. Konkretisierung von Beschreibungsmerkmalen entlang von Vererbungsbäumen. Man spricht in diesem Zusammenhang auch von abgeleiteten bzw. von Ober- und Unterklassen. Eine Klasse K_2 ist abgeleitet

von einer Klasse K_1 bedeutet dabei: K_2 ist „spezieller" als K_1 bzw. K_1 ist Oberklasse von K_2 und K_2 ist Unterklasse von K_1. Die abgeleitete Klasse übernimmt alle Attribute und Methoden ihrer Oberklasse, sie „erbt" also die Eigenschaften der Oberklasse. Für die Objekte gilt, dass alle Instanzen einer Klasse K gleichzeitig Instanz aller Oberklassen von K sind. Das Konzept der Vererbung unterstützt die Modellierung komplexer Strukturen, da jede Abstraktionsebene der Modellbildung durch eine Ebene einer Klassenhierarchie repräsentiert wird. Attribute und Methoden, die für eine Menge von Klassen identisch sind, können zu einer gemeinsamen Oberklasse zusammengefasst werden und müssen nur einmal implementiert werden. Mit dem Abstraktionsgrad einer Klasse steigt auch die Wiederverwendbarkeit der Klasse, da die Spezialisierung auf ein konkretes Problem erst durch eine Ableitung in der Applikation stattfindet. Die Vererbungsschritte repräsentieren insofern Generalisierungen in Richtung der Basisklassen und Spezialisierungen in Richtung der abgeleiteten Klassen.

- **Polymorphismus:** Polymorphismus bzw. Vielgestaltigkeit ist ein besonderes Phänomen der Vererbung. Eine Basisklasse kann durch sogenannte *virtuelle Methoden* Schnittstellen definieren, ohne diese zu implementieren. Eine solche Schnittstelle muss in allen abgeleiteten Klassen implementiert werden. Zwei Klassen, die eine gemeinsame Basisklasse besitzen, können somit die gleiche Schnittstelle besitzen, die für die Instanzen der Klassen unterschiedliches Verhalten auslöst.

- **Beziehungen:** Grundsätzlich können zwei Typen von Beziehungen unterschieden werden: *Klasse-Unterklasse-Beziehung* und *Klasse-Komponenten-Beziehungen*. Der erste Typ wird durch das Konzept der Vererbung realisiert. *Klasse-Komponenten-Beziehungen* beschreiben Beziehungen, die die Objekte untereinander haben können oder müssen. Die Struktur und Kardinalität der jeweiligen Beziehung ($1:1$, $1:n$ oder $n:m$) wird in den zugehörigen Klassendefinitionen festgelegt. Dies ist durch die rekursive Konstruktion von Datentypen möglich, die als Domänen für Klassenattribute verwendet werden können:

  1. Die Basistypen eines Systems sind Datentypen. Dies sind die fest implementierten und einfach strukturierten Datentypen eines Systems, wie z.B. *char*, *int*, *float*oder *bool*.
  2. Die strukturelle Gruppierung unterschiedlicher Datentypen zu Tupeln, z.B. in Form einer Klasse oder der einfacheren *struct* Konstruktion, führt zu einem neuen Datentyp, der den Namen der Klasse oder des *struct* trägt.

3. Wenn t ein Datentyp ist, dann ist *set(t)* ein Datentyp, der beliebige Mengen von Elementen aus der Domäne von t unbestimmter Kardinalität zur Domäne hat.
4. Wenn t ein Datentyp ist, dann ist *list(t)* ein Datentyp, der beliebige geordnete Listen von Elementen aus der Domäne von t unbestimmter Kardinalität zur Domäne hat.

Durch die Verwendung von *Klasse-Komponenten-Beziehungen* wird es möglich, auch komplexe Beziehungen des Modells, die z.B. fachlicher, räumlicher oder zeitlicher Art sein können, direkt zu implementieren.

Objektorientierte Datenbanken zeichnen sich dadurch aus, dass sie das durch Klassen- und Objektstrukturen aufgebaute Modell in seiner ganzen Komplexität direkt im sekundären Speicher eines Systems ablegen und angefragte Objekte mittels Zeigerarithmetik oder Objektidentitäten sehr performant zur Bearbeitung in den Arbeitsspeicher laden können.

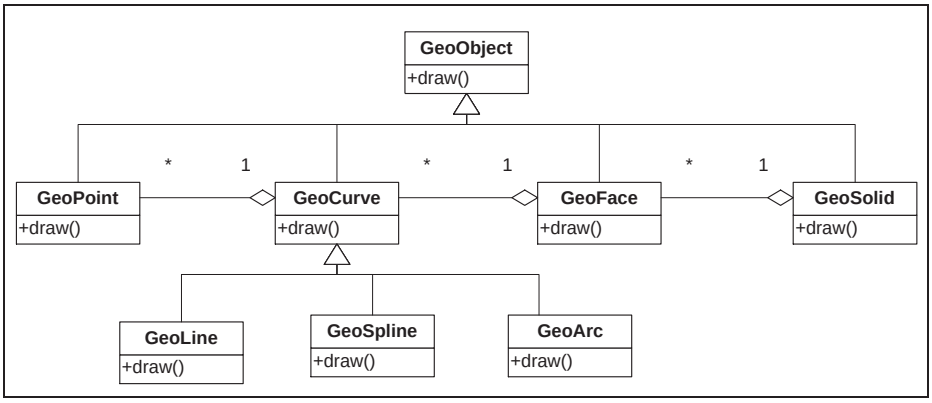

Abbildung 3.6: *Vererbungshierarchie eines Modells für Geo-Objekte mit einer polymorphen Methode* draw(UML-Notation in Anhang A).

Darüber hinaus ist der Sprachumfang objektorientierter Programmiersprachen durch die *API* einer objektorientierten Datenbank um datenbanktypische Operationen - wie *openDB*, *closeDB*, *search* und *retrieve* - erweitert. Die am häufigsten eingesetzten objektorientierten Datenbanksysteme sind derzeit *ObjectStore*, *Objectivity*, *O2* und *POET FastObjects*.

3.5.3 Vergleich zwischen relationalem und objektorientiertem Modellierungsansatz

Obwohl auch die Hersteller relationaler Datenbanken den industriellen Bedarf an Objektspeichern erkannt haben und versuchen, diesen mittels sogenannter

objektrelationaler Datenbanken zu bedienen, bleiben die konzeptionellen Unterschiede zwischen objektorientierten und relationalen Datenbanken bestehen. Bei der Wahl eines Datenbankmodells bzw. eines konkreten Datenbanksystems zur dauerhaften Speicherung von Informationen muss das Anwendungsgebiet, Art und Häufigkeit der zu erwartenden Anfragen an die Datenbank, die zu erwartenden Datenmengen und die Komplexität des Datenmodells berücksichtigt werden.
Der große Vorteil des Relationenmodells liegt in der Definition durch einen strengen mathematischen Formalismus und den daraus abgeleiteten mathematischen Anfragekonzepten: Relationenalgebra und Relationenkalkül. Der allgemein anerkannte Standard der Anfragesprache SQL verhilft relationalen Datenbanken zu einer weiten Verbreitung, da Relationenschemata relativ einfach definiert, modifiziert und damit für eine konkrete Aufgabenstellung angepasst werden können, ohne dass Kenntnisse einer speziellen Programmiersprache oder der Systemarchitektur benötigt werden. Als nachteilig ist festzustellen, dass zur Modellierung von Beziehungen aufwendige Fremdschlüsselkriterien definiert oder Beziehungstabelle erzeugt werden müssen. Insbesondere zur Speicherung und Verwaltung von Geodaten erweisen sich relationale Datenbanken nur bedingt als geeignet, da diese (in der Regel) fachliche und topologische Beziehungen hoher Komplexität aufweisen, die über eine Reihe von Fremdschlüsseln beschrieben werden müssten:
Flächen haben begrenzende Linien, *Linien haben linke und rechte Maschen*, *Linien haben Anfangs- und Endpunkte*, *Punkte haben anhängende Linien*, *Fachobjekte haben definierende Geometrie*.
Die Folge sind Speicherstrukturen, die sich mit wachsender Komplexität immer weiter von der Semantik des fachlichen Datenmodells entfernen und nur durch aufwendige Algorithmen wieder in eine Anwendungssicht abgebildet werden können. Insbesondere führen Modifikationen sowie das Hinzufügen oder Löschen von komplex strukturierten Objekten zu hohem Rechenaufwand bei der Identifizierung zusammengöriger Datensätze durch die Belegung von Schlüsselattributen. Ähnliche Probleme bestehen für den Erhalt von Integritätsregeln in komplex modellierten relationalen Datenbanken. In der Praxis wird die mangelnde Effizienz im Datenzugriff durch geeignete *Hash*-Verfahren und Indizierungen ausgeglichen.
Objektorientierte Datenbanken bieten den Vorteil einer *realwelt-konformen* Modellierung bei hoher Homogenität zwischen den Schemata der drei Datenbankebenen. Komplexe Beziehungen werden durch Attributbildung zu Eigenschaften der Objekte einer Klasse. Durch Mengen- oder Listenstrukturen können beliebige Kardinalitäten von Beziehungen abgebildet werden. Dadurch kann die semantische Beschreibung von Beziehungen konsistent in die Syntax der objektorientierten (Datenbank-)Programmiersprache übersetzt werden. Die Zeigerarithmetik und die Existenz eindeutiger Objekt-Indentifikatoren ver-

meidet den Bedarf an redundanter Datenhaltung und ermöglicht effizienten Datenzugriff, da Objekte der Datenbank direkt über ihre unveränderliche persistente Adresse oder ihren eindeutigen Identifikator referenziert und diese zur direkten Definition von Beziehungen zwischen Objekten verwendet werden können. Eine der herausragenden objektorientierten Paradigmen ist die Aufhebung der Trennung von Daten und Methoden. Diese findet sich auch in objektorientierten Datenbanken wieder. Durch die Definition von Datenbank-Klassen mit der Syntax und Semantik objektorientierter Programmiersprachen werden die Methoden für den Zugriff auf Objekteigenschaften innerhalb der entsprechenden Klasse festgelegt. *Private* Objektinformationen werden nach außen gekapselt und sind nur über die definierten, *öffentlichen* Methoden in der vorgesehen Weise zugreifbar. Die Sicherheit der gespeicherten Informationen wird entscheidend erhöht, da die Benutzer nie direkt mit privaten Eigenschaften interagieren, sondern nur über fest definierte Kommunikationsschnittstellen Zugriff auf Objekteigenschaften haben, die ihnen durch die sichtbaren Methoden zur Verfügung stehen.
Da sich objektorientierte Datenbanken dadurch auszeichnen, dass sie die Klassen einer objektorientierten Programmiersprache sowie deren Instanzen direkt als Datenbankobjekte speichern können, setzt die Definition eines Datenbankschemas häufig fundierte Kenntnisse dieser Programmiersprache und der Datenbankarchitektur voraus. Weiterhin ist kritisch anzumerken, dass ein definiertes Datenbankschema im allgemeinen nur mit Einschränkungen und erhöhtem Aufwand nachträglich erweitert oder modifiziert werden kann. Darüber hinaus bedingt der fehlende Standard einer Anfragesprache die Entwicklung von Schema-gebundenen Applikationen.
Insgesamt lässt sich feststellen, dass beide Datenbanktypen ihre Berechtigung, in Abhängigkeit von der zu lösenden Fachaufgabe haben. Mit steigender Komplexität des Datenmodells und der zu erwartenden Datenmenge gewinnen die Argumente für objektorientierte Datenbanken an Bedeutung. Daher bietet sich besonders bei der Modellierung von Geodaten die Verwendung von objektorientierten Datenbanken an.

3.6 Objektorientierte Modellierung von Geodaten mit SupportGIS

Um für die Entwicklung von Geoinformationssystemen die Vorteile objektorientierter Datenbank-Konzepte hinsichtlich der Modellierung von Geodaten nutzen zu können, ohne sich dabei durch ein starres Applikationsschema und einer fehlenden Anfragesprache in der Flexibilität und Wiederverwendbarkeit des Modells einschränken zu lassen, erscheint es sinnvoll, die objektorientierten

Paradigmen auf einem generischen objektorientierten Datenbank-Kernel zu implementieren. Ein solcher Ansatz wird seit einigen Jahren am Institut für Kartographie und Geoinformation der Universität Bonn verfolgt. Als Ergebnis entstand das Programmsystem SupportGIS, das aus einer Forschungs- und Entwicklungskooperation mit der Firma CPA Geo-Information hervor ging und mit Unterstützung des Thüringer Ministeriums für Landwirtschaft, Naturschutz und Umwelt realisiert wurde. Die zentralen Zielsetzungen des Datenmodells lassen sich in folgenden Punkten zusammenfassen:

- Ein generisches Applikationsschema beschreibt das Datenmodell auf abstrakter Ebene. Statische Beschreibungsklassen dienen der Definition von Anwendungsklassen, die als Instanzen der Beschreibungsklassen das Anwendungsschema abbilden. Die Fachobjekte des Anwendungsschemas sind Ausprägungen einer statischen Instanzenklasse, die den Beschreibungen einer generischen Anwendungsklasse genügt und über Beziehungen mit diesen verbunden ist.

- Die Konfiguration der Fachdatenschamata verlangt weder nach Kenntnissen einer Programmiersprache, noch nach Zugangsmöglichkeiten zum Quellcode.

- Die binären Bestandteile des GIS sind unabhängig vom Fachdatenschema.

- Durch einen ganzheitlichen Ansatz in der Modellierung werden alle drei Aspekte eines GIS - Fachdaten, Geometrie und Anfragekonzept - als integrativer Bestandteil einer Datenbank behandelt.

Ein Datenmodell, dass die genannten Zielsetzungen unterstützt, soll im folgenden skizziert werden. Das Datenmodell sieht eine Strukturierung des Modells in drei Teilen vor, um damit zunächst Fachdatenmodell, räumliche Ausprägung und Auswertung in getrennten logischen und physikalischen Bereichen zu verwalten. Im ersten Teil wird die räumliche Komponente des Datenmodells in Form von Instanzen geometrischer Klassen hinterlegt. Der Zugriff auf diese erfolgt über räumlich-geometrische Auswahlkriterien, die mittels einer *R-Tree*-Struktur ausgewertet werden. Der zweite Teil der Modellstruktur verwaltet ein Fachdatenschema zusammen mit den Fachobjekten des Modells als *Extents* der generischen Klassen, die ihrerseits Beziehungen untereinander und zu geometrischen Objekten unterhalten. Im dritten Teil des Modells werden fachliche Anfragen zur Lösung von Fachaufgaben definiert, gespeichert und verwaltet.

3.6.1 Modellobjekte der Geometrie

Abbildung 3.7 zeigt einen allgemeingültigen objektorientierten Ansatz zur vektororientierten Modellbildung geometrischer Objekte. Das Modell unterscheidet zunächst punkthaft, linienhaft, flächenhaft und gegebenenfalls körperartige Geometrie.

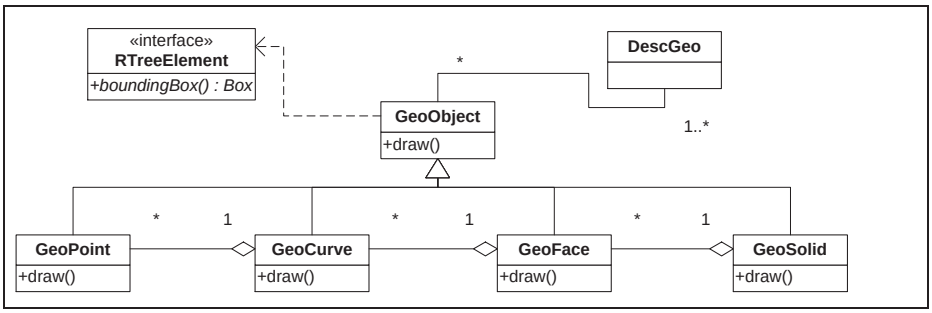

Abbildung 3.7: *Modell für Geo-Objekte erweitert um ein RTree Interface zur Implementierung einer performanten Zugriffsverwaltung.*

Diese werden durch sogenannte geometrische Primitive - mit den Klassen `GeoPoint` , `GeoCurve`, `GeoFace`, und `GeoSolid` - modelliert. Die Instanzen der geometrischen Primitive stehen in Beziehungen zueinander, die die Topologie der Geometrie beschreiben. Art und Kardinalität der Beziehungen werden in den jeweiligen Klassen beschrieben: Die Topologie einer Instanz von `GeoFace` wird beschrieben durch eine Liste ihrer begrenzenden Instanzen von `GeoCurve`. Jede `GeoCurve` Instanz unterhält Beziehungen zu den `GeoFace` Instanzen, für die sie Begrenzungslinie ist. Ihre eigene Topologie ist definiert durch die Beziehung zu zwei Instanzen der Klasse `GeoPoint`, die den Anfangs- und den Endpunkt der Linien-Geometrie markieren. Ein `GeoPoint` besitzt Attribute, die seine Raum-Koordinaten beschreiben. Darüber hinaus steht er in Beziehung zu allen Instanzen von `GeoCurve`, von denen er Anfangs- oder Endpunkt ist. Durch Vererbung ihrer geomerischen Basiseigenschaften werden die Geometrie-Klassen innerhalb des Datenbankmodells weiter spezifiziert. So bildet `GeoCurve`, als linienhafte Geometrie, die Basisklasse für Klassen zur Beschreibung der Geometrietypen Linie, Kreis, Kreisbogen und Spline. In den abgeleiteten Geometrie-Klassen bestehen je nach Bedarf des Datenmodells, noch weitere topologische oder semantische Beziehungen zwischen den Instanzen gleicher oder unterschiedlicher Klassen, die beispielsweise den Mittelpunkt von Kreisbögen, die Stützpunkte von Splines oder die Nachbarschaft von Flächen modellieren. Die geometrischen Primitive besitzen alle die Basisklasse `GeoObjekt`, in der die grundlegenden Eigenschaften (*id, Koordinatensystem, Erstellungsdatum, Änderungsdatum, ...*) und Methoden (*Erzeugen,*

Löschen, Verschieben, Drehen,...) geometrischer GIS-Daten beschrieben sind. Eine semantische Spezialisierung erfahren Geometrie-Instanzen durch ihre Fachbedeutung, die als Instanzen der generischen Fachbedeutungsklasse `DescGeo` eine „einfache Attributierung" der Objekte darstellen. Die Instanzen der Klasse `DescGeo` beschreiben und spezifizieren die fachliche Funktion von Geometrie-Objekten und definieren die Abhängigkeiten zwischen den Fachbedeutungen verschiedener Geometrietypen. Jede Instanz vom Typ `GeoObjekt` genügt den Beschreibungen mindestens einer Fachbedeutung. Geometrische Elemente werden damit zu Repräsentanten modellierter Realwelt-Objekte. In den Modellen einer Liegenschaftsverwaltung repräsentiert ein punkthaftes Geo-Objekt beispielsweise einen vermarkten oder unvermarkten Grenzpunkt. Eine Linie kann Flurstücksgrenze, Flurgrenze oder Gemarkungsgrenze sein, und eine Fläche wird durch ihre Fachbedeutung beispielsweise zur Gebäude- oder Gewässerfläche. Die Fachbedeutungen unterschiedlicher Geometrietypen stehen ihrerseits in definierenden Beziehungen zueinander (In Abbildung 3.7 nicht dargestellt). Mit diesen wird festgelegt, welche Fachbedeutung eine linienhafte Geometrie haben muss, um Begrenzungslinie einer Fläche mit einer bestimmten Fachbedeutung zu sein, und welche Fachbedeutungen die definierenden Endpunkte dieser linieförmigen Geometrie haben müssen.
Aus objektorientierter Sicht entspricht die Zuweisung von Fachbedeutungen durch Attribute oder Beziehungen der Vererbung geometrischer Basistypen und -informationen auf die raumbeschreibenden Klassen einer fachlichen Weltsicht. Durch die Verwendung von Beziehungen zwischen geometrischen Primitiven und den Fachbedeutungsinstanzen wird jedoch die Flexibilität bei Veränderungen der Ansprüche an das Datenmodell gegenüber einer starren Vererbungshierarchie erhöht.

3.6.2 Fachobjektverwaltung

Um eine möglichst hohe Flexibilität in der Modellierung von Fachaufgaben zu erreichen, wird mit der SupportGIS-Lösung ein generischer, objektorientierter Modellansatz vorgeschlagen. Die geforderte fachliche Sicht der Daten wird vom Anwender problemorientiert in einem verfügbaren Schemagenerator als konzeptuelles Datenschema formuliert. Die Kernel-Architektur des Systems (Abbildung 3.8) sorgt dafür, dass dieses Anwenderschema in interne Datenbankstrukturen übersetzt wird.

Änderungen oder Ergänzungen des Schemas können jederzeit von einem Anwender bzw. Administrator, auch ohne Programmierkenntnisse, vorgenommen werden. Ein Aktor/Reaktor-Mechanismus sorgt dabei für die konsistente Aktualisierung aller internen Datenbankstrukturen.

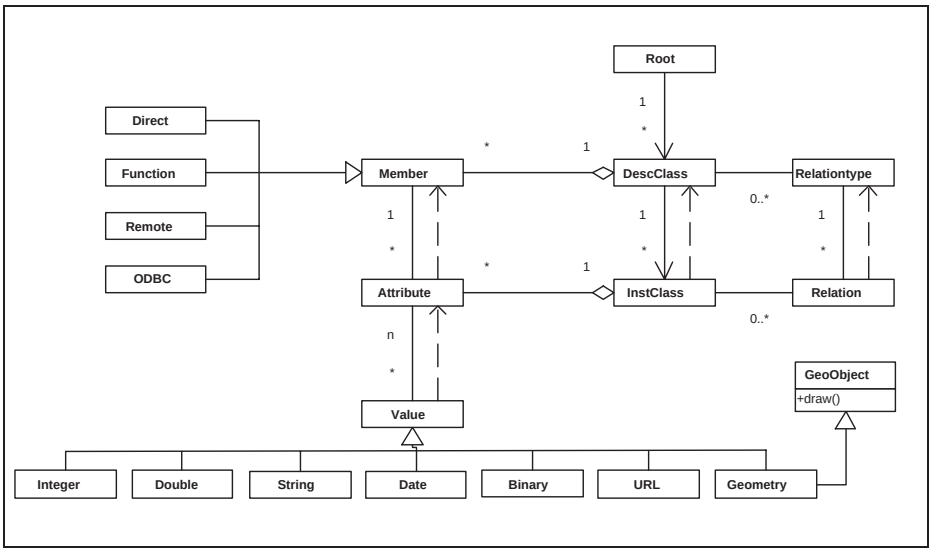

Abbildung 3.8: *Klassendiagramm eines generischen Kernels zur Modellierung raumbezogener Daten.*

Die Kernel-Architektur unterteilt sich, neben einem ausgezeichneten Einstiegsknoten (`Root`), in eine Schemaebene und eine Instanzebene. Die Schemaebene dient den SupportGIS-Applikationen zur Definition von bzw. zur Navigation durch die Fachobjekte der Datenbank. Das Fachdatenschema bildet dabei eine Art Katalog, der die Instanzen der Fachobjektklassen thematisch strukturiert verwaltet. Auf dieser Ebene ist die Beschreibung aller generischer Klassen (`DescClass`) abgelegt. Die Instanzen der statischen `DescClass` bilden die generischen Fachobjektklassen der Instanzebene. Die Instanzen der `DescClass` beinhalten jeweils die vollständige Beschreibung der Fachobjektklasse, die sie repräsentieren. Die Beschreibung einer Fachobjektklasse setzt sich aus ihren Attributen und den Beziehungen zu anderen Fachobjektklassen zusammen. Attribut- und Beziehungsbeschreibungen sind wiederum durch Datenbank-Klassen - `Member` und `Relationtype` - modelliert. Zur Attribut-Klasse `Member` existieren Spezialisierungen durch abgeleitete Attributtypen, wie `Direct` oder `Function`. Zu jeder generischen Fachobjektklasse sind auf der Instanzebene die Fachobjekte der Datenbank gespeichert. Aus Sicht des internen Datenbankschemas sind alle Fachobjekte, unabhängig von ihrer Fachobjektklasse, Instanzen der gleichen statischen Klasse `InstClass`. Jede Instanz der `InstClass` beschreibt ein Fachobjekt, das den Definitionen der zugehörigen `DescClass` genügt. Insofern muss jedes Fachobjekt seine eigene Identität und Klassenzugehörigkeit kennen. Aus der Klassenzugehörigkeit ergeben sich unmittelbar die Beschreibungen der Attribute (`Attribute`) und Relationen (`Relation`) der Fachobjektinstanzen. Jedes Attribut eines Fachobjektes

kann in Abhängigkeit von seiner Multiplizität ein oder mehrere Werte (`Value`) eines bestimmten Datentyps besitzen. Neben den Basistypen, wie `Integer` und `Double`, befinden sich unter diesen auch komplexere Datentypen, wie z.B. `URL`, zur Verknüpfung von Fachobjekten mit eindeutigen Internetadressen oder `Geometry`, zur Zuweisung eines Raumbezugs zu Fachobjekten.

3.6.3 Visuelle Notation einer räumlichen Anfragesprache

Neben der Erfassung sowie der dauerhaften und zuverlässigen Speicherung von Daten sind Möglichkeiten zur strukturierten Auswertung und Analyse der Datenbestände wesentliche Anforderungen an ein GIS. Dieser Aspekt gewinnt mit der Menge der auf digitalen Medien gespeicherten Informationen zunehmend an Bedeutung. Daten werden erst dann zu wertvollen Informationen, wenn Sie bei Bedarf und zur Beantwortung bestimmter Fragestellungen gezielt und strukturiert abgerufen werden können. Solche gezielten Anfragen an ein System sollen vom Anwender, im Rahmen seiner Berechtigungen, in komfortabler und verständlicher Weise formuliert und ausgeführt werden können. Diese Anforderung macht das Konzept einer Anfragesprache zu einem wichtigen Bestandteil eines jeden Informationssystems.
Im Bereich der kommerziell genutzten relationalen Datenbanken hat sich die Anfragesprache SQL [38] als Standard etabliert. Sie integriert die Relationenalgebra und das Relationenkalkül in die Syntax eines gemeinsamen Spachkonzeptes und garantiert damit funktionale Vollständigkeit bei der Auswertung von Relationen. Um den Anforderungen moderner Objekt- und Geodatenbanken gerecht zu werden, wurde der Sprachumfang von SQL mit *OQL*, *Object-SQL* und *Spatial-SQL* in den vergangenen Jahren um entsprechende Datentypen und Konzepte erweitert. Dennoch erscheint SQL und seine Erweiterungen nur bedingt als geeignetes Werkzeug zur Anfrage und Auswertung objektorientierter Geoinformationssysteme im allgemeinen und GIS, die dem in den Abschnitten 3.6.1 und 3.6.2 vorgestellten generischen Ansatz folgen, im Besonderen.
Die textuelle Notation von SQL ist rein syntaxorientiert und in ihrer Struktur für die Auswertung von Tabellen optimiert. Komplexe Anfragen über mehrere Tabellen, unter Einbeziehung fachlicher Relationen, werden schnell unübersichtlich und nur schwierig nachvollziehbar. Der fachliche Kontext der gespeicherten Informationen findet in SQL keine Berücksichtigung. Der Benutzer kann an das System nur dann gezielte Anfragen stellen, wenn er die Speicherstrukturen der hinterlegten Datenbank kennt. Die Anforderungen an eine für die Auswertung objektorientierter GIS geeignete Anfragesprache, deren Syntax sich an der fachlichen Modellsicht orientiert, lassen sich in folgenden Punkten zusammenfassen:

- Anfragen sollten in deklarativer Form definierbar sein: Anfragen werden

durch eine Beschreibung der Ergebnismenge definiert.

- Die Anfragen sollten über eine leicht bedienbare grafische Benutzerschnittstelle (GUI) definierbar sein.
- Die Syntax der Anfragesprache sollte möglichst auf einer gut lesbaren und nachvollziehbaren (grafischen) Notation basieren.
- Die Syntax der Anfragesprache muss unabhängig von Anwendungen und Anwendungsschemata sein.
- Die Semantik der Anfragesprache sollte sich intuitiv aus der Anfragegrafik ableiten lassen.
- Die Anfragesprache muss der Anforderung der Orthogonalität genügen und insofern vollständig im Sinne der bool'schen Algebra sein: Integration der Operatoren $\neg$, $\wedge$, $\vee$ un der Quantoren $\forall$, $\exists$.
- Die Anfragesprache muss operational vollständig sein: Operationale Vollständigkeit wird durch Integration der mathematischen Operatoren - $<$, $\leq$, $>$, $\geq$, $=$, $\neq$ - erreicht.
- Die Anfragesprache soll geometrische und topologische Funktionen unterstützten: *length* , *width*, *area*, *distance*, *overlap*, *intersects*, ...
- Die Anfragesprache soll sinnvolle und erweiterbare Funktionen zur Bilanzierung der Ergebnisse unterstützten.

Als ein zentraler Bestandteil wurde im Rahmen der SupportGIS-Entwicklung ein Anfragekonzept entworfen und implementiert, das diesen Anforderungen genügt und hinsichtlich der Auswertung raumbezogener Daten optimiert ist. Dazu wurde eine grafische Notation zur Beschreibung von Ergebnismengen definiert und ein grafisches *Frontend* entwickelt, das die Formulierung von Anfragen im Sinne dieser Notation erlaubt. Eine vereinfachte Beschreibung der SupportGIS-Anfragesprache ergibt sich aus ihren Eigenschaften:

- Gültige Anfragen werden durch einen zyklenfreien, zusammenhängenden, gerichteten Graph mit einer Knotenmenge V und einer Kantenmenge E beschrieben.
- Die Knotenmenge besteht aus Klassenknoten KK, Operatorknoten OK, Selektionstestknoten SK und Bilanzierungsknoten BK: $V \subseteq KK \cup OK \cup SK \cup BK$.

- Die Klassenknoten spezifizieren Fachobjektklassen in Verbindung mit logischen Quantoren zu *SucheAlle*-, *FürAlle*- und *EsGibt*-Knoten.
- Für jede Anfrage existiert genau ein *SucheAlle*-Knoten, welcher der Startknoten der Anfrage ist.
- Die Menge der Operatorknoten OK beinhaltet die logischen Operatoren UND, $ODER$, $NICHT\ UND$ und $NICHT\ ODER$.
- Selektionstestknoten schränken die Ergebnismenge übergeordneter Klassenknoten, durch die Auswertung einzelner Attributeigenschaften ein.
- Bilanzierungsknoten werten Anfrageergebnisse anhand von Funktionen, wie $COUNT$, AVG, MAX, MIN oder SUM, aus.
- Die Kanten aus E verknüpfen die Knoten operational, werten Funktionen aus und definieren so einschränkende Bedingungen für den nächsten untergeordneten Knoten.

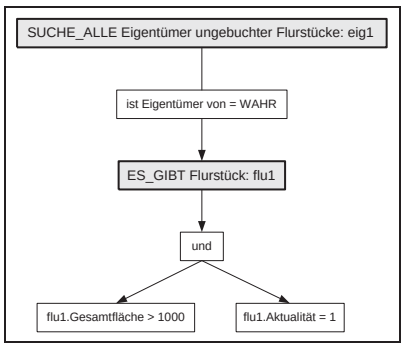

Abbildung 3.9: *Anfragesprache - Die Anfrage liefert als Ergebnis die Menge aller Eigentümer ungebuchter Flurstücke, für die es Flurstücke gibt, deren Gesamtfläche größer als 1000 m^2 ist und deren Aktualität den Wert 1 hat.*

Die SupportGIS-Anfrage stellt dem Anwender Werkzeuge zur Verfügung, mit denen Anfragen nach Geodaten problemorientiert und im fachlichen Kontext intuitiv formuliert werden können. Der Anfragegraph impliziert die Semantik der Anfrage durch die Beschreibung der Eigenschaften der Ergebnisobjekte. Der Graph einer Anfrage legt den Pfad der einzelnen Auswertungsschritte fest. In jedem Anfrageknoten werden (in einem Bereich der Datenbank) Zwischenergebnisse generiert, die dann zur weiteren Auswertung oder Einschränkung an den nächsten Knoten des Graphen weitergereicht werden. Die elementaren Operationen der Anfrageknoten werden auf Methoden des Datenbanksystems abgebildet. Die Abbildungen 3.9 und 3.10 zeigen zwei exemplarische Anfragen

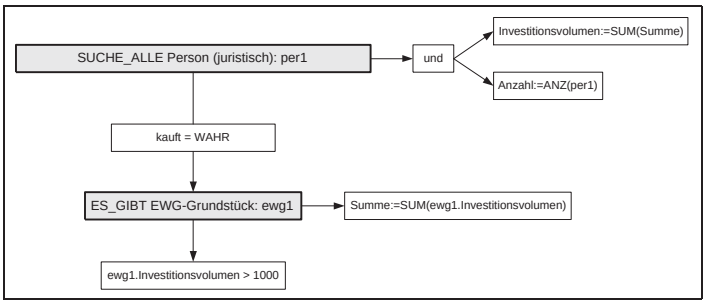

Abbildung 3.10: *Anfragesprache - Die Anfrage sucht nach allen juristischen Personen, die ein EWG-Grundstück kaufen, dessen Investitionsvolumen größer ist, als der Wert des Parameters* Volumen .

in grafischer Darstellung nach den Vorgaben der SupportGIS-Notation.
Die Eigenschaft von Anfragen, einen Ausschnitt eines vorgegebenen Modells und damit eine Sicht der Daten zu definieren, wird in Kapitel 8 wieder aufgegriffen, um damit die Zugriffsbeschränkungen für Benutzer aus Instanzenebene zu definieren.

3.7 Zusammenfassung

Mit der Komplexität einer zu modellierenden Fachaufgabe wachsen auch die Anforderungen an die zugrunde gelegten Datenmodelle. In diesem Kapitel wurde dargelegt, dass eine realweltnahe Abbildung raumbezogener Informationen in besonderer Weise Möglichkeiten zur Beschreibung komplexer Objekte sowie zur stufenweisen Abstrahierung der Objekte benötigt. Objektorientierte Datenmodelle liefern hierzu mit dem Konzept der Klassenvererbung und den Beziehungstypen Aggregation und Assoziation geeignete Modellierungswerkzeuge. Um einen möglichst homogenen Informationsfluss zwischen der internen, der konzeptuellen und der externen Modellsicht klassischer Drei-Ebenen-Architekturen zu ermöglichen liegt es nahe, den objektorientierten Ansatz bis zur Ebene der Datenhaltung aufrecht zu erhalten. Eine solche Modellierungsstrategie wurde in den vergangenen Abschnitten mit der SupportGIS-Architektur vorgestellt. Neben den offensichtlichen Vorteilen objektorientierter Konzepte im Bereich der fachlichen Modellierung muss bei objektorientierten Datenhaltungskomponenten insbesondere das Fehlen einer einheitlichen modellorientierten Anfragesprache sowie die unzureichenden Strategien der Zugriffskontrolle angemerkt werden.

Kapitel 4

Begriffe und Modelle der IT-Sicherheit

Die Definition der Sicherheit von Informationssystemen kann auf drei Begriffe zurück geführt werden: „Integrität", „Vertraulichkeit" und „Verfügbarkeit". Diese zentralen Begriffe werden auch als „Säulen" der IT-Sicherheit bezeichnet. Der Erhalt dieser Prinzipien ist die Zielsetzung jeder Sicherheitsstrategie für IT-Systeme.

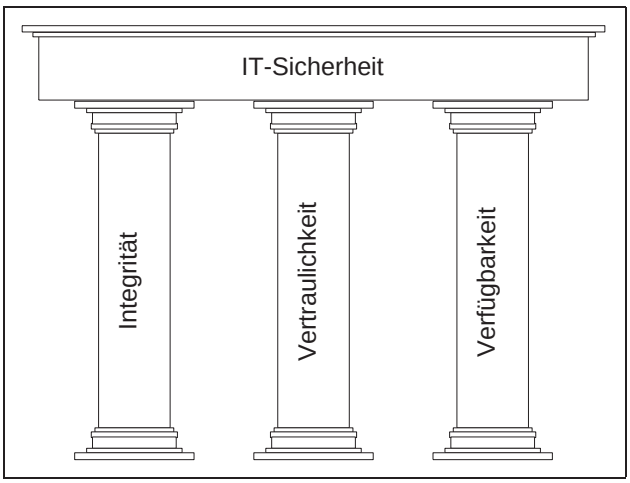

Abbildung 4.1: *Integrität, Vertraulichkeit und Verfügbarkeit, als tragende Säulen der IT-Sicherheit.*

Integrität, Vertraulichkeit und Verfügbarkeit beschreiben die grundsätzlichen Anforderungen an jede DV-technische Verarbeitung sicherheitsrelevanter Informationen.

Integrität

Der Nutzen eines DV-Systems wird in hohem Maße dadurch mitbestimmt, dass der Anwender von der Korrektheit und Konsistenz der gespeicherten Informationen ausgehen kann. Die Daten sollen also zu jedem Zeitpunkt einen gültigen Zustand haben. Darüber hinaus soll jede zulässige Transaktion die Daten von einem gültigen Zustand in einen neuen, gültigen Zustand überführen. In diesem Zusammenhang wird im allgemeinen von der Integrität der Informationen gesprochen. Die Integrität unterliegt unterschiedlichen Bedrohungen: Beabsichtigte oder versehentliche Fehlbedienung eines Systems, fehlerhafte Algorithmen oder Systemabstürze sind nur einige Beispiele. Dementsprechend sind auch mögliche Maßnahmen zum Erhalt der Integrität vielseitig und komplex. Zunächst bedarf es einer informellen - und darauf aufbauend einer formalen Beschreibung der Integrität. Sogenannte Integritätsregeln formalisieren die Eigenschaften gültiger Zustände von Daten. Sie erlauben u.a. die Spezifikationen von Eindeutigkeitsbedingungen, Wertebereichen, erforderlichen Beziehungen und deren Kardinalitäten. Mit Datenbankmethoden werden einfache Integritätsregeln als *Constraints* (*PRIMARY KEY*, *INDEX*, *NOT NULL*, *UNIQUE*), bei umfangreicheren Regeln als *Trigger* (*PL/SQL*) definiert. Integritätsregeln können nur formale Randbedingungen und inhaltliche Spielräume für Datenbankobjekte beschreiben. Sie sind nicht in der Lage, die inhaltliche Übereinstimmung mit den modellierten Realweltobjekten zu verifizieren oder die Zuverlässigkeit des Datenbanksystems zu gewährleisten. Daher sind weitreichendere Maßnahmen zum Erhalt der Integrität - sei es durch gezielte und nachhaltige Schulung der Anwender, *Backup*-Strategien oder Transaktionsmanagement - bei sensiblen Daten empfehlenswert.

Vertraulichkeit

Nicht selten werden in Datenbanken sensible Informationen gespeichert, die sich dadurch auszeichnen, dass es ein (mehr oder weniger) berechtigtes Interesse gibt, diese Daten nur bestimmten Personen und nur zur Erfüllung bestimmter Aufgaben zugänglich zu machen. Die betroffenen Daten unterliegen also einer besonderen Geheimhaltungsforderung, die sich aus privaten, persönlichkeitsrechtlichen, politischen, wirtschaftlichen, militärischen oder anderen Erwägungen heraus begründet. Daten, die einer Geheimhaltungsforderung unterliegen, werden mit unterschiedlicher Gewichtung als vertraulich eingestuft. Die Vertraulichkeit von Informationen korrespondiert mit der Vertrauenswürdigkeit der autorisierten Anwender und des verwendeten Systems. Insofern stellen die Eigentümer und Betroffenen vetraulicher Informationen Anforderungen an das Informationssystem, dem die entsprechenden Daten zur Verarbeitung unterstellt werden. Zu diesen Anforderungen gehört die Vertrauenswürdigkeit des

Systems und seines Betreibers, der vertrauliche und zweckgebundene Umgang mit den Daten sowie ein zuverlässiges Benutzer- und Berechtigungssystem.

Verfügbarkeit

In der modernen Informationsgesellschaft sind Systeme zur elektronischen Datenverarbeitung in immer mehr industrielle, organisatorische und verwaltungstechnische Abläufe vollständig integriert. Deren Funktionalität ist von zentraler Bedeutung. Die zentrale Aufgabe von Informationssystemen ist es, die im Rahmen dieser Abläufe benötigten Daten bei Bedarf - verzögerungs- und verlustfrei - bereit zu stellen. Demzufolge ist die Verfügbarkeit von Daten und Diensten ein wesentlicher Faktor für die Qualität des Gesamtsystems und unter Berücksichtigung möglicher Kosequenzen, auch für die Sicherheit der Abläufe. Ein umfassendes Sicherheitskonzept für Informationssysteme muss auch Maßnahmen zur Sicherstellung der Verfügbarkeit einbeziehen. Die Verfügbarkeit von Daten und Diensten wird häufig gefährdet durch Systemausfälle, durch Überlastung des *Servers* oder des Netzwerks oder einfach durch schlechte Algorithmen und ungeeignete Datenstrukturen, mit denen vertretbare Antwortzeiten des Informationssystems bei gezielten Anfragen verhindert werden.

Trotz einer formalen Unterscheidung möglicher Sicherheitsbedürfnisse hinsichtlich ihrer Anforderungen an die Integrität, Vertraulichkeit oder Verfügbarkeit eines Informationssystems bzw. dessen Informationen stehen diese in enger Beziehung zueinander. Der Zusammenhang zwischen den Sicherheitsbedürfnissen findet seinen Ausdruck in der Vergabe oder Verweigerung von Zugriffsrechten und Privilegien an Benutzer. So setzt die grundlegende Forderung der Integrität eines Datenbestandes voraus, dass das Schreiben und Verändern von Informationen nur durch speziell autorisierte Personen und ausschließlich in einem vorgegebenen Umfang durchgeführt werden kann. Die Erhaltung der Integrität wird also durch eine gezielte Beschränkung von Schreibrechten realisiert. Besteht dagegen eine Forderung nach vertraulicher Behandlung von Informationen, so bedingt dies eine gezielte Beschränkung der Leserechte. Eine nachhaltige Verwehrung von Leserechten bestimmter Benutzer, setzt für diese Benutzer (in der Regel) auch die gezielte Beschränkung der Schreibrechte an den zu schützenden Informationen voraus. Wird andererseits einem Benutzer das Recht zum lesenden Zugriff auf Informationen zugestanden, so wird dieser bei Bedarf - möglicherweise - sein Recht einfordern und damit die Verfügbarkeit der entsprechenden Daten verlangen.

4.1 Bedrohungen für die IT-Sicherheit

Beim Entwurf, der Entwicklung und dem Einsatz von Informationssystemen erhalten Aspekte der Sicherheit zunehmende Beachtung - nicht zuletzt vor dem Hintergrund der hohen finanziellen Schäden, die Systembetreibern und -anbietern in den vergangenen Jahren aufgrund mangelder Sicherheitskonzepte entstanden sind. Insbesondere der Umfang personenbezogener und wirtschaftlich relevanter Daten, die auf digitalen Medien gespeichert sind, ist heute so groß wie nie zuvor. An verschiedenen Stellen von Verwaltung, Industrie und Gesundheitswesen werden umfangreiche Informationen über Bürger, Unternehmen und Organisationen sowie deren Eigentumsverhältnissen gesammelt.
Zur Steigerung der Effizienz betrieblicher Abläufe, zur Verkürzung von Kommunikationswegen und zur gemeinsamen Nutzung verteilter Ressoucen findet eine zunehmende Vernetzung zwischen Computersystemen statt - dabei hat die voranschreitende Nutzung des Internets eine besonders hohe Relevanz für die IT-Sicherheit. Aktuelle Anwendungsgebiete mit einem Fokus auf vernetzte Systeme sind E-Commerce, Online Banking und Mobile Computing. Mit der Vernetzung der Systeme untereinander steigt auch die Zahl der potenziellen Anwender eines Systems und die Zahl der Personen, die sich für die Daten und Dienste eines Anbieters interessieren. Mit der technischen und physikalischen Möglichkeit des Zugriffs und dem zunehmenden Interesse an netzbasierten Diensten steigt auch Gefahr des Missbrauchs sensibler Daten.
In einer Informationsgesellschaft kann der Zugang zu Daten fremder Systeme Wettbewerbsvorteile erbringen, da aus den Daten abgeleitete Informationen einen messbaren Marktwert besitzen und der rechtmäßige Erwerb von Informationen bzw. deren Generierung mit Zeit, Kosten und Aufwand verbunden ist. Daraus folgt als Umkehrschluss, dass der Verlust von Informationen zu einem messbaren Schaden für die betroffenen führen kann. Computerkriminalität und (Industrie-) Spionage, unter Verwendung von DV-Techniken sind zu einer deutlichen Bedrohung der Datensicherheit geworden. Die Angriffe auf IT-Systeme durch Hacker und die Verbreitung von Viren, Würmern, Trojanern oder vergleichbaren unerwünschten Programmteilen können erhebliche Schäden verursachen. Aber auch innerhalb eines Betriebs oder einer Verwaltung entstehen Sicherheitsrisiken durch bewusste oder unbewusste fehlerhafte Behandlung oder Zerstörung von Daten. Die Ursachen für vorhandene Sicherheitslücken sind häufig durch fehlendes Sicherheitsbewusstsein, durch die Arglosigkeit der Anwender im Umgang mit sensiblen Daten und Software sowie durch Vertrauen in die trügerische Sicherheit kommerzieller Produkte zu erklären. Die formale Festlegung einer Sicherheitspolitik und die Definition von Sicherheitsmodellen wird häufig aus zeitlichen und finanziellen Gründen vernachlässigt.

Grundsätzlich existieren für IT-Systeme und damit für die gespeicherten Daten der Systeme zwei Formen potenzieller Gefahren: bewusste und unbewusste Angriffe. Die bewussten Angriffe umfassen gezielte Manipulationen, Zerstörungen und unbefugte Zugriffe auf Daten oder Systemkomponenten mit dem Ziel der Schädigung Anderer oder der Verschaffung von Wettbewerbsvorteilen. Weniger kriminell, aber ebenso bedrohlich ist der Bereich der unbewussten Angriffe, der auf fehlerhafte Systemkomponenten oder fehlerhafte Bedienung zurück zu führen ist. Gerade in der Eigenschaft des Unbewussten liegt eine erhebliche Gefahr, da aufgetretene Fehler und Inkonsistenzen in den Daten häufig erst verspätet bemerkt werden.

- Bewusste Angriffe
  - Manipulation und Zerstörung.
    - Angriffe durch Viren und Würmer.
    - Vernichtung und Zerstörung von Datenträgern und Systemkomponenten.
    - Gezielte Manipulation durch den Missbrauch von Zugriffsrechten.
  - Unbefugter Zugang zu Informationen.
    - Verdeckte Informationskanäle.
    - Trojanische Pferde.
    - Mangelnde Zugangs- und Zugriffskontrolle.
  - Behinderung der Verfügbarkeit.
    - *Denial of Service* - Angriffe.
    - Angriffe auf Hardware- und Softwarekomponenten.
- Unbewusste Angriffe
  - Fehler in der Hardware.
  - Fehler in der Software.
    - Verwendung nicht-vertrauenswürdiger Software.
    - Trügerische Vertrauenswürdigkeit von Software.
  - Fehlerhafte Bedienung.
    - Mangelndes Sicherheits- und Verantwortungsbewusstsein.
    - Unzureichend geschultes Personal.
    - Unzureichende oder fehlende Sicherheitsstrategie und Maßnahmen zur Datensicherung, Rechtevergabe, etc.

4.2 Ebenen sicherer Informationssysteme

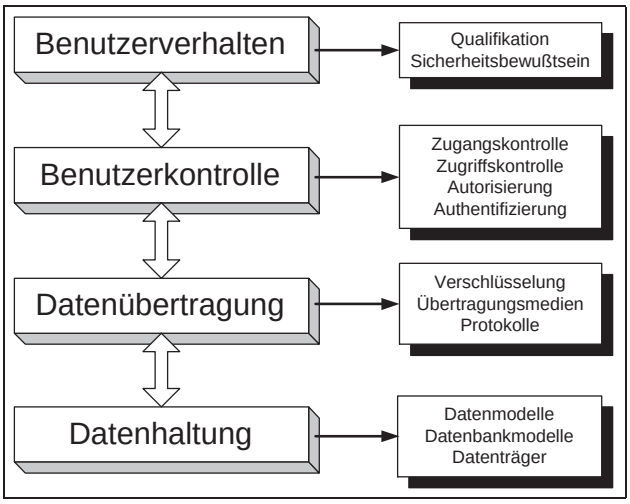

Abbildung 4.2: *Ebenen der Datenbanksicherheit.*

Um die Maßnahmen zur Entwicklung sicherer Informationssysteme methodisch zu strukturieren ist es sinnvoll, diese in Bezug auf vier Ebenen der DV-Sicherheit zu entwerfen: Datenhaltung, Datenübertragung, Benutzerkontrolle, Benutzerverhalten. Diese in Abbildung 4.2 dargestellten Ebenen repräsentieren die sicherheitskritischen Bereiche bei der Verarbeitung digitaler Informationen innerhalb eines Informationssystems. Prizipiell können die Ansatzpunkte für Maßnahmen zur Datensicherheit immer in dieser Weise charakterisiert werden: Auf der untersten Ebene befindet sich die eigentliche Datenhaltung innerhalb einer Datenbank oder einem Dateisystem. Die Sicherheitskonzepte dieser Ebene müssen unmittelbar auf die Medien, Methoden und Modelle der Datenhaltung aufsetzen. Die gespeicherten Informationen verlassen und erreichen die Ebene der Datenhaltung grundsätzlich über ein Medium des Datentransfers auf der Grundlage fest definierter Protokolle. Auf dieser Ebene müssen geeignete Maßnahmen ergriffen werden, um das unbefugte Abhören, Umlenken oder Verfälschen der übertragenen Informationen zwischen Sender und Empfänger zu vermeiden. Auf der nächsthöheren Ebene erreichen die Daten den Benutzer, der sich zuvor über einen Anwendungs-*Client* bei einem Informationssystem angemeldet hat. Die Sicherheitskonzepte dieser Ebene befassen sich mit der Identifizierung und Authentifikation der Benutzer und der Zugangskontrolle zu entsprechenden *Client*- und *Server*-Arbeitsplätzen. Die letzte Ebene der Datenverarbeitung entzieht sich der Kontrolle durch mathematisch-technische Konzepte. Im Mittelpunkt dieser Ebene steht das individuelle Verhalten der Benutzer: Was passiert mit den Informationen, die ein autorisierter Benut-

zer auf Anfrage von einem System im Rahmen seiner Rechte erhalten hat? Wie kann ein vertraulicher und vertrauenswürdiger Umgang mit den Daten gewährleistet werden? Folgt aus einer korrekten Autorisierung eines Benutzers auch eine Garantie für die korrekte Bearbeitung der Daten durch den Benutzer? Der Faktor „Mensch" stellt den am schwierigsten (bzw. garnicht) zu kontrollierenden Teil einer Sicherheitspolitik dar, da sich sein Verhalten auf kein deterministisches Modell abbilden lässt.

4.2.1 Ebene der Datenhaltung

Die Ebene der Datenhaltung wird durch den dauerhaften Speicher einer Datenbank oder eines Dateisystems repräsentiert. In diesem sind die zu schützenden Daten nach den Vorgaben eines Datenbankschemas abgelegt. Ein Datenbankschema entsteht durch die Abbildung eines fachlichen Datenschemas zur formalen Beschreibung eines Realweltmodells, auf die internen Strukturen und Datentypen eines spezifischen Datenbanksystems. Maßnahmen zur Sicherheit, die auf der Ebene der Datenhaltung ansetzen, haben die Aufgabe, eine zuverlässige und dauerhafte Speicherung der Daten zu gewährleisten und diese Daten, entsprechend einem Zugriffs- und Berechtigungsmodell, logisch und physikalisch zu organisieren. Auf Anfrage eines Benutzers sollen die Daten, für die der Benutzer ausreichende Zugriffsrechte besitzt, der nächsten Verwaltungsebene übergeben werden. Aus Sicht der Datensicherheit kommt der Ebene der Datenhaltung die Rolle der innersten Kommunikationsschnittstelle zwischen Daten und Benutzern zu. Insofern definiert eine Berechtigungsstruktur, die direkt auf der Datenhaltung ansetzt, die maximal zulässigen Benutzeraktionen auf den Daten eines Informationssystems und beschreibt benutzerabhängige Teilmodelle in Form von Sichten. Benutzer- und Berechtigungsstrukturen auf Ebene der Datenhaltung werden in der Regel in Bezug auf ein Ausgangs-Datenmodell spezifiziert und mit den Methoden des verwendeten Datenbanksystems oder eines zentralen Systemkernels umgesetzt. Berechtigungsprofile und Benutzeraktionen können somit zentral und einheitlich verwaltet, kontrolliert und überwacht werden. Sämtliche Zugriffe von Benutzern auf die Daten eines Systems erfolgen, unabhängig von der jeweiligen Applikation, über die Zugriffsstrukturen der Datenhaltungsebene.

4.2.2 Ebene der Datenübertragung

Auf der Ebene der Datenübertragung sind für die Sicherheit eines Informationssystems sämtliche Einheiten, Komponenten und Protokolle relevant, mit deren Unterstützung digitale Daten von einem logischen oder physikalischen

Knoten der Datenverarbeitung zu einem anderen Knoten übertragen werden. Im Kontext von GIS interessieren uns dabei insbesondere die Kommunikationswege zwischen der Datenhaltung (Führung) und den Anwendungs- bzw. Verarbeitungskomponenten (Fortführung, Auskunft). Dabei kann die Übertragung zwischen Datenhaltung und Datenverarbeitung auf der lokalen *Client/Server*-Architektur eines Rechners erfolgen, als netzbasierte Lösung innerhalb eines LANs realisiert sein oder im Rahmen einer verteilten Architektur im Internet stattfinden. Insbesondere die Möglichkeiten webbasierter GIS werden derzeit intensiv untersucht. Dabei werden Lösungen angestrebt, bei denen die Applikationen auf Seiten der *Clients* in einem marktüblichen *Internet-Browser* (MS Internet-Explorer, Netscape Commuticator) als *Plug-Ins* oder *JAVA-Applets* ablaufen. Der *Server* einer solchen Lösung kann ein *Web Feature Server* sein, der XML-*Filterencoding (Extended Markup Language)* als Eingabe akzeptiert und GML-kodierte (*Geography Markup Language*) Ausgaben generiert und an den *Client* zurück schickt. Die Maßnahmen der Datensicherheit auf diesen Übertragungswegen sind so vielfältig wie die Übertragungswege selbst. Als übergeordnetes Ziel kann festgehalten werden: Eine Information i, die von einem Sender s an einen Empfänger e gesendet wird, soll derart Übertragen werden, dass e die Information i' erhält. Die Übertragung besitzt dabei die Eigenschaften:

- Die empfangene Information i' ist identisch zur gesendeten Information i ($i' = i$) oder der Empfänger kann die gesendete Nachricht i durch ein vereinbartes Verfahren v aus i' ableiten ($v(i') = i$).
- Nur diejenigen Empfänger e erhalten die gesendete Information, die von s addressiert wurden.
- Der Empfänger e kann die Identität des Senders s und die Authentizität der Nachricht i überprüfen.

Für eine, in diesem Sinne sichere Datenübertragung sind sowohl die physikalisch-technischen Eigenschaften der verwendeten Übertragungsmedien und der Aufbau der Übertragungswege, als auch Zustand, Grösse und Format der Datenpakete selbst von Bedeutung. In den sicherheitsrelevanten Bereich der physikalisch-technischen Eigenschaften von Übertragungsmedien gehören die Abhörsicherheit und die Störanfälligkeit von Datenleitungen. Konzepte sicherer Kommunikations-Architekturen wurden in Abschnitt 2.3 bereits vorgestellt. Insbesondere der Einsatz von *Firewall*-Systemen hat weitreichende Bedeutung erlangt. Eine *Firewall* dient als regelbasierter Filter mit dem die Rechner eines Teilnetzes vor unerwünschten Zugriffen von Rechnern außerhalb des Teilnetzes geschützt werden können. Eine *Firewall* eines Teilnetzes ist in der Regel die

einzige physikalische Verbindung des Teilnetzes mit anderen Teilnetzen oder dem Internet. Somit muss jeder Datentransfer zwischen einem Rechner innerhalb des Teilnetzes und einem beliebigen Rechner ausserhalb des Teilnetzes über die *Firewall* ablaufen und deren Filter-Regeln passieren. Mit den Filter-Regeln ist es zum Beispiel möglich, externe Zugriffe auf bestimmte IP-Bereiche einzuschränken. Durch die Anwendung kryptographischer Verfahren auf die übertragenen Daten wird ein zusätzlicher Schutz der Informationen gegen missbräuchliche Verwendung erreicht. Asymmetrische Verschlüsselungsalgorithmen erreichen ein hohes Maß an Sicherheit und ermöglichen die Einrichtung von *Public-Key*-Infrastrukturen und die Generierung digitaler Signaturen ([6] und [35]), mit denen die Authentizität von Sendern und Nachrichten sichergestellt werden kann.

4.2.3 Ebene der Benutzerkontrolle

Die Vergabe von Privilegien und das damit einhergehende Zugeständnis der Vertrauenswürdigkeit steigert nur dann die Sicherheit einer Datenbank, wenn sich das System der Identität seiner Anwender und der Authentizität von Anfragen und Aufträgen sicher sein kann. Sichere Informationssysteme benötigen also Möglichkeiten, die Identität eines Benutzers erfragen und verifizieren zu können. Ein gängiges und einfaches Verfahren der Zugangs- und Zugriffskontrolle von Benutzern besteht in der Abfrage des Benutzernamens und eines dazugehörigen „geheimen" Passworts, wenn ein Benutzer Zugriff auf die Daten und Dienste eines Systems wünscht. Für die meisten Systeme, deren Daten als eher gering schützenswürdig einzustufen sind, kann dieses Verfahren als ausreichend betrachtet werden. Eine Sicherheitslücke entsteht weniger durch das Verfahren des Passwort-Schutzes selbst, als vielmehr durch die Wahl von Passwörtern und dem - häufig nachlässigen - Umgang mit diesen, wodurch unbefugtes Eindringen in Systeme erheblich erleichtert wird. So sind Kombinationen von Vor- und Nachname, Namen von Freunden, Verwandten oder Prominenten, das Geburtsdatum oder leicht zu merkende Wörter aus dem täglichen Sprachgebrauch beliebte Passwörter. Darüber hinaus werden die geheimen Zugangsdaten zu einem System gerne in der Nähe des Arbeitsplatzes als Notiz aufbewahrt. Dieses Verhalten ist besonders dann zu beobachten, wenn ein System die Zugangssicherheit durch besondere Anforderungen an zulässige Passwörter (Mindestlänge, Sonderzeichen,...) steigern soll. Aufwendigere und effizientere Verfahren der Zugangskontrolle werden zur Zeit nur vereinzelt eingesetzt, da sie sich größtenteils noch im Forschungsstadium befinden und nur bedingt Marktreife besitzen. Hinzu kommt bei vielen technisch möglichen Verfahren der Identitätskontrolle mangelnde Akzeptanz der Öffentlichkeit.
Neben dem Prinzip des „Wissens", wie es durch das Konzept von Passwörtern

umgesetzt wird, basieren neuere Ansätze zur Benutzerkontrolle auf den Prinzipien „Sein" und „Haben".

- Ein bekanntes, wenn auch wenig sicheres Verfahren der Zugangskontrolle nach dem Prinzip des „Wissens" besteht in einer Art Frage-Antwort-Spiel mit dem Benutzer. Durch eine geeignete Kombination von Fragen, die individuell auf den Benutzer zugeschnitten sind und nach Angabe eine Benutzerkennung auf dem Monitor als Zugangskontrollmaske erscheinen, soll die Identität des Anwenders verifiziert werden. Die Sicherheit dieses Verfahrens basiert auf der Annahme, dass nur ein Benutzer in der Lage ist, die Fragen richtig zu beantworten. Häufig begegnet einem dieses fragwürdige Verfahren bei Informationssystemen im Internet zur Überbrückung vergessener Passwörter, die dem Benutzer dann per *e-mail* zugeschickt werden.

- Das Prinzip des „Seins" wird durch biometrische Verfahren umgesetzt. Diese nutzten die Tatsache, dass jeder Mensch eindeutige, unveränderliche Merkmale besitzt, über die jedes Individuum eindeutig identifizierbar ist. Wünscht ein potenzieller Anwender Zugriff zu einem System, so kann seine Identität festgestellt werden, indem entsprechende Geräte die charakteristischen Merkmale aufnehmen und mit den gespeicherten Profilen einer Datenbank vergleichen. Zu den bekanntesten biometrischen Verfahren gehören: Der Vergleich von Sprachmustern, die Überprüfung von Fingerabdrücken, die Gesichtserkennung sowie die Retinaanalyse.

- Zusätzliche Sicherheit gegen unbefugten Zugriff auf Informationen kann der Besitz von Chipkarten oder Magnetstreifenkarten liefern. Die Sicherheit basiert dabei auf dem Identitätsprinzip des „Habens". Wünscht ein Benutzer Zugang zu bestimmten Informationen, so kann eine Karte, die sich in seinem Besitz befindet, von einem geeigneten Lesegerät ausgewertet und die Identität des Benutzers anhand der auf der Karte gespeicherten Daten festgestellt werden. Bekanntestes Anwendungsbeispiel sind Geldausgabeautomaten: Nach dem Einschieben einer Karte in das Lesegerät des Geldautomaten wird die PAN (*Primary Account Number*) vom Magnetstreifen ausgelesen. Damit ist die Identität des Kunden festgestellt. Der Kunde wird nun aufgefordert seine Identität durch Eingabe einer vierstelligen PIN (*Personal Identification Number*) zu verifizieren. Das Sicherheitsmodul berechnet aus Teilen der PAN mit einem geheimen DES-Schlüssel einen Wert, der mit der PIN übereinstimmen muß.

Das letzte Beispiel zeigt, dass beim Zugriff auf besonders sensible Daten eine Kombination verschiedener Verfahren der Benutzerkontrolle sinnvoll sein kann und gegebenfalls die Sicherheit der Daten erhöht.

4.2.4 Ebene des Benutzerverhaltens

Aus Gründen der Vollständigkeit, soll die Ebene des Benutzerverhaltens in diese Liste mit aufgenommen werden. Grundsätzlich unterliegen die gespeicherten Informationen, die einen Benutzer mit ausreichenden Privilegien erreicht haben, weitestgehend dessen Willkür. Der Missbrauch von Benutzerrechten kann nur in sehr eingeschränktem Umfang - beispielsweise durch geeignete Integritätsregeln - vermieden werden. Die Weitergabe vertraulicher Informationen an nicht-autorisierte Personen ist grundsätzlich nicht kontrollierbar. Durch Maßnahmen zur Überwachung und Protokollierung von Transaktionen können jedoch Erfolge in der späteren Aufklärung von Fehlverhalten und Rechtemissbrauch erzielt werden: Wem wurden wann welche Informationen erteilt? Durch wen wurden welche Veränderungen in den Datenbeständen vorgenommen?
Neben dem gezielten Missbrauch erteilter Privilegien stellt das versehentliche Fehlverhalten von Benutzern ein weiteres Problemfeld dieser Ebene dar. Nicht selten sind die Ursachen in mangelhafter Qualifikation für den Umgang mit den Systemen zu suchen. Darüber hinaus fehlt es häufig an dem Bewusstsein für die Verantwortung, die mit der Erteiltung von Privilegien auf den Benutzer übertragen wurde. Der Bedarf an Sicherheit und Vertraulichkeit der verarbeiteten Informationen wird dabei unterschätzt. Ein solches Bewusstsein kann nur durch Qualifizierungsmassnahmen für Mitarbeitern und Anwender, beispielsweise mit Hilfe von Schulungen, geschaffen werden. Außerdem muss eine konzipierte Sicherheitspolitik konsequent umgesetzt werden und sich demnach auch in der Vergabe von Zugriffsrechten widerspiegeln. Im Einzelnen bedeutet dies, den Benutzern eines Systems nur die minimal notwendigen Rechte zur Bewältigung einer Aufgabe zu erteilen.
Die beschriebenen Ebenen können noch weiter unterteilt werden. Beispielsweise kann auf der Ebene der Datenhaltung noch zwischen logischen Datenmodellen und physikalischen Datenträgern unterschieden werden. Für die Zwecke der vorliegenden Arbeit ist die gewählte Aufteilung jedoch ausreichend, da im Weiteren Sicherheit als Teil logischer Konzepte behandelt wird.

4.3 Maßnahmenkatalog zum Schutz personenbezogener Daten

Es gibt eine Reihe von Möglichkeiten, Daten eines Informationssystems gegen unbefugte oder unerwünschte Zugriffe zu schützen. Dabei werden die zu ergreifenden Massnahmen wesentlich vom Schutzbedarf und der Sensibilität der Daten bestimmt. Enthält ein Informationssystem personenbezogene Da-

ten, so sind Persönlichkeitsrechte einzelner Personen betroffen. In Deutschland ist der Schutzbedarf dieser Daten rechtlich durch das Bundesdatenschutzgesetz (BDSG) festgeschrieben. In anderen demokratisch ausgerichteten Rechtsräumen finden sich vergleichbare rechtliche Regelungen. In der Anlage zum BDSG [12] werden zehn erforderliche technische und organisatorische Massnahmen zum Schutz personenbezogener Daten genannt. Um dem Schutzbedarf und -anspruch solcher Daten gerecht zu werden, müssen Informationssysteme diese Massnahmen in geeigneter Weise unterstützen.

1. Zugangskontrolle: Nur befugten Personen ist der Zugang zu den Datenverarbeitungsanlagen zu gewähren.

2. Datenträgerkontrolle: Datenträger dürfen von Unbefugten nicht gelesen, kopiert, verändert oder entfernt werden.

3. Speicherkontrolle: Die unbefugte Eingabe in den Speicher sowie die unbefugte Kenntnisnahme, Veränderung oder Löschung gespeicherter personenbezogener Daten ist zu verhindern.

4. Benutzerkontrolle: Es ist zu verhindern, dass die Datenverarbeitungsanlagen von Unbefugten benutzt werden können.

5. Zugriffskontrolle: Der Zugriff berechtigter Personen darf ausschließlich im Rahmen der Zugriffsrechte dieser Personen stattfinden.

6. Übermittlungskontrolle: Es muss überprüfbar und nachvollziehbar sein, durch wen und an wen personenbezogene Daten mit den Einrichtungen zur Datenübertragung übermittelt wurden.

7. Eingabekontrolle: Es muss überprüfbar und nachvollziehbar sein, wann und durch wen personenbezogene Daten in Datenverarbeitungssysteme eingegeben worden sind.

8. Auftragskontrolle: Es muss gewährleistet sein, dass personenbezogene Daten, die im Auftrag verarbeitet werden, nur entsprechend den Weisungen des Auftraggebers verarbeitet werden können.

9. Transportkontrolle: Bei der Übertragung personenbezogener Daten oder dem Transport von Datenträgern muss gewährleistet sein, dass die Daten nicht unbefugt gelesen, kopiert, verändert oder gelöscht werden können.

10. Organisationskontrolle: Die innerbehördliche oder innerbetriebliche Organisation ist so zu gestalten, dass sie den besonderen Anforderungen des Datenschutzes gerecht wird.

Die genannten Maßnahmen beziehen sich in ihrem Ursprung auf den Schutz von Persönlichkeitsrechten. Dennoch sind sie, auch beim Entwurf von Strategien zum Schutz sensibler Daten jenseits datenschutzrechtlicher Aspekte, ein nützlicher Leitfaden. Mit den 10 Punkten werden die sicherheitskritischen Bereiche beim Umgang mit schutzwüdigen Informationen erfasst und denkbare Sicherheitslücken aufgezählt.

4.4 Bewertung von IT-Sicherheitsstrategien

Um IT-Systeme hinsichtlich ihrer Sicherheitskonzepte und -strategien bewerten zu können, bedarf es einheitlicher und allgemein anerkannter Kriterien, die von einer neutralen, unabhängigen und kompetenten Instanz formuliert und formalisiert werden. Die Kriterien sollen als Orientierung für die Entwicklung sicherer und vertrauenswürdiger Systeme dienen und eine objektive Bewertung dieser Systeme ermöglichen. Seit den 1980ger Jahren wurden von verschiedenen öffentlichen Einrichtungen solche Kriterien vor dem Hintergrund des aktuellen Kenntnisstandes entwickelt. Zunächst schien das Thema der IT-Sicherheit ausschließlich für militärische Zwecke interessant zu sein: Die ersten Ansätze wurden vom US-Verteidigungsministerium (**DoD**=*Department of Defense*) entwickelt oder in Auftrag gegeben. Auslöser dafür war nicht zuletzt eine gezielte Abgrenzung der militärischen Sicherheitsbedürfnisse von den Möglichkeiten kommerzieller Produkte. Das machte den Weg frei für eigene Forschungs- und Entwicklungsarbeiten auf diesem Gebiet. Im Laufe der Jahre mussten die Kriterien den rasanten Entwicklungen im IT-Bereich angepasst werden, was schließlich auch zu einer Internationalisierung der Kriterien führte.

4.4.1 1983: Trusted Computer Security Evaluation Criteria - TCSEC

Die *Trusted Computer Security Evaluation Criteria* (**TCSEC**) bildeten den ersten Meilenstein in der Historie der IT-Sicherheitskriterien und haben noch heute, trotz deutlich veränderter Voraussetzungen, einen hohen Stellenwert. In ihren Kernaussagen lieferten sie die Grundlage für alle späteren, bedeutenden Sicherheitskriterien. Als erste bedeutende Kriteriensammlung wurden TCSEC vom DoD aufgestellt und 1983 im sogenannten *Orange Book* veröffentlicht. Die Kriterien ermöglichen eine Zuordnung von IT-Systemen zu einer von vier Klassen, die zum Teil weitere Subklassen enthalten. Insbesondere bestehen zwischen den Sicherheitsklassen der TCSEC Inklusionsbeziehungen. Die Zugehörigkeit eines Systems zu einer Klasse bringt einen anerkannten Sicherheitsstandard zum Ausdruck.

- D - Minimaler Schutz: Dieser Klasse gehören alle Systeme an, die die Kriterien einer höheren Klasse nicht erfüllen können. Der Schutz von Daten kann als minimal bzw. als nicht gegeben angesehen werden. Systeme dieser Klasse verfügen über keine nennenswerten Sicherheitsstrategien.
- C - Freiwilliger Schutz: *Discretionary Access Control* - DAC
  - C1 - Freiwilliger Zugriffsschutz: Systeme der Klasse C1 treffen eindeutige Unterscheidungen zwischen Benutzern und Daten. Es obliegt den jeweiligen Besitzern von Daten, diese zu schützen. Das Recht eines Benutzers auf die Daten eines anderen Benutzers zuzugreifen, wird von diesem explizit erteilt oder verweigert.
  - C2 - Kontrollierter Zugriffsschutz: Systeme dieser Klasse erfüllen die Kriterien der Klasse C1 und unterstützen darüber hinaus Mechanismen der Zugriffskontrolle (Passwort-geschützte Anmeldeprozeduren, Überprüfung und Isolierung sicherheitsrelevanter Aktionen).
- B - Zwingender Schutz: *Mandatory Access Control* - MAC
  - B1 - Verbindliche Vergabe von Sicherheitsstufen: Systeme der Klasse B1 erfüllen die Kriterien der Klasse C2. Ausserdem existiert für diese Systeme ein informelles Modell einer Sicherheitspolitik, das die Einstufung von Daten und Benutzern auf Sicherheitsebenen beschreibt.
  - B2 - Strukturierter Schutz: Der Klasse B2 gehören alle Systeme an, die die Kriterien der Klasse B1 erfüllen und zusätzlich ein formales Modell zur Beschreibung der Sicherheitspolitik besitzen. Alle Subjekte und Objekte des Systems werden einer Sicherheitsebene zugeordnet. Die Systeme der Klasse B2 unterscheiden strukturell ihre sicherheitskritischen und ihre sicherheitsunkritischen Elemente. Verdeckte Informationskanäle werden formal von den Modellen erfaßt.
  - B3 - Sicherheitsbereiche: Der Klasse B3 werden Systeme zugeordnet, die die Kriterien der Klasse B2 erfüllen und über einen Referenz-Monitor verfügen. Dieser protokolliert und überwacht alle Zugriffe von Subjekten auf Objekte des Systems. Die Wiederherstellung früherer, konsistenter Zustände eines B3-Systems ist jederzeit möglich.
- A - Verifizierte Sicherheit: Der Klasse A werden Systeme zugeordnet, die der Klasse B3 angehören und ein formales, mathematisches Modell besitzen, auf dessen Grundlage die korrekte Arbeitsweise der angebotenen Sicherheitsfunktionen bewiesen werden kann.

4.4.2 1989: IT-Sicherheitskriterien - ITS

In den Jahren 1989 und 1990 entwickelte die damalige Zentralstelle für Sicherheit in der Informationstechnik (ZSI) für den deutschen Anwendungsraum gültige Sicherheitskriterien und veröffentlichte diese unter dem Namen *Grünbuch*. Das ZSI war als Zweigstelle des Bundesnachrichtendienstes (BND) das zuständige staatliche Organ für Sicherheitsaspekte in der Informationstechnik. Es wurde mit dem BSI-Errichtungsgesetz zum 1.1.1991 in ein staatliches Institut umgewandelt und direkt dem Bundesministerium des Inneren (BMI) unterstellt.
Das ITS führte 10 sicherheitsrelevante Funktionsklassen ein. Je nach Existenz und Stärke der Sicherheitsfunktionen kann ein IT-System einer von 8 hierarchischen Qualitätsstufen (Q0,...,Q7) zugeordnet werden. Q0 stellt die geringste, Q7 die höchste Qualitätsstufe mit jeweils steigenden Anforderungen an Art und Umfang der Sicherheitsprüfung dar. Die Qualitätsstufen bilden somit ein Maß für die Beurteilung der Sicherheit eines IT-Systems und machen implizit die Sicherheit zu einem Qualitätsmerkmal des Gesamtsystems. Die 10 Funktionsklassen beinhalten jeweils die Grundmechanismen sicherer Systeme mit abgestuften Anforderungen. Aus Sicht der ITS umfassen die Grundmechanismen der Sicherheit:

- Identifikation und Authentisierung von Anwendern.
- Verwaltung und Prüfung von Benutzerrechten.
- Beweissicherung, Wiederaufbereitung und Fehlerüberbrückung vor, während und nach Transaktionen.
- Gewährleistung der Funktionalität des Systems.
- Übertragungssicherung
  - durch Authentisierung von Sendern und Empfängern sowie
  - durch Zugriffskontrolle.

Die für ein IT-System erreichbaren 8 Qualitätssufen $Q0 - Q7$ können folgendermaßen dargestellt werden:

- Q0: Unzureichende Qualität.
- Q1: Die Sicherheitsanforderungen sind verbal formuliert. Das System wurde mit einfachen Tests auf die Erfüllung der Sicherheitsanforderungen geprüft.

- $Q2$: Die Sicherheitsanforderungen sind verbal formuliert. Das System wurde anhand der Spezifikationen methodisch getestet.
- $Q3$: Die Sicherheitsanforderungen sind detailliert verbal formuliert. Die Systemspezifikationen wurden teilweise informell analysiert. Das System wurde anhand der Spezifikationen methodisch getestet.
- $Q4$: Die Sicherheitsanforderungen sind detailliert verbal formuliert. Die Systemspezifikationen wurden vollständig informell analysiert. Das System wurde anhand der Spezifikationen methodisch getestet.
- $Q5$: Die Sicherheitsanforderungen sind detailliert verbal formuliert, die wichtigsten Sicherheitsanforderungen sind formal spezifiziert. Die Systemspezifikationen wurden mit semiformalen Methoden analysiert. Das System wurde anhand der Spezifikationen methodisch getestet.
- $Q6$: Die Sicherheitsanforderungen und Systemspezifikationen sind mit einer formalen Notation festgeschrieben. Die Systemspezifikationen wurden mit formalen Methoden analysiert. Das System wurde anhand der Spezifikationen methodisch getestet.
- $Q7$: Ergänzend zu $Q6$ wurde die Konsistenz zwischen dem Quellcode und den Spezifikationen formal verifiziert.

4.4.3 1991: Information Technology Security Evaluation Criteria - ITSEC

Mit dem Verschmelzen des europäischen Wirtschaftsraums zur Europäischen Union (EU) wurde Anfang der 1990er Jahre eine gegenseitige Anerkennung der Mitgliedstaaten von Zertifikaten und damit auch eine Vereinheitlichung von Bewertungskriterien angestrebt. Als Ergebnis der Vereinheitlichung von Sicherheitskriterien für IT-Systeme entstanden die **ITSEC**. Zur Bewertung eines IT-Systems wurde hierbei zwischen der Funktionalität und der Vertrauenswürdigkeit des Systems unterschieden. Die Mechanismen zur Steigerung der Vertrauenswürdigkeit wurden hinsichtlich der **Wirksamkeit** und ihrer **Korrektheit** beurteilt. Die Wirksamkeit beschreibt die Fähigkeit der Sicherheitsmechanismen, potenziellen Bedrohungen für die Sicherheit des Systems entgegenzuwirken. Die Beurteilung der Wirksamkeit der Sicherheitsmechanismen wird bei ITSEC durch die Zuordnung zu einer von drei Stufen ausgedrückt.

- Niedrig: Das System besitzt Mechanismen zum Schutz gegen zufällige Angriffe.

- Mittel: Das System ist durch seine Sicherheitsmechanismen gegen Angreifer geschützt, die ein eingeschränktes Risikopotential darstellen.
- Hoch: Die Sicherheitsmechanismen des Systems bieten ausreichenden Schutz gegen gezielte Angriffe mit hohem Risikopotenzial.

Zur Beurteilung der Vertraulichkeit des Systems hinsichtlich der Korrektheit werden sechs hierarchische Evaluierungsstufen $E1$ bis $E6$ eingeführt:

- $E1$: Informelle Beschreibung des Architekturentwurfs und funktionale Tests der Sicherheit.
- $E2$: Informelle Beschreibung des Feinentwurfs.
- $E3$: Bereitstellung des Quellcodes sowie des Hardwareentwurfs.
- $E4$: Existenz eines formalen Sicherheitsmodells und eine semiformale Beschreibung der sicherheitsrelevanten Funktionen.
- $E5$: Abbildbarkeit des Feinentwurfes auf den Quellcode und den Hardwareentwurf.
- $E6$: Existenz einer formalen Sicherheitsspezifikation.

4.4.4 1998: Common Criteria Version 2.0 - CC

Unter dem Titel *Common Criteria* (CC) wurden 1998 die gemeinsamen Kriterien mehrerer bedeutender Industrienationen zur Beurteilung der IT-Sicherheit zusammengefasst und von der internationalen Standardisierungsorganisation ISO zum formalen Standard (ISO 15408) erhoben ([9] und [7]). Beim Entwurf dieser Kriterien waren Vertreter Frankreichs, Deutschlands, Großbritanniens, der Niederlande, Kanadas und der USA beteiligt. Die CC können als formale Weiterentwicklung und Vereinheitlichung von TCSEC und ITSEC interpretiert werden. Die Sicherheitskriterien von TCSEC und ITSEC sind vollständig in den CC der Version 2.0 enthalten. Die CC basieren auf sieben Stufen der Vertrauenswürdigkeit eines Systems ($EAL1$,...,$EAL7$), die inhaltlich den Evaluierungsstufen der ITSEC entsprechen. Dabei bildet $EAL1$ eine Hierarchiestufe unterhalb von $E1$. Die übrigen Stufen $EALi$ sind jeweils equivalent mit den Evaluierungsstufen $E(i-1)$. CC sieht vor, dass die Anforderungen an die Funktionalität und die Vertrauenswürdigkeit von IT-Systemen in sogenannten Schutzprofilen für Produkttypen zusammengefasst werden.

Die CC bilden heute die allgemein anerkannte Basis für die Erteilung von Sicherheitszertifikaten. Die Vergabe derartiger Zertifikate an Softwareprodukte

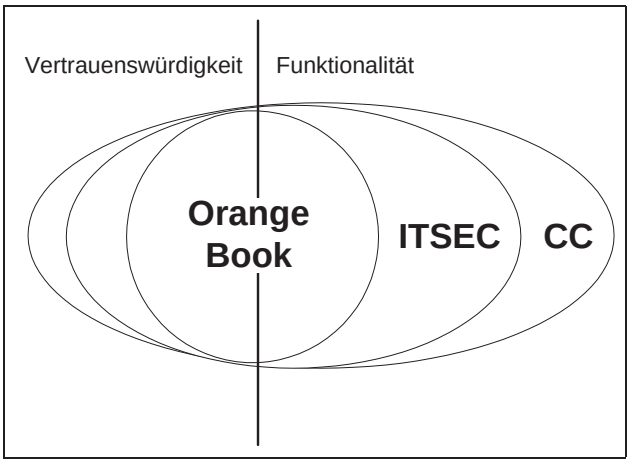

Abbildung 4.3: *Ausdrucksstärke der formalen Sicherheitskriterien. (Quelle:[9])*

oder Gesamtsysteme, mit der die Erfüllung einheitlicher und allgemein anerkannter Sicherheitskriterien bestätigt wird, ist sowohl für die Hersteller und Anbieter von Systemen, als auch für die Kunden und Anwender eine hilfreiche Orientierung. Hersteller können den Sicherheitsanspruch ihrer Produkte entsprechend den Sicherheitskriterien einstufen und Entwicklungsziele festlegen. Für Kunden und Anwender eröffnen sich Möglichkeiten des objektiven Vergleichs der Sicherheitsmechanismen verschiedener Produkte.

4.5 Formale Sicherheitsmodelle

Die Anforderungen an die Sicherheit eines IT-Systems werden in der Regel während der Entwurfsphase des Systems, zunächst informell, z.B. in einem Konzeptpapier, festgelegt. Diese Beschreibung der Sicherheitsanforderungen definiert die angestrebte Sicherheitspolitik, auf deren Grundlage die Daten und Dienste des Systems hinsichtlich ihrer sicherheitsrelevanz klassifiziert und die zukünftigen Anwender nach ihren Aufgaben, Erwartungen und ihren minimal benötigten Zugriffsrechten strukturiert werden. Die Sicherheitspolitik beschreibt aus abstrakter fachlicher Sicht die erforderliche Sicherheitsstrategie eines Systems, ohne diese hinsichtlich der einzusetzenden Methoden und der technischen Umsetzung zu konkretisieren. Um die konzeptionelle Sicherheitspolitik auf einem konkreten System umzusetzen, muss sie vor der Implementierung der Software, also der Übersetzung des Systementwurfs in den Quellcode einer Programmiersprache bzw. der Konfiguration von Systemkomponenten (*Bottom-up* Entwurf), durch ein geeignetes formales Sicherheitsmodell spezifiziert werden. Der Abschnitt 4.4 hat gezeigt, dass ein IT-System nur durch die

Existenz eines formalen Sicherheitsmodells eine hohe sicherheitsbezogene Qualität im Sinne allgemein anerkannter Kriterien erreichen kann. Zwei formale Zugriffsmodelle, die sich in den Bereichen Betriebssysteme und Datenbanken etabliert haben, sollen in diesem Abschnitt vorgestellt werden. Zunächst wird das *Access Matrix* Modell skizziert, das eine freiwillige Zugriffskontrolle im Sinne von DAC formalisiert. Systeme, die über eine *Access Matrix* verfügen, können in Verbindung mit einem geeigneten Gesamtkonzept eine Einstufung in die Sicherheitsklasse C der TCSEC erreichen. Im Anschluss daran werden die zentralen Eigenschaften des Modells von Bell und LaPadula dargestellt, mit dem eine Zugriffsstrategie mit verbindlicher Sicherheitsklassifizierung vorgeschlagen wird. Die Realisierung einer solchen MAC-Strategie ist eine minimale Vorraussetzung für eine Einstufung in die Klasse B der TCSEC. Am Schluss dieses Kapitels wird kurz ein erster wissenschaftlicher Ansatz zur Vergabe von Zugriffsrechten für objektorientierte Datenmodelle skizziert.

4.5.1 Das *Access-Matrix* Modell

Die freiwillige Zugriffskontrolle (DAC), wie sie im *Access Matrix* Modell angewendet wird, unterteilt die für die Sicherheit relevanten Komponenten einer Datenbank oder eines Betriebssystems in:

- Objekte O: Die Menge der zu schützenden Elemente eines Systems. Damit sind Laufwerke, Verzeichnisse, Dateien, Datenbanken, Datensätze oder atomare Daten gemeint.
- Subjekte S: Die Menge der Benutzer oder Prozesse, die auf die Objekte eines Systems zugreifen.
- Zugriffsrechte AR: Die Menge der möglichen Zugriffsrechte (Lesen, Schreiben, Ändern, Löschen, ...), die einem Subjekt für ein Objekt zugewiesen werden können.

Mit der Unterscheidung der Datenbankbestandteile nach Subjekten, Objekten und Zugriffsrechten wird es möglich, die Privilegien von Benutzern in die Felder einer Matrix einzutragen, mit der die Subjekte einer Datenbank den Objekten gegenüberstellt werden. Neben den Subjekten, die bestimmte Aktionen auf Objekten ausführen möchten, existiert für jedes Objekt $o \in O$ des Systems genau ein Besitzer, der für das Objekt ein besonderes Subjekt $s \in S$ darstellt. Dieses besitzt als einziges das Privileg, Zugriffsrechte an seinen eigenen Objekten anderen Subjekten zu erteilen.

Die Basis des DAC lässt sich in drei Aussagen zusammenfassen:

- Jedes Subjekt $s \in S$ ist selber verantwortlich für den Schutz der Objekte $o \in O$, deren Besitzer es ist.
- Jedes Subjekt $s \in S$ kann Zugriffsrechte $r \in AR$ für seine eigenen Objekte $o \in O$ an andere Subjekte $s \in S$ vergeben.
- Jedes Subjekt $s \in S$ kann auf einem Objekt $o \in O$ die Operationen ausführen, die innerhalb seiner Zugriffsrechte für das Objekt liegen.

Verlangt ein Subjekt Zugriff auf ein Objekt, so überprüft und verifiziert eine Zugriffsfunktion f seine Berechtigungen an diesem Objekt:

$$f : S \times O \times AR \longrightarrow \{true, false\} \tag{4.1}$$

Für $s \in S$, $o \in O$ und $r \in AR$ bedeutet dies: Wenn $f(s, o, r) = true$, dann ist r Bestandteil der Zugriffsrechte, die das Subjekt s an dem Objekt o hat.

Das *Access-Matrix* Modell implementiert die Zugriffsfunktion f in einer Zugriffsmatrix M.

	o1	**o2**	**o3**	**o4**
s1	r	w	rw	-
s2	a	rw	-	rx
s3	x	-	rw	- r

Abbildung 4.4: *Zugriffsmatrix mit 3 Subjekten und 4 Objekten.*

Die Zugriffsmatrix in Abbildung 4.4 beschreibt die Zugriffsrechte der Subjekte $S = \{s1, s2, s3\}$ für die Objekte $O = \{o1, o2, o3, o4\}$ mit der Rechtemenge $AR = \{r, w, x, a, -\}$ (Lesen, Schreiben, Löschen, Vollzugriff, keine Rechte). Zugriffsstrukturen, die sich an den Ideen des *Access Matrix* Modells orientieren, haben weite Verbreitung in kommerziellen Betriebssystemen und Datenbanken gefunden. Zugriffsmatrizen sind einfach zu implementieren und auszuwerten. Die Verantwortung für die Erteilung von Privilegien wird auf die Eigentümer der Daten verteilt und ist für diese leicht nachvollziehbar. So kann ein effizienter Schutz von Daten mit relativ geringem Aufwand realisiert werden.

4.5.2 Das Zugriffsmodell von Bell und LaPadula

Sicherheitskonzepte für Informationssysteme auf der Grundlage von Zugriffsmatrizen stellen für vergleichsweise einfach strukturierte Datenmodelle mit einer relativ geringen Anzahl von Einzelobjekten einen sinnvollen Ansatz zur

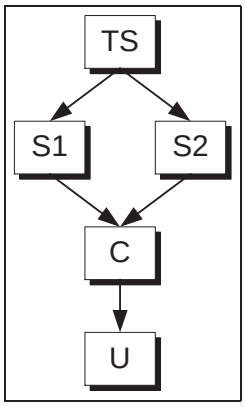

Abbildung 4.5: *Eine mehrsufige Zugriffshierarchie nach Bell/LaPadula.*

Verwaltung von Benutzerrechten dar. Erhöht sich jedoch die Komplexität des zu Grunde liegenden Datenmodells, mit dem Ziel einer realitätsnahen Beschreibung von Objekten einer Fachaufgabe, so sind die Möglichkeiten eines umfassenden Berechtigungskonzeptes mit Zugriffsmatrizen sehr eingeschränkt: Die Privilegien jedes Benutzers an jedem elementaren Objekt des Informationssystems müssen explizit definiert und gepflegt werden. Zwischen den einzelnen Feldern der Zugriffsmatrizen können keine Beziehungen beschrieben werden. Abhängigkeiten zwischen verschiedenen Privilegien eines Benutzers, die sich aus dem Datenmodell ergeben, können somit nur durch aufwendige Zusatzstrukturen und Algorithmen abgebildet werden.
Vor dem Hintergrund dieser Einschränkungen bei der Benutzung von Zugriffsmatrizen lag das zentrale Anliegen von Bell und LaPadula in der Konzeption einer Zugriffsstrategie, die es ermöglichte, mit einer geringen Anzzahl expliziter Festlegungen ein breites Spektrum untereinander konsistenter Privilegien abzubilden. Ausgangspunkt einer solcher Strategie ist die Einführung von Zugriffsstufen, denen die Subjekte und Objekte eines Systems zugewiesen werden. Aus den Zugriffsstufen von Subjekten und Objekten sollen die Zugriffsrechte implizit ableitbar sein. Aufgrund der sicherheitsspezifischen Vergleichbarkeit der Zugriffsstufen und durch das Konzept, jedes Subjekt und jedes Objekt einer Stufe zuzuweisen, ergibt sich eine Berechtigungsstrategie nach MAC-Kriterien mit dem Ziel verbindlicher Zugriffskontrollen für alle Benutzer hinsichtlich aller Objekte.

Die Menge der Zugriffsrechte AR aus dem *Access Matrix* Modell wird ersetzt durch die Zugriffstypen $ACC = \{read, write, append\}$ sowie einer Menge von Klassifikationen L, auf der eine Halbordnung mit den Operatoren $<$ und $\sim$ definiert wird. Dabei gilt für je zwei $l_1, l_2 \in L$: $l_1 < l_2 \vee l_2 < l1 \vee l_1 \sim l_2$.

Mit $\sim$ wird die Nicht-Vergleichbarkeit zweier Stufen zum Ausdruck gebracht. Abbildung 4.5 zeigt ein Beipiel einer solchen Halbordnung auf Zugriffsstufen. Die Zuweisung von Subjekten und Objekten zu Zugriffsstufen ist eine Funktion *level*, die als Klassifikation (*Classificaton*) bzw. Vertrauenswürdigkeit (*Clearance*) bezeichnet wird:

$$level : S \cup O \longrightarrow L \tag{4.2}$$

Die Menge der Operationen OP definiert alle Aktionen, die ein Subjekt auf einem Objekt potenziell ausführen kann. Jede Operation lässt sich eindeutig einem Zugriffstyp zuordnen.

$$acc : OP \longrightarrow ACC \tag{4.3}$$

Bell und LaPadula charakterisieren die Zulässigkeit eines Datenzugriffs (s, o, op) $\in S \times O \times OP$ mit drei Eigenschaften:

- *Simple Secure Property* : $acc(op) = read \Rightarrow level(o) \leq level(s)$ (*„no read up"*).
- ** Property* : $acc(op) = append \Rightarrow level(o) \geq level(s)$; $acc(op) = write \Rightarrow level(o) = level(s)$ (*„no write down"*).
- *Discretionary Secure Property* : Die Zelle in der Zugriffsmatrix enthält *op*.

Das Zugriffsmodell von Bell und LaPadula kann als inhaltliche Erweiterung der freiwilligen Zugriffskontrolle um Konzepte der verbindlichen Vertrauenswürdigkeit von Subjekten und der Klassifikation von Objekten interpretiert werden. Die *Simple Secure Property* liefert den grundsätzlichen Ansatz zur Geheimhaltung von Informationen. Ein Subjekt - in der Regel ein Benutzer - soll nur Daten einsehen können, deren Klassifikationen von der Vertrauenswürdigkeit des Benutzers dominiert werden. Im Sinne der Zugriffshierarchie darf der Benutzer nur Daten einsehen, deren Zugriffsstufe von der des Benutzers auf einem Pfad erreichbar sind.
Die ** Property* ist aus der Absicht heraus motiviert, verdeckte Informationskanäle (*Covert Channels*) aufzudecken und zu vermeiden, die durch schreibenden Zugriff auf Datenbanken entstehen können. Einem Benutzer oder Prozess soll es nicht möglich sein, Daten auf eine niedrigere Klassifikations-Ebene zu schreiben und damit potenziell vertrauliche Informationen (versehentlich oder beabsichtigt) auf niedrigeren Zugriffsstufen sichbar zu machen. Eine typische

Bedrohung dieser Art geht von „Trojanischen Pferden" aus, die sich als versteckter Prozess an ein Subjekt hoher Klassifikation anhängen, um dann im Hintergrund geheime Informationen an eine niedrigere Klassifikations-Ebene zu schicken.
Mit der *Discretionary Secure Property* behält der Eigentümer von Daten die Möglichkeit, für diese zusätzlich einschränkende Zugriffsrechte nach den Methoden des *Access-Matrix* Modells zu erteilen.
Mit dem Bell/LaPadula Modell werden Zugriffsrechte auf Objekte für Subjekte durch die Zuweisung und die Vergleichbarkeit von Zugriffsstufen implizit erteilt. Zu jedem Subjekt wird lediglich deren Vertrauenswürdigkeit und zu jedem Objekt deren Klassifikation gespeichert. Durch Änderungen in den Zuweisungen können Zugriffsrechte auf einfache und konsistente Weise einer neuen Sicherheitspolitik angepasst werden.
Die Ansätze des Bell/LaPadula Modells eignen sich insbesondere für die Erteilung von Zugriffsrechten auf Objektstrukturen mit relativ geringer Komplexität. Solche Strukturen finden sich in der Datei- und Verzeichnisverwaltung von Betriebssystemen und in relationalen Datenbanken mit wenigen Fremdschlüsseln zwischen den einzelnen Tabellen wieder. Mit steigender Komplexität des Datenmodells steigt jedoch die Gefahr des Auftretens verdeckter Informationskanäle oder der Nichtdurchsetzbarkeit von Geheimhaltungsforderungen. Ursächlich für diese Gefahr ist die Möglichkeit der Existenz von Beziehungen bekannter Kardinalität, zwischen Objekten unterschiedlicher Klassifikationen. Mit objektorientierten Realweltmodellen werden gerade derart komplexe Objektstrukturen in geeigneter Weise beschreibbar. Insofern wird auch für die Beschreibung von Zugriffsrechten für objektorientierte Datenmodelle ein erweitertes und geeignetes Konzept benötigt, das die Beziehungen und Abhängigkeiten zwischen Objekten, deren Teilaspekten und ihren Beschreibungen berücksichtigt und in das Gesamtkonzept mit einbezieht.
Mehrstufige Zugriffsstrukturen nach den Ideen von Bell und LaPadula haben sich, aufgrund des erhöhten Kosten und Entwicklungsaufwands bei Anbietern kommerzieller Datenbanken nur vereinzelt und in Ansätzen durchgesetzt. Dennoch gibt es eine Reihe spezieller Entwicklungen für hochsensible Anwendungen, insbesondere aus dem militärischen Bereich, bei denen auf diese Ideen zurückgegriffen wurde.

4.6 Ansatz objektorientierter Berechtigungen

Bei der Modellierung von Pänomenen der realen Welt steht die Semantik der Daten und weniger deren Speicherstruktur im Vordergrund. Folglich wäre es sinnvoll, über ein Zugriffs- und Berechtigungskonzept zu verfügen, mit dem

es möglich ist, explizite Berechtigungsstrukturen nach fachlichen und semantischen Aspekten zu definieren und konsistenzerhaltende Eigenschaften sowie implizite Rechte und Verbote direkt aus dem Datenmodell abzuleiten. Objektorientierte Datenmodelle eignen sich in besonderer Weise zur Definition implizierender Berechtigungsstrukturen, da sich neben den fachlichen Abhängigkeiten auch die Vererbungshierarchien von Klassen in ein Berechtigungsmodell integrieren lassen. Rabitti[28] und Jajodia[24] haben in ihren Arbeiten Ansätze zur Abbildung von Zugriffsstrukturen auf objektorientierte Datenmodelle vorgestellt.
Ausgangspunkt der Betrachtungen ist zunächst wiederum die Vorstellung, die Autorisierungen durch eine Menge von Tripeln aus $S \times O \times A$ mit der Menge der Subjekte S, der Menge der Objekte O und der Menge der Autorisierungstypen A (z.B. lesen, schreiben oder löschen) zu definieren. Jedes in der Datenbank existierende Recht $a \in A$, das ein Subjekt $s \in S$ an einem Objekt $o \in O$ haben soll, wird als Tripel (s,o,a) kodiert und explizit in der Datenbank gespeichert. Um den Verwaltungsaufwand und den Speicherbedarf zu reduzieren und gleichzeitig das Rechtesystem mit einer semantischen Struktur zu versehen, wird das Konzept der *impliziten Autorisierung* - also der Ableitung weiterer Berechtigungen aus vorhandenen Berechtigungen - auf den drei Ebenen Subjekt, Objekt und Autorisierungstyp eingeführt. Dabei wird berücksichtigt,

- in welchem Verhältnis die Subjekte zueinander stehen,
- welche Abhängigkeiten zwischen den Objekten bestehen und
- in welcher Weise die Autorisierungstypen voneinander abhängen.

Als Ergebnis werden Autorisierungen nach drei Aspekten charakterisiert:

1. **Explizite und implizite Autorisierungen:** *Explizite Autorisierungen* umfassen alle unmittelbar definierten Berechtigungen aus A, die eindeutig benannte Subjekte aus S an eindeutig benannten Objekten aus O besitzen sollen und zu diesem Zweck in geeigneter Form als Tripelmenge $A_E \subseteq S \times O \times A$ gespeichert werden. Die Menge der tatsächlichen Berechtigungen, die ein Subjekt an einem Objekt besitzt, ergibt sich dadurch, dass seine explizit definierten Berechtigungen um die Berechtigungen ergänzt werden, die sich *implizit* und rekursiv aus seinen Berechtigungen und dem verwendeten Datenmodell ableiten lassen. Die Menge aller abgeleiteten Berechtigungen von Subjekten an Objekten können ebenfalls als Tripelmenge $A_I \subseteq S \times O \times A$ dargestellt werden. Diese Menge wird als *implizite Autorisierung* bezeichnet. Die Menge aller Autorisierungen - also die Gesamtheit aller Berechtigungen der Datenbank -

ergibt sich aus der Vereinigung expliziter und impliziter Autorisierungen zu $A_{EI} = A_E \cup A_I$. Wenn eine Autorisierung $r \in A_{EI}$ eine Autorisierung $p \in A_{EI}$ impliziert, kann dafür auch kurz $r \rightarrow p$ geschrieben werden.

2. **Positive und negative Autorisierungen:** Bei den bisherigen Zugriffsmodellen wurde ausschließlich von der Vergabe positiver Berechtigungen ausgegangen - also der expliziten Gewährung des beschriebenen Zugriffstyps eines Subjektes für ein Objekt. Ein nicht aufgeführtes Zugriffsrecht bedeutete implizit immer ein Verbot. Das hier beschriebene Konzept beinhaltet zusätzlich negative Autorisierungen - also die explizite Definition von verbotenen Zugriffen von Subjekten auf Objekte der Datenbank. Der Nutzen dieser Form der Autorisierung wird durch die nachfolgenden Aspekte verdeutlicht.

3. **Starke und schwache Autorisierungen:** Um die Gefahr möglicher Konflikte im Autorisierungsmodell zu minimieren, werden Autorisierungen nach *starker* und *schwacher* Autorisierung unterschieden. Konflikte können zum Beispiel entstehen, wenn sich aus zwei expliziten Autorisierungen widersprüchliche implizite Autorisierungen ableiten lassen. Starke Autorisierungen stellen grundlegende und dominierende Eigenschaften des Berechigungsmodells dar. Von starken Autorisierungen abgeleitete Autorisierungen sind ebenfalls stark. Starke Autorisierungen können von anderen Autorisierungen nicht überschrieben werden. Dem gegenüber sind schwache Autorisierungen und deren Ableitungen mehr als Vorschlag einer Berechtigungsstruktur zu verstehen. Sie behalten nur so lange ihre Gültigkeit, bis sie von einer anderen, unvereinbaren (starken oder schwachen) Autorisierung überschrieben werden. Konflikte im Berechtigungsmodell können vermieden werden, wenn nur wenige, konfliktfreie und für die Durchsetzung einer Sicherheitspolitik unverzichtbare Autorisierungen als stark definiert werden. Die übrigen, angestrebten Autorisierungen werden als schwach eingestuft, um sich somit sofort einem veränderten Kontext anzupassen. In diesem Zusammenhang ist die Unterscheidung zwischen positiven und negativen Autorisierungen von besonderem Nutzen, da mit Ihrer Hilfe eine erteilte, schwache Autorisierung durch Überschreibung in ihr Gegenteil verändert werden kann - Rechte werden zu Verboten, Verbote werden zu Rechten.

Abbildung 4.6 zeigt das Beispiel einer Struktur zur Autorisierung von der Systemebene bis zu Instanzen einzelner Klassen der Datenbank. Verfolgt man einen Pfad im dargestellten Graphen, so wird deutlich, wie starke und schwache bzw. explizite und implizite Autorisierungen sich gegenseitig beeinflussen. Beispielsweise wird die explizite schwache Autorisierung der *Bonn* Datenbank

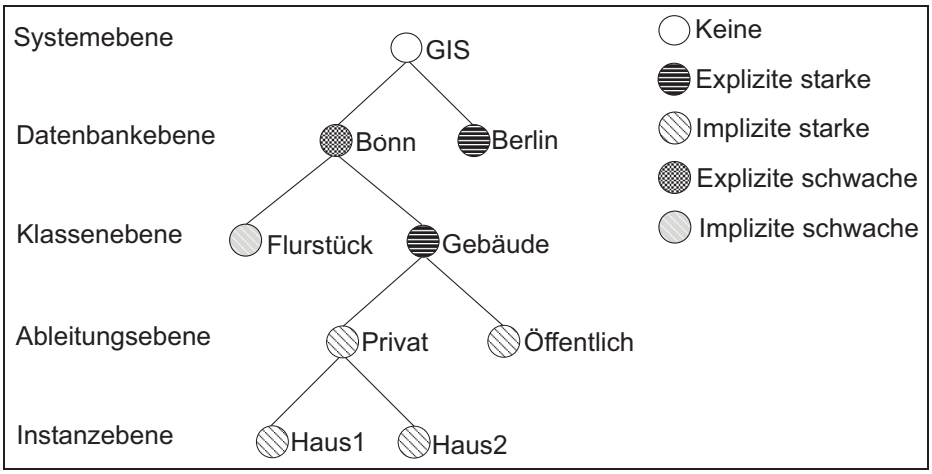

Abbildung 4.6: *Formen der Autorisierung und deren Abhängigkeiten.*

auf die untergeordnete Klassenebene abgebildet. Die definierte Autorisierung wird von der Klasse *Flurstück* implizit übernommen, von der Klasse Gebäude dagegen explizit als starke Autorisierung überschrieben. Diese explizite starke Autorisierung wird dann von allen abgeleiteten Klassen und deren Instanzen implizit übernommen. Die Menge der starken expliziten Autorisierungen wird als Autorisierungsbasis AB, die Menge der schwachen expliziten Autorisierungen wird als schwache Autorisierungsbasis SAB bezeichnet. Zur Gewährleistung der Widerspruchsfreiheit werden zusätzlich folgende Regeln festgelegt:

1. **Konsistenz:** Für jede Autorisierung $(s, o, a) \in AB$ (positiv oder negativ) gilt: Wenn ein $(s_i, o_i, a_i) \in A_{EI}$ existiert mit $(s, o, a) \rightarrow (s_i, o_i, a_i)$, dann darf kein $(s_j, o_j, a_j) \in AB$ mit $(s_j, o_j, a_j) \rightarrow (s_i, o_i, \neg a_i)$ existieren. Für jede Autorisierung $(s, o, a) \in SAB$ (positiv oder negativ) gilt: Wenn ein $(s_i, o_i, a_i) \in A_{EI}$ existiert mit $(s, o, a) \rightarrow (s_i, o_i, a_i)$, dann darf kein $(s_j, o_j, a_j) \in SAB$ mit $(s_j, o_j, a_j) \rightarrow (s_i, o_i, \neg a_i)$ existieren.

2. **Redundanzfreiheit:** Für jede Autorisierung $(s, o, a) \in AB$ (positiv oder negativ) gilt: Wenn ein $(s_i, o_i, a_i) \in A_{EI}$ existiert mit $(s, o, a) \rightarrow (s_i, o_i, a_i)$, dann ist $(s_i, o_i, a_i) \notin AB$

3. **Vollständigkeit:** Für jede starke Autorisierung $(s_i, o_i, a_i) \in A_{EI} \setminus AB$ (positiv oder negativ) muss eine starke Autorisierung $(s, o, a) \in A_{EI}$ existieren, so dass $(s, o, a) \rightarrow (s_i, o_i, a_i)$. Für jede schwache Autorisierung $(s_j, o_j, a_j) \in A_{EI} \setminus SAB$ (positiv oder negativ) muss eine schwache Autorisierung $(s, o, a) \in SAB$ existieren, so dass $(s, o, a) \rightarrow (s_j, o_j, a_j)$

Die *Konsistenz*-Regel sorgt dafür, dass aus der gleichen Autorisierungsbasis keine zwei Autorisierungen abgeleitet werden können, die für das gleiche Subjekt an dem gleichen Objekt den gleichen Autorisierungstyp, sowohl positiv als auch negativ beschreiben. Die Regel der *Redundanzfreiheit* minimiert die Autorisierungsbasis durch Verzicht auf die explizite Beschreibung jeder Berechtigung, die sich aus bereits bestehenden Berechtigungen ableiten lässt. Durch die *Vollständigkeits*-Regel wird der wesentliche Charakter von Autorisierungsbasen festgelegt: Jede starke bzw. schwache Autorisierung ist entweder Element der starken bzw. schwachen Autorisierungsbasis oder kann aus dieser abgeleitet werden. Die Autorisierungen des gesamten Systems sind also durch die starke und schwache Autorisierungsbasis vollständig beschrieben.
Die Implikationen der Autorisierungsregeln können in dem vorliegenden Berechtigungsmodell aus den sogenannten *Gittern* für die drei Komponenten Subjekte, Objekte und Autorisierungstypen abgeleitet werden. Diese Gitter sind drei zyklenfreie gerichtete Graphen:

1. Für Subjekte das *Autorisierungs-Rollen-Gitter* $ARG = (V_R, E_R)$ mit der Knotenmenge $V_R = S$ und der Kantenmenge $E_R \subset S \times S$,

2. für Objekte das *Autorisierungs-Objekte-Gitter* $AOG = (V_O, E_O)$ mit der Knotenmenge $V_O = O$ und der Kantenmenge $E_O \subset O \times O$ und

3. für Autorisierungstypen das *Autorisierungs-Typen-Gitter* $ATG = (V_A, E_A)$ mit der Knotenmenge $V_A = A$ und der Kantenmenge $E_A \subset A \times A$.

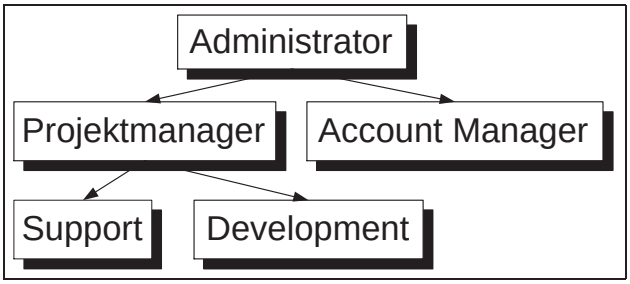

Abbildung 4.7: *Autorisierungs-Rollen-Gitter.*

Ein Beispiel eines *Autorisierungs-Rollen-Gitters* ist in Abbildung 4.7 dargestellt. Abbildung 4.8 zeigt das *Autorisierungs-Objekte-Gitter*, der bereits in Abbildung 4.6 dargestellten Autorisierungs-Implikationen.

In Abbildung 4.9 ist ein einfaches Beispiel eines *Autorisierungs-Typen-Gitters* skizziert.

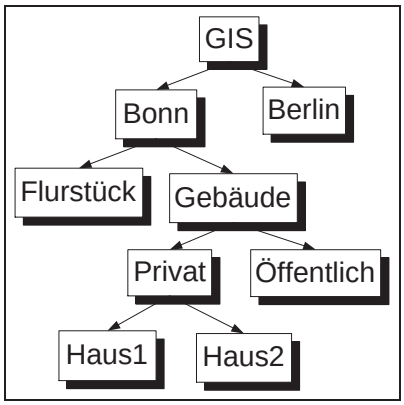

Abbildung 4.8: *Autorisierungs-Objekte-Gitter.*

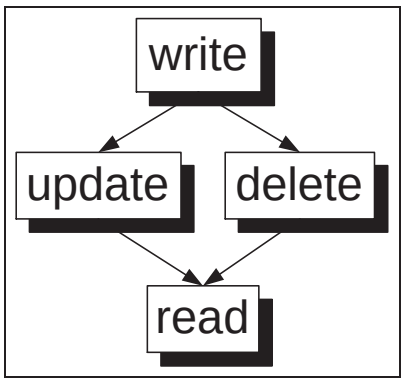

Abbildung 4.9: *Autorisierungs-Typen-Gitter.*

Wenn in einem gerichteten Graphen $G = (V, E)$ für zwei Knoten $v_i, v_j \in V$ eine Kante von v_i nach v_j existiert $(v_i, v_j) \in E$, dann schreiben wir kurz $v_i > v_j$.

Für zwei identische Knoten $v_i, v_j \in V$ schreiben wir kurz $v_i = v_j$. Zusätzlich sind die Relationen $>$ und $=$ transitiv. Die Semantik der Graphen hinsichtlich des Berechtigungsmodells ist in den folgenden Regeln zusammengefasst.

- **1.Regel :**Für alle $o \in O, a \in A$ und $s_i, s_j \in S$ gilt: $(s_i \geq s_j) \Longrightarrow (s_j, o, a) \rightarrow (s_i, o, a) \wedge (s_i, o, \neg a) \rightarrow (s_j, o, \neg a)$.

- **2.Regel :**Für alle $o \in O, s \in S$ und $a_k, a_l \in A$ gilt: $(a_k \geq a_l) \Longrightarrow (s, o, a_k) \rightarrow (s, o, a_l) \wedge (s, o, \neg a_l) \rightarrow (s, o, \neg a_k)$.

- **3.Regel :**Für alle $s \in S, a \in A$ und $o_m, o_n \in O$ gilt: $(o_m \geq o_n) \Longrightarrow (s, o_m, a) \rightarrow (s, o_n, a) \wedge (s, o_n, \neg a) \rightarrow (s, o_m, \neg a)$.

Das hier beschriebene Autorisierungsmodell, wie es von Rabitti und Jajodia entworfen wurde, liefert geeignete Methoden, um die charakteristischen Eigenschaften objektorientierter Datenmodelle, wie z.B. Klassenvererbung oder Objektbeziehungen, auf hierarchische Berechtigungsstrukturen abzubilden. Die grundsätzlichen Ideen dieses Berechtigungskonzeptes werden in Kapitel 6, beim Entwurf eines Berechtigungskonzeptes für objektorientierte Geodatenmodelle, wieder aufgegriffen. Kritisch ist anzumerken, dass beim Entwurf der Autorisierungsgitter nicht auf die komplexen fachlichen und semantischen Abhängigkeiten eingegangen wird, die bei einer Realweltmodellierung zwischen Klassen, Instanzen und Benutzern entstehen. Werden diese Abhängigkeiten bei der Modellierung der Autorisierungsgitter nicht berücksichtigt, können Sicherheitslücken entstehen. Die Vermeidung dieses Schwachpunktes im Berechtigungskontept ist eine der Zielsetzungen, die mit den Zugriffsstrategien der folgenden Abschnitte erreicht werden sollen.

4.7 Zusammenfassung

Die etablierten formalen Sicherheitsmodelle für Datenbanken beziehen sich im allgemeinen auf die Anforderungen, die im Umfeld relationaler Datenmodelle entstehen. Die theoretischen Konzepte gehen dabei von verhältnismäßig flachen Modellstrukturen aus. Die praktische Umsetzung dieser Konzepte basiert in der Regel auf einer Definition von Sichten mittels geeigneter Anfragesprachen. Für Nicht-Standardanwendungen, denen insbesondere auch objektorientierte GIS angehören, müssen Zugriffsbeschränkungen den Anforderungen komplexer Sachverhalte und Datenmodelle genügen, wie sie im Kapitel 3 dargestellt wurden. Hierbei ist das zentrale Problem neben dem Fehlen geeigneter deklarativer Anfragesprachen für objektorientierte Modelle, die Abbildung von Vererbungen und komplexen, zusammengesetzten Objekten. Die in der Literatur verfolgten Ansätze zur Beschreibung von Zugriffsrechten für objektorientierten Datenbanken sind für die praktischen Anforderungen der Anwender von GIS nicht ausreichend, da in diesen Konzepten die Granularität von Zugriffseinheiten auf Klassen- oder *Custer*-Größe reduziert wird, um Konflikte und Zugriffskanäle zu vermeiden. Im Folgenden soll das Problem der Zugriffsbeschränkung für Objektstrukturen näher betrachtet und die Auswirkungen einer Erteilung von Zugriffsrechten für ausgewählte Objekte oder Objektteile untersucht werden.

Kapitel 5

Zugriffe auf objektorientierte Geodaten

In den vergangenen Kapiteln - insbesondere in Kapitel 3 - wurden die Vorteile objektorientierter Techniken bei der Beschreibung raumbezogener Sachverhalte hervorgehoben. Objektorientierte Datenmodelle und in letzter Konsequenz die entsprechenden Daten*bank*-Modelle, orientieren sich bei der Abbildung räumlicher und fachlicher Informationen an der Semantik der abzubildenden Phänomene. Dabei können die Anforderungen von Fachaufgaben und spezielle Interessenslagen unmittelbar in das Datenmodell einfließen. Objektcharakter, fachliche Beziehungen, Abhängigkeiten und Funktionen sind impliziter Bestandteil objektorientierter Modelle. Diese Modelleigenschaft hat unmittelbare Auswirkungen auf die Definition und Durchsetzbarkeit von Zugriffsbeschränkungen, da mit dem Datenmodell eine fachliche und strukturelle Intention und Interpretation verknüpft ist. In den folgenden Abschnitten werden die grundlegenden objektorientierten Modellierungstechniken hinsichtlich ihrer Bedeutung für die Beschreibung raumbezogener Daten erläutert und die durch Anwendung der jeweiligen Technik implizierten Abhängigkeiten bei der Erteilung von Zugriffsrechten untersucht. Anhand von Modellierungs-Beispielen sollen dabei die besonderen Anforderungen objektorientierter Geodaten-Modelle an geeignete Berechtigungskonzepte verdeutlicht werden.

5.1 Klassen und Instanzen

Klassen beschreiben Eigenschaften und Verhaltensmuster von thematisch und strukturell zusammengehörigen (Geo-)Objekten. Die Objekte, die den Beschreibungen einer Klasse genügen, werden als Instanzen dieser Klasse bezeich-

net. Die Menge aller Instanzen einer Klasse bildet den *Extent* der Klasse. Formate, Datentypen, Wertebereiche, Kardinalitäten und inhaltliche Abhängigkeiten konkreter Eigenschaften von Objekten einer Klasse werden mit der Definition von Attributen festgelegt. Dabei bezieht sich die Definition eines Attributs auf Basisdatentypen wie Ganzzahl, Gleitkommazahl, Logischer Wert und Zeichenkette oder auf komplexere Datentypen wie Geometrie, Dokument oder Verwaltungsbezirk, die sich wiederum aus der Definition anderer Klassen ergeben. Mit jeder Definition einer Klasse wird ein neuer Datentyp eingefügt. Diese können einfacher oder komplexer Natur, raumbezogen oder nicht-raumbezogen sein. Klassenmethoden definieren und implementieren die Verhaltensweisen von Objekten der jeweiligen Klasse, die als Reaktion auf externe Benachrichtigungen oder Ereignisse ausgelöst werden. Ein zentraler Zugriffsschutz für die Daten objektorientierter Modelle ergibt sich unmittelbar aus dem Konzept der Datenkapselung. Dabei werden die schutzwürdigen Informationen von Instanzen einer Klasse hinter entsprechenden Zugriffsmethoden verborgen. Die Definition von Klassen unterteilt sich zu diesem Zweck in einen öffentlichen (*public*) und einen privaten (*private*) Bereich.
Für Zugriffe auf Objekte einer Klasse steht grundsätzlich nur der öffentlichen Bereich zur Verfügung. Dieser definiert die externe Kommunikationsschnittstelle der Klasse. Um kontrollierten Zugriff auf Objekte zu gewährleisten, sollten sich die Attribute im privaten Bereich der Klassenbeschreibungen befinden. Im öffentlichen Bereich der Klassendeklaration können dann Zugriffsmethoden definiert werden, die nicht nur den Zugriff auf Attribute hinter Methoden verbergen, sondern auch die Art des Zugriffs festlegen. So kann bereits in der Schnittellenfinition festgelegt werden, dass nur bestimmte Attribute abgefragt, andere nach bestimmten Kriterien ausgewertet (*Objektwert* < *Grenzwert ?*) und wieder andere nur in vorgegebener Weise modifiziert werden dürfen. Konzeptionell können die Zugriffsmethoden einer Klasse entsprechend ihrer Zugriffsart nach lesenden und schreibenden Methoden unterschieden werden. In Programmiersprachen wie C++ wird diese Unterscheidung durch die prototypische `const`-Deklaration von Klassenmethoden spezifiziert. Damit werden Methoden deklariert, die keine Veränderungen an den Eigenschaften eines Objektes auslösen dürfen. Aus Sicherheitserwägungen sind schreibende Methoden grundsätzlich kritischer einzustufen als `const`-Methoden. Gestützt wird diese Annahme inbesondere durch zwei Feststellungen:

1. Durch wiederholtes Einfügen und Modifizieren von Daten mit dem Ziel, bekannte Integritätsregeln (zum Beispiel Eindeutigkeits- oder Beziehungseigenschaften) zu verletzen und damit die Ablehnung einer Transaktion zu provozieren, kann der Inhalt geheimer Daten, zumindest in Teilen, ausspioniert werden.

2. Durch schreibenen Zugriff können Daten nicht nur unbefugt gelesen und zweckentfremdet, sondern auch gezielt verändert oder zerstört und damit unbrauchbar gemacht werden.

Darüber hinaus ist schon intuitiv einsichtig, dass es wenig sinnvoll ist, wenn ein Benutzer Daten herstellen oder verändern kann, die anschließend vor ihm geheim gehalten werden sollen. Insofern kann als erste Regel für sicherheitsrelevante Eigenschaften objektorientierter Datenmodelle festgehalten werden:

Zugriffsregel 1
Besitzt eine Klasse Methoden, die sich modifizierend auf bestimmte Eigenschaften der jeweils ausführenden Instanzen auswirken, so kann die gleiche Klasse auch Methoden anbieten, die Auskunft über die betroffenen Eigenschaften oder über deren Veränderungen erteilen.

Zugriffsregel 1 hat unmittelbare und weitreichende Auswirkungen auf die Vergabe von Benutzerrechten: Jedes explizit erteilte Schreibrecht für einen Benutzer impliziert immer entsprechende Leserechte.

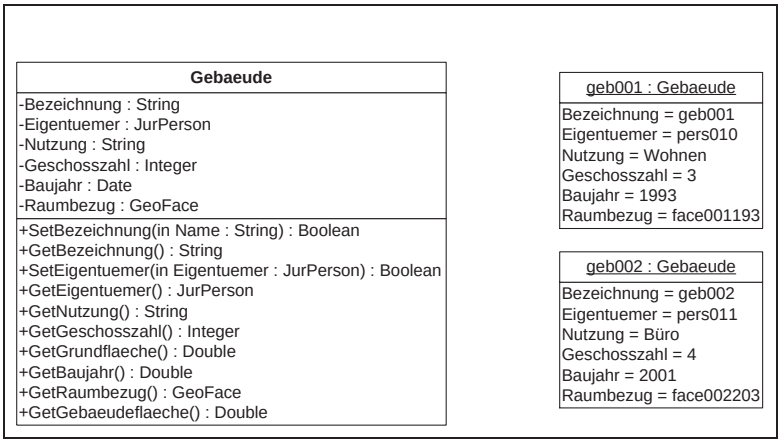

Abbildung 5.1: *Modellierung von Klassen und Instanzen.*

In Abbildung 5.1 ist eine Gebäudeklasse mit den privaten Attributen `Bezeichnung`, `Eigentuemer`, `Nutzung`, `Geschosszahl`, `Baujahr` und `Raumbezug` definiert. Korrespondierende `Get...`und `Set...`Methoden definieren den externen Zugang zu diesen Attributen. Die zwei Gebäudeobjekte `geb001` und `geb002` bilden den *Extent* der Klasse `Gebaeude` und belegen dabei die Attribute mit konkreten Werten.
Der charakteristische Zusammenhang zwischen Klassen und deren Instanzen erzeugt weitere, unmittelbar relevante Bedingungen für die Durchsetzbarkeit von Zugriffsrechten: Mit der Veränderung von Klassendefinitionen wird der

Zustand, die Existenz und die Semantik der Instanzen dieser Klassen und gegebenenfalls weiterer betroffener Klassen beeinflusst. Somit gilt:

Zugriffsregel 2
Eine Berechtigung zur Generierung und Veränderung von Klassenbeschreibungen geht einher mit der Berechtigung Instanzen dieser Klassen erzeugen, verändern oder löschen zu können.

Umgekehrt führt die Einsichtnahme in die Eigenschaften von Objekten nur dann zu sinnvollen Interpretationen, wenn gleichzeitig die Beschreibung der Objekte, im Sinne der Modellklassen eingesehen werden kann. Somit gilt:

Zugriffsregel 3
Der Zugriff auf Instanzen einer Klasse erfordert die Einsicht in die Definition der Klasse oder deren relevanten Teile.

5.2 Vererbung

Die Vererbung gehört zu den mächtigsten Werkzeugen der objektorientierten Modellierung. Sie ermöglicht die abstrahierte Beschreibung wiederverwendbarer Eigenschaften und Methoden. Die Konkretisierung erfolgt in fachlich spezifizierten, abgeleiteten Klassen. Bei der Vererbung werden sämtliche Eigenschaften einer Basisklasse an alle erbenden Klassen weitergegeben. Die wiederholte Deklaration von Eigenschaften, fachlich oder strukturell „ähnlicher" Klassen kann damit eingespart werden. Dies führt neben einer vereinfachten Definition von Datenschemata, auch zu einer Abbildung der Realwelt-Semantik in das Datenmodell und somit zu konsistenten Modellstrukturen. Die Intention der Vererbung ist, dass jede Instanz einer abgeleiteten Klasse gleichzeitig auch eine Instanz der Basisklasse(n) ist. Die Klassendiagramme aus Abbildung 5.2 verdeutlichen diesen Zusammenhang.

Die Basisklasse zur Beschreibung von Gebäuden wird spezifiziert und sowohl attributiv, als auch funktional durch abgeleitete Klassen zur Beschreibung von Wohn-, Gerbe- und öffentlichen- Gebäuden erweitert. Eine weitere, multiple Ableitung von `Wohngebaeude` und `GewerblichesGebaeude` führt zu einer attributfreien Klasse zur Definition von Gebäuden mit gemischter Nutzung. Hinsichtlich des kontrollierten Zugriffs auf die Daten einzelner Objekte, fallen zwei Aspekte auf:

1. Die abgeleiteten Klassen definieren eine Methode `SetNutzung`, die jeweils im privaten Bereich deklariert ist und somit nur von den Objekten selber,

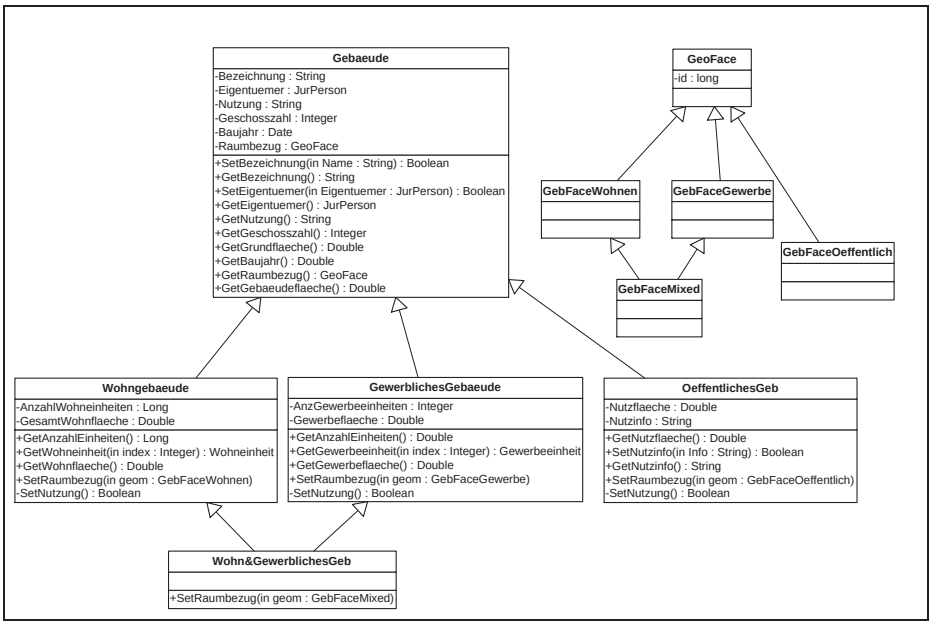

Abbildung 5.2: *Vererbung von Eigenschaften der Gebäudeklasse.*

z.B. im Konstruktor der Klasse, ausgeführt werden kann. Die Aufgabe der Methode, den Wert des Attributs `Nutzung` zu setzen, bleibt also den Instanzen selbst überlassen. Das Attribut ist somit vor fehlerhaften Belegungen durch Anwender der Klasse geschützt.

2. Der Raumbezug von Gebäude-Instanzen kann nur durch Verwendung der entsprechenden Methode `SetRaumbezug` gesetzt werden. Dabei besitzt jede der abgeleieteten Klassen eine eigene Definition des Funktionsprototypen und eine eigene Implementierung der Methode. Entscheidend ist in diesem Zusammenhang, dass die Methode in jeder Klasse nur räumliche Objekte eines bestimmten flächenartigen Datentyp als Übergabeparameter akzeptiert. Durch diese Art der Modellierung wird der Semantik des Datenmodells Rechnung getragen und insbesondere sichergestellt, dass Fachobjekte und deren Raumbezug semantisch zusammenpassen und ein konsistentes Datenmodell bilden.

Im Zusammenhang mit diesen beiden Punkten sind die bereits erläuterten Konzepte des Polymorphismus und der virtuellen Methoden (siehe auch: 3.5.2) von Bedeutung, die es Methoden ermöglichen, sich in Abhängigkeit von der Klasse des ausführenden Objekten unterschiedlich zu verhalten, ohne dass dies nach aussen sichtbar wird. Die Vererbung erzeugt unmittelbare Abhängigkeiten zwischen den beteiligten Klassen, die auch für die Vergabe von Zugriffsrechten

relevant sind: Die Struktur und Semantik einer abgeleiteten Klasse stützt sich auf die Definitionen und Beschreibungen ihrer Basisklassen, was zu folgenden Regeln führt.

Zugriffsregel 4
Ist die Beschreibung einer Klasse sichbarkeit, so sind können die Beschreibungen der Basisklassen nicht durchgängig und verborgen werden.

Zugriffsregel 5
Kann auf die Definitionen und Beschreibungen einer Klasse verändernd zugegriffen werden, so können auch die Eigenschaften abgeleiteter Klassen verändert werden (z.B. Veränderung der Vererbungshierarchie).

Zusammen mit Zugriffsregel 3 ergibt sich eine weitere Regel:

Zugriffsregel 6
Mit dem Zugriff auf ein Objekt einer bestimmten Klasse sind sämtliche Methoden und Eigenschaften des öffentlichen Bereichs der instanziierten Klasse sowie die Methoden und Eigenschaften sämtlicher Basisklassen zugreifbar.

Zur Formalisierung der Vererbung wählen wir im Weiteren folgende Formulierung: Sei Cl die Menge aller Klassen eines Datenschemas und $A, B \in Cl$ Klassen, die so definiert sind, dass B von A abgeleitet ist, dann erbt die Klasse B die beschreibenden Eigenschaften der Klasse A, die durch Attribute, Beziehungen und Methoden festgelegt sind. Die Vererbungsbeziehung zwischen A und B wird dargestellt mit: $base(A, B) =$ „A ist Basisklasse von B".

5.3 Assoziation

Assoziationen beschreiben Beziehungen, die zwischen Instanzen der beteiligten Klassen bestehen können. Es handelt sich dabei um fachliche oder topologische Beziehungen, die einen Zusammenhang zwischen zwei oder mehreren Objekten herstellen, dabei aber die strukturellen Eigenschaften und die Eigenständigkeit der beteiligten Objekte nicht behindern. Assoziationen können reflexiv sein (Objekt a steht in der gleichen Beziehung zu Objekt b, wie b zu a) und Zyklen beschreiben (a steht in der gleichen Beziehung zu b, wie b zu c, c zu d und d zu a). Das Klassenschema aus Abbildung 5.3 definiert drei typische Assoziationen: Instanzen der Klasse `Gebaeude` können topologisch zu anderen Gebäuden *benachbart* sein. Dabei gilt die Nachbarschaft zwischen zwei Gebäuden immer in beide Richtungen entlang der Beziehungskante. Des Weiteren sind Instanzen der Klasse `Wohngebaeude` immer eindeutig mit einer postalischen `Adresse`

assoziiert. Hingegen beschreibt eine gültige Adresse nicht zwingend den postalischen Standort eines Wohngebäudes. Die Beziehung `gehoert_zu/hat` zwischen `Wohngebaeude` und `Adresse` besitzt folglich eine $1 : [0..1]-$ Kardinalität. Schließlich definiert die $n : m-$ Beziehung `hat/wohnt_in` den Zusammenhang zwischen `Wohngebaeude` und `Anwohner` - Wohngebäude haben 0, einen oder mehrere Anwohner, während Personen in ihrer Rolle als Anwohner mehreren Wohngebäuden angehören können.

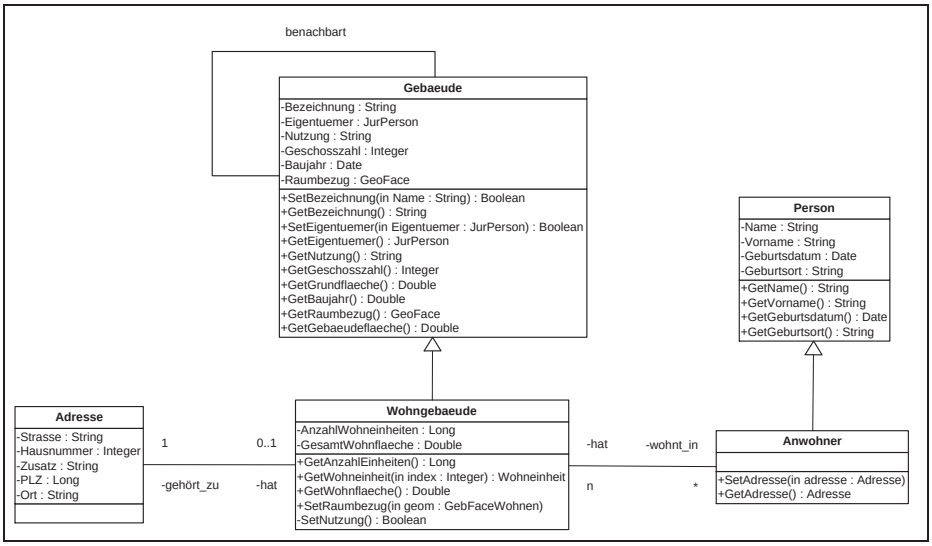

Abbildung 5.3: *Fachliche Beziehungen zwischen* `Adresse`, `Wohngebäude` und `Anwohner`, bzw. topologische Nachbarschaftsbeziehung zwischen `Gebäuden`.

Da Assoziationen keine strukturellen oder semantischen Abhängigkeiten zwischen Objekten definieren, sondern Beziehungen zwischen bereits bestehenden Klassen und Objekten beschreiben, können auch Zugriffsrechte für die an einer Assoziation beteiligten Klassen und Objekte unabhängig voneinander erteilt werden. Soll dagegen die Assoziation selbst bekannt gegeben werden, so muss dies zwangsläufig die in Beziehung gesetzten Klassen und deren betroffene Instanzen mit einbeziehen.

Zugriffsregel 7
Der Zugriff auf eine definierte Assoziation zweier Klassen, erfordert mindestens Einsicht in die Definition der betroffenen Klassen.

5.4 Aggregation

Aggregationen beschreiben abhängigkeitsrelevante Zusammensetzungen von Teilaspekten der Instanzen einer Klasse aus Instanzen einer anderen Klasse. Dabei wird die Aggregation zum immanenten Bestandteil der aggregierten Klassenstruktur. Aufgrund der sich ergebenden Teilmengenbeziehung sind Aggregationen, im Gegensatz zu Assoziationen, grundsätzlich zyklenfrei und irreflexiv. Aus diesen Eigenschaften ergeben sich weitreichende aber eindeutig gerichtete Abhängigkeiten für die Erteilung von Zugriffsrechten, die an den Aggregationen aus Abbildung 5.4 verdeutlicht werden können: Gebäude besitzen sowohl Aussenwände, als auch Innenwände, die unmittelbarer Bestandteil der Gebäude selbst sind. Weder in der Realität, noch im Modell können Wände verändert oder entfernt werden, ohne dass dies Auswirkungen auf die Eigenschaften des Gebäudes hat. Gleichzeitig kann mit dem Vollzugriff auf ein Gebäude auch der Zugriff auf die aggregierenden Wände in Verbindung gebracht werden. Dies ist schon deshalb möglich, weil eine Untersuchung der internen Modellierung (z.B. als C++ Klassen) zeigt, dass Aggregationen durch Klassenattribute beschrieben werden können, deren Instanzen durch Werte des Attributs referenziert sind. Die Beziehung zwischen Gebäuden und ihren Innenwänden, ebenso wie die Beziehung zwischen Wohngebäuden und ihren Wohneinheiten bzw. zwischen Wänden und ihren linienhaften Geometrieteilen zeigt die Komposition, als eine besondere Art der Aggregation. Diese definiert Beziehungen von Objekten einer bestimmten Klasse zu exklusiven und existenzabhängigen Teilobjekten. So existieren die Innenwände eines Gebäudes nur solange, wie das Gebäude selbst existiert. Das gleiche gilt auch für Wohneinheiten, die mit der Auflösung des Wohngebäudes ebenfalls verschwinden.

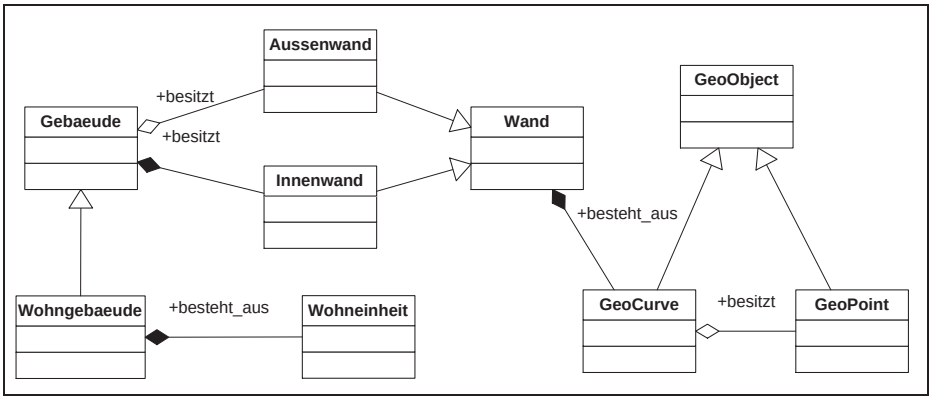

Abbildung 5.4: *Aggregationen und Komponitionen von Gebäudeteilen.*

Für die Erteilung von Zugriffsrechten bedeutet dies, dass der Zugriff auf Ag-

gregationen über den Zugriff auf die aggregierten Objekte gesteuert wird. Der Zugriff auf die aggregierenden (Teil-)Objekte wird durch den Zugriff auf eine Aggregation mit einbezogen. In der Umkehrung bedeutet dies, dass der Zugriff auf ein Teilobjekt nur zusammen mit dem Zugriff auf das aggregierte Objekt möglich ist oder wenn die verfügbaren Methoden keine Relevanz für den Zustand des aggregierten Objekts haben (z.B. nur lesender Zugriff). Im Fall von Kompositionen folgt aus dem Zugriff auf ein aggregiertes Objekt implizit auch der Zugriff auf die Teilobjekte. Als Zugriffsregeln formuliert ergibt sich:

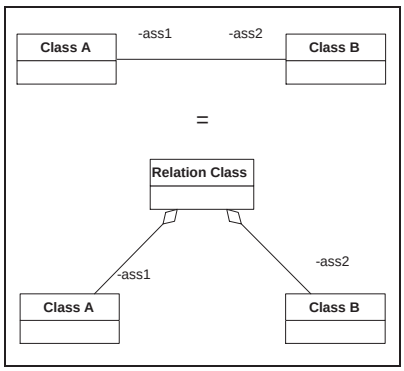

Abbildung 5.5: *Modellierung einer Assoziation durch die Aggregation einer Beziehungsklasse.*

Zugriffsregel 8
Besitzt eine Klassendefinition Aggregationen, so impliziert der Zugriff auf die Beschreibung der Klasse auch Zugriffe auf die Beschreibungen der aggregierenden Klassen.

Zugriffsregel 9
Aggregiert sich ein Objekt aus Teilobjekten, so ist die Art des Zugriff auf die Teilobjekte abhängig vom Recht des Zugriffs auf das aggregierte Objekt: Greift ein Benutzer schreibend auf Teilobjekte zu, so verändert er damit auch Eigenschaften aggregierter Objekte. Kann er umgekehrt auf aggregierte Objekte (lesend) zugreifen, so kann er auch auf deren Teilobjekte lesen zugreifen.

Die Suche nach einem Zusammenhang zwischen Aggregationen und Assoziationen führt zu einem Modellierungsansatz nach dem Vorbild aus Abbildung 5.5. Mit der Einführung einer Beziehungsklasse und der Definition der entsprechenden Beziehungsinstanzen können Assoziationen auf Aggregationen zurückgeführt werden. Es gelten dann auch die gleichen Zugriffseigenschaften, wie bei der Aggregation. Dabei wird insbesondere deutlich, dass Assoziationen nachträglich auf bestehende Klassen aufmodelliert werden (können) und deren

Existenz keinen Einfluss auf die beteiligten Objekte hat. Der Zugriff auf die beteiligten Objekte dagegen kann die Eigenschaften der Assoziation verändern.

5.5 Zusammenfassung

Die objektorientierten Paradigmen implizieren Zusammenhänge zwischen den modellierten Objekten, die bei der Erteilung von Zugriffsrechten berücksichtigt werden müssen. Die klassische Definition von Sichten als Basis für Zugriffsbeschränkungen erweist sich somit für objektorientierte Geo-Datenbanken als unzureichend. Besonders das Konzept der Vererbung im Sinne der Spezialisierung von Objekteigenschaften sowie die Möglichkeiten zur Beschreibung komplexer Objektstrukturen mittels Aggregationen müssen von einem geeigneten Zugriffs- und Berechtigungskonzept erfasst werden. Die in diesem Kapitel dargestellten Zugriffsregeln erfassen modular die strukturellen Abhängigkeiten zwischen Objekten und Objektteilen, die mit der Anwendung einer objektorientierten Beschreibungsstechnik einhergeht. Sie liefern damit eine Basis für den Entwurf eines Berechtigungskonzeptes, das eine flexible, an der Semantik des fachlichen Modells angelehnte Zugriffsbeschränkung ermöglicht und den Benutzern in sich abgeschlossene Teilmodelle liefert.

Kapitel 6

Berechtigungskonzept für Geodaten

Die Erkenntnisse über die rechterelevanten Zusammenhänge und Abhängigkeiten in objektorientierten Modellstrukturen, die in Kapitel 5 untersucht wurden, sollen in diesem und den folgenden Abschnitten Einzug in den Entwurf einer geeigneten Zugriffs- und Berechtigungsstrategie für Geodaten finden. Dabei steht ein objektorientierter Ansatz zur konsistenten und effizienten Beschreibung und Auswertung von Zugriffsrechten, im Sinne der Definition von Benutzer- und Berechtigungsprofilen im Mittelpunkt des Interesses.

6.1 Zielsetzungen einer objektorientierten Zugriffsarchitektur

Es hat sich gezeigt, dass objektorientierte Geodatenmodelle in ihrer Eigenschaft, die Zusammenhänge der realen Welt abzubilden, diverse semantische Abhängigkeiten direkt in den Strukturen des Modells verankern. Gleichzeitig liefern die gleichen Modelle die Intention einer Semantik durch die Definition von Klassen und deren Attributen, Vererbungen von Klasseneigenschaften sowie Beziehungen zwischen Klassen. Aus diesen Konzepten ergibt sich ein Ansatzpunkt, Teilaspekte objektorientierter Geodatenmodelle für bestimmte Benutzer oder Benutzergruppen auszublenden, dabei aber die intendierte Semantik für die verbleibende Information zu erhalten und die Konsistenz des Gesamtmodells nicht zu gefährden. Die hier skizzierte Berechtigungsstrategie zur modellkonformen Beschränkung von Zugriffsrechten basiert auf folgenden Zielsetzungen: Im Sinne der genannten objektorientierten Paradigmen und einer am fachlichen Datenmodell ausgerichtreten Sicht auf die Daten soll die

Verwaltung der Benutzerrechte hinsichtlich der Vermeidung semantischer Sicherheitslücken optimiert und hinsichtlich des Aufwands bei der Beschreibung und Auswertung von Zugriffsrechten minimiert werden. Das angestrebte Berechtigungskonzept soll eine mehrstufige Zugriffsstruktur im Sinne einer *MAC*-Politik insofern realisieren, als dass eine Vergleichbarkeit der Rechte von Benutzern und der Sicherheitsrelevanz von Daten erreicht wird.
In der Datenhaltung selbt greift das Berechtigungskonzept auf die verifizierten und zertifizierten Sicherheitsfunktionen eines geeigneten DBMS (Anhang B) zurück und integriert diese in die fachlichen Zugriffsstrukturen.
Die Strategie der Zugriffsbeschränkung setzt an verschiedenen Ebenen der Datenmodelle an.

1. Datenschema: In dynamischen Modellumgebungen steht am Anfang die Frage, welche Benutzer in welchem Umfang das Datenschema durch hinzufügen oder verändern von Klassendeklarationen modifizieren dürfen. Dabei übernehmen Benutzer mit entsprechenden Rechten die Rolle von (Teil-)Administratoren.

2. Semantikräume: Innerhalb eines Datenmodells bestimmt der Semantikraum, in dem sich ein Benutzer befindet, den semantischen Kontext des Zugriffs auf Geodaten. Ein Benutzer kann nur auf die Objekte zugreifen und dabei nur die Methoden und Eigenschaften verwenden, die ihm in seinem Semantikraum-beschränkten Datenschema zur Verfügung stehen.

3. Innerhalb des Semantikraum-beschränkten Datenmodells sollen Benutzer nicht grundsätzlich auf alle Objekte einer Datenquelle zugreifen können. Weitere Zugriffsbeschränkungen sollen dabei nach räumlichen und fachlichen Kriterien erfolgen können. So soll der Zugriff auf Objekte eines bestimmten geografischen Ausschnitts oder auf die Objekte einer ausgewählten Gemarkung oder Flur einschränkbar sein.

6.2 Beschreibung von Benutzern und Benutzergruppen

Der Schutz von Geodaten meint im allgemeinen den Schutz dieser Daten gegen Missbrauch und Zerstörung durch unbefugte Benutzer eines GIS oder eines Portals, das den Zugang zu einer entsprechenden Datenverwaltung ermöglicht. Auch wenn Angreifer in Form von Prozessen auftreten, verbergen sich dahinter in der Regel Benutzer oder Benutzergruppen mit einem gezielten Interesse an den Daten bzw. an dem Vorteil, der sich aus der Kenntnis der Informationen

ergibt oder einem Interesse an einer Veränderung bzw. Zerstörung der Daten oder des Gesamtsystems. Ausgenommen davon sind Schäden, die durch technisches Versagen - wie Stromausfall oder defekte Hardware - oder als Nebenwirkung einer anderen Ursache entstehen. Derartige Einflüsse werden in dieser Arbeit nicht weiter berücksichtigt. Im Mittelpunkt der Betrachtungen stehen Informationen, Methoden und deren Benutzer. Die Definition von Benutzerprofilen und die Charakterisierung der Rechte von Benutzern, die einem solchen Profil zugeordnet werden können, ist eine zentrale Problemstellung bei der Erstellung von Sicherheits- und Zugriffsstrategien.
Mehrstufige Zugriffsmodelle im Sinne einer *MAC*-Strategie basieren - wie in den Abschnitten 4.5.2 und **??** dargestellt - auf Zugriffsstufen, für die eine Halb- oder Totalordnung definiert ist.

Definition 6.1 *(Halbordnung) Eine binäre Relation R auf einer Menge M heisst Halbordnung, wenn R irreflexiv, asymmetrisch und transitiv ist.*

Mit einer Halbordnung kann zum Ausdruck gebracht werden, dass für ein Paar $a, b \in M$ die Relation aRb gilt. In diesem Fall ist bRa nicht gültig. Bei einer Halbordnung müssen nicht alle Elemente aus M bezüglich R vergleichbar sein.

Definition 6.2 *(Totalordnung) Eine Halbordnung R auf einer Menge M, bei der alle Paare $(a, b) \in M \times M$ bezüglich R vergleichbar sind, heisst Totalordnung.*

Benutzer und Daten werden jeweils einer Zugriffsebene zugeordnet, wodurch eine Vergleichbarkeit der Benutzer untereinander, sowie zwischen Benutzern und den Objekten einer Datenquelle entsteht. Diese Vergleichbarkeit definiert letztlich die Zugriffsrechte. In einem objektorientierten Ansatz wird das Ziel der Vergleichbarkeit und der damit verbundenen impliziten Rechtevergabe auf direktem Weg über Vererbungs- und Beziehungseigenschaften erreicht.
Den objektorientierten Paradigmen folgend, sollen Benutzergruppen im Sinne von Klassen der objektorientierten Modellierung und Programmierung interpretiert werden. Demzufolge werden im Datenmodell zunächst Benutzerklassen definiert, deren Instanzen die Benutzer eines GIS repräsentieren. Eine Benutzerklasse beschreibt die grundlegenden und stereotypen Eigenschaften von Benutzern dieser Gruppe einschliesslich ihrer Beziehungen zu anderen Benutzergruppen und bildet damit einen Rahmen zur Definition von Benutzerprofilen.

Benutzer gehören der selben Benutzergruppe an, bzw. sind Instanzen der gleichen Benutzerklasse, wenn sie in Bezug auf die betroffene Datenbasis gleiche oder ähnliche Rollen annehmen können. In der Praxis wird schnell klar, dass

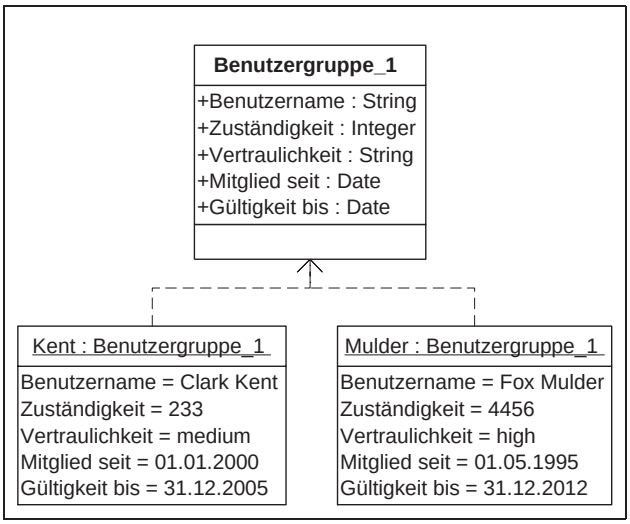

Abbildung 6.1: *Benutzer als Instanzen von Benutzergruppen.*

ein Benutzer häufig mehr als nur einer Gruppe angehört. In Abhängigkeit von seiner zu erfüllenden Aufgabe muss er unterschiedliche Rollen annehmen können, ohne über verschiedene Benutzerkennungen zu verfügen, zwischen denen er nach Bedarf wechseln muss. Obwohl ein Objekt, in den meissten objektorientierten Sprachen aufgrund von Konstruktoren-Semantik, Speichermodellen und Objektidentitäten, nur eine Klasse unmittelbar instanziieren kann, stellt die objektorientierte Modellierung optimale Techniken zur Beschreibung von Benutzern mit multiplen Gruppenzugehörigkeiten bereit. Mit dem Konzept der Vererbung und insbesondere der Mehrfachvererbung werden aus bestehenden Klassen neue, abgeleitete Klassen konstruiert. Dabei „erbt" die abgeleitete Klasse sämtliche Eigenschaften, Beschreibungsmerkmale und Methoden ihrer Basisklassen. Für die Modellierung von GIS-Benutzern bedeutet dies, dass Benutzerklassen mit Hilfe von gerichteten, zyklenfreien Vererbungsbäumen konstruiert werden können. Benutzer als Repräsentanten einer Klasse erhalten ihre Privilegien durch Ableitung von Privilegien aus Basisklassen, die entweder selber Benutzerklassen sind oder elementare Zugriffsrechte darstellen. Eine mögliche Ableitung von Benutzerrechten könnte folgendermassen aussehen: Zunächst werden auf oberster Ebene die Optionen (lesen, schreiben) mit denen z.B. eine Datenbank geöffnet werden kann als eigene Klassen festgelegt. Somit ist durch eine Ableitung von einer dieser Klassen die grundsätzliche Art des Zugriffs auf die Datenquelle (lesend oder schreibend) festgelgt. Als nächstes werden sämtliche atomaren Basisoperationen auf der Datenbank, wie zum Beispiel „Anlegen neuer Klassen", „Lesen der Klasse X", „Einfügen von Instanzen der Klasse X", „Ändern von Instanzen der Klas-

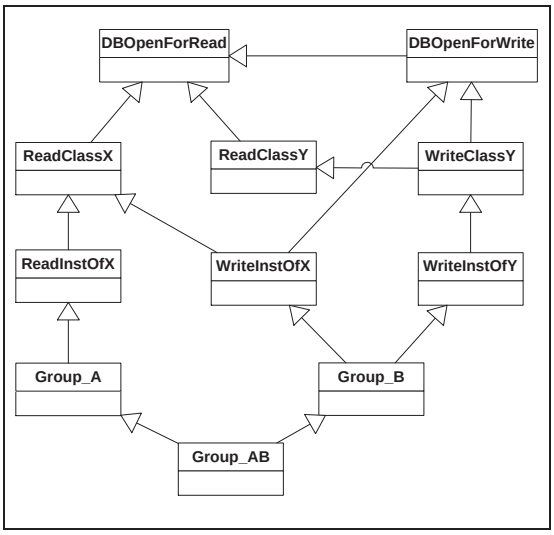

Abbildung 6.2: *Gruppendefinition durch Vererbung von Privilegien.*

se X" oder „"Vergabe von Zugriffsrechten", als elementare Zugriffsrechte in Form von (Benutzer-) Klassen definiert. Auf der nächsten Ebene können nun Benutzerklassen mit rollenbezogenen Profilen definiert werden, die ihre Privilegien durch Mehrfachvererbung aus den Klassen der Basisoperationen beziehen. Auf diese Weise entstehen aufgabenspezifische Profile, wie „Datenbankadministrator", „Backup-Administrator", „Planung", „Auswertung" oder „Wartung". In den darauf folgenden Ebenen können immer wieder neue Benutzerklassen definiert werden, die ihre Aufgaben und Berechtigungen aus verschiedenen, bereits definierten Basisklassen beziehen. In Abbildung 6.2 ist der beschriebene Aufbau von Vererbungsbäumen für Benutzergruppen dargestellt. Durch die Einführung von Vererbungskonzepten in die Definition von Benutzerprofilen ist implizit ein Ansatz für eine hierarchische Zuweisung und Verwaltung von Benutzerrechten gegeben. Ist eine Benutzerklasse B abgeleitet von einer Benutzerklasse A, so erbt die Klasse B neben den allgeinen Eigenschaften auch alle Zugriffsrechte der Klasse A. Für die Benutzer folgt daraus, dass ein Benutzer b der Gruppe B - also eine Instanz der Klasse B - mindestens die Privilegien besitzt, die ein beliebiger Benutzer a der Klasse A besitzt. Oder in objektorientierter Terminologie: Der Benutzer b, der eine Instanz der Klasse B darstellt, ist gleichzeitig eine Instanz der Klasse A. In der MAC-Semantik bedeutet diese Konstellation eine höhere Klassifizierung der Benutzer der Klasse B gegenüber denen der Klasse A. Zusammengefasst ergibt sich für die Modellierung von Benutzergruppen, dass die Benutzer eines GIS alle Privilegien besitzen, die sowohl ihrer eigenen Benutzerklasse, als auch deren Basisklassen erteilt wurden.

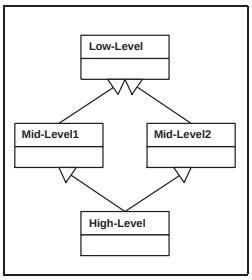

Abbildung 6.3: *Berechtigungshierarchie von Benutzergruppen mit Vererbung.*

Abbildung 6.3 zeigt, wie mit Vererbung und Mehrfachvererbung eine (Halb)-ordnung auf der Menge der Benutzergruppen definiert werden kann. Die Basisklasse `Low-Level` verfügt über eine minimale Menge an Zugriffsrechten. Benutzer der Gruppen `Mid-Level1` und `Mid-Level2` erben mindestens die Berechtigungen der Klasse `Low-Level` und können darüber hinaus noch weitere Zugriffsrechte besitzen. Durch Mehrfachvererbung auf die Klassen `Mid-Level1` und `Mid-Level2` erbt die Klasse `High-Level` die Rechte aller niedrigeren Benutzergruppen und kann zusätzlich noch eigene Berechtigungen besitzen. Somit erhält die in der Ableitungshierarchie unterste Benutzerklasse die höchste Einstufung in der Berechtigungshierarchie.

6.3 Erteilung von Zugriffsrechten für Benutzer

Prinzipiell erhalten Benutzer ihre Zugriffsrechte über die Methoden der Benutzerklassen, deren Instanzen sie sind bzw. deren Rolle sie annehmen können. Diese kapseln ihrerseits die Zugriffe auf die entsprechenden Klassen und deren individuell gefilterten Instanzen. Abbildung 6.4 zeigt die Deklaration von Zugriffsmethoden für Klassen. Solche Methoden werden gegebenfalls als virtuelle Methoden definiert, um sich in den Implementierungen der einzelnen Benutzerklassen unterscheiden zu können. Namentlich identische Methoden können sich demnach unterschiedlich verhalten und entsprechend unterschiedliche Ergebnisse zurückliefern, ohne dass der Benutzer sich dessen bewusst ist. Diese Eigenschaft ist insbesondere bei Methoden von Bedeutung, die den Zugriff auf Instanzen einer Datenquelle implementieren.

Um den Instanzen und Methoden einer Benutzerklasse Zugriffsrechte für die privaten (und damit geschützten) Elemente einer Schemaklasse zu erteilen, bietet sich das Konzept der *friend*-Deklaration von Klassen und Methoden an, das auch in objektorientierten Programmiersprachen wie C++ Anwendung findet. Setzt eine Klasse eine andere Klasse oder Methode als *friend*, so stuft sie diese damit als vertrauenswürdig ein und gewährt ihren Instanzen Zugriff

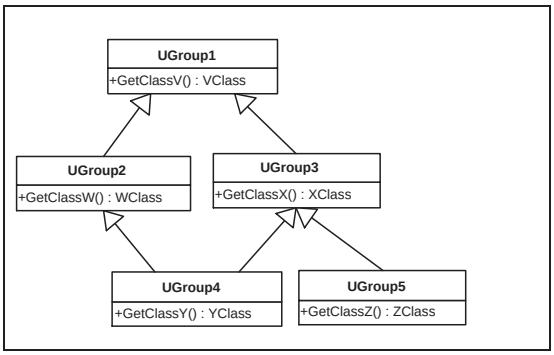

Abbildung 6.4: *Zugriffsmethoden für Benutzergruppen.*

auf den privaten Bereich ihrer Klassendeklaration. Mit der *friend*-Deklaration wird gewissermaßen ein Zugriffstunnel für ausgewählte Klassen geöffnet. Zur Veranschaulichung ist hier eine Anwendungsklasse `Flur` mit zwei Methoden `GetFlurnummer` und `SetFlurnummer` definiert.

```
class Flur{
        friend class UGroup1;
private:
        int GetFlurnummer();
        bool SetFlurnummer(long nummer);
}
```

Die Klasse `Flur` deklariert die Benutzerklasse `UGroup1` als *friend*, mit der Folge, dass die Methoden `GetFlnummer` und `SetFlnummer` der Klasse `UGroup1` in ihren Implementierungen auf die entsprechenden Methoden der Klasse `Flur` zugreifen können. Für Klassen, die nicht als *friend* deklariert sind, bleiben diese Methoden verborgen und unzugänglich.

```
class UGroup1{
        long GetFlnummer(Flur f){
              return f.GetFlurnummer();
        }
        bool SetFlnummer(Flur f, long nummer){
              return f.SetFlurnummer(nummer);
        }
};
```

Von besonderer Bedeutung ist dabei, dass die Eigenschaft, *friend* einer Klasse zu sein, nicht vererbt werden kann. Will also in dem genannten Beispiel,

eine von `UGroup1` abgeleitete Benutzerklasse auf die privaten Methoden der Klasse `Flur` zugreifen, ohne die von `UGroup1` geerbten Zugriffsmethoden zu verwenden (denn diese werden vererbt), so muss sie dazu von `Flur` ebenfalls als *friend* deklariert werden.
Neben der direkten Vergabe von Zugriffsrechten für die öffentlichen Methoden einer Klasse durch Vererbungssemantik, kann mit der *friend*-Deklaration ein sicherer und kontrollierbarer Zugriffskanal für die internen Strukturen von Klasseninstanzen definiert werden. Der Entwurf einer Zugriffsarchitektur in Kapitel 9 wird zeigen, dass der Zugriff von Benutzern auf die Objekte des Datenmodells über sogenannte „Zugriffsklassen" gekapselt werden soll. Diese befinden sich im Semantikraum der jeweiligen Benutzergruppe und können somit direkt über ihre öffentlichen Methoden angesprochen werden. Die Zugriffsklassen haben die Aufgabe Sichtendefinitionen bezüglich des *Extents* einer Klasse auszuwerten und den Zugriff auf diesen zu reglementieren. Mithilfe von *friend*-Deklarationen kann den Zugriffsklassen der Zugriff auf die internen Eigenschaften und Methoden der Instanzen der jeweiligen Fachobjektklasse ermöglicht werden. Diese sind für die Mitglieder einer Benutzergruppe nicht sichtbar und somit auch nicht zugreifbar. In der Zugriffsarchitektur dient die *friend*-Deklaration dazu, den Zugriff auf Eigenschaften und Methoden von Fachobjektklassen nur über sichere Schnittstellen zu ermöglichen, die für unterschiedliche Benutzergruppen unterschiedlich implementiert und reglementiert sein können. Bei der Erteilung von Zugriffsrechten auf die Informationen einer Datenquelle, müssen die drei genannten Zugriffsebenen in zwei Bereiche unterschieden werden:
Auf der einen Seite stehen definierende Zugriffe auf Datenschemata im Sinne der Definition von Klassen, Vererbungen und Beziehungen. Damit werden Administratoren und Teiladministratoren eines Systems festgelegt, die mit dem Funktionsumfang einer DDL auf ein Datenschema zugreifen können. Auf der anderen Seite stehen die Zugriffe auf die Fachobjekte einer Datenquelle in einem gegebenen fachlichen Kontext. Dabei wird sowohl die Menge der zugreifbaren Objekte, als auch deren Informationsgehahlt benutzerabhängig beschränkt. Die Zusammenhänge dieser Bereiche der Zugriffskontrolle werden im Weiteren näher spezifiziert. Zunächst werden die Auswirkungen von Berechtigungen zur Veränderung des Schemas behandelt.

6.3.1 Zugriffsrechte für die Schemaebene

Metadaten oder Metainformationen sind Daten, die der Beschreibung von Daten dienen. Dazu gehören neben den Schemabeschreibungen auch Informationen über die Herkunft der Daten, den Aktualitätsstatus oder das verwendete räumliche Bezugssystem. Metadaten bilden die Basis für Interpretatio-

nen und die Qualität der beschriebenen Nutzdaten. Insofern sind Konsistenz, Vollständigkeit und Korrektheit von Metadaten wesentliche Voraussetzung für die Vertrauenswürdigkeit und den Wert von Daten. Als Konsequenz ergibt sich, dass der Schutzbedarf von Metadaten gegenüber unbefugten Veränderungen im allgemeinen sehr hoch einzuschätzen ist. Insbesondere die unzulässige Modifikation von Datenschemata kann zur Unbrauchbarkeit und damit zum Verlust großer Datenmengen führen. Das Datenschema hebt sich insofern von den übrigen Metainformationen ab, als das es - zusammen mit dem Datenbankschema - Bestandteil der Systemarchitektur und damit wesentliche Voraussetzung für eine sinnvolle Erfassung und Auswertung von Informationen ist. Bei objektorientierten Datenmodellen bildet das Datenschema die Interpretationsbasis für die Semantik der beschriebenen Daten, die sich in der modellierten Klassenstruktur widerspegelt. Metadaten, die im Sinne inhaltlicher oder fachlicher Charakterisierungen der Daten über die Beschreibung des Datenschemas hinausgehen, können als eigene Informationseinheiten in das Gesamtdatenmodell integriert und über fachliche Relationen mit den Inhalten verknüpft werden, deren Beschreibung sie darstellen. Folglich entsprechen auch die Regeln zur Vergabe von Zugriffsrechten für derartige Metainformationen denen fachlicher Beziehungen, wie sie in Abschnitt 6.3.2 behandelt werden. Für die Vergabe von Zugriffsrechten für Schemadaten gelten andere und weitreichendere Beschränkungen. Das Erstellen, Verändern oder Entfernen von Schemainformationen darf grundsätzlich nur speziell dafür autorisierten und besonders vertrauenswürdigen Personen erlaubt sein. Mit der Berechtigung, schreibend auf ein Datenschema zugreifen zu dürfen, wird einem Benutzer eine sehr hohe Vertrauenswürdigkeit und Verantwortung zuerkannt, die ihm weitreichende Einflussnahme auf alle Daten ermöglicht: So kann der Benutzer beispielsweise Datentypen von Attributen verändern, Attribute hinzufügen oder entfernen, Schlüssel- oder Indexattribute definieren und Beziehungsarten festlegen oder aufheben. Aber nicht nur einzelne Attribut- und Beziehungseigenschaften können verändert werden. Gerade bei raumbezogenen Datenmodellen können grundlegende Eigenschaften beschriebener Geo-Objekte verändert werden, wenn eine Basisklasse eines bestimmten Geometrietyps (z.B. flächenhaft) durch eine Klasse eines anderen Geometrietyps (z.B. punkthaft) ersetzt wird oder topologische Bedingungen verändert werden (z.B. „Linie begrenzt zwei Flächen" wird ersetzt durch „Linie begrenzt drei Flächen") . Die Auswirkungen des verändernden Zugriffs auf Klassenbeschreibungen sind in Zugriffsregel 2 aus Abschnitt 5.1 formalisiert. Aus den Abhängigkeiten zwischen Klassenbeschreibungen objektorientierter Strukturen ergeben sich unmittelbar Möglichkeiten zur Vergabe von fachlich und thematisch eingeschränkten Rechten der Schema-Administration. Dabei wird mit dem Recht einer Benutzergruppe, administrierend auf eine bestimmte Klassen innerhalb einer Hierarchie zugreifen zu dürfen, eine Rolle als Teil- oder Fachadministrator zugeteilt. Die Zugriffsre-

geln objektorientierter Klassenarchitekturen implizieren, auf welche Teile einer Schemabeschreibung die Mitglieder der Benutzergruppe definierend zugreifen können. Ein Administrator des Basissystems legt Basisklassen sowie deren Strukturen und Beziehungen fest. Weiterhin vergibt er, wie in Abbildung 6.5 dargestellt, Berechtigungen für die Basisklassen an Fachadministratoren mit unterschiedlichen Zuständigkeitsbereichen, die diese Basisstrukturen verwenden können, um davon abgeleitete Fachobjektklassen zu konstruieren, ohne dabei Zugriff auf den Zuständigkeitsbereich und die Schemadaten anderer Fachadministratoren zu haben. Dabei wirken sich die Zugriffsregeln 4 und 5 unmittelbar aus.
Die Beziehungstypen Aggregation und Assoziation können in gleicher Weise von Fachadministratoren, unter Beachtung von Zugriffsregel 8 verwendet und interpretiert werden: Der Fachadministrator kann strukturbeschreibende Aggregationen zwischen Klassen festlegen, die in seinem administrativen Zugriff liegen. Dabei muss er mindestens für die aggregierten Klassen Rechte zur Schemadefinition oder -veränderung besitzen. Da mit dem Setzen oder Ändern einer Beziehung R von einer Klasse A zu einer Klasse B, eine Eigenschaft der Klasse A verändert wird, muss ein Benutzer, der dieses Privileg besitzt, mindestens auch Schreibrechte für die Klasse A haben. Assoziationen kann ein Fachadministrator bezüglich zweier Klassen formulieren, deren Definition er einsehen kann.

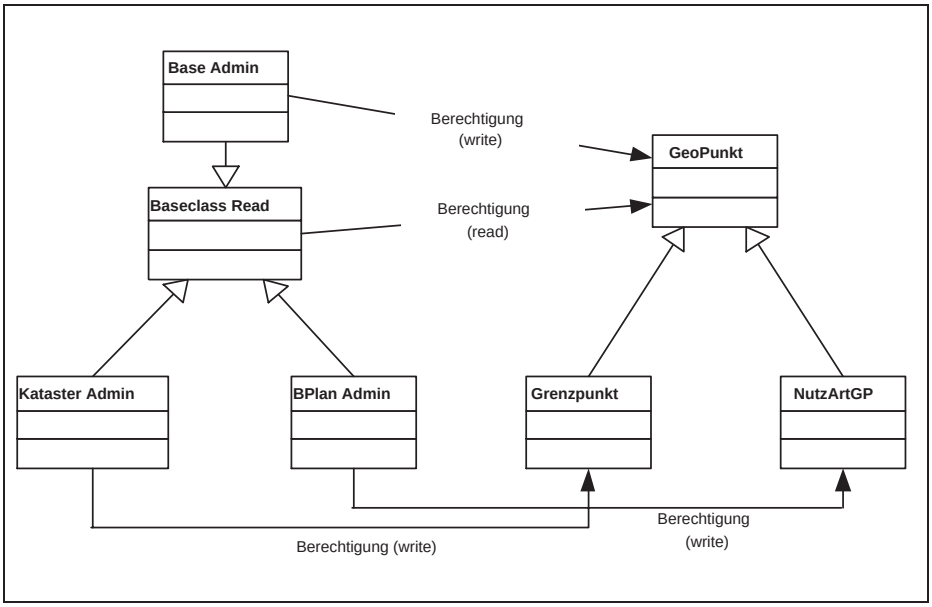

Abbildung 6.5: *Fachadministratoren mit der Zuständigkeit für Teilbereiche des Schemas.*

Im Gegensatz zur hohen Sensibilität von Schemadaten bei schreibendem Zugriff, verlangt die Bedeutung des Datenschemas, jedem Benutzer mit Zugriffsrechten für bestimmte Nutzdaten, auch lesenden Zugriff für deren formale Beschreibung zu erteilen. Dies ergibt sich aus den Zugriffsregeln 3 und 6 aus Abschnitt 5.1. Nur im Kontext des Datenschemas ist für den Benutzer eine sinnvolle Interpretation der verfügbaren Daten möglich. Hat ein Benutzer beispielsweise ein Geo-Objekt im Zugriff, das Referenzen auf zwei weitere Objekte erkennen lässt, so benötigt der Benutzer Einsicht in die Schemabeschreibung der betroffenen Objekte, um einerseits die Objekte und deren referenzierte Beziehungen richtig interpretieren zu können, andererseits aber auch, um nicht aus Unkenntnis über die Bedeutung der Referenzen die referenzierten Objekte fehlerhaft zu bearbeiten.
Das Konzept der Klassenvererbung innerhalb einer Schemabeschreibung realisiert die Semantik der „Generalisierung" und „Spezialisierung" der modellierten Objekte. Mit dem Recht zur Einsicht in ausgewählte Ausschnitte eines gegebenen Schemas wird die Detailtiefe festgelegt, die den betroffenen Benutzern an den verfügbaren Objekten zuerkannt wird. Attribute und Methoden, die im Sinne der Vererbung unterhalb der festgelegten Hierarchieebene definiert werden, sind für die betroffenen Benutzer weder sichtbar noch anwendbar.
Eine weitere Bedingung an die Sichtbarkeit von Klassenbeschreibungen wird von dem Beziehungstyp Aggregation und dessen Zugriffsregeln 8 und 9 festgelegt. Besitzt die Klassenbeschreibung eines sichtbaren Objektes Aggregationen, so müssen die betroffenen Benutzer auch Zugriffsrechte für die Beschreibungen der aggregierenden Klassen besitzen. Hat ein Benutzer Einsicht in die Definitionen zur Beschreibung von Beziehung zwischen zwei Klassen, so kann er daraus unmittelbar Informationen über die Klassen selber ableiten: Ihm ist mindestens die Existens der beteiligten Klassen sowie der fachliche Zusammenhang zwischen den Klassen bekannt. Um nicht von einer trügerischen Sicherheit getäuscht zu werden, sollten einem Benutzer nur dann Leserechte auf Beziehungen zwischen Klassen erteilt werden, wenn der Benutzer auch die an der Beziehung beteiligten Klassen einsehen darf. Dieser Zusammenhang ergibt sich unmittelbar aus Zugriffsregel 7 bezüglich des Zugriffs auf Assoziationen in objektorientierten Modellen.

6.3.2 Zugriffsrechte für Sachdaten und deren räumlichen Ausprägungen

Befassen wir uns mit den Endanwendern eines GIS und deren Zugriffsrechten, so lenken wir unser Interesse von der Ebene der Schemadaten zur Ebene der Nutzdaten. Hier befinden sich die jeweiligen Instanzen der zuvor beschriebenen Klassen und Beziehungen. Benutzer, die mit einem System arbeiten, verlangen

Zugriff auf modellierte Fachobjekte und deren räumliche Ausprägungen. Der Endanwender greift in der Regel über eine Applikationsschicht auf die Nutzdaten eines GIS zu. Diese Fachanwendung stellt ihm Werkzeuge zur gezielten Anfrage und Bearbeitung von Geodaten bereit. Dabei definiert die Fachanwendung selbst eine erste Stufe der Datensicherheit, in dem sie die Möglichkeiten von Benutzern, schreibend oder lesend auf die Daten zuzugreifen, auf die verfügbaren Werkzeuge beschränkt. In diesem Sinne erfolgt eine Zugriffsbeschränkung grundsätzlicher Art durch die Unterscheidung zwischen reinen Auskunftssystemen - zum Beispiel Routenplaner oder Stadtpläne im Internet - und GIS, die sowohl lesende als auch schreibende Zugriffe auf die Informationen einer Datenquelle erlauben - dabei werden Funktionen zur Datenerfassung, Fortführung, Planung, Auswertung und Reporterstellung angeboten. Wird auf Geodatenbestände über reine Auskunftssysteme zugegriffen, so werden diese in erster Instanz mittels der Verfahren aus Abschnitt 2.3 vor unsachgemäßer Modifikation und gezielter oder versehendlicher Zerstörung geschützt. Die Integrität der Daten und die Konsistenz des Gesamtbestandes unterliegt außchliesslich der Kontrolle des Datenanbieters. Regelungen müssen in diesem Fall für die Erteilung von Auskünften getroffen werden. Erfolgt der Zugriff der Anwender auf Geodaten über ein GIS, das neben den Auskunfts-Funktionen auch über Methoden zur Veränderung und Ergänzung der Daten verfügt und diese den Anwendern anbietet, so muss das Rechtesystem alle mittelbaren und unmittelbaren Konsequenzen für die Integrität von Objekten, deren Beziehungen und die Durchsetzbarkeit von Geheimhaltungsforderungen, die sich aus der Erteilung eines Zugriffsrechts ergeben, erkennen, berücksichtigen und gegebenfalls ablehnen.
Bei der Erteilung und Interpretation von Zugriffsrechten, soll innerhalb eines GIS nach Fachobjekten, deren Raumbezug sowie den fachlichen und räumlichen Beziehungen von Objekten unterschieden werden. Fachobjekte werden als Instanzen sogenannter Fachobjektklassen modelliert. Diese beschreiben die relevanten fachbezogenen Attribute der darzustellenden Objekte sowie die fachlichen Beziehungen, die zwischen Fachobjekten der gleichen oder unterschiedlichen Klassen bestehen können. Der Raumbezug von Fachobjekten wird durch räumliche Objekte als Geometrie mit einer bestimmten Fachbedeutung in einem räumlichen Bezugssystem modelliert. Besitzen die Fachobjekte einer Klasse eine räumliche Ausprägung, zum Beispiel Gebäude, Flurstück oder Gemarkung, so enthält die zugehörige Fachobjektklasse entsprechende „Geo"-Attriute, über die die Fachobjekte mit den jeweiligen Geometrie-Elementen verknüpft werden können. Geometrie-Elemente sind Instanzen vergleichsweise einfach strukturierter Klassen. Diese basieren auf den elementaren Geometrietypen Punkt, Linie (enthält: Strecke, Kreisbogen und Spline) und Fläche und erhalten eine fachbezogene Spezifikation durch die Ableitung von Fachbedeutungsklassen (zum Beispiel Flurstücksgrenze, Gebäudefläche,

Grenzpunkt). Die Fachbedeutungsklassen beschreiben neben ihrer intuitiven Fachbedeutung geometrische und fachliche Abhängigkeiten zwischen Fachbedeutungen: „Gebäudeflächen werden gegrenzt durch Gebäudegrenzen", „Die Endpunkte von Flurstücksgrenzen sind Grenzpunkte".
Bei der Vergabe und Auswertung von Zugriffsrechten soll zwischen Fachobjekten und Geometrie-Elementen nicht weiter unterschieden werden. Dies ist möglich, weil sich die Definition generischer Klassen zur Beschreibung von Geometrie-Elementen nur dadurch von Fachobjekt-Klassen unterscheidet, dass sie von statischen Basisklassen zur Beschreibung der elementaren Geometrietypen abgeleitet werden und deren definierende Beziehungen fachlich spezifizieren. Aus Sicht der objektorientierten Modellierung handelt es sich in beiden Fällen um Klassen, die mit ihrer Definition zu Datentypen werden und damit zur Beschreibung von Attributen anderer Klassen verwendet werden können. Der Raumbezug von Fachobjekten wird somit durch die Referenzierung räumlicher Klasseninstanzen hergestellt. In Kapitel 7 werden die Besonderheiten räumlicher Objekte im Kontext von Zugriffsrechten genauer betrachtet.

6.3.2.1 Definitionen

Seien $A, B \in Cl$ zwei Klassen zur Beschreibung von Fach- bzw. Geo-Objekt. Ausserdem sei $R \in Rel$ eine im Schema definierte Beziehung zwischen den Klassen A und B. Ausserdem bezeichne O die Menge aller Objekte einer Datenquelle, die jeweils Instanzen einer ausgewählten Klasse aus Cl sind.

Definition 6.3 (Anfrage) *Eine Anfrage q ist eine logische Formel mit einer festgelegten Syntax und Semantik. Das Ergebnis einer Anfrage ist die Menge von Objekten $O_q \subseteq O$, für die q erfüllt ist: $\forall o \in O_q : q$. Alle Elemente der Menge O_q sind Instanzen der gleichen Klasse. Wir bezeichnen die Menge aller Anfragen eines Informationssystems mit Q.*

Anfragen bilden die Basis für den strukturierten Zugriff auf Objekte einer Datenquelle. Mit ihnen können Objektfilter definiert werden, die eine eingeschränkte Sicht auf Objekte mit festgelegten Eigenschaften ermöglichen. Unter der Menge aller Anfragen Q eines Informationssystems definieren wir eine ausgezeichnete Anfrage $*$, die für alle Instanzen einer ausgewählten Klasse erfüllt ist: $\forall o \in O : *$.

Definition 6.4 (Objektcontainer) *Sei $q \in Q$ ein Anfrage. Wir bezeichnen mit SET_A^q die Ergebnismenge der Anfrage q auf der Klasse A. SET_A^q enthält genau die Menge der Fach- oder Geo-Objekte, die Instanzen der Klasse A sind und die Bedingungen von q erfüllen.*

Für SET_A^* schreiben wir kurz SET_A und beschreiben damit die Menge aller Instanzen einer Klasse A.

Definition 6.5 (Instanzen) *Ein einzelnes Fach- oder Geo-Objekt a einer Klasse A wird als Element eines entsprechenden Objektcontainers mit* $a \in SET_A$ *dargestellt oder als* $a \in SET_A^q$*, wenn a aus der Ergebnismenge einer Anfrage q gewählt wird.*

Definition 6.6 (Relationen) *Ist zwischen zwei Klassen* $A, B \in Cl$ *eine Relation* $R(A, B)$ *im Schema definiert, so werden die Relation R zwischen zwei Instanzen* $a \in SET_A, b \in SET_B$ *der Klassen A und B durch* $r(a, b)$ *dargestellt.*

6.3.2.2 Zugriffe auf Fachobjekte

Bezüglich der Zugriffsrechte auf einzelne Fachobjekte kann festgestellt werden: Hat ein Benutzer das Recht, mindestens lesend auf eine Instanz einer bestimmten Klasse zuzugreifen, so hat dies zwei mögliche Auswirkungen.

1. Der Benutzer kann aus einem zugreifbaren Objekt Rückschlüsse auf dessen Schemabeschreibung ziehen und damit auch auf die Klassenbeschreibung, oder zumindest auf wesentliche Teile davon lesend zugreifen.

2. Der Benutzer erhält keine Schemabeschreibungen zu den zugreifbaren Objekten und kann diese auch nicht aus verfügbaren Informationen ableiten. Die einsehbaren Informationen sind damit für den Benutzer wertlos und können nicht in sinnvoller Weise ausgewertet werden, da erst durch Schemabeschreibungen aus Daten Informationen im Sinne einer modellierten Semantik werden.

Diese Feststellung hat unmittelbare Auswirkungen auf die Vergabe von Zugriffsrechten für Fachobjekte, wie sie bereits im vergangenen Abschnitt dargestellt wurden: Aus der Erteilung von Zugriffsrechten für Geo-Objekte folgt ein entsprechendes Zugriffsrecht für die zugehörigen Schemabeschreibungen. Negativ ausgedrückt bedeutet dies, dass Zugriffsrechte an Geo-Objekten nur Benutzern zuerkannt werden können, die auch über entsprechende Zugriffsrechte für die Schemabeschreibungen dieser Objekte verfügen. Diese Einschränkung kann nun verwendet werden, um zusammen mit der Definition von Objektcontainern Zugriffsrechte für Geo-Objekte an Benutzergruppen zu vergeben. Dazu werden zu jeder Klasse des sichtbaren Schemas einer Benutzergruppe Anfragen formuliert, die der Definition von Objektcontainern der jeweiligen Klasse dienen. Die Mitglieder von Benutzergruppen erhalten nun Zugriffsrechte auf die Instanzen von Objektcontainern. Die Art des erlaubten Zugriffs auf Objekte,

wird durch die verfügbaren Methoden, bzw. den öffentlichen Deklarationsteil der jeweiligen Klassenbeschreibung festgelegt.

6.3.2.3 Zugriffe auf Anfragen

Da der Objektzugriff an der Anwenderschnittstelle immer über geeignete Anfragen erfolgt, muss sichergestellt sein, dass mit der Forderung, Zugriffsrechte für Geo-Objekte deskriptiv an Benutzergruppen zu vergeben, Anfragen formulierbar sind, die genau die gewünschte Menge an Objekten enthält.

Zugriffsregel 10
Wenn $Pot(A)$ die Potenzmenge der Instanzen einer beliebigen Klasse $A \in Cl$ definiert, dann muss für jedes Element $p \in Pot(A)$ eine Anfrage $q \in Q$ existieren, so dass $SET_A^q = p$ gilt.

Mit der Erfüllung der Forderungen aus Zugriffsregel 10 wird es ermöglicht, den Zugriff auf einzelne Instanzen und jede beliebige Teilmenge des *Extents* einer Klasse über die Deklaration von Anfragen zu steuern. Für jede sichbare Fachobjektklasse eines Benutzers oder einer Benutzergruppe definiert eine Anfrage die Menge der zugreifbaren Instanzen dieser Klasse. Anfragen definieren Objektmengen durch die Beschreibung von einfachen oder komplexen, räumlichen oder nicht-räumlichen Eigenschaften der Ergebnisobjekte. Durch die Erteilung von Zugriffsrechten auf Objektmengen (Anfrageergebnisse) werden die Benutzerrechte in der Weise parametrisierbar, dass der Zugriff auf einzelne Objekte nicht auf eine Liste von Objekt-Identitäten, sondern auf Objekte mit beschriebenen Eigenschaften beschränkt wird. Das reduziert den Aufwand der Rechteverwaltung und erhöht gleichzeitig die Flexibilität des Rechtesystems. Insbesondere bei dynamischen GIS, deren Inhalte häufig aktualisiert werden, hat dieser Vorteil besonderes Gewicht. Die Ergebnismengen der zugewiesenen Anfragen einer Benutzergruppe definieren den zulässigen Arbeitsbereich von Mitgliedern dieser Gruppen. Beim Zugriff auf Objekte über Anfragen unterscheiden wir zwischen Systemanfragen und Benutzeranfragen. Systemanfragen sind von Administratoren als Bestandteil des Fachdatenschemas vordefinierte Anfragen zur Definition von Zugriffsrechten für Instanzen der Ergebnisklassen. Systemanfragen definieren *Views* auf Objekte, die den beschriebenen Eigenschaften genügen.
Benutzeranfragen werden von den Anwendern selbst, mit dem Ziel der Datenauswertung im Rahmen ihrer Zugriffsrechte definiert. Systemanfragen und Benutzeranfragen basieren in der Regel auf der gleichen Anfragesyntax und -semantik. Unterschiedlich sind insbesondere die Eingabemengen, auf die Anfragen angewendet werden. Während Systemanfragen unmittelbar auf eine

Datenquelle zugreifen, werten Benutzeranfragen im allgemeinen bereits eingeschränkte Datenmengen aus. Die Eingabemengen von Benutzeranfragen werden in diesen Fällen durch die Ergebnismengen von Systemanfragen definiert. Die Details der beiden Ansätze werden in Kapitel 8 dargestellt und schließlich in einem einheitlichen Modell dargestellt.

6.3.2.4 Zugriffe auf Objektrelationen

Fachliche, topologische und geometrische Beziehungen zwischen Objekten werden sowohl hinsichtlich des Beziehungstyps, als auch hinsichtlich der zulässigen oder geforderten Kardinalität unterschieden. Die Beziehungstypen werden mit den Begriffen „Assoziation" und „Aggregation" bezeichnet.

- Assoziationen beschreiben fachliche Beziehungen zwischen in sich abgeschlossenen Objekten. Beziehungen dieses Typs sind kein elementarer Bestandteil der Objektbeschreibung, sondern die Darstellung einer Relation zwischen existierenden Objekten. Assoziationen repräsentieren beispielsweise Beziehungen zwischen Flurstücken und deren Eigentümern, zwischen Schuldnern und Gläubigern oder zwischen benachbarten Einwohnern.
- Aggregationen dagegen haben für die Objekte definierenden Charakter. Sie beschreiben die Zusammensetzung von Objekten aus Teilobjekten, bzw. die Zusammenfassung bestehender Objekte zu komplexeren Objekten. Derartige Beziehungen bestehen zwischen Städten und deren Gemeinden, zwischen Gemarkungen und ihren Fluren und zwischen Fluren und ihren Flurstücken.

Mit der Kardinalität einer Beziehung wird die Anzahl der Objekte beschrieben, die jeweils in der beschriebenen Relation zueinander stehen können. Bei der Definition der Beziehung werden die Kardinalitäten $1:1-$, $1:n-$ und $n:m-$ unterschieden. Typische Vertreter dieser Beziehungsformen sind:

- $1:1$: Die Beziehung zwischen einem Bundesland und seinem Ministerpräsidenten oder zwischen einem Regierungsbezirk und seinem Regierungspräsidenten ($regiert \longleftrightarrow wird\_regiert\_von$).
- $1:n$: Die Beziehung zwischen Personen und deren Hauptwohnsitz ($wohnt\_haupt \longleftrightarrow wird\_haupt\_bewohnt\_von$).
- $n:m$: Die Nachbarschaftsbeziehung zwischen Personen oder Flurstücken ($ist\_nachbar\_von$).

Assoziationen definieren thematische Zusammenhänge zwischen Objekten, ohne dass die beteiligten Objekte strukturell von der Existenz und der Bedeutung der Assoziationen abhängig sind. Folglich ist auch die Erteilung von lesenden Zugriffsrechten für einzelne Objekte unabhängig von den für diese Objekte definierten Assoziationen. Anders verhält es sich bei der Erteilung von Schreibrechten für Objekte, die an Assoziationen beteiligt sind. Aus der Möglichkeit, die Eigenschaften von Objekten zu verändern oder sogar ihre Existenz in Frage zu stellen, ergeben sich Möglichkeiten zur Einflussnahme auf Eigenschaften oder die grundsätzliche Gültigkeit der definierten Beziehungen. Verwiesen sei hierbei nur auf das Verhalten einer Beziehung *neighbour* (mit der die topologische Nachbarschaft von Objekten definiert wird), wenn die betroffenen Objekte in ihrer Lage verändert oder gelöscht werden können. Entsprechend der Zugriffsregel 7 sind Objekte und der Zugriff auf dieselben unabhängig von Assoziationen, die zwischen diesen Objekten definiert sind. Umgekehrt sind aber modellierte Assoziationen zwischen Objekten abhängig von der Existenz und dem Zustand der in Beziehung gesetzten Objekte. Insofern sind von dem Recht eines Benutzers, schreibend auf ein Objekt zuzugreifen, auch Assoziationen betroffen, an denen dieses Objekt beteiligt ist. Sollen nun explizite Rechte zur Definition oder Veränderung von Beziehungen gesetzt werden, so setzt dies vorraus, dass die in Beziehung gesetzten Klassen im Schema der entsprechenden Benutzer liegen. Der Zugriff auf Instanzen dieser Klassen ist dann Voraussetzung, wenn die Benutzer das Recht erhalten sollen, vorhandene Relationen zwischen Objekten einzusehen, zu verändern oder zu entfernen. So wird hierbei mindestens die Kenntnis von der Existenz der beteiligten Objekte vorausgesetzt. Wenn für den Benutzer keine Instanzen der jeweiligen Klassen existieren, so kann er keine Beziehungsinstanz gründen. Das Setzen und Verändern von Beziehungen hat nur Auswirkungen auf die Beziehungen selbst, nicht aber auf die Definition der beteiligten Objekte.
Häufig dienen Assoziationen dazu, kontextrelevante Beziehungen zwischen Objekten zu definieren. Dabei kann der Zustand des einen Objektes nicht unabhängig von den in Beziehung gesetzten Objekten betrachtet werden. Dies hat unmittelbare Auswirkungen auf die Möglichkeit, sinnvolle Zugriffsrechte für die beteiligten Objekte zu erteilen, obwohl die Objekte für sich genommen, nicht strukturell voneinander abhängen. In Abbildung 6.6 ist ein Ausschnitt dargestellt, in dem zwei Straßen zu jeweils zwei Brücken führen, die ihrerseits einen Fluss überqueren. Dabei bestehen zwischen den einzelnen Objektklassen die Assoziationen: *„Straße führt zu Brücke"*, *„Brücke verbindet Straßen"* und *„Brücke überquert Fluss"*. Wird nun, wie in Abbildung 6.7 dargestellt, versucht, den Fluss oder dessen Lage und Verlauf vor bestimmten Benutzern zu verbergen, so führt dies nur bedingt zum Erfolg, wenn die betroffenen Benutzer die Brücken entlang des Flusslaufes sehen können. Das gleiche Problem entsteht, wenn wie in Abbildung 6.8 versucht wird, die Brücken, die den Fluss

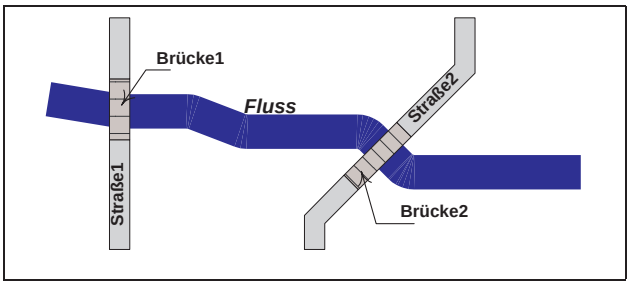

Abbildung 6.6: *Einen Fluss wird von zwei Brücken überquert.*

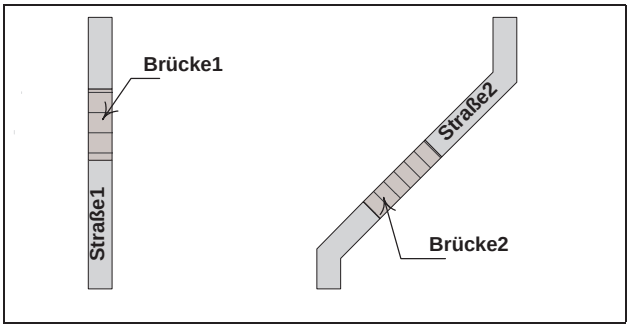

Abbildung 6.7: *Ausblendung des Flusses, bei gleichzeitiger Sichbarkeit der überquerenden Brücken.*

überqueren, bzw. deren Lage vor bestimmten Benutzern zu verbergen, wenn sowohl der Fluss mit seinem Verlauf, als auch die Strassen, die auf den Fluss zuführen, für die Benutzer erkennbar sind. Aus dem Gesamtkontext, in dem sich einzelne Objekte befinden, lassen sich Rückschlüsse auf einzelne Eigenschaften dieser Objekte ziehen. Der Kontext wird dabei mit den definierten Assoziationen eines Datenmodells beschrieben. Aus diesen Assoziationen lässt sich also erkennen, welche Eigenschaften bestimmter Objekte in direktem Zusammenhang mit Eigenschaften anderer Objekte stehen, obwohl zwischen diesen Objekten keine direkte, strukturelle Abhängigkeit definiert ist. Insofern werden die definierten Assoziationen zu Objekten des Datenmodells, die ihrerseits festlegen müssen, inwieweit ihre Existenz den Zustand eines Datenmodells bestimmt.
Handelt es sich bei den Beziehungen zwischen ausgewählten Objekten um Aggregationen, so sind die Abhängigkeiten zwischen den Zugriffsmöglichkeiten für einzelne Objekte direkt aus den Objektstrukturen ableitbar. Die Abhängigkeiten für Zugriffsrechte auf Geodaten ergeben sich unmittelbar aus den Zugriffsregeln 8 und 9 für Aggregationen in objektorientierten Klassenarchitekturen. Zunächst sei die Vergabe von expliziten Zugriffsrechten für das aggregierte Objekt angenommen. Wird einem Benutzer explizit das Recht erteilt, lesend

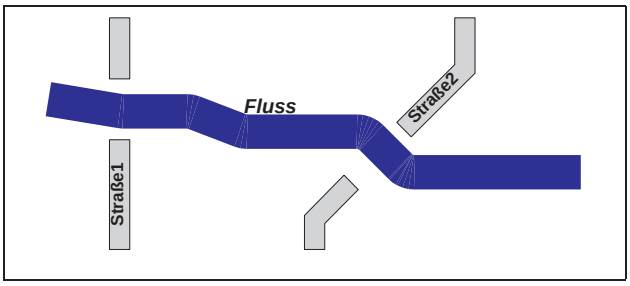

Abbildung 6.8: *Ausblendung der Brücken, bei gleichzeitiger Sichbarkeit des Flusses und der Strassen.*

oder schreibend auf ein Objekt zuzugreifen, und bezieht dieses Objekt zumindest Teile seiner Eigenschaften aus der Aggregation von anderen Objekten, so können diese Teilobjekte i.d.R. nicht vollständig vor Zugriffen durch den Benutzer geschützt werden, da der Benutzer mit jedem Zugriff auf das Gesamtobjekt immer auch auf Eigenschaften der Teilobjekte zugreift. Die Aggregationen von Flurstücken zu Fluren veranschaulichen diesen Zusammenhang: Wird einem Benutzer explizit das Recht erteilt, auf eine Flur lesend und schreibend zuzugreifen, so kann er mit jedem lesenden Zugriff auf die Daten der Flur auch Eigenschaften über die enthaltenen Flurstücke sammeln, ohne dass ihm die Berechtigung hierzu explizit erteilt wurde. Wird dem Benutzer das Verändern von Eigenschaften der Flur erlaubt, so setzt dies vorraus, dass er auch auf Daten der aggregierten Flurstücke schreibend zugreifen darf.
Sei nun der umgekehrte Fall, der expliziten Vergabe von Zugriffsrechten für die aggregierenden Objekte - also die Berechtigungen für Zugriffe auf Objekte, aus denen komplexere Objekte aufgebaut werden - angenommen. Mit dem Recht, lesend auf ein Teilobjekt zuzugreifen, wird zwar auch die Einsicht in bestimmte Teilaspekte des zusammengesetzten Objektes gewährt, diese bleibt aber eingeschränkt auf die Eigenschaften des ausgewählten Teilobjektes. Aus einer expliziten Leseberechtigung für Objekte lassen sich also weder Voraussetzungen, noch implizite Berechtigungen für die Zugriffe auf aggregierte Objekte ableiten. Anders verhält es sich bei Schreibberechtigungen: Wird ein Objekt aus anderen Objekten aggregiert, so ergeben sich daraus unmittelbare Abhängigkeiten des aggregierten Objektes von den Eigenschaften der aggregierenden Objekte. Im Einzelfall bedeutet dies: Soll einem Benutzer die Berechtigung erteilt werden, Teile zusammengesetzter Objekte verändern oder löschen zu können, so kann diesem Benutzer der schreibende Zugriff auf die aggregierten Objekte nicht grundsätzlich untersagt werden. Insbesondere besteht die Gefahr, dass jedes der Teilobjekte unverzichtbar für den Bestand des zusammengesetzten Objektes ist. Die Zugriffsregel 9 zeigt, dass das Vorkommen von Aggregationen in Datenmodellen hinsichtlich der Erteilung von Zugriffsrechten Abhängigkeiten

erzeugt, die sowohl in Richtung der Teilobjekte, als auch in Richtung der aggregierten Objekte wirken und berücksichtigt werden muss. Da die Art der Zugriffe auf ein Objekt mit der Verfügbarkeit von Klassenmethoden festgelegt wird, kann eine ungünstige Bereitstellung und Implementierung von Methoden dazu führen, dass die von Aggregationen betroffenen Objekte nur gemeinsam oder garnicht für Benutzer zugänglich gemacht werden dürfen. Die Interpretation von Zugriffsrechten aus Abschitt 6.3.2 wird zeigen, wie das Konzept der Klassenvererbung eingesetzt werden kann, um diese gegenseitigen Abhängigkeiten auf Basiseigenschaften der beteiligten Objekte zu reduzieren und damit die Flexibilität in der Erteilung von Zugriffsrechten zu erhöhen.

6.3.3 Definition von Zugriffsrechten

Im Folgenden soll nun der Vorgang der Erteilung von Zugriffrechten für Geodaten, unter Berücksichtigung der untersuchten Eigenschaften, konkretisiert werden. Die Vergabe von DV-relevanten Zugriffsrechten begründet sich aus der Anforderung, bestimmte Bereiche in Datenbanken bzw. Informationssystemen auf die Zugriffe durch ausgewählte Benutzer oder Benutzergruppen einschränken zu können bzw. vor den Zugriffen bestimmter Benutzer oder Benutzergruppen schützen zu können. Dabei soll ein Höchstmaß an Integrität, Vertraulichkeit und Verfügbarkeit der Daten gewährleistet werden, um damit den an das System gestellten Sicherheitsanforderungen sowie den Interessen und Rechten direkt oder indirekt Betroffener gerecht zu werden. Eine minimale Anforderung an die Vergabe von Zugriffsrechten ist es, betroffene Anwender in die Lage zu versetzen, die an sie gestellten Aufgaben erfüllen und für sie bestimmte Informationsangebote nutzen zu können. Da die Vergabe von Zugriffsrechten für Daten, die nur einem eingeschränkten Personenkreis zugänglich sein sollen, immer auch eine Form von persönlichem Vertrauen des Administrators in den sachgemäßen und vertraulichen Umgang des Anwenders mit den Daten voraussetzt, sprechen wir im Zusammenhang mit Zugriffsmodellen auch von der „Vertraulichkeit" der Informationen und der entsprechenden „Vertrauenswürdigkeit" der Benutzer.
Es ist naheliegend, dass eine sinnvolle Interpretation des positiv formulierten Begriffs des „Zugriffsrechts" eine Definition des Begriffs „Zugriffsverbot" beinhalten muss. In diesem Sinne soll ein Zugriffsverbot als Abwesenheit eines entsprechenden expliziten oder impliziten Zugriffsrechts definiert werden. Diese Definition stützt sich auf den Grundsatz des *Negation as Failure* : Die Summe der möglichen Zugriffe auf ein System ergibt sich aus der Summe der Zugriffsrechte eines Benutzers und der Summe seiner Zugriffsverbote. Einem Benutzer ist also grundsätzlich jede Form des Zugriffs verboten, für die kein Zugriffsrecht explizit erteilt wurde oder aus vorhandenen Zugriffsrechten ein-

deutig ableitbar ist.
Eine Beschränkung des Datenzugriff durch die Auswertung von Zugriffsrechten entspricht im Modellansatz der Definition von Sichten. Dem Anwender wird, wie auch bei unbeschränktem Zugriff auf Datenquellen, ein Ausschnitt der Objekte der realen Welt präsentiert. Dieser Realweltausschnitt repräsentiert ein Modell. So wird auch bei eingeschränkten Zugriffsrechten angestrebt, dem Anwender ein plausibles Modell der realen Welt zu präsentieren, das jeweils um Detailinformationen gegenüber dem Ausgangsmodell der Datenbank reduziert ist. Ein solches, auf Zugriffsrechte eingeschränktes Benutzermodell, wird jeweils durch entsprechende Benutzersichten definiert. Eine wesentliche Voraussetzung für diese Annahme ist, dass aus den rechteabhängigen Zugriffsbeschränkungen jeweils in sich geschlossene und konsistente Benutzersichten abgeleitet werden können. Benutzer sollen also immer Zugriff auf ein Modell der realen Welt erhalten, dass aus ihrer Sicht und für ihre Ansprüche vollständig ist. Mit einem solchen Ansatz ist es im Idealfall sogar möglich, die Existenz von Zugriffsbeschränkungen vor den betroffenen Benutzern zu verbergen. Mit dem Einsatz objektorientierter Datenmodelle und insbesondere durch die Verwendung des objektorientierten Vererbungskonzeptes kann ein Berechtigungsverfahren gefunden werden, das die genannten Bedingungen in optimaler Weise unterstützt und sich auf natürliche Weise in ein vorhandenes Fachdatenmodell integrieren lässt. Im wesentlichen wird ein solches Berechtigungsverfahren von zwei Ideen bestimmt:

1. Objekte sind die Träger von Informationen einer Datenquelle, deren Struktur und Semantik durch die Klassen beschrieben werden, deren Instanzen sie sind. Als umgekehrte Folge ergibt sich, dass ein Objekt immer nur die Informationen trägt, die durch die Klasse charakterisiert werden, deren Repräsentant das Objekt im aktuellen Kontext ist. Durch die Anwendung des Konzeptes der Klassenvererbung, übernimmt eine Klasse sämtliche Attribute, Beziehungen und Methoden ihrer Basisklassen und ergänzt diese gegebenenfalls um eigene Definitionen. Wird einem Benutzer der Zugriff auf Instanzen einer Klasse A gewährt, so kann er von den Objekten der Klasse A auf genau die Attribute, Methoden und Beziehungen zugreifen, die von A definiert werden, ohne dabei auf die Informationen und Beschreibungen von Klassen zugreifen zu können, die von A abgeleitet sind - dies gilt selbst dann, wenn er auf Instanzen der Klassen A zugreifen kann, die gleichzeitig Instanz einer von A abgeleiteten Klasse sind.

2. Der Zugriff auf Objekteigenschaften ist ausschließlich über die öffentlichen Methoden der Klassen möglich: die Zugriffe auf die Daten müssen also nicht länger nach lesenden und schreibenden Transaktionen unter-

schieden werden. Objektbeschreibungen können damit auf Klassen verteilt werden, die in Abhängigkeit von der sicherheitsrelevanten Unbedenklichkeit ihrer Methoden in Vererbungshierarchien modelliert werden. Dabei stellen Basisklassen Basismethoden (auch hinsichtlich der Sicherheitsrelevanz) zur Verfügung. Erst in abgeleiteten Klassen werden Attribute und Methoden beschrieben, die zur Erfüllung spezifischer Interessen oder Aufgaben definiert werden und daher gegebenfalls nur bestimmten Benutzern zugänglich sein sollen. Diese Annahme folgt aus den Vererbungsprizipien der Generalisierung in Richtung der Basisklassen und der Spezialisierung in Richtung der abgeleiteten Klassen. Entlang des Vererbungspfades werden also immer mehr Eigenschaften der Realwelt-Objekte modelliert.

Zur Veranschaulichung der Problematik und des Lösungsansatzes dienen die Klassen `Personen` und `Anschrift`sowie deren Beziehung zueinander, aus dem ALKIS® Datenschema. Diese sind in Abbildung 6.9 dargestellt.

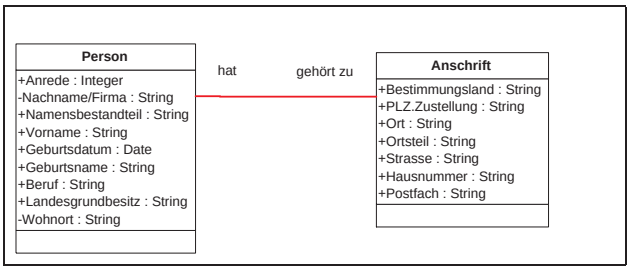

Abbildung 6.9: *Die Klassen* `Person` und `Anschrift` ohne Vererbungen.

Wird einem Benutzer oder einer Benutzergruppe der Zugriff auf Instanzen der Klasse `Personen` gewährt, so folgt daraus eine Zugriffsberechtigung auf alle öffentlichen (*public*) Attribute und Methoden der Instanzen dieser Klassen. Im Fall der Klasse `Personen` beinhaltet dies sowohl die Namen aller gespeicherten Instanzen, als auch die Daten bezüglich der Geburt, des Berufs und des Wohnorts dieser Personen. Es sind nun aber Situationen denkbar, in denen einem Benutzer zwar grundsätzlich der Zugriff auf die Daten gespeicherter Personen erlaubt sein soll, jedoch eingeschränkt auf bestimmte Attribute, die er zur Erfüllung seiner Aufgaben benötigt. Für die Instanzen anderer Klassen, wie zum Beispiel `Anschrift` , sind vergleichbare Situationen denkbar. Nach der herkömmlichen Methodik relationaler Datenbanken würde eine solche Anforderung der Zugriffsbeschränkung durch die Ausblendung von Spalten (z.B. mit der Definition von *Views*) erfüllt. Dabei besteht das grundsätzliche Problem, dass Tabellen und deren Spalten keine eigene Semantik besitzen. Als Konsequenz können Spalten (oder Zeilen) unabhängig vom fachlichen Kontext der

Datensätze und den Eigenschaften der verbleibenden Datensätze ausgeblendet werden. Neben einer Gefährdung der Datenkonsistenz durch schreibende Zugriffe auf Datenfelder, die im Kontext mit verborgenen Attributen eine andere Bedeutung besitzen, muss auch die generelle Durchsetzbarkeit von Geheimhaltungsforderungen in Frage gestellt werden. Die Ausblendung von Spalten erlaubt keine Rückschlüsse darauf, welche „geheimen" Informationen aus den verbleibenden Daten abgeleitet werden können.
Objektorientierte Datenmodelle eröffnen die Möglichkeit, inhaltliche Zusammenhänge und Abhängigkeiten von Objekteigenschaften bereits in das fachliche Datenmodell zu integrieren. So werden in der Deklaration einer Klasse die Attribute und Beziehungen zusammengefasst, die der untrennbaren Beschreibung eines Sachverhaltes dienen. Mit den Methoden der Klassenverbung können elementare Module zu komplexeren Modellen zusammengefasst und weitere Abhängigkeiten beschrieben werden. Abbildung 6.10 zeigt eine mögliche Zerlegung der Klassen `Person` und `Anschrift` nach dem Aspekt der Vertraulichkeit der Informationen.

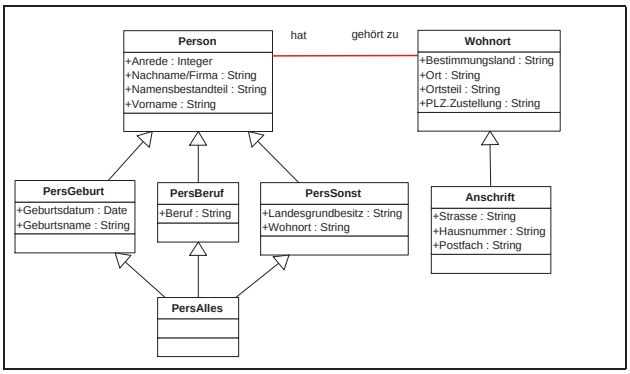

Abbildung 6.10: *Die Klassen* `Person` und `Anschrift` mit Vererbungen.

Mit der Definition einer Klassen- und Vererbungsstruktur wird die erste Stufe einer Zugriffshierarchie festgelegt. Dabei wird mit dem Zugriffsrecht für eine Klasse, bzw. deren Instanzen, das Recht zur Ausführung der öffentlichen Methoden dieser Klasse bezüglich eines zugreifbaren Objektes erteilt. Wird der Zugriff auf die Objekte einer Datenquelle auf die öffentlichen Methoden der jeweiligen Klassenbeschreibungen begrenzt, so entsteht damit eine entscheidende Eigenschaft sicherer und konsistenter Datenbestände: Die Zugriffe auf Objekte sind kontrollierbar. Selbst mit uneingeschränkten Zugriffsrechten können Benutzer nur auf die Weise auf die Daten zugreifen, wie sie von einer entsprechenden Methode unterstützt wird. Der Bedarf einer Unterscheidung zwischen lesendem und schreibendem Zugriff entfällt somit. Art und Umfang möglicher Zugriffe unterliegen der Kontrolle und Implementierung eines System- oder

Schemadesigners. Dieser kann die Methoden einer Klassenbeschreibung definieren und im öffentlichen Teil der Klassendeklaration für Benutzer freigeben. So können Methoden definiert werden, die den Wert von Attributen ausgeben, oder andere, die das Setzen bestimmter Werte, in Abhängigkeit von bestimmten Randbedingungen ermöglichen. Weitere Methoden können so definiert sein, dass sie mehrere Attribute auswerten und ein Funktions- oder Bilanzierungsergebnis zurückliefern, ohne dabei Einsicht in die einzelnen Werte des Objektes zu ermöglichen.

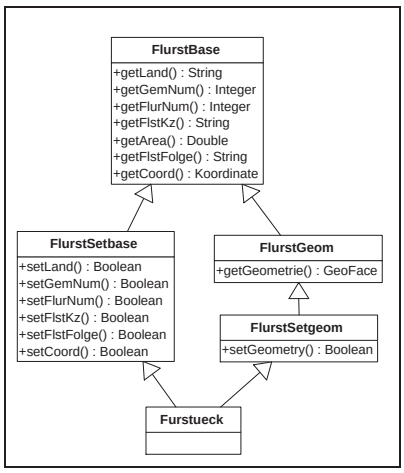

Abbildung 6.11: *Aufsplittung der Klasse* `Flurstueck` nach der Sicherheitsrelevanz ihrer Methoden.

In Abbildung 6.11 ist am Beispiel der Modellierung von Flurstücken dargestellt, wie die Methoden zum Zugriff auf Informationen hinsichtlich ihrer Sicherheitsrelevanz unterschieden und auf Basis- oder abgeleitete Klassen verteilt werden können: Eine Basisklasse (hier: `FlurstBase`) stellt zunächst die relativ unkritischen Methoden zur Abfrage der Attributbelegungen ihrer Instanzen zur Verfügung. Methoden zum Erfragen der Werte von Attributen mit vertraulicheren Informationen werden in abgeleiteten Klassen (hier: `FlurstGeom`) definiert. Die Methoden mit schreibendem Datenzugriff, mit denen Attributwerte eingefügt oder verändert werden, sind in Abhängigkeit von dem jeweiligen Potenzial ihrer Bedrohung für die Datensicherheit in weiteren Ableitungsschritten (hier: `FlurstSetbase` und `FlurstSetgeom`) implementiert. Um über eine Klasse zu verfügen, die ohne Zugriffsbeschränkung alle Methoden und Eigenschaften einer zerlegten Fachobjekt-Klasse besitzt, wird eine zusätzliche (künstliche) Klasse eingefügt, die direkt oder indirekt von allen Teilklassen abgeleitet ist (hier: `Flurstueck`). Wird einem Benutzer des GIS der Zugriff auf die Instanzen einer bestimmten (Teil-)Klasse gewährt, so kann der Benutzer mit dem Zugriff auf eine Instanz dieser Klasse sämtliche Methoden der Instanz

ausführen, die in dieser Klasse oder einer ihrer Basisklassen definiert sind.
Aus den Regeln und Definitionen des Abschnitts 6.3.2 geht bereits hervor, dass die Erteilung von Zugriffsrechten für Objekte nicht auf deren Eigenschaft einer Klassenzugehörigkeit eingeschränkt sein soll. Von einem mächtigen und flexiblen Berechtigungskonzept wird erwartet, dass die Zugriffsbefugnisse von Benutzern auch auf einzelne Objekte oder auf Objektmengen, die einer bestimmten Beschreibung genügen, begrenzt werden können. Entsprechend dieser Forderung, wird die Zuweisung von Zugriffsrechten zu Benutzergruppen in zwei zentrale Schritte unterteilt:

1. Die Beschreibung benutzerabhängiger Datenschemata und den damit verbundenen Semantikräumen.

2. Die Auswahl schemakonformer Objekte, die für die Mitglieder einer Benutzergruppe zugreifbar sein sollen.

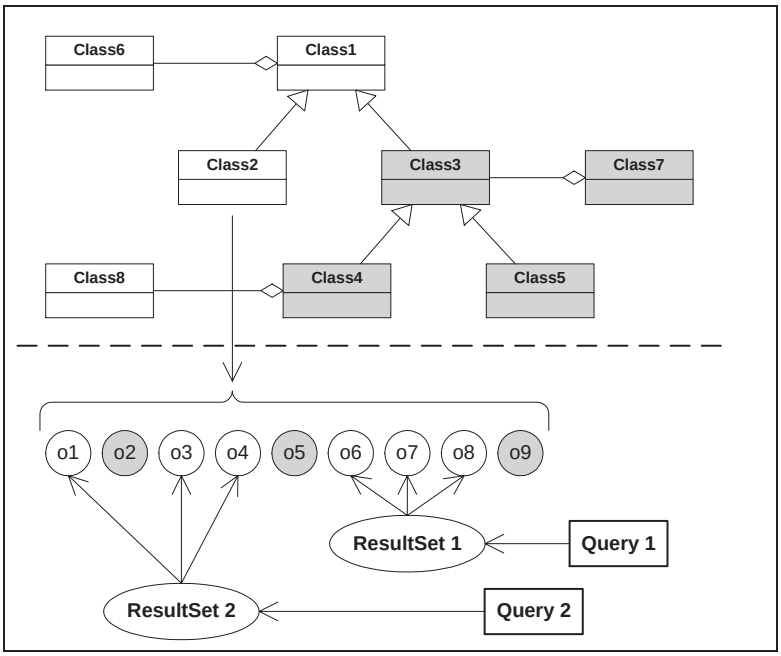

Abbildung 6.12: *Erteilung von Benutzerrechten auf zwei Zugriffsebenen.*

Ausgangspunkt für die Erteilung von Zugriffsrechten ist ein Datenschema und eine verfolgte Sicherheitspolitik. Entsprechend der Sicherheitspolitik werden aus dem Schema die Klassen entfernt, die nicht zum Semantikraum einer Benutzergruppe gehören sollen oder müssen (restriktiver Ansatz). Dabei werden die beschriebenen Eigenschaften und Abhängigkeiten objektorientierter Klas-

senarchitekturen ausgewertet und entsprechende weitere Klassen und Beziehungen ausgeblendet:

1. Wird eine Klasse aus dem Schema entfernt, so werden auch alle von dieser Klasse abgeleiteten Klassen aus dem Schema entfernt.

2. Wird eine Klasse aus dem Schema entfernt, so werden auch alle Klassen, die Aggregationen dieser Klasse enthalten, aus dem Schema entfernt.

Das Ergebnis ist ein strukturell abgeschlossenes und unabhängiges Teilschema. Für das Beispiel aus Abbildung 6.12 bedeutet dies: Mit dem Entfernen der Klasse `Class3` aus dem Schema werden zunächst die abgeleiteten Klassen `Class4` und `Class5` automatisch ausgeblendet. Da die Klasse `Class7` eine Aggregation zur Klasse `Class3` definiert und damit strukturell von dieser abhängig ist, wird auch `Class7` aus dem Schema entfernt. Der Schritt der benutzerabhängigen Schemabeschreibung kann auch „positiv" ablaufen: Ausgangspunkt ist dabei ein leeres Benutzerschema. Wird nun aus dem Ausgangsschema eine Klasse zum Benutzerschema hinzugefügt, so werden zunächst deren Basisklassen automatisch in das Benutzerschema aufgenommen. Anschliessend werden alle Klassen in das Benutzerschema eingetragen, die durch Aggregation im Ausgangsschema als Teil einer hinzugefügten Klasse definiert wurden. Beziehungen zwischen den sichtbaren Klassen werden entsprechend den Definitionen im Ausgangsschema gesetzt.
Relativ zu einem Teilschema werden nun die Instanzen beschrieben, die einer Benutzersicht angehören sollen. Dazu werden für die Klassen des Benutzerschemas - soweit Instanzen dieser Klasse explizit freigegeben werden sollen - Anfragen formuliert, deren Ergebnismengen Instanzen der jeweiligen Klasse auflisten. Anfragen zur Auswahl von Instanzmengen beschreiben mehr oder weniger komplexe Eigenschaften, die von Objekten der Ergebnismenge erfüllt werden müssen. Somit können auch große Mengen von Einzelobjekten definiert werden, ohne dass diese explizit aufgelistet und in diesen Listen aufwändig gepflegt werden müssen. Die Aufnahme eines Objektes in eine Benutzersicht kann Auswirkungen auf die Sichtbarkeit anderer Objekte haben. Aggregiert sich ein Objekt aus anderen Objekten (z.B. „Linie besitzt Endpunkte", „Flur besteht aus Flurstücken",...), so werden auch die Teilobjekte in die Sicht mit aufgenommen. Die Kompatibilität der Instanzen mit dem Schema bleibt in jedem Fall erhalten, da mögliche Aggregationen bereits in der Klassenbeschreibung erkannt und in das Schema mit aufgenommen wurden.

6.3.4 Integration und Erhalt von Konsistenzregeln

Ein wichtiger Bestandteil raumbezogener Datenmodelle sind Konsistenzregeln. Diese legen fachliche, geometrische oder topologische Eigenschaften fest, die zu jedem Zeitpunkt von den entsprechenden Datenmodellen eingehalten werden müssen. Konsistenzregeln sind Teil des Qualitätsmanagements und sorgen insofern für den Werterhalt von Datenbeständen. In konkreten Anwendungen wie ALK und ALKIS® finden solche Konsistenzregeln zur Beschreibung von Bedingungen der vollständigen Flächendeckung, der Überlappungsfreiheit von Flurstücksflächen oder der Eindeutigkeit abgeleiteter Flurstückskennzeichen Anwendung. Konsistenzregeln werden in Bezug auf ein Gesamtdatenmodell definiert und haben die Aufgabe, dessen Gültigkeit zu erhalten. Gleichzeitig werden aber die meissten Transaktionen in benutzerabhängigen Sichten des Modells ausgelöst. Um weder Geheimhaltungsforderungen, noch die Konsistenz des Gesamtdatenmodells zu gefährden, muss bei der Sichtenerstellung dafür gesorgt werden, dass innerhalb eines Teilmodells die konsistenzbedingte Ablehnung von Benutzeraktionen aus dem Teilmodell heraus begründet werden kann.

Zugriffsregel 11
Die Ablehnung von Transaktionen in Teilmodellen, aufgrund einer Verletzung von Konsistenzregeln im Gesamtmodell muss in den auslösenden Teilmodellen nachvollziehbar sein.

Um diese Regel zu erfüllen, bedienen wir uns wieder der Semantik von Aggregationen und führen in das Datenmodell Konsistenzklassen und -objekte ein. Diese beschreiben jeweils Konsistenzregeln und stehen in Beziehung zu den Klassen und Objekten, die von der Konsistenzregel betroffen sind. Eine Klasse bzw. die Instanzen, die eine Konsistenzverletzung auslösen können, besitzen eine Aggregation eines entsprechenden Konsistenzobjektes. Die Konsistenzregel wird somit zu einem Teil der Definition der auslösenden Klasse und deren Instanzen. Das Konsistenzobjekt selbst enthält neben der Beschreibung der Regel selbst, Aggregationen aller Objekte, auf deren Zustand sich die Regel bezieht. Nach dem beschriebenen Verfahren der Erteilung von Zugriffsrechten wird damit sichergestellt, dass ein bestimmtes Objekt nur zusammen mit allen Konsistenzregeln, die durch Methoden des Objektes verletzt werden können, einer Sicht und damit einem bestimmten Teilmodell angehören kann. Die Konsistenzregel selbst sorgt dafür, alle Objekte, die zur Überprüfung ihrer Einhaltung benötigt werden, in das Teilmodell mit aufzunehmen oder sämtliche Transaktionen grundsätzlich abzulehnen.

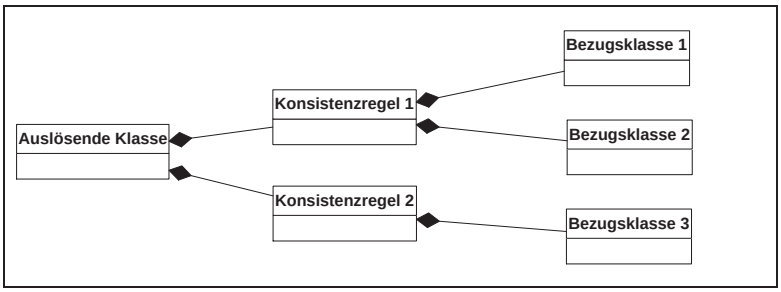

Abbildung 6.13: *Aggregation spezieller Klassen zur Beschreibung von Konsistenzregeln.*

6.4 Zusammenfassung

In Bezug auf die in den vorangegangenen Kapiteln dargestellten strukturellen und semantischen Schwierigkeiten bei der Erteilung und Auswertung von Zugriffsrechten für objektorientierte, raumbezogene Daten, wurde in diesem Kapitel ein Zugriffskonzept entworfen, das durch konsequente Umsetzung der objektorientierten Paradigmen den Aufbau geeigneter und kontrollierbarer Zugriffsstrukturen erlaubt.
Das Konzept umfasst drei Ebenen der Zugriffskontrolle:

1. Den kontrollierten Zugriff auf Schemadaten.

2. Den Aufbau von Semantikräumen im Sinne einer Spezialisierung bzw. Verallgemeinerung von Detailwissen.

3. Die Beschränkung des Zugriffs auf Klassenextents durch deskriptive räumliche und fachliche Einschränkungen.

Zentrale Strategien der Erteilung von Zugriffsrechten sind:

1. Die Identifizierung konsistenter Vererbungsebenen für die Definition gültiger Teilschemata.

2. Der Entwurf einer objektorientierten, räumlichen Anfragesprache für die Beschreibung von Teilmodellen.

3. Die Beschränkung aller Objektzugriffe auf die öffentlichen Methoden der beschreibenden Klassen zur Gewährleitung des kontrollierten Zugriffs auf Attributwerte.

Benutzer werden als Instanzen von Benutzerklassen in homogener Weise in das fachliche Datenmodell eingefügt. Diese erhalten ihre Berechtigungen implizit durch Ableitung von bereits vorhandenen Benutzerklassen oder von elementaren Privilegienklassen, die ihrerseits einzelne Zugriffsmethoden implementieren. Der Zugriff von Benutzer- und Privilegienklassen auf die geschützten (privaten) Methoden der Fachobjekt-Klassen wird mittels *friend*-Deklarationen oder vergleichbaren Konzepten realisiert. Dabei kann eine Fachobjekt-Klasse ausgewählten anderen Klassen - und damit auch Benutzerklassen - den Zugriff auf ihren privaten Deklarationsbereich erlauben, in dem sie diese Klassen als *friend* markiert. Mit der Eigenschaft, *friend* einer anderen Klasse zu sein, wird Benutzergruppen im übertragenen Sinne eine Vertrauenswürdigkeit ausgesprochen.
Mit dem entworfenen Berechtigungskonzept wird es möglich, den Zugriff auf objektorientierte, raumbezogene Daten in Anlehnung an ein gegebenes räumliches Fachdatenmodell gezielt und effizient zu beschränken. Die Granularität der Zugriffsbeschränkung entspricht im allgemeinen der Auflösung der jeweiligen Fachobjekt-Klassen und deren fachlichen Beziehungen.

Kapitel 7

Eingeschränkte Sichten auf Geo-Objekte

In den vorangegangenen Abschnitten wurde dargestellt, wie Zugriffsrechte für objektorientierte Geodaten erteilt und dabei Zugriffsregeln ausgewertet werden können. Nun sollen die Auswirkungen von Zugriffsbeschränkungen auf die Sichten einzelner Benutzer näher betrachtet werden. Insbesondere die Darstellbarkeit und der Informationsgehalt des zugriffsbeschränken Raumbezugs soll untersucht werden.
Die Beschreibung von Geo-Objekten setzt sich aus alphanumerischen Sachdaten, Beziehungen zwischen Objekten und der Modellierung des Raumbezugs zusammen. Der Raumbezug beschreibt dabei einen direkten Zusammenhang zwischen Fachobjekten und deren geometrischen Ausprägung in einem gewählten Bezugssystem. Es gibt verschiedene Möglichkeiten, den Sachdaten eines GIS ihre räumliche Ausprägung zu zuweisen: Bei präsentationsorientierten Systemen wird die Geometrie eines Datenbestandes i.d.R. direkt mit den Sachdaten attributiert, während sich bei informationsorientierten Systemen der Raumbezug als geometrisches Attribut der Sachdaten wieder findet. Unabhängig von der Art der Zuweisung, existiert zwischen den Sachdaten und deren Raumbezug eine im Datenmodell verankerte fachliche Beziehung.
Der Raumbezug vektorbasierter Geo-Objekte wird durch geometrische Elemente repräsentiert, die in ihrer Modellierung auf geometrischen Primitiven (Basis-Geometrietypen) basieren und erst durch thematische Spezifizierung zur Modellierung räumlicher Phänomene der Realwelt geeignet sind. Bei objektorientierten GIS liegt es nahe, die Objekte des Raumbezugs eines Datenmodells mit geeigneten Geometrie-Klassen zu modellieren und damit räumliche Datentypen zu definieren, die als Attribute den Raumbezug entsprechender Fachobjektklassen definieren können. Die Instanzen der Geometrie-Klassen

können nun mit den jeweiligen Fachobjekten verbunden werden. Im Datenmodell gibt es somit keinen Unterschied mehr zwschen räumlichen und nichträumlichen Attributen. Der modellierte Raumbezug besteht selber wieder aus Geo-Objekten, die auf den Beschreibungen mehr oder weniger spezifizierten räumlichen Klassen basieren. Den Ausführungen aus Kapitel 6 und insbesondere denen aus Abschnitt 6.3.3 folgend kann der Zugriff auf den Raumbezug eines Geo-Objektes und dessen Informationsgehalt (Repräsentation) somit ebenfalls durch Einschränkung des Zugriffs auf Schemainformationen gesteuert werden. Die Modellierung von Klassen für geometrische Elemente ist hier exemplarisch angelehnt an die SupportGIS-Modellierung, wie sie in Abschnitt 3.6 erläutert wurde. Dabei soll gezeigt werden, wie die Semantik der objektorientierten Vererbung verwendet werden kann, um thematische Spezifizierungen räumlicher Objekte zu beschreiben und diese vor bestimmten Benutzergruppen zu verbergen. Es wird insbesondere gezeigt, dass der Zugriff auf räumliche Objekte - insbesondere im Kontext einer Forderung nach Zugriffsbeschränkungen - nicht isoliert betrachtet werden kann: Der Zugriff auf die Geometrie eines Punktes hat Konsequenzen für die Geometrie angehängter Linien. Ebenso hat die Veränderung einer linienförmigen Geometrie unmittelbaren Einfluss auf die Eigenschaften einer definierten flächenförmigen Geometrie.

7.1 Klassenstruktur räumlicher Objekte

Die geometrischen Basistypen (Punkt, Linie, Fläche) definieren räumliche und topologische Grungeigenschaften, wie die Position eines Punktes, die Endpunkte einer Linie oder die Begrenzungslinien einer Fläche. Die Modellierung der geometrischen Ausprägung von Fachobjekten dient der Beschreibung und Repräsentation räumlicher Gegebenheiten und Abhängigkeiten sowie als Grundlage zur Ableitung grafischer Darstellungen ausgewählter räumlicher Ausschnitte. Zur Erfüllung dieser Anforderungen benötigt die Geometrie von Geo-Objekte eine semantische Komponente, die es ermöglicht, die Objekte hinsichtlich Ihrer Funktionen und thematischen Abhängigkeiten im Kontext des jeweiligen Fachdatenmodells zu spezifizieren. Die Modellierung der fachlichen Spezifizierung räumlicher Objekte kann durch die Verwendung der objektorientierten Vererbung und die Definition von räumlichen und topologischen Klassenbeziehungen auf spezifizierter Vererbungsebene erfolgen. Damit wird der objektorientierte Ansatz einer Modellierung von Geodaten konsequent fortgesetzt.
In diesem Sinne definieren die Klassen `GeoPoint`, `GeoCurve` und `GeoFace` die geometrischen Grundtypen Punkt, Linie und Fläche als Basisklassen al-

ler räumlichen Klassen. Diese sind ihrerseits von einer gemeinsamen geometrischen Oberklasse `GeoObject` abgeleitet. Dreidimensionale Geo-Objekte auf der Basis einer Klasse `GeoSolid` werden aufgrund der geringen Relevanz für eine Zugriffsstruktur hier nicht weiter berücksichtigt.
In Abhängigkeit von der zu modellierenden Fachaufgabe werden raumbezogene Fachbedeutungsklassen definiert, die ihren Geometrietyp und ihre elementaren räumlichen Eigenschaften durch Vererbung von einer `GeoObjekt`-Klasse beziehen. Durch nachfolgende Vererbungsschritte werden weitere fachliche Spezialisierungen in die jeweiligen Fachbedeutungsklassen modelliert. Die so beschriebenen räumlichen Fachbedeutungsklassen werden im Weiteren kurz als Fachbedeutungen bezeichnet. Die Vererbung zwischen Fachbedeutungen kann auch als vertikale Beziehung zwischen Fachbedeutungen gleicher Basisgeometrie behandelt werden. Diese wird dann im Sinne der Vererbungssemantik objektorientierter Paradigmen interpretiert. In dem exemplarischen Modell aus Abbildung 7.1 ist demnach eine Instanz von `Flurgrenze` immer zugleich auch Instanz der Klasse `Flurstuecksgrenze` und jede Instanz von `Flurstuecksgrenze` ist immer auch eine Instanz von `GeoCurve`.

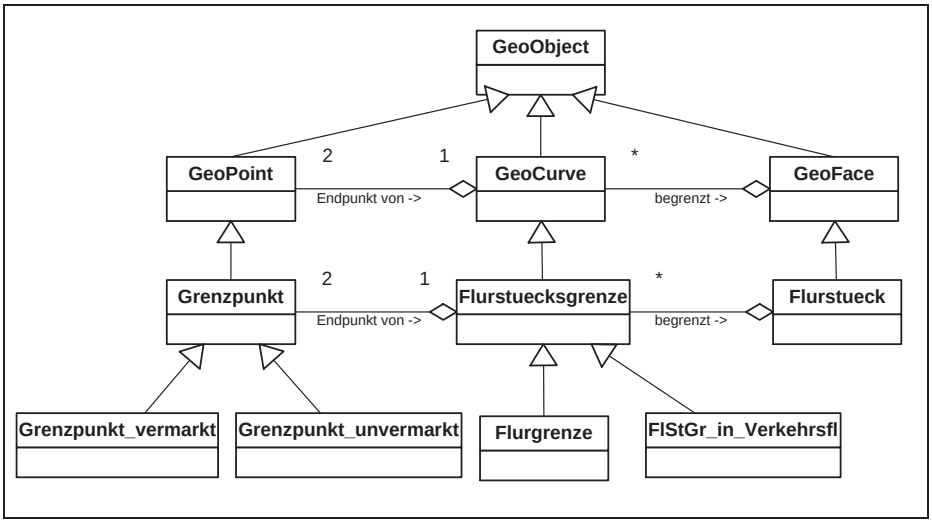

Abbildung 7.1: *Geometriefachbedeutungen als abgeleitete Geometrietypen.*

Den vertikalen Beziehungen zwischen Fachbedeutungen des gleichen Geometrietyps stehen die horizontalen Beziehungen zwischen Fachbedeutungsklassen unterschiedlicher Geometrietypen gegenüber. Diese beschreiben die für die geometrische Modellierung der Erdoberfläche benötigten, Geometrietyp-übergreifenden Definitions- und Abhängigkeitsbeziehungen. Auf der Ebene der Basis-Geometrietypen werden solche Beziehungen von dem gewählten Geometriemodell vorgegeben: Im einfachsten Fall wird ein flächenförmiges Geo-Objekt

von einem geschlossenen Umring begrenzender linienförmiger Geo-Objekte definiert. Diese werden ihrerseits jeweils durch zwei Endpunkten definiert. Auf fachlicher Ebene werden diese Beziehungen durch die Abhängigkeiten zwischen Fachbedeutungen überlagert und erfahren damit selbst eine fachliche Ausprägung. In einem konkreten Beispiel kann dies bedeuten: Eine Flurstücksfläche wird von einen geschlossenen Umring aus Flurstücksgrenzen definiert und jede Flurstücksgrenze wird durch mindestens zwei Grenzpunkte definiert. Horizontale Beziehungen der Form „A ist bestimmend für B" oder in umgekehrter Formulierung „B ist abhängig von A" werden ebenfalls vor dem Hintergrund einer definierten Vererbungshierarchie interpretiert: Mit einer Fachbedeutung A ist demnach die Klasse A oder jede von A abgeleitete Klasse gemeint. Mit der Verwendung von vertikalen und horizontalen Beziehungen eröffnet sich die Möglichkeit, Vererbungsbeziehungen und definierende Abhängigkeiten zwischen Fachbedeutungen einheitlich, im Sinne von aggregierenden Relationen zu beschreiben und dadurch Abhängigkeiten bei der Erteilung von Zugriffsrechten für die Instanzen von Fachbedeutungsklassen in einem einheitlichen Modell aufzudecken.

7.2 Zugriffsstrategie für Geo-Objekte im Kontext der Abhängigkeiten von Fachbedeutungen

Die beschriebenen Strukturen und Abhängigkeiten von geometrischen Objekten und deren Fachbedeutungen haben unmittelbare Auswirkungen auf die Festlegung und Durchsetzbarkeit von Zugriffsbeschränkungen für geometrische Daten. Dabei sind zwei Aspekte sicherer Zugriffsstrukturen besonders zu berücksichtigen:

1. Aus verfügbaren Informationen dürfen keine Rückschlüsse auf geheime Informationen möglich sein und

2. die Veränderung zugreifbarer Objekte darf keine unzulässigen Auswirkungen auf nicht zugreifbare Objekte haben.

Diese beiden Punkte sind zwar grundsätzliche Voraussetzungen für jede sinnvolle Zugriffsstrategie, stellen aber für eine Berechtigungsstruktur für Geo-Objekte, aus den bereits genannten Gründen eine besondere Herausforderung dar.
Wie bereits in Kapitel 6 bezüglich der Sachdaten erläutert, impliziert auch die Vererbung zwischen Fachbedeutungsklassen wesentliche Konzepte der Berechtigungsstruktur für Geo-Objekte. Berechtigungen für den schreibenden

Zugriff auf das generische Schema der Fachbedeutungsklassen entspricht den Ausführungen des Kapitels 6. Somit muss in diesem Abschnitt nur der Zugriff auf die Instanzen der Geometrie betrachtet werden.
Die Beschränkung des Zugriff auf Geo-Objekte erfolgt nach thematischen und räumlichen Aspekten. Zu diesem Zweck werden bei der Selektion zugreifbarer räumlicher Objekte die Lage und Ausdehnung sowie die zugewiesene Fachbedeutung der Objekte ausgewertet. Die Eigenschaften der räumlichen Lage einer Geometrie hinsichtlich eines Bezugssystems - z.B. die vollständige Lage eines Geo-Objekts innerhalb eines räumlichen Ausschnitts, definiert durch ein Rechteck oder Polygon - werden mit Hilfe einer Beschreibung von Bearbeitungsausschnitten, durch Sichtendefinitionen mit Anfragelogik ausgewertet. Die Konzepte hierzu werden in Kapitel 8 erläutert.
Damit wird die Fachbedeutung einer Geometrie zum zentralen Argument bei der Erteilung von Zugriffsrechten und der interpretierbaren Semantik der sichtbaren Objekte. Zwei Aspekte stellen bei der fachlichen Beschränkung von Zugriffsrechten für räumliche Objekte eine besondere Schwierigkeit dar:

- Geo-Objekte besitzen häufig mehr als nur eine Fachbedeutung (z.B. Flurgrenze und Flurstücksgrenze).
- Geo-Objekte sind aufgrund ihrer Fachbedeutung, von Objekten anderer Fachbedeutungen abhängig oder für diese definierend.

Aus dem Recht, auf ein geometrisches Objekt aufgrund seiner Fachbedeutung zugreifen zu dürfen, ergeben sich potenziell Zugriffsrechte für Geo-Objekte mit anderen Fachbedeutungen. Diese fachlichen Abhängigkeiten müssen im Rahmen einer Berechtigungsstrategie erkannt und nötigenfalls durch geeignete Maßnahmen vermieden werden. Dabei finden die Zugriffsregeln objektorientierter Strukturen in vollem Umfang Anwendung (Aggregationen, Vererbungen). Als Vorteil für die Berechtigungsstrategie erweist sich dabei, dass die Position einer zugreifbaren Fachbedeutung innerhalb des Vererbungsbaums unmittelbare Rückschlüsse auf die Zugriffsrechte für die Instanzen weiterer Fachbedeutungen zulässt. Somit kommt die Rechteverwaltung für Geometriedaten mit vergleichsweise wenigen Einzeldefinitionen aus und spiegelt zugleich den fachlichen Charakter der Datenbank wider.
Im Fall der Geometrie kann von einem Vererbungsbaum gesprochen werden, da alle Fachbedeutungen von einem Basis-Geometrietyp und schließlich von `GeoObject` abgeleitet sind. Isolierte Fachbedeutungen ohne Vererbungsstrukturen können somit nicht existieren.
Die Prinzipien der Rechtevergabe auf der Grundlage vertikaler Beziehungen (Vererbung), können am Beispiel der Abbildung 7.1 folgendermaßen dargestellt werden:

- Wird einem Benutzer das Recht erteilt, aus einer Menge von Geo-Objekten Instanzen einer Klasse `Flurstuecksgrenze` einzusehen, so darf dieser alle Geo-Objekte der spezifizierten Menge einsehen, die der Klasse `Flurstuecksgrenze` oder einer von `Flurstuecksgrenze` abgeleiteten Klasse (`Flurgrenze` oder `FlStGr_in_Verkehrsfl`) angehören. Bei diesen darf der autorisierte Benutzer auf genau die Eigenschaften zugreifen, die von der Klasse `Flurstuecksgrenze` oder einer ihrer Basisklassen - `GeoCurve` oder `GeoObject`- beschrieben werden.
- Besitzt ein Benutzer Schreibrechte für die Instanzen einer Klasse `Grenzpunkt` innerhalb einer ausgewählten Menge von Geo-Objekten, so darf dieser auf alle Instanzen der Menge zugreifen, die der Klasse `Grenzpunkt` oder einer von `Grenzpunkt` abgeleiteten Klasse (`Grenzpunkt_vermarkt` und `Grenzpunkt_unvermarkt`) angehören. Unter diesen kann er jeweils auf die Eigenschaften schreibend zugreifen, die von der Klasse `Grenzpunkt` oder einer ihrer Basisklassen (`GeoPoint` und `GeoObject`) beschrieben werden.

Unter der Annahme, dass jede Art des Zugriffs auf Objekteigenschaften (lesend oder schreibend) ausschließlich über die öffentlichen Methoden der betroffenen Klasse erfolgen kann, lassen sich für vertikale Abhängigkeiten zwischen Fachbedeutungen folgende, rechterelevante Zusammenhänge festhalten:

1. Wenn A, B, C Fachbedeutungen sind, für die die Mengen SET_A, SET_B und SET_C der Instanzen (*Extents*) von A, B und C sowie FCT_A, FCT_B und FCT_C der Methoden von A, B und C existieren, dann folgt aus den Vererbungsbeziehungen $base(A, B)$ und $base(B, C)$ zwischen den Fachbedeutungen (A ist Basisklasse von B; B ist Basisklasse von C) : $e \in SET_C \Longrightarrow e \in SET_B$ und $f \in FCT_A \Longrightarrow f \in FCT_B$.
2. Sei $u \in U$ ein Benutzer mit dem Recht des Zugriffs auf SET_B, dann folgt für u die Berechtigung, für alle $e \in SET_B$ und $f \in FCT_B$, die Methode $e.f$ auszuführen.

Zusammengefasst folgt aus diesen Eigenschaften:

Zugriffsregel 12
Wird einem Benutzer das Recht erteilt, auf die Instanzen einer bestimmten Fachbedeutgung A zuzugreifen, so impliziert dies für den Benutzer Berechtigungen zum Ausführen der Methoden der Fachbedeutungsklasse A und aller - direkten oder indirekten - Basisklassen von A, für die Instanzen der Fachbedeutung A und aller - direkt oder indirekt - von A abgeleiteten Klassen.

Durch eine einfache Zuweisung von Fachbedeutungsklassen zu Benutzergruppen lassen sich diese intuitiven Zusammenhänge in eine Berechtigungsstruktur integrieren.
Nun sind aber Situationen denkbar (und auch gewollt), in denen die Instanzen spezialisierter - also abgeleiteter Klassen - in ihrer gesamten Ausprägung - und nicht nur in ihren spezialisierenden Eigenschaften - besonderen Zugriffsbeschränkungen unterliegen: Ein Benutzer soll beispielsweise das Recht haben, die Lage von allgemeinen Flurstücksgrenzen zu verändern. Dieses Recht soll aber nur für solche Flurstücksgrenzen gelten, die nicht gleichzeitig eine Flurgrenze oder eine noch höherwertigere Begrenzung darstellen. Da die Lage einer linienförmigen Geometrie aber zu den Eigenschaften der Basisklasse `GeoCurve`, bzw. der abstrakten Klasse `GeoObject` gehört, liegt die Vermutung nahe, dass die Vergabe von Zugriffsrechten auf Klassenebene derartige Zusammenhänge nicht abbilden kann. Bei der Lösung des Problems hilft uns wiederum die grundsätzliche Beschränkung von Objekt-Zugriffen auf die öffentlichen Methoden der entsprechenden Klassen, bzw. auf die Deklarationen des *public*-Bereichs.

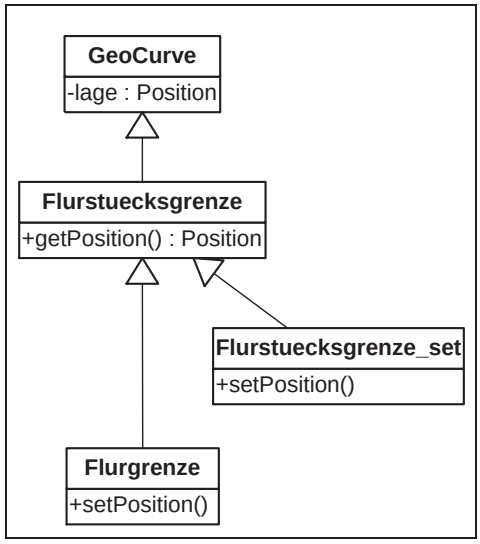

Abbildung 7.2: *Zugriff auf private Basiseigenschaften mit öffentlichen Methoden abgeleiteter Klassen.*

Abbildung 7.2 zeigt, wie am Beispiel der Flur- und Flurstücksgrenzen das Problem der klassenabhängigen Zugriffsbeschränkung ohne Paradigmenwechsel oder einem aufwendigen Regelwerk gelöst werden kann. Die Idee hinter dieser Art der Klassenmodellierung sieht vor, Klassenmethoden hinsichtlich ihres sicherheitskritischen Verhaltens im Sinne der angestrebten Zugriffsstrategie zu ordnen und nötigenfalls aus dem direkten Vererbungspfad in eigene Klassen

auszugliedern. Diese Klassen beinhalten alle sicherheitskritischen Methoden der Ausgangs-Klasse, deren Instanzen bearbeitet werden sollen, und leitet sich von dieser ab. Die neu gegründete Klasse ist selbst zunächst keine Basisklasse für weitere Fachbedeutungen. Dadurch ergibt sich die Möglichkeit, auch den Zugriff auf die Eigenschaften gemeinsamer Basisklassen in Abhängigkeit von der Klassenzugehörigkeit der Instanzen zu reglementieren. In der dargestellten Vererbungshierarchie können die Zugriffsrechte auf Flurstücksgrenzen nun in der gewünschten Weise erteilt werden, indem den betreffenden Benutzern der Zugriff auf Instanzen der Klasse `Flurstuecksgrenze_set` eingeräumt wird. Mithilfe einer Methode dieser neuen Klasse ist es den Benutzern nun möglich, die Lage von solchen Flurstücksgrenzen zu setzen, die zum *Extent* der Klasse `Flurstuecksgrenze_set` gehören. Obwohl Flurgrenzen auch Flurstücksgrenzen sind, können die für Flurstücksgrenzen autorisierten Benutzer nicht automatisch auch schreibend auf Flurgrenzen zugreifen, da sich diese nicht im *Extent* von `Flurstuecksgrenze_set` befinden.
Allgemein kann für vertikale und horizontale Abhängigkeiten von Fachbedeutungsklassen festgestellt werden:

Seien A und B zwei Fachbedeutungsklassen mit einer fachlich definierten Abhängigkeit „A ist bestimmend für B" oder $defines(A, B)$. Wenn es nun Instanzen $a \in SET_A$ und $b \in SET_B$ mit $defines(a, b)$ gibt und eine Funktion $f \in FCT_A$ existiert, so dass $a.f$ eine Eigenschaft von a verändert, dann kann $a.f$ auch eine Eigenschaft von b verändern.

Bei der Vergabe von Zugriffsrechten für die Instanzen von Fachbedeutungsklassen ist demnach von Bedeutung, ob diese

1. bestimmend für andere Fachbedeutungen sind und
2. die Klasse oder eine ihrer Basisklassen Methoden besitzt, die schreibend auf die Eigenschaften der ausführenden Instanz zugreift.

Wenn beide Punkte erfüllt sind, werden einem Benutzer mit der Erteilung von Zugriffsrechten für Instanzen einer bestimmten Fachbedeutung implizit Schreibrechte für die Instanzen abhängiger Fachbedeutungen erteilt. Eine Abhängigkeitsbeziehung der Form „A ist bestimmend für B" ($defines(A, B)$) kann durch eine Aggregation der Form „B wird bestimmt von A" (*defined_by(B,A)*) in umgekehrter Richtung modelliert werden. Demnach ergibt sich der oben beschriebene Automatismus direkt aus den in Abschnitt 5.4 definierten Zugriffsregeln 8 und 9.
Da die geometrische Ausprägung eines Realwelt-Objektes im Datenmodell durch

ein Attribut des entsprechenden Fachobjektes beschrieben ist, ergibt sich der grundsätzliche - und damit mindestens lesende - Zugriff auf die Geometrie aus den Zugriffsrechten für das jeweilige Fachobjekt (Abbildung 7.3): Die Einsicht in eine Gebäudegeometrie ist abhängig von den Zugriffsrechten auf das Geometrieattribut eines entsprechenden Gebäudeobjekts; Einblick in die Geometrie eines Flurstücks ist nur über das Flurstücksobjekt mit Geometrie-Attribut möglich. Auf diesem Grundsatz basiert auch die Zugriffsbeschränkung auf Objekte, die in einem bestimmten räumlichen Ausschnitt liegen.

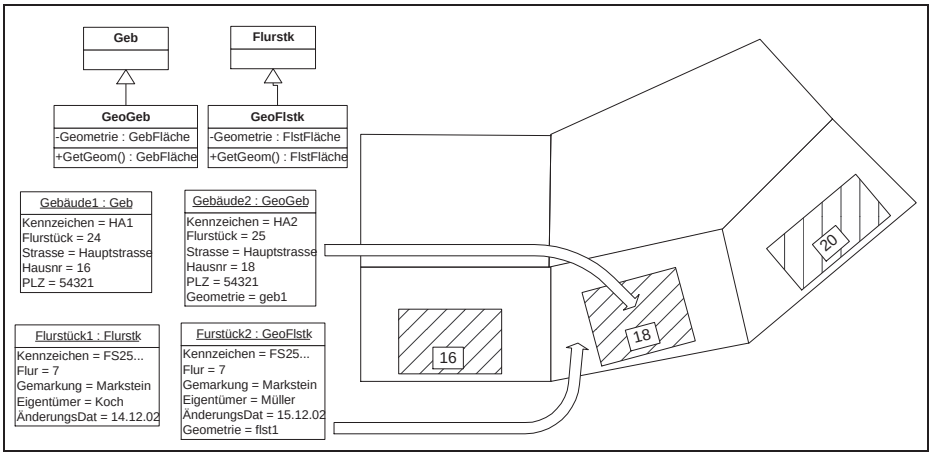

Abbildung 7.3: *Zugriff auf die Objektgeometrie über Attribute und Methoden des entsprechenden Fachobjektes.*

7.3 Zugriffsbeschränkungen für die Objekt-Geometrie

Die beschriebene Berechtigungsstrategie für räumliche Objekte kann von einem entsprechenden GIS so interpretiert werden, dass sie unmittelbare Auswirkungen auf die grafische Repräsentation bei Anfrage der Objekte durch einen Benutzer hat. Ein einfaches Beispiel soll die Idee der rechteabhängigen Darstellung räumlicher Objekte erläutern. Abbildung 7.4 zeigt eine Vererbungshierarchie für flächenförmige Fachbedeutungen.

Dabei bildet wiederum die Klasse `GeoFace` die Basisklassen für alle weiteren Flächen-Fachbedeutungen. Die oberste themenspezifische Fachbedeutung wird durch die Klasse `Gebäude`, zur Beschreibung der Geometrie von Gebäudeflächen, repräsentiert. Eine weitere thematische Spezialisierung erfahren Gebäudeflächen durch eine Unterscheidung in Gebäuden, die als Wohnraum genutzt werden und solchen, die einer anderen, noch nicht weiter spezifizierten

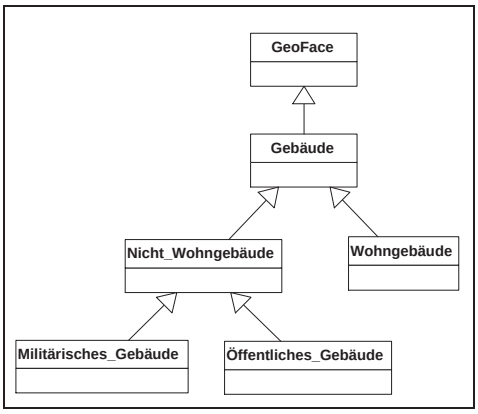

Abbildung 7.4: *Vererbungshierarchie von Flächen-Fachbedeutungen.*

Nutzung dienen. Dafür werden von `Gebäude` die Klassen `Wohngebäude` und `Nicht_Wohngebäude` abgeleitet. In einem weiteren Vererbungsschritt werden nun die `Nicht_Wohngebäude` in militärisch und öffentlich genutzte Gebäude unterschieden. Die Klassen `Militärisches_Gebäude` und `Öffentliches_Gebäude` beschreiben diese Unterscheidung.
Auf der Basis der so modellierten Fachbedeutungen werden die Gebäudeflächen eines Kartenausschnitt in unterschiedlichen Graustufen dargestellt. Wenn a ein Anwender des GIS ist, dessen Zugriffsrechte für Flächenfachbedeutungen (unabhängig von anderen Geometrietypen) auf die Beschreibung und die Instanzen der Fachbedeutung `Gebäude` des fraglichen Ausschnitts beschränkt ist, so erhält er eine Benutzersicht auf die vorhandenen Gebäude, entsprechend der Darstellung in Abbildung 7.5. Der Benutzer a kann alle Gebäude des Ausschnitts sehen, ohne aus der Darstellung der Geometire weitere Informationen über die spezielle Nutzung eines Gebäudes erkennen zu können. Die Sicht des Benutzers a auf die Daten erlaubt ihm also nur, Gebäude mit ihren allgemeinen Eigenschaften zu identifizieren. Eine wesentlich differenzierte Sicht auf die Da-

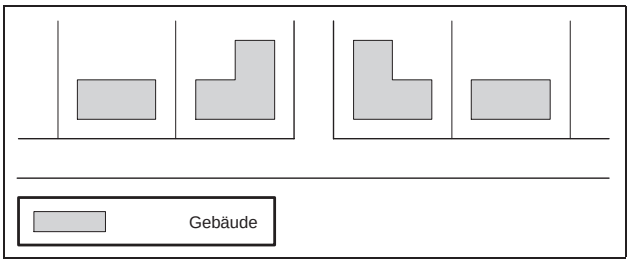

Abbildung 7.5: *Benutzersicht a, eingeschränkt auf die Fachbedeutung* `Gebäude`.

ten des gleichen Ausschnitts wird einem Benutzer b ermöglicht, dem Zugriffs-

rechte für die Instanzen der Klassen `Wohngebäude` und `Nicht_Wohngebäude` des gewählten Ausschnitts erteilt wurden. Implizit wird ihm damit - aufgrund der Vererbung - auch der Zugriff auf die allgemeinen `Gebäude`-Eigenschaften der Instanzen des Ausschnitts gewährt. Die Sicht des Benutzers b auf die Gebäude des Kartenausschnitts ist in Abbildung 7.6 dargestellt. Die erteilten Zugriffsrechte ermöglichen es dem Benutzer b, Gebäude zu erkennen und diese hinsichtlich ihrer Nutzung in Wohngebäude und Nicht-Wohngebäude zu unterscheiden. Ein im Sinne der dargestellten Klassenstruktur vollständiger Zugriff

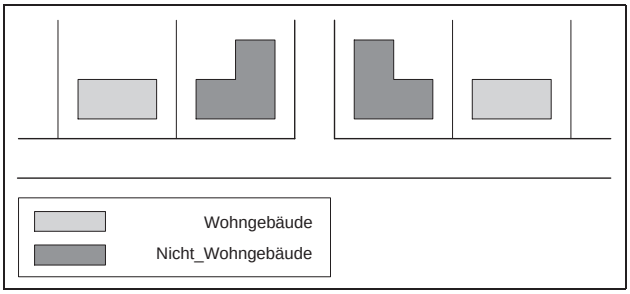

Abbildung 7.6: *Benutzersicht b, eingeschränkt auf* `(Nicht-)Wohngebäude`.

auf die Gebäudeflächen des Ausschnitts wird einem Anwender c zuerkannt, indem diesem Zugriffsrechte für die Beschreibungen und Instanzen der Klassen `Wohngebäude`, `Militärisches_Gebäude` und `Öffentliches_Gebäude` des gewählten Kartenausschnitts zugewiesen werden. Die Sicht des Benutzers c auf die Gebäudeflächen des Ausschnitts zeigt Abbildung 7.7. Die Zugriffsrechte des Benutzers c entsprechen nur solange einem Vollzugriff auf die Geometrie des Kartenausschnitts, solange die Klassenstruktur der Flächen-Fachbedeutungen nicht um weitere Klassen erweitert wird. In diesem Sinne muss auch die sichtbare Datenmenge eines Benutzers mit Vollzugriff als Benutzersicht beschrieben werden. Mit der hier beschriebenen Zugriffsstrategie für Geo-Objekte

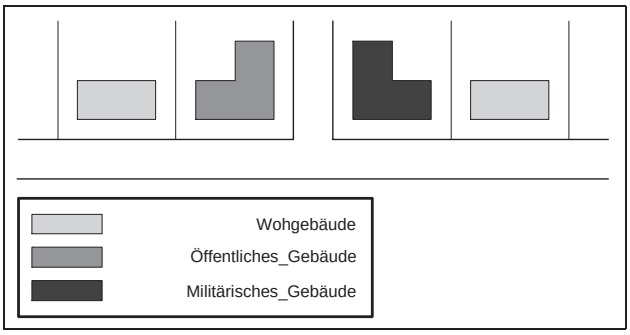

Abbildung 7.7: *Benutzersicht c, vollständige Ansicht der Gebäudemodelle.*

kann der Zugriff auf die spezialisierenden Eigenschaften räumlicher Objekte

und der grundsätzliche Zugriff auf Geo-Objekte bestimmter Fachbedeutungen und Geometrietypen reglementiert werden. Dabei ist die Information, die ein Benutzer über ein Objekt erhält, abhängig von dem konkreten Datenschema, das ihm bezüglich der zugreifbaren Objekte zur Verfügung steht. Allein diese Komponente der Zugriffsbeschränkung kann für die Realisierung netzbasierter Auskunftssysteme, im Sinne OGC-konformer *Web-Services* sinnvoll genutzt werden: Ein registrierter Benutzer stellt auf der Basis des ihm zugänglichen Datenschemas eine Anfrage an einen entsprechenden Server. Dieser ermittelt die angefragten und zulässigen Objekte, bereitet diese in Kompatibilität zu der im Benutzerprofil abgelegten XSD (*XML Schema Definition*) oder DTD (*Document Type Definition*) als XML-Datenformat (z.B. GML 2.0) auf und schickt sie zurück an den Benutzer. In Kapitel 8 wird erläutert, wie die Logik einer Anfragesprache verwendet werden kann, um die Zugriffsrechte für Benutzer auf einzelne Instanzen oder Instanzmengen von Geo-Objekten bzw. Fachobjekten, in Abhängigkeit der Objekteigenschaften zu beschränken.

7.4 Zusammenfassung

Die Identifizierung von Vererbungsebenen im Rahmen des entworfenen Berechtigungskonzeptes, ist insbesondere für den Entwurf und die Präsentation thematisch-orienierter, räumlicher Ausschnitte relevant. In diesem Kapitel wurde gezeigt, wie mithilfe der Vererbungssemantik Klassen zur Beschreibung von Geo-Objekten fachlich spezialisiert werden können. Zusammen mit den definierten Beziehungstypen innerhalb einer Vererbungsstufe ergeben sich Vererbungsebenen, die fachlich geschlossenene und konsistente Schemata definieren. Mit der Beschränkung der Zugriffsrechte von Benutzern oder Benutzergruppe auf ausgewählte Vererbungsebenen werden die sichtbaren Instanzen der Geometrie-Klassen in einer verallgemeinerten oder in einer spezialisierteren Form dargestellt und schließlich als solche von den Benutzern interpretiert.

Kapitel 8

Zugriffsbeschränkte Arbeitsbereiche

Beim realen Einsatz von GIS, verfolgen Benutzer in der Regel eine bestimmte thematische Intention. Sie melden sich bei einem Sytem an, um eine fachlich begrenzte Aufgabe zu erfüllen oder eine konkrete Interessenslage zu bedienen. Dabei erwarten sie eine fachlich motivierte Sicht auf die verfügbaren Daten. Häufig überschneiden sich die formalen Berechtigungen der Benutzer an den Daten des System mit der fachlichen Organisation dieser Daten. Insofern erscheint es sinnvoll, die formale Beschreibung von Benutzerrechten in die fachliche Organisationsstruktur einer Datenhaltungskomponente zu integrieren. Eine dahingehende Zielsetzung kann durch zwei Bedingungen zusammenfassend dargestellt werden:

1. Ein Benutzer, der mit einer konkreten Aufgabe betraut ist, soll aus der Vielzahl der in der Datenbank gespeicherten Informationen möglichst in Echtzeit Zugriff auf die für seine Aufgabe relevanten Daten erhalten.
2. Gleichzeitig soll dieser Benutzer nur die Daten bearbeiten können, die für seine Aufgabe relevant sind - selbst dann, wenn seine Benutzerrechte innerhalb der Datenbank darüber hinaus gehen.

Mit der Erfüllung dieser Bedingungen wird die Gefahr der versehendlichen Zerstörung und fehlerhaften Modifikation von Daten durch einen Benutzer reduziert. Gleichzeitig wird das unnötige sperren von Objekten für andere Benutzer vermieden. Benutzer wählen bei der Anmeldung an ein GIS gezielt einen fachlichen Ausschnitt zur Einsichtnahme oder zur Bearbeitung aus. Die Zusammenhänge zwischen Geo-Objekten, Benutzersichten, Fachthemen und

Präsentationen sollen zu diesem Zweck in fachlichen Arbeitsbereichen, sogenannten *GIS-Workspaces*, verwaltet werden. Wenn ein Benutzer innerhalb eines GIS ein definiertes Fachthema auswählt, so muss eine übergeordnete Zugriffsverwaltung die Rechte des Benutzers und die Einstellungen des gewählten Themas überlagern und sowohl aus dem Datenschema als auch aus den Nutzdaten die „Schnittmengen" aus Zugriffsrechten und Fachthematik bilden. Das Ergebnis der Verschneidung kann dem angemeldeten Benutzer als aktueller Arbeitsbereich zur Verfügung gestellt werden. Daten und Schemainformationen müssen dabei von der Zugriffsverwaltung derart ausgewertet werden, dass jedem Benutzer immer ein in sich geschlossenes und konsistentes Teilmodell präsentiert wird.

8.1 Definition und Bedeutung von Arbeitsbereichen

Im Rahmen von *Workspaces* sollen die Anwender von Geodaten, entsprechend ihren unterschiedlichen fachlichen Aufgaben und Anforderungen und im Rahmen ihrer Berechtigungen, Zugriff auf die Informationen einer oder mehrerer Geo-Datenbanken erhalten. Die Arbeitsbereiche ergeben sich dabei als Sicht auf Teilbereiche des Datenbestandes und dessen Schemainformationen. Der Begriff der „Sicht" ist dabei durchaus im Sinne der Terminologie relationaler Datenbanken zu verstehen: Nach festen algebraischen Vorschriften werden aus einem Modell der realen Welt themen- und benutzerabhängige Teilmodelle ermittelt und präsentiert. Der aktuelle Arbeitsbereich eines Benutzers ergibt sich aus der zum Anmeldezeitpunkt ermittelten Verschneidung von Zugriffsrechten mit dem ausgewählten Themenbereich:

1. Zugriffsrechte definieren Benutzer- oder Benutzergruppen-spezifische Datenschemata als Teilmenge eines Ausgangsdatenschemas und spezifizieren für die betroffenen Benutzer zugreifbare Objekte, die den Beschreibungen des Benutzerschemas genügen.

2. Themenbereiche beschreiben die Teile eines vorgegebenen Datenmodells, die zur Bearbeitung spezifischer Fachaufgaben oder zur Darstellung themenspezifischer Aspekte eines Datenbestandes benötigt werden. Dazu wird auf der Basis eines Ausgangsdatenschemas ein Teilschema definiert. Die Objekte, die einem Themenbereich angehören sollen, werden formal beschrieben. Themenbereiche dienen der thematischen Strukturierung von Datenmodellen.

Formal können Zugriffsrechte und Themenbereiche mit den gleichen Methoden beschrieben werden. Somit lässt sich auch das Ergebnis einer Verschneidung

von Zugriffsrechten und Themenbereichen formal ableiten. Die Sichten von Benutzern und Themenbereichen schränken die Menge aller verfügbaren Daten nach räumlichen und fachlichen Kriterien ein. Diese Kriterien können mit räumlichen und fachlichen Anfragen formal beschrieben werden. Teilschemata und Systemanfragen zur Definition von Sichten haben also zwei prinzipielle Einsatzbereiche:

1. Die Erteilung von Benutzerrechten und
2. die Beschreibung von Themenbereichen.

Themenbereiche werden unabhängig von Benutzern oder Benutzerklassen definiert. Der grundsätzliche Zugang von Benutzern zu bestimmten Themen wird durch spezielle Zugriffsrechte reglementiert. Die Themenbereiche selber benötigen keine zusätzliche Verwaltungsstruktur für die Benutzer, die auf sie zugreifen dürfen. Somit können Benutzern oder Benutzergruppen neue Zuständigkeiten zugewiesen werden, ohne dass die Definition der Themenbereiche verändert werden muß. Erst bei der Auswertung der Anfragen zum Zwecke der Präsentation eines Themas müssen die Zugriffsrechte des anfragenden Benutzers mit einbezogen werden. Zwischen den verschiedenen Themenbereichen eines GIS können inhaltliche Überlappungen existieren, in denen die ermittelten Sichten gemeinsame Objekte enthalten. Wenn mehrere Benutzer - mit ausreichenden Berechtigungen - gleichzeitig auf solche Themen zugreifen, besteht die Gefahr, dass sich einzelne Objekte im Zugriff mehrerer Benutzer befinden. Konflikte, die durch das *Update*-Verhalten derartiger Objekte entstehen, müssen durch geeignete Tansaktions- bzw. *Locking*-Mechanismen aufgelöst werden.
Nach der Anmeldung eines Benutzers bei einem Themenbereich wird der aktuelle *Workspace* des Benutzers ermittelt. Die Daten dieses Arbeitsbereichs sollen dann entsprechend einer gewählten Konfiguration in der *Client*-Applikation präsentiert werden. Konfigurationen zur Definition der Darstellung von Geo-Objekten sollen im Zusammenhang mit Berechtigungsstrukturen und Zugriffsverwaltungen in dieser Arbeit nicht weiter berücksichtigt werden. Daher wird die Präsentation aus dem Management von Arbeitsbereichen ausgegliedert und den lokalen Einstellungen des *Clients* (z.B. durch Konfigurationsdatenbanken) überlassen.

8.1.1 Konzeptioneller Aufbau von Arbeitsbereichen

Die Erstellung eines aktuellen Arbeitsbereichs für einen Benutzer beginnt mit einer Auswahlliste verfügbarer Themen. Aus diesen kann der angemeldete Benutzer das fachliche Thema auswählen, dessen Daten er einsehen oder bearbeiten möchte. Die Auswahllisten sind Bestandteile der definierten Zugriffsrechte

von Benutzergruppen. Somit kann ein einzelner Benutzer als Repräsentant einer Benutzerklasse nur unter den Themen auswählen, für die seiner Benutzergruppe explitiz entsprechende Zugriffsrechte erteilt wurden.

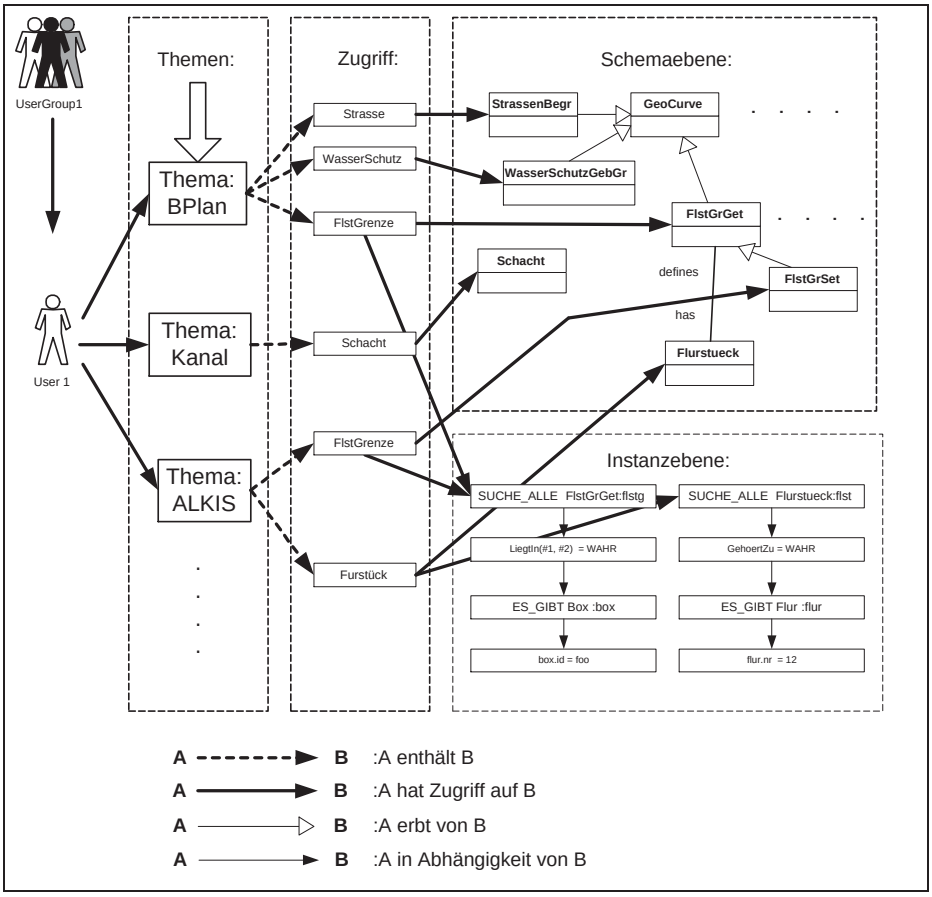

Abbildung 8.1: *Zuweisung von Schema- und Instanzbeschreibungen zu Themenbereichen.*

In Abbildung 8.1 ist skizzenhaft dargestellt, wie ein Benutzer auf die Themen eines GIS Zugriff erhält und wie diesen Themen Schema- und Objektbeschreibungen zugewiesen werden. Die dargestellten Themen 'BPlan' ,'Kanal' und 'ALKIS' stellen typische Fachanwendungen in der raumbezogenen Datenverarbeitung dar, die bei unterschiedlicher fachlicher Ausrichtung auf gemeinsame Ressourcen zurückgreifen müssen. Die Teilschemata und Objektmengen, die den Themen zugewiesen werden sollen, sind durch zusätzliche Zugriffsstrukturen spezifiziert. Diese repräsentieren jeweils die für ein spezielles Thema relevanten Informationen einer Fachobjektklasse. So enthält jede Zugriffsklasse eines Themas einen Verweis auf genau eine Fachobjektklasse des Schemas

sowie Referenzen der Beschreibungen der benötigten Instanzen dieser Fachobjektklasse. Mithilfe dieser Zugriffsklassen wird sichergestellt, dass nur solche Instanzen ausgewählt werden, zu denen innerhalb der Sicht des Themas die Schemainformationen vorliegen. An dieser Stelle soll angenommen werden, dass die Beschreibung von Instanzen durch die Definition von Anfragen erfolgt. Auch wenn andere Formen der Beschreibung von Instanzen denkbar sind (z.B. durch explizite Listen), so kann mit der Verwendung deskriptiver Anfragesprachen ein allgemeingültiges Beschreibungskonzept realisiert werden. Die Anfragen zur Beschreibung der Instanzen von Themenbereichen müssen Auswertungen nach Eigenschaften des Raumbezugs, nach Eigenschaften der Sachdaten und hinsichtlich der Beziehungen zwischen Objekten unterstützen. Auf diese Weise werden aus den vorhandenen Datenbeständen Objektmengen ermittelt, die ein geschlossenes Modell der Daten repräsentieren und aufgrund der vorgegebenen Objekteigenschaften an die Anforderungen bestimmter Aufgabenstellungen angepasst sind. Konflikte in der Datenbearbeitung durch konkurierende Zugriffe können dadurch bereits weitestgehend vermieden werden. So werden zum Beispiel Sachbearbeiter unterschiedlicher Regionen, selbst bei gleicher fachlicher Thematik, nie oder nur selten die gleichen Geo-Objekte im Zugriff haben, da ihre Zuständigkeitsbereiche räumlich voneinander getrennt sind.

8.1.2 Arbeitbereiche als rechtebeschränkte Themenbereiche

Bevor die Daten und Schemainformationen eines Themenbereichs dem angemeldeten Benutzer präsentiert werden, müssen diese mit den Rechten des Benutzers abgeglichen bzw. auf diese eingeschränkt werden. Der Benutzer hat möglicherweise nur auf einen Teil der Daten des ausgewählten Themas ausreichende Zugriffsrechte. Aus mengentheoretischer Sicht muss vor der Weitergabe der Daten an den Benutzer die Schnittmenge aus Zugriffsrechten und dem Themenbereich ermittelt werden. In einem naiven Ansatz werden aus dem Gesamtdatenbestand des GIS zunächst alle Daten und Schemainformationen ermittelt und gesondert gespeichert, für die der angemeldete Benutzer ausreichende Zugriffsrechte besitzt. Anschließend werden die Daten des ausgewählten Themas ermittelt und ebenfalls zwischengespeichert. Die so erhaltenen Mengen von Fachobjektklassen und deren Instanzen werden nun sequenziell miteinander verschnitten. Das Ergebnis dieses Vorgangs repräsentiert den aktuellen Arbeitsbereich eines bestimmten Benutzers nach der Wahl eines Fachthemas. Dieser Ansatz birgt jedoch erhebliche Nachteile:

1. Die Vorgehensweise muss als wenig effizient eingeschätzt werden, da zunächst alle Anfragen auf den vollständigen Datenbestand ausgeführt wer-

den müssen, was allein schon sehr aufwendig sein kann. Im Anschluss daran müssen (potenziell) sehr grosse Mengen miteinander verschnitten werden, die bei einer geeigneten Modellierung von Benutzerrechten und Themenbereichen weitestgehend Deckungsgleichheit oder zumindest Teilmengenbeziehungen aufweisen.

2. Während des Ermittlungsprozesses werden Objekte bei der Datenbank angefragt und in temporären Mengen gesammelt, die entweder nicht dem gewählten Thema zugeordnet sind oder durch die Zugriffsrechte des anfragenden Benutzers nicht abgedeckt werden. Damit ist die Sicherheit der Daten - mindestens für kurze Zeit - gefährdet, da vertrauliche Informationen von Transaktionen des Benutzers betroffen sind und damit entgegen der festgelegten Sicherheitspolitik in dessen Zugriff sind.

3. Je nach verwendetem DBMS kann es auch zu unnötigen *Locking*-Situationen kommen, die entweder den Prozess der Datenermittlung behindern und fehlerhaft abbrechen, wenn abgefragte oder betroffene Daten bereits im Zugriff anderer Benutzer sind, oder diese am Zugriff auf ihre Daten durch einen *Lock* gehindert werden.

In einem verbesserten Ansatz wird dem Erhalt der Sicherheitsstrategie höchste Priorität eingeräumt. Dabei werden zunächst alle Daten und Schemainformationen entsprechend den Zugriffsrechten des angemeldeten Benutzers ermittelt. Diese Daten werden in einem physikalisch oder logisch getrennten Bereich der Datenbank - temporär - abgelegt oder unter die Kontrolle einer geeigneten Zugriffsstruktur gestellt und dienen als Datenbasis für alle weiteren Auswertungen. Wählt der Benutzer nun einen Themenbereich aus, so werden die Daten, die im Kontext dieses Arbeitsbereichs zur Präsentation kommen sollen, aus der benutzerspezifischen Datenbasis ausgewählt. Dazu werden die Systemanfragen, mit denen die Fachobjekt-Instanzen eines Themas beschrieben werden, auf der Menge der benutzerverfügbaren Daten ausgeführt. Die Datenmenge, die einem Benutzer auf dem Anwendungs-*Client* zur Bearbeitung angeboten wird, repräsentiert somit die Schnittmenge aus Benutzerrechten und dem ausgewählten Themenbereich. Neben einer höheren Gewährleistung der vertraulichen Behandlung der Daten werden mit diesem Verfahren auch die Laufzeiten der Anfragealgorithmen verkürzt, da die zweite Anfragephase auf einen bereits reduzierten Datenbestand aufsetzt. Auch dieses Vorgehen genügt noch nicht den Anforderungen einer zuverlässigen, effizienten und konsistenzerhaltenden Zugriffsarchitektur für Geodaten. In Abschnitt 8.2 wird daher ein Ansatz vorgeschlagen, dessen Motivation darin liegt, nicht die Ergebnisse von Zugriffsrechten und Themenbereichen zu verschneiden, sondern die enthaltenen Beschreibungen zu vereinen und somit, nach der Auswahl eines Themas

durch einen Benutzer, eine zusammenhängende Darstellung der zugreifbaren Daten- und Schemainformationen zu erhalten.

8.2 Deskriptive Anfragen zur Beschreibung von Sichten

Wie bereits dargestellt, basiert die zentrale Methode zur Beschreibung von Geodaten, die einem Themen- oder Rechtebereich angehören sollen, auf der Formulierung räumlicher, fachlicher und wertbezogener Anfragen. Mit den Konzepten einer deskriptiven Anfragesprache, die sowohl räumliche als auch objektorientierte Strukturen unterstützt, ist eine geeignete Grundlage zur Interpretation von Geodaten gegeben. In Abschnitt 3.6.3 wurde eine grafische Notation zur Darstellung und Formulierung raum-, themen- und wertbezogener Anfragen vorgestellt. Diese soll hier wieder aufgegriffen und näher erläutert werden. Anfragen die der Definition von Sichten dienen, werden von Administratoren des GIS definiert und Benutzern oder Themenbereichen zugewiesen. Solche Systemanfragen besitzen eine Syntax und Semantik in Analogie zu Benutzeranfragen. Diese werden jedoch im Gegensatz zu Systemanfragen vom Benutzer selbst auf Basis der ihm verfügbaren Schemainformationen definiert und ausgeführt, um mit diesen die verfügbaren Daten hinsichtlich bestimmter Kriterien auszuwerten und Objekte mit den beschriebenen Eigenschaften im Datenbestand zu finden. Anfragen, die vom Benutzer selbst definiert und ausgeführt werden können, dürfen natürlich nur die Informationen berücksichtigen, die der betreffende Benutzer auch einsehen darf. Insofern ist es notwendig, zwei Aspekte der Auswertung von Anfragen näher zu untersuchen:

1. Welcher Zusammenhang besteht zwischen den Anfragen zur Definition von Zugriffsrechten und denen zur Festlegung von Themenbereichen und welche Konsequenzen oder Optimierungsmöglichkeiten ergeben sich daraus für die Auswertung von Anfragen.

2. Wie können die Daten ermittelt werden, die bei der Ausführung von Benutzeranfragen innerhalb eines Themenbereichs berücksichtigt werden dürfen.

Diese Aspekte ergeben sich aus zwei Beweggründen: Einerseits soll die Rechenzeit bei der Ermittlung der Daten reduziert werden, indem zum Beispiel einschränkende Konditionen nicht mehrfach abgefragt werden ($a < 5 \wedge a < 6 \equiv a < 5$). Darüber hinaus steht die Sicherheit der Daten im Sinne von Integrität, Vertraulichkeit und Verfügbarkeit im Mittelpunkt der Untersuchungen

dieser Arbeit. Insofern soll die korrekte Zusammenfassung von Anfragen mit gleichen Ergebnisklassen sowie die Reduzierung ihrer Komplexität nach Regeln der mathematischen Logik und der damit verbundenen Minimierung der Objektzugriffe dazu beitragen, die Verfügbarkeit der Daten durch Reduzierung der *Locking*-Zustände zu erhöhen, die Nachvollziehbarkeit der Anfragelogik zu steigern und potenzielle Fehlerquellen in der Auswertung aufzudecken. Insgesamt wird dadurch eine verbesserte Durchsetzbarkeit von Sicherheitsanforderungen erreicht.
Mit dem Ziel der Zusammenfassung und Optimierung von Anfragen müssen zunächst die Komponenten der in Abschnitt 3.6.3 eingeführten Anfragesprache formalisiert werden, indem sie eine Interpretation im Sinne logischer Formeln erhalten.

8.2.1 Komponenten der Anfragesprache

Eine Anfrage q wird durch einen gerichteten zyklenfreien Graphen $G = (V, E)$ beschrieben. Die Knotenmenge V setzt sich zusammen aus:

- Klassenknoten (KK): KK verknüpfen die Klassen des Fachdatenschemas mit Quantoren der Aussagenlogik, erweitert um einen *SucheAlle* -Quantor. Jede Fachobjektklasse kann also in Verknüpfung mit einer der drei Quantoren *SucheAlle*, *FuerAlle* ($\forall$) oder *EsGibt* ($\exists$) als Knoten in der Anfrage vorkommen.
- Operatorknoten (OK): Die Operatorknoten aus V sind durch die Menge der logischen Operatoren - *UND*, *ODER*, *NICHT*, bzw. Kombinationen aus diesen definiert.
- Selektionstestknoten(SK): SK können nur in Verbindung mit einem KK auftreten. Sie definieren einschränkende Bedingungen für Attribute der Fachobjektklasse des assoziierten KKs.

Bilanzierungsknoten (BK), die ebenfalls in die Anfragesyntax integriert sind, können im Zusammenhang mit Sichtendefinitionen vernachlässigt werden, da sie als Anhang an Klassenknoten ausschließlich definiert werden, um die erhaltenen (und damit zugreifbaren) Anfrageergebnisse nach bestimmten Eigenschaften (Minimum, Maximum, Anzahl,...) auszuwerten.
Damit ergibt sich die Knotenmenge zu: $V \subseteq KK \cup OK \cup SK$.
Die Kantenmenge $E \subset V \times V$ definiert die Menge aller gerichteten Verbindungen zwischen den Knoten aus V. Kanten zwischen Klassenknoten (KK) (möglicherweise unter Einbeziehung von Operatorknoten) können über ihre

pfadbeschreibenden Eigenschaften hinaus zur Auswertung fachlicher und topologischer Beziehungen zwischen den Instanzen der beteiligten Klassen verwendet werden. Derartige Kanten, denen eine auswertende Funktion zugewiesen ist, bezeichnen wir als Beziehungskanten: $E_{Rel} \subset (KK \cup OK) \times KK$. Alle übrigen Kanten $E \setminus E_{Rel}$ dienen ausschließlich der Verknüpfung von Anfragekomponenten und der Festlegung der Auswertungsreihenfolge: Die Eingabemenge eines Anfrageknotens wird nach den angegebenen Eigenschaften ausgewertet. Objekte, die die Kriterien des Knotens erfüllen, werden entlang der Ausgangskanten an die Nachfolgeknoten weitergeleitet.

8.2.2 Interpretation der Anfragekomponenten

Jede Anfrage q, die durch einen Graphen mit den beschriebenen Komponenten definiert wird, besitzt genau einen *SucheAlle KK*, der immer der Startknoten der Anfrage ist. Der Startknoten bestimmt die Klassenzugeörigkeit der Ergebnismenge der Anfrage. Im Folgenden wird die Interpretation der Anfragekomponenten im einzelnen erläutert.

8.2.2.1 Der *SucheAlle* Klassenknoten

Der *SucheAlle KK* besitzt keine Eingangs- und maximal eine Ausgangskante. Wenn die Ausgangskante existiert, dann führt sie zu den einschränkenden Bedingungen der Anfrage q. Das Ergebnis der Anfrage q mit einem *SucheAlle KK* bezüglich einer Klasse A ist eine Menge SET_A^q (Definition 6.3). Hat der Knoten *SucheAlle A* keine Ausgangskante, so ist das Anfrageergebnis die Menge SET_A^* aller Datenbankinstanzen der Klasse A. Die logische Interpretation der Anfragen aus Abbildung 8.2 lautet Links: $SET_A^q = SET_A$ und Rechts: $SET_A^q = \{a | a \in SET_A \wedge SubQuery\}$.

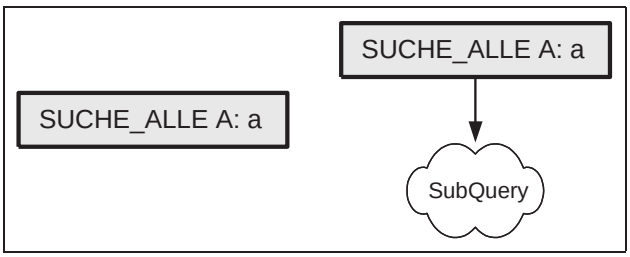

Abbildung 8.2: SucheAlle Startknoten suchen nach Instanzen der Klasse A. Links: ohne weitere Bedingungen. Rechts: mit weiter einschränkender Teilanfrage.

8.2.2.2 Der *EsGibt* Klassenknoten

*EsGibt KK*s entsprechen funktional den Existenzquantoren der Aussagenlogik. Sie beziehen sich auf eine Klasse des Schemas (z.B. *EsGibt B*) und formulieren eine Bedingung, nach der in der Datenbank mindestens eine Instanz der genannten Klasse mit den beschriebenen Eigenschaften gefunden werden muss. Der *EsGibt KK* hat eine Eingangskante, deren Ursprung in dem Knoten liegt, für dessen Ergebnisinstanzen jeweils die Existenz weiterer Datenbankinstanzen mit den beschriebenen Eigenschaften überprüft werden soll. Daher ist die Eingangskante immer eine Beziehungskante, die zugleich das Auswahlkriterium definiert. Der *EsGibt* Knoten besitzt maximal eine Ausgangskante, die für jede gefundene Instanz weitere einschränkende Bedingungen beschreibt. Die Teilanfrage aus Abbildung 8.3 hat die Bedeutung: $\exists b \in SET_B : relation(a, b) \wedge SubQuery$.

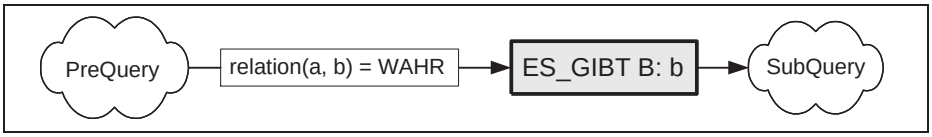

Abbildung 8.3: EsGibt Klassenknoten zur Verifikation der Existenz mindestens einer Instanz der Klasse B, die in der Beziehung *relation(a,b)* zur anfragenden Instanz a des letzten Vorgängerknotens aus *PreQuery* steht, und für die *SubQuery* erfüllt ist.

8.2.2.3 Der *FuerAlle* Klassenknoten

Die *FuerAlle* Klassenknoten werden im Sinne von Allquantoren der Aussagenlogik interpretiert. Der Kontext ihres Auftretens entspricht dem der *EsGibt* Klassenknoten. Im Unterschied zu diesen ist es aber nicht ausreichend, eine Instanz der Klasse zu finden, die den beschriebenen Eigenschaften genügt. Vielmehr müssen alle gefundenen Instanzen der Klasse des *FuerAlle* Knotens die gestellten Bedingungen erfüllen. Der *FuerAlle* Klassenknoten besitzt maximal eine Ausgangskante, die zu einem Knoten führt, an dem die Eigenschaften der Instanzen des *FuerAlle KK*s weiter spezifiziert werden. Die Eingangskante ist wiederum eine Beziehungskante, durch deren Definition die Instanzen des *FuerAlle* Knotens ausgewählt werden. Eine mathematische Darstellung des *FuerAlle*-Knotens aus Abbildung 8.4 ist:
$\forall b \in SET_B : \neg relation(a, b) \vee SubQuery$.

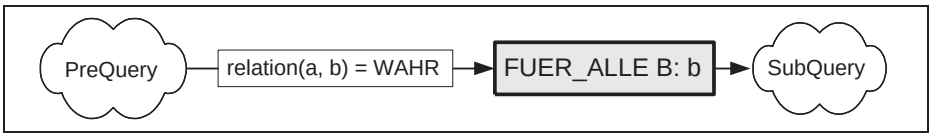

Abbildung 8.4: FuerAlle Klassenknoten überprüfen alle eintreffenden Instanzen der Klasse B hinsichtlich der in der *SubQuery* vorgegebenen Eigenschaften.

8.2.2.4 Der Selektionstestknoten

Selektionstestknoten dienen der Auswahl von Instanzen einer vorgegebenen Klasse anhand bestimmter Eigenschaften wie den Werten oder Wertebereichen einzelner Attribute. Dazu werden die über die Eingangskante eintreffenden Objekte mit den Eigenschaften des Selektionstests verglichen und bei erfolgreicher Überprüfung an den Vorgängerknoten zurückgegeben. Selektionstestknoten besitzen genau eine Eingangskante und keine Ausgangskante, da sie atomare Eigenschaften validieren. Als logischer Ausdruck stellt sich der Selektionstestknoten aus Abbildung 8.5 dar als : $a. < Eigenschaft > op < Wert >$.

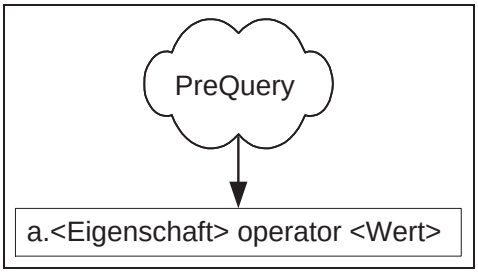

Abbildung 8.5: *Selektionstestknoten überprüfen alle eintreffenden Instanzen der* PreQuery hinsichtlich atomarer Eigenschaften.

8.2.2.5 Der Operatorknoten

Operatorknoten im Anfragebaum entsprechen den logischen Operatoren der boolschen Algebra. Durch die Verfügbarkeit der Operatoren *UND*, *ODER*, *NICHT UND* und *NICHT ODER* handelt es sich bei der beschriebenen Anfragesprache um eine vollständige Algebra. Operatorknoten haben immer eine Eingangskante, die von einer Voranfrage (*PreQuery*) kommt, die über eine logische Verknüpfung zweier Teilanfragen (*SubQuery*) (zu denen die zwei Ausgangskanten führen) weiter eingeschränkt werden soll. Operatorknoten wie in Abbildung 8.6, implizieren die Interpretation: *PreQuery: SubQuery1 Operator SubQuery2* .

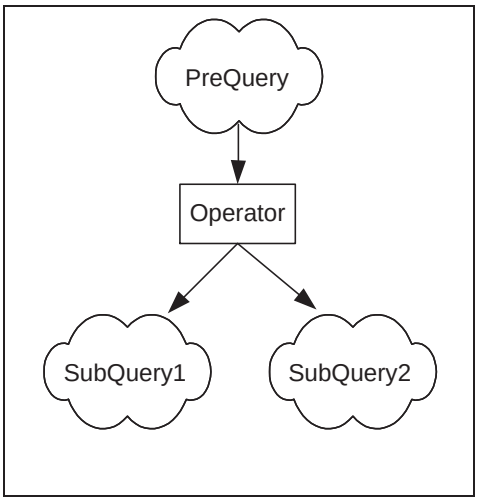

Abbildung 8.6: *Operatorknoten repräsentieren die logischen Operatoren, mit denen jeweils zwei Teilanfragen verknüpft werden.*

8.2.3 Berechnung von Arbeitsbereichen aus Zugriffsrechten und Themenbereichen

In Abschnitt 8.1.2 wurden zwei einfache Methoden skizziert, mit denen Daten- und Schemainformationen von Benutzergruppen und Themen synchronisiert werden können, um sie dann in einem Arbeitsbereich dem angemeldeten Benutzer zur Verfügung zu stellen. Das eine Verfahren sah vor, Benutzerrechte und Themenbereiche getrennt voneinander auszuwerten und die Ergebnisse zu verschneiden. In dem zweiten Verfahren sollten erst alle Daten im Sinne der Benutzerrechte ausgewertet werden, um damit eine neue Datenbasis für die Auswertung von Themenbereichen zu erhalten.
In diesem Abschnitt soll nun ein Verfahren präsentiert werden, bei dem nicht die Inhalte, sondern die Beschreibungen von Benutzerrechten und Themenbereichen verschnitten werden. Die Vorteile eines solchen Ansatzes sind darin zu sehen, dass die Verschneidung der Beschreibungen zu einer vollständigen Definition des Arbeitsbereichs führt, ohne dabei auf große Datenmengen zugreifen zu müssen. Die endgültige Beschreibung des Arbeitsbereichs muss nur einmal abgeleitet werden und ist unabhängig von den aktuellen Inhalten der Datenbasis. Darüber hinaus wird eine durchgängige Modellbeschreibung generiert, die sich auf den aktuellen Benutzer im Kontext eines Themas bezieht und anhand geeigneter Regeln auf Konsistenz geprüft werden kann.
Wählt ein Benutzer mit vorgegebenen Zugriffsrechten ein fachlich motiviertes Thema aus, so muss zunächst das Schema ermittelt werden, dass ihm innerhalb eines Arbeitsbereichs zugänglich ist. Dabei ist entscheidend, welche Klassen

verfügbar gemacht werden. Die Beziehungen, die im Ausgangsdatenschema zwischen den Klassen definiert sind, werden dann in Abhängigkeit von den sichtbaren Klassen des Arbeitsbereichs freigegeben.
Wird die Vererbung von Klassenbeschreibungen im Sinne einer Halbordnung mit $base(A, B) \Longrightarrow A \leq B$ interpretiert, so gilt für jede Klasse A innerhalb eines Arbeitsbereichs WS: alle Klassen B mit $B \leq A$ gehören zu dem ausgewählten Thema und liegen innerhalb der Zugriffsrechte des Benutzers. Die beschriebene Forderung ergibt sich aus der Interpretation eines Arbeitsbereichs als Beschreibung eines Teilmodells, dass sowohl von den Rechten des angemeldeten Benutzers, als auch von dem ausgewählten Themenbereich abgedeckt wird. Im einzelnen bedeutet dies:

1. Liegt eine Klasse A in den Zugriffsrechten des angemeldeten Benutzers und gehört A gleichzeitig zum Schema des ausgewählten Themas, so gehört A auch zu dem Schema des zu generierenden Arbeitsbereich.

2. Liegt eine Klasse A in den Zugriffsrechten des angemeldeten Benutzers und enthält das ausgewählte Thema gleichzeitig eine Klasse B, für die

   (a) $base(A, B)$ bzw. $A \leq B$ gilt, so gehört A zum Schema des zu generierenden Arbeitsbereich.

   (b) $base(B, A)$ bzw. $B \leq A$ gilt, so gehört B zum Schema des zu generierenden Arbeitsbereich.

3. Gibt es eine Klasse C, so dass eine Klasse A mit $C \leq A$ in den Zugriffsrechten des angemeldeten Benutzers liegt und gleichzeitig eine Klasse B mit $C \leq B$ in dem ausgewählten Thema beschrieben ist, so enthält das Schema des zu generierenden Arbeitsbereichs die gemeinsame Basisklasse A.

Das so erzeugte Schema eines Arbeitsbereichs entspricht den objektorientierten Paradigmen der Klassenmodellierung im Kontext von Zugriffsregel 4 aus Abschnitt 5.2: Mit der Sicht auf eine Klassenbeschreibung sind auch die Beschreibungen aller Basisklassen dieser Klasse lesbar. Somit werden bei der Verschneidung der beiden Ausgangsschemata alle Vererbungsschritte nachvollzogen, die in beiden Schemata modelliert sind.
Auf der Grundlage eines Fachdatenschemas können Fachobjekt-Instanzen formal beschrieben werden, die zum Einen innerhalb der Zugriffsrechte eines Benutzers (bzw. einer Benutzergruppe) liegen und zum Anderen der Modellierung eines fachlich motivierten Themas angehören. Dabei können nur solche Instanzmengen beschrieben werden, die eine Ausprägung einer Fachobjektklasse des verfügbaren Schemas darstellen. Als Folge gilt dieser Grundsatz in gleicher

Weise für alle Instanzen, die innerhalb eines aktuellen *Workspaces* einem Benutzer präsentiert werden. Das Schema des aktuellen Arbeitsbereichs definiert somit den Semantikraum für den Zugriff auf Instanzen durch einen Benutzer innerhalb des gewählten Themas. Der formalen Beschreibung von Instanzmengen des Arbeitsbereichs dient eine deskriptive Anfragesprache, wie sie in Abschnitt 8.2 dargestellt wurde. Die entprechenden Anfragen zur Sichtendefinition des Arbeitsbereichs sollen aus den Beschreibungen der Zugriffsrechte des angemeldeten Benutzers und den Beschreibungen der Instanzen des ausgewählten Themas abgeleitet werden. Dabei definiert der Arbeitsbereich die Sicht des Gesamt-Datenmodells, die einem Benutzer alle Informationen des gewählten Themenbereichs zur Verfügung stellt, die durch seine Zugriffsrechte abgedeckt sind. Sämtliche Instanzen einer Fachobjektklasse des Arbeitsbereichs, die damit einer bestimmten Beschreibung genügen, müssen gleichzeitig durch eine Beschreibung der Zugriffsrechte des Benutzers für Instanzen der gleichen Klasse erfasst werden. Bei der hier verwendeten Anfragelogik wird die Klasse, deren Instanzen durch eine Anfrage beschrieben werden, vom Startknoten dieser Anfrage bestimmt. Bei der durchgängigen Beschreibung von Fachobjektinstanzen eines Themenbereichs, die gleichzeitig innerhalb der Zugriffsrechte eines Benutzers liegen, besteht die Aufgabe also darin, Anfragen mit Startknoten gleicher Klassen unterhalb eines gemeinsamen Startknotens logisch zu verknüpfen. Die Zusammenfassung von sichtendefinierenden Anfragen erfolgt auf Basis einer Unterscheidung zwischen Anfragekopf und Anfragerumpf. Der *SucheAlle*-Knoten einer Anfrage repräsentiert hierbei den Anfragekopf. Dieser definiert die Klassenzugehörigkeit der Anfrageeregebnisse. Der Anfragerumpf wird durch den Rest der Anfrage definiert, die vom Startknoten aus über einen Pfad erreichbar ist. Der Anfragerumpf definiert die einschränkenden Selektionskriterien, die von den Instanzen der Ergebnisklasse erfüllt sein müssen, um zur Ergebnismenge der Anfrage zu gehören. Anfragen, die nur durch den Startknoten definiert sind, sogenannte $*-$ Anfragen, haben demzufolge einen leeren Rumpf, der auch als Rumpf mit einem einzigen Knoten mit dem statischen Wert *WAHR* interpretiert werden kann und insofern für alle Instanzen der Ergebnisklasse erfüllt ist. Der Anfragerumpf ist ein logischer Ausdruck, der für jede Instanz der Startknotenklasse ausgewertet wird. Eine Instanz der Anfrageklasse gehört zur Ergebnismenge, wenn die Auswertung des Anfragerumpfes für diese Instanz das Ergebnis *WAHR* liefert. Der Anfragerumpf ist rekursiv aus Unteranfragen aufgebaut, für die jeweils ein Klassenknoten (*EsGibt* oder *FuerAlle*) einen (Unter-) Startknoten bildet. Die Unter- oder Teilanfragen stehen jeweils in einer durch Kanten definierten Beziehung zu Ihrem übergeordneten Klassenknoten und werden untereinander durch Operatorknoten verknüpft. Die Rekursionen enden an den Blattknoten des Anfragebaums, die entweder Klassenknoten oder Selektionstestknoten sind und atomare Anfragen darstellen, die für das aktuell auszuwertende Objekt

unmittelbar und eindeutig beantwortet werden können.
Nach den beschriebenen Eigenschaften von Anfragebäumen, erfolgt die Zusammenfassung von Anfragen in zwei Schritten:

1. Verknüpfung der Anfragerümpfe unterhalb des Startknotens durch einen Operatorknoten und
2. Aufdeckung und Entfernung von Redundanzen und logischen Widersprüchen, die im Zuge der Zusammenfassung entstanden sein können.

Das Prinzip der Verknüpfung von Instanzbeschreibungen kann in zwei Punkten zusammengefasst werden:

1. Die Anfragerümpfe ergänzender Beschreibungen von Instanzen der gleichen Klasse, die alle einer Sicht angehören sollen, werden als Disjunktion mit einem *ODER*- Knoten verbunden.
2. Die Anfragerümpfe mit Beschreibungen, die von allen Instanzen der Startknotenklasse erfüllt sein sollen, werden als Konjunktion mit *UND*- Knoten verbunden.

Für die Instanzen eines Arbeitsbereichs ergibt sich daraus eine einfache logische Vorschrift mit der festgelegt wird, dass jede Instanz, die einem Anfrageergebnis einer Themenbeschreibung angehört, mindestens eine Anfrage der Definition der Zugriffsrechte erfüllen muss. Formal findet das Prinzip der Zusammenfassung die Darstellung:
Seien $qk_1^p, ...qk_n^p$ die Anfragerümpfe zur Beschreibnung der Instanzen einer Klasse k eines Themas t. $qk_1^u, ...qk_m^u$ seinen die Anfragerümpfe zur Festlegung der Zugriffsrechte einer Benutzersgruppe u auf Instanzen der Klasse k. Wählt nun ein Benutzer der Benutzergruppe u das Thema t aus, so müssen die Instanzen der Klasse k des Arbeitsbereichs die Bedingung $(qk_1^p \vee ... \vee qk_n^p) \wedge (qk_1^u \vee ... \vee qk_m^u)$ erfüllen.

Im einfachsten Fall werden für Zugriffsrechte und Themenbereiche nur Instanzen von Klassen der gleichen Vererbungsebene beschrieben. Somit ist eine direkte Zusammenfassung der Anfragerümpfe im Arbeitsbereich entsprechend der genannten Vorschrift möglich (Abbildungen 8.7 - 8.9).

Im ersten Schritt der Zusammenfassung (Abbildung 8.8) werden innerhalb der Themenbeschreibung und der Rechtebeschreibung jeweils die Anfragen mit gleichen Startknoten ausgewählt. Diese definieren die Instanzen einer Klasse, die im Kontext einer Beschreibung zugreifbar sein sollen. Die Rümpfe der

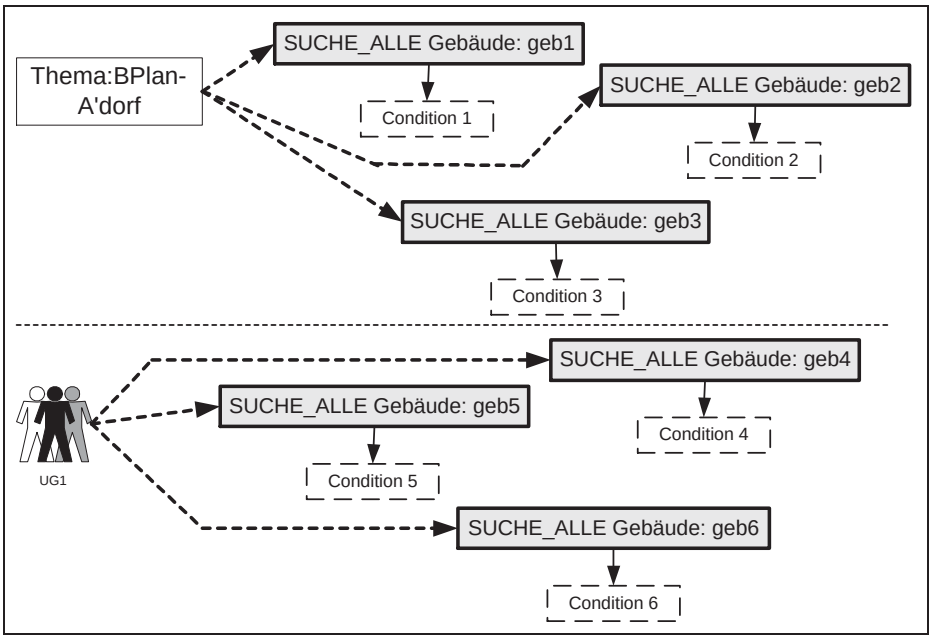

Abbildung 8.7: *Beschreibung eines Themenbereichs und der Zugriffsrechte einer Benutzergruppe bezüglich der Instanzen einer Klasse.*

gefundenen Anfragen werden unterhalb eines gemeinsamen Startknotens mit binären *ODER*-Operatoren verknüpft.

Im nächsten Schritt der Zusammenfassung (Abbildung 8.9) werden nun die Rümpfe der Anfragen des Themenbereichs mit denen der Zugriffsrechte für gleiche Startknotenklassen durch einen binären *UND*-Operator verknüpft. Als Ergebnis entsteht eine vollständige Beschreibung der Instanzen einer Klasse, die im aktuellen Arbeitsbereich verfügbar sein sollen.

Werden im Berechtigungs- oder Themenbereich Anfragen verwendet, die Instanzen von Klassen beschreiben, die zwar nicht identisch sind aber in einer Vererbungsbeziehung zueinander stehen (Abbildung 8.10), so ist vor der Zusammenfassung der Anfragen zunächst zu überprüfen, welche Anfragen zusammengefasst werden können, ohne dabei Zugriffsrechte zu überschreiten. Dabei wird die Vorgehensweise von der fachlichen Zielsetzung der Zusammenfassung geprägt: Innerhalb des gewählten Themenbereichs ist festgelegt, welche Informationen (Fachobjektklassen und zugehörige Fachobjekte) eines Datenbestandes von einem Bearbeiter behandelt werden können, wenn er über ausreichende Zugriffsrechte verfügt. Diese Informationen des Themenbereichs müssen also vor dem Zugriff durch einen Benutzer hinsichtlich seiner Zugriffsrechte gefiltert werden. Insofern werden die Startknoten den Beschreibungen des Themas ent-

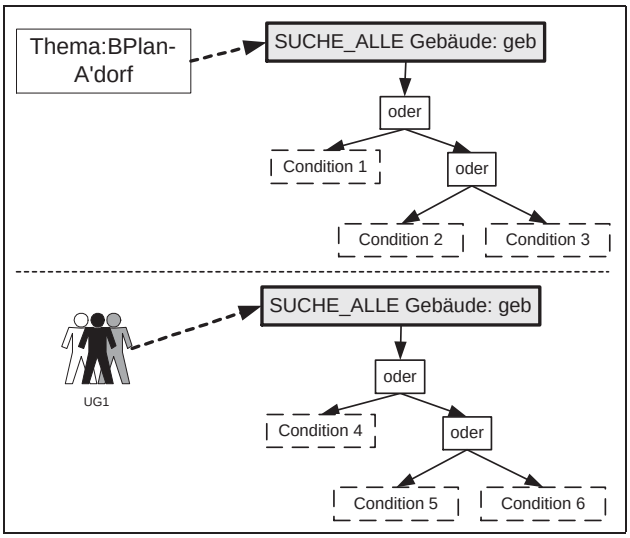

Abbildung 8.8: *Zusammenfassung von Anfragerümpfen ergänzender Teilanfragen gleicher Startknoten mit* ODER-Operatoren.

nommen. Dabei finden nur solche Startknoten Berücksichtigung, deren Fachobjektklassen innerhalb des Benutzerschemas liegen. Die Zusammenfassung von Anfragerümpfen unterhalb eines gemeinsamen Startknotens erfolgt innerhalb des Themenbereichs durch disjunktive Verknüpfungen. Mit den Klassen der Startknoten des Themenbereichs wird festgelegt, welche Informationen und Erscheinungsformen der beschriebenen Objekte für den gewählten Themenbereich relevant sind. Daher werden jeweils nur Instanzenbeschreibungen zusammengefasst, die sich auf die gleiche Klasse der gleichen Vererbungsebene beziehen. Die resultierenden Anfragen der Themenbeschreibung werden nun nacheinander konjunktiv mit den disjunkt verknüpften Beschreibungen der Zugriffsrechte zusammengefasst. Dazu ist zunächst zu jeder Themenanfrage ein entsprechender Startknoten der gleichen Klasse im Bereich der Zugriffsrechte auszuwählen. Falls nur Anfragen für Basisklassen oder abgeleitete Klassen der Startknotenklasse existieren, muss ein neuer Startknoten definiert werden. Unterhalb dieses Startknotens müssen die Beschreibungen der Zugriffsrechte zusammengefasst werden, deren Ergebnis Instanzen der im Startknoten angegebenen Klasse sind. Die neu generierte Anfrage dient nun der Verknüpfung von Zugriffsrechten und Themenbeschreibungen mittels eines *UND*-Knotens. Bei der disjunktiven Verknüpfung von Anfragerümpfen unterhalb eines einheitlichen Startknotens im Bereich der Zugriffsrechte müssen drei Gruppen von Anfragen berücksichtigt werden:

1. Anfragen, die sich direkt auf die Klasse des angegebenen Startknotens

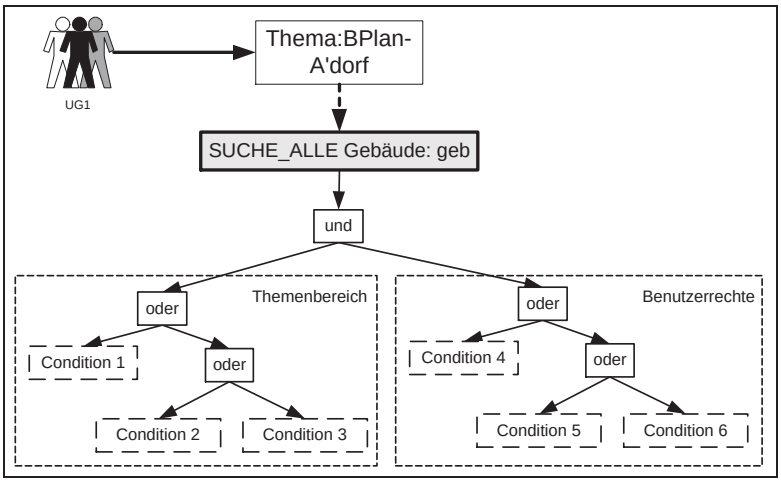

Abbildung 8.9: *Zusammenfassung von Anfragerümpfen zu verschneidender Teilanfragen gleicher Startknoten mit* UND-Operatoren.

beziehen,

2. Anfragen, die sich auf eine Basisklasse der Klasse des angegebenen Startknotens beziehen und

3. Anfragen, die sich auf eine abgeleitete Klasse der Klasse des angegebenen Startknotens beziehen.

Da die Sicht eines Benutzers den Semantikraum, in Form des zugreifbaren Datenschemas, angewendet auf die zugreifbaren Objekte repräsentiert, gehört ein Fachobjekt eines Themas dem aktuellen Arbeitsbereich an, wenn der angemeldete Benutzer für das Objekt Zugriffsrechte besitzt und die beschriebene Klasse innerhalb seines Sprachraums liegt. Für Objektbeschreibungen, die sich auf die gleiche Fachobjektklasse beziehen, ergibt sich unmittelbar eine Verschmelzung in der bereits beschriebenen Weise. Bezieht sich eine Beschreibung von Zugriffsrechten für Fachobjekte auf eine Basisklasse der vorgegebenen Startknoten-Klasse, so wird damit eine allgemeingültigere Beschreibung der zugreifbaren Objekte formuliert, als dies bei der Instanzenbeschreibung des Themenbereichs gegeben ist (Abbildung 8.11). Die Objekteigenschaften, auf die innerhalb der allgemeingültigeren Beschreibung zurückgegriffen wird, sind ebenso über die Methoden und Attribute aller abgeleiteten Klassen zugreifbar, da die öffentlichen und damit sichtbaren Methoden und Attribute zwischen Klassen vererbt werden. Insofern kann die Klasse des Startknotens einer Beschreibung von Zugriffsrechten durch eine geforderte, spezialisiertere Klasse des Themenbereichs ersetzt werden, wenn der Benutzer grundsätzlich das Recht besitzt, auf diese abgeleitete Klasse zuzugreifen. Bezieht sich hingegen die Beschreibung der

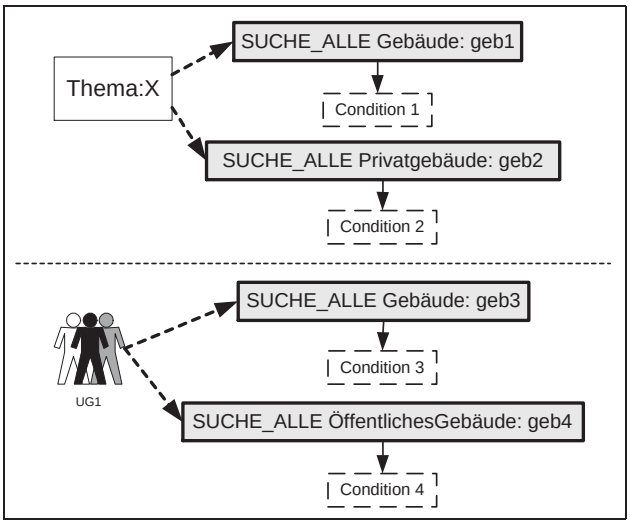

Abbildung 8.10: *Zugriff auf Instanzen von Klassen unterschiedlicher Vererbungsebenen.*

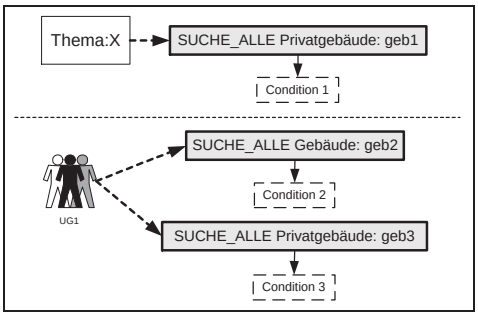

Abbildung 8.11: *Beschreibungen von Klassen-Instanzen und Basisklassen-Instanzen.*

Zugriffsrechte eines Benutzers auf eine abgeleitete Klasse hinsichtlich einer Beschreibung von Instanzen des Themenbereichs, so besteht die Möglichkeit, dass Instanzen des Themenbereichs durch die Zugriffsrechte abgedeckt sind, diese aber aus spezialisierteren Eigenschaften der Objekte hervorgehen (Abbildung 8.13), die innerhalb des Themenbereichs nicht zugänglich sind. Daher müssen die sichtbaren Instanzen des Arbeitsbereichs, die den Klassen des Themenbereichs entnommen sind, gleichzeitig Instanzen der abgeleiteten Klasse sein und deren Methoden unterstützen. Bevor also eine Beschreibung von Zugriffsrechten auf die Instanzen einer Basisklasse der Startknoten-Klasse angewendet werden kann, müssen die jeweiligen Anfragen um geeignete (Selektionstest-)Knoten erweitert werden, mit deren Hilfe die Typsicherheit der erhaltenen Objekte gewährleistet werden kann. Mit der Einführung einer Methode *isA*

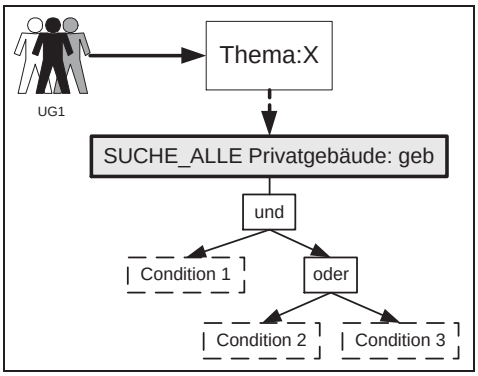

Abbildung 8.12: *Zusammenfassung der Beschreibungen von Klassen-Instanzen und Basisklassen-Instanzen.*

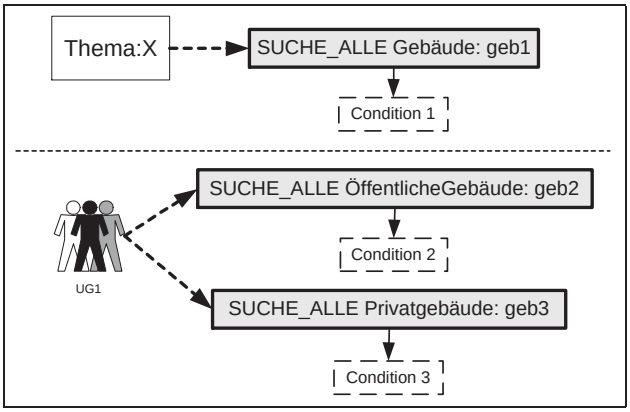

Abbildung 8.13: *Beschreibungen von Klassen-Instanzen und Instanzen abgeleiteter Klassen.*

wird ein abstraktes Werkzeug beschrieben, mit dem die Klassenzugehörigkeit einer Instanz abgefragt werden kann. Die Implementierung einer solchen Methode wird von gängigen objektorientierten Programmiersprachen unterstützt und kann daher als realisierbar angenommen werden. Eine Überprüfung der jeweiligen Instanzen auf ihre Zugehörigkeit zu der geforderten abgeleiteten Klasse kann somit als zusätzliche Bedingung (Selektionskriterium) formuliert und über eine Konjunktion mit den übrigen Kriterien für diese Klasse verknüpft werden (Abbildung 8.14). Zum Zeitpunkt der Auswertung einer Anfrage muss also die Möglichkeit bestehen, intern auf die Eigenschaften abgeleiteter Klassen zuzugreifen, wenn für eine Instanz die entsprechende *isA*-Bedingung erfüllt ist. Auf der Grundlage der zusammengefassten Beschreibungen von Instanzen bestimmter Themenbereiche, eingeschränkt auf die Zugriffsrechte des jeweiligen Benutzers an den Instanzen dieser Klassen, ergibt sich im allgemeinen der Bedarf,

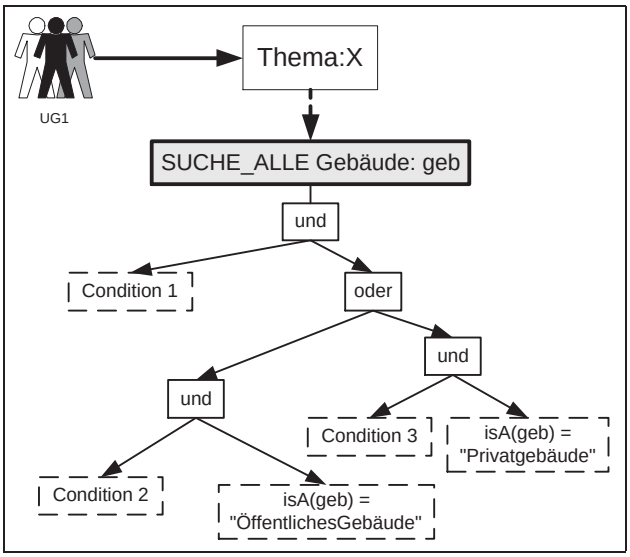

Abbildung 8.14: *Zusammenfassung der Beschreibungen von Klassen-Instanzen und Instanzen abgeleiteter Klassen.*

diese deskriptiven Objektfilter hinsichtlich ihrer Erfüllbarkeit und Redundanzfreiheit weiter zu Untersuchen. Eingedenk der Tatsache, dass Anfragerümpfe geschlossene logische Klauseln repräsentieren, eröffnet sich die Möglichkeit, die Bedingungen für die Instanzen eines Arbeitsbereichs in konjunktiven oder disjunktiven Normalformen darzustellen und darauf die Gesetze der boolschen Algebra (Tabelle 8.1) anzuwenden. Durch die Anwendung dieser Gesetze lassen sich die durch Verschmelzung mehrerer Teilanfragen komplex gewordenen Anfragebäume in vielen Fällen erheblich vereinfachen. Insbesondere führt die Anwendung der Idempotenz-, Komplement- und Absorptionsgesetze häufig zum Wegfall ganzer Teilbäume. Dadurch ergeben sich neben übersichtlicheren Anfragen auch die angestrebte Reduzierung wiederholter Zugriffe auf die gleichen Objekte oder Objektmengen und die Aufdeckung widersprüchlicher und redundanter Anforderungen an die Eigenschaften der Objekte bereits vor der algorithmisch teuren Ausführung der Anfragen. Typische Anfragestrukturen, die dies veranschaulichen, ergeben sich, wenn:

1. Zwei Teilbäume unterhalb eines Operatorknotens implizierende Bedingungen beschreiben, so dass aus der Gültigkeit des einen Teilbaums die Gültigkeit des anderen folgt. Bei *UND*-Verknüpfungen fällt dann der Teilbaum mit der schwächeren (ableitbaren) Bedingung weg, während bei *ODER*-Verknüpfungen nur der Teilbaum mit der schwächeren Bedingung erhalten bleibt (Abbildung 8.15).

Kommutativgesetze	$a \wedge b = b \wedge a$ $a \vee b = b \vee a$
Assoziativgesetze	$a \wedge (b \wedge c) = (a \wedge b) \wedge c$ $a \vee (b \vee c) = (a \vee b) \vee c$
Distributivgesetze	$a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$ $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$
Idempotenzgesetze	$a \wedge a = a$ $a \vee a = a$
Komplementgesetze	$a \wedge \neg a = 0$ $a \vee \neg a = 1$
Neutrale Elemente	$a \wedge 1 = a$ $a \vee 1 = 1$ $a \wedge 0 = 0$ $a \vee 0 = a$
Absorptionsgesetze	$a \wedge (a \vee b) = a$ $a \vee (a \wedge b) = a$ $a \wedge b) \vee (a \wedge \neg b) = a$ $a \vee b) \wedge (a \vee \neg b) = a$
De Morgansche Regeln	$\neg(a \wedge b) = \neg a \vee negb$ $\neg(a \vee b) = \neg a \wedge negb$

Tabelle 8.1: *Gesetze der boolschen Algebra.*

2. Zwei Teilbäume unterhalb eines Operatorknotens unvereinbare bzw. komplementäre Bedingungen beschreiben, so dass nicht gleichzeitig die Bedingungen beider Teilbäume erfüllt sein können bzw. aus der Gültigkeit des einen Teilbaums die Ungültigkeit des anderen Teilbaums folgt. Bei einer *UND*-Verknüpfung derartiger Teilbäume ergibt sich für den übergeordneten Knoten grundsätzlich eine leere Menge. Im Fall einer *ODER*-Verknüpfung der Teilbäume ergibt sich die Ergebnismenge des übergeordneten Knotens aus der Vereinigung der Objekte, die entweder die Bedingungen des linken oder die des rechten Teilbaums erfüllen. Für komplementäre Bedingungen bedeutet dies, dass beide Teilbäume wegfallen (Abbildung 8.16).

Nachdem für einen Benutzer als Repräsentanten einer Benutzerklasse nach den Verfahren dieses Abschnitts die Sicht auf einen Arbeitsbereich entsprechend seinen Zugriffsrechten und den fachlichen Aspekten eines Themenbereichs ermittelt wurde, steht dem Benutzer ein (Teil-)Modell der Datenbasis zu Zwecken der weiteren Auswertung und Bearbeitung zur Verfügung. Eine Möglichkeit die sich dem Benutzer bietet, ist zum Beispiel die Definition eigener Anfragen als Werkzeug zur Auswertung und Analyse der verfügbaren Daten. Für diese Definition von Benutzeranfragen verfügt der Benutzer formal über den vollen Sprachumfang der verwendeten Anfragesprache. Er kann dabei jedoch nur

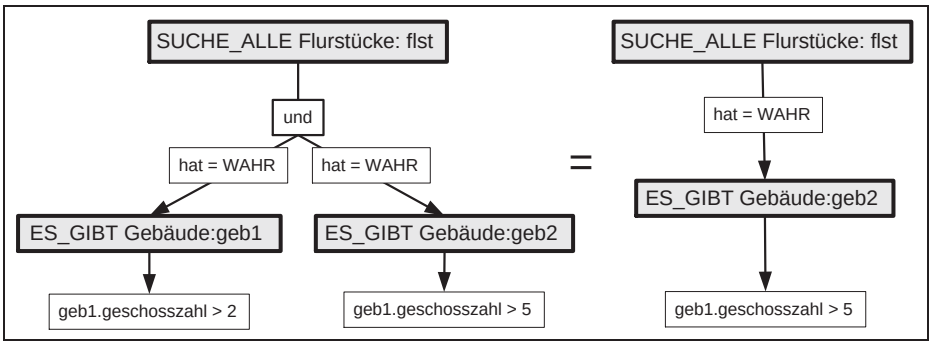

Abbildung 8.15: *Beispiel für den Wegfall von Teilbäumen, wenn die Bedingungen des einen Teilbaums die Bedingungen des anderen Teilbaums implizieren.*

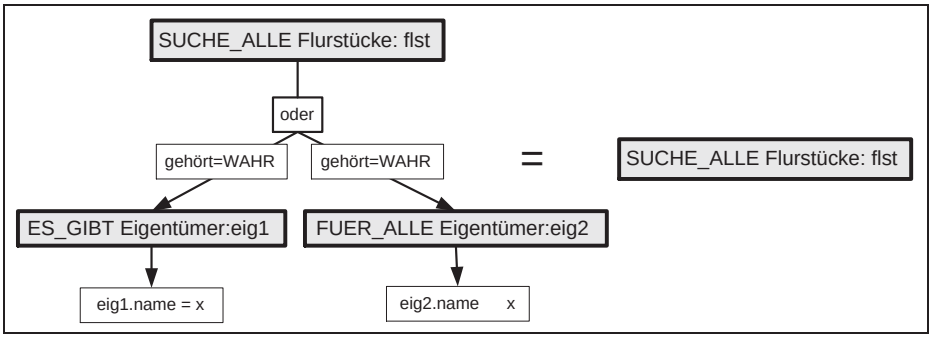

Abbildung 8.16: *Beispiel für den Wegfall von Teilbäumen, bei unvereinbaren Bedingungen in den Teilbäumen.*

die Klassen, Beziehungen und Vererbungen verwenden, die innerhalb seines Arbeitsbereichs bekannt sind - also zum Semantikraum des aktuellen *Workspaces*' gehören. Wird eine Anfrage durch einen Benutzer (oder einen *Client*-Prozess) ausgeführt, so darf sich die Auswertung ausschliesslich auf die Daten des aktuellen Arbeitsbereichs beziehen. Vor dem Hintergrund dieser Anforderung bieten sich grundsätzlich zwei Möglichkeiten der Realisierung an:

1. Die formulierten Benutzeranfragen werden entsprechend ihrem Startknoten analog zu dem beschriebenen Verfahren mit den Beschreibungen der Instanzen des Arbeitsbereichs konjunktiv verknüpft und bei der Ausführung durch die verknüpfte Anfrage ersetzt.

2. Eine geeignete Systemarchitektur sorgt dafür, dass die Daten des Arbeitsbereichs ausschließlich über einen logisch und physikalisch getrennten Bereich zugreifbar sind (z.B. durch Objektreferenzen oder -zeiger innerhalb des Arbeitsbereichs).

Die Zugriffsbeschränkung mit den Mitteln einer Systemarchitektur bietet dabei erhebliche Vorteile: Der Zugriff auf Daten durch Benutzer erfolgt grundsätzlich auf den thematisch und rechtlich beschränkten Datenbereich. Die Daten des Arbeitsbereichs müssen nur einmal zum Zeitpunkt der Anmeldung des Benutzers berechnet werden. Zudem kann die Ausführung von Anfragen durch einen Benutzer deutlich performanter erfolgen, da nur die Daten berücksichtigt werden, die für den Benutzer innerhalb des aktuellen Themenbereichs zugreifbar und relevant sind. Die Ergebnismengen von Anfragen eines Benutzers müssen nicht hinsichtlich der Zugriffsrechte des Benutzers gefiltert werden.

8.2.4 Anfragebasierte Benutzerprofile für *Web-Services*

Die beschriebene Vorgehensweise zur Ermittlung und Auswertung von Arbeitsbereichen unter Berücksichtigung definierter Benutzerrechte, eignet sich in besonderer Weise für die Realisierung rechtebeschränkter, OGC-konformer *Web-Services*, wie zum Beispiel einem *Web Feature Server*. Benutzer beziehen dabei ihre fachlich relevanten Anwendungsdaten über ein einheitliches Portal, das beispielsweise über das Internet zugänglich ist. Zur gezielten Anforderung von Daten des Servers bedienen sich die Benutzer einer XML-basierten, objektorientierten Anfragesprache: Für OGC-konforme *Web Feature Server* handelt es sich dabei um das sogenannte *OGC Filter Encoding* - einer Abwandlung der von der W3C (*World Wide Web Consortium*) entworfenen *XML-Query*. In der abgewandelten Form der Architektur eines *Web Feature Server*s sendet dieser eintreffende Anfragen authentisierter Benutzer an einen *UserService* weiter. Dieser hat Zugriff auf serverseitig[1] hinterlegte Benutzerprofile. Die Benutzerprofile enthalten sowohl die für den Benutzer sichtbaren Schemainformationen in Form einer XSD-konformen Beschreibung, als auch eine Menge XML-basierter Anfragen zur Definition der für den Benutzer zugreifbaren Instanzen der sichtbaren Schemaklassen. Die für eine eintreffende Benutzeranfrage relevanten Teile des entsprechenden Benutzerprofils werden vom *UserService* ausgewählt und zusammen mit der Benutzeranfrage an einen *QueryProcessor* weitergeleitet. Dessen Aufgabe ist es, die Benutzeranfrage unter Beachtung des hinterlegten Benutzerprofils auszuwerten, die durch entsprechende Benutzerrechte abgedeckten Ergebnis-Objekte aus der Datenquelle zu ermitteln und diese der Kontrolle des *Web Feature Server*s zu übergeben. Die Antwort-Objekte können nun als GML-kodierte Struktur an den Benutzer geschickt werden.

[1] Der Server des *UserServices* muss nicht der gleiche physikalische Server, wie der des *Web-Services* sein

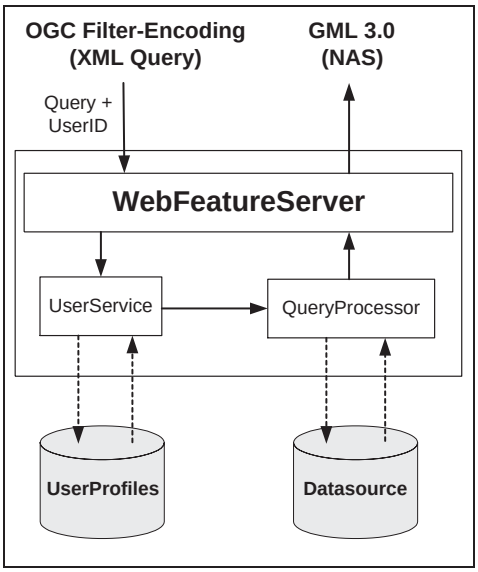

Abbildung 8.17: *Modifizierte Architektur eines* Web Feature Servers*, ergänzt um Komponenten zur Auswertung von Benutzerrechten.*

Da auch hier die Syntax und Semantik der verwendeten Anfragesprache mit der zur Beschreibung von Benutzerprofilen eingesetzten Anfragesprache übereinstimmt, kann die Auswertung von Benutzerrechten bei Zugriffen auf den *Web Feature Server* in Analogie zu den Ausführungen in Abschnitt 8.2.3 erfolgen. Auf eine zusätzliche Filterkomponente, die jedes potenzielle Ergebnis-Objekt einer Anfrage mit den Zugriffsrechten des anfragenden Benutzers vergleicht, kann verzichtet werden.

8.3 Zusammenfassung

Arbeitsbereiche definieren thematische und räumliche Organisationsstrukturen für GIS-relevante Datenquellen. Diese erleichtern und reglementieren den gezielten Zugang zu Informationen vor dem Hintergrund einer fachlichen Aufgabenstellung. Insofern können Arbeitsbereiche zunächst unabhängig von vorhandenen Benutzerprofilen definiert werden.
Soll nun ein bestimmter Benutzer bzw. ein Mitglieder einer Benutzergruppe fachliche Aufgaben innerhalb eines Arbeitsbereichs wahrnehmen, so müssen die zur Verfügung gestellten Informationen und die bedienbaren Methoden hinsichtlich der erteilten Zugriffsrechte des Benutzers überprüft werden: Ein Benutzer soll innerhalb eines gewählten Arbeitsbereichs nur die Transaktionen ausführen können, die durch seine Zugriffsrechte abgedeckt sind.

Die Untersuchungen in diesem Kapitel haben gezeigt, dass sowohl Benutzerprofile als auch Arbeitsbereiche auf Grundlage der gleichen Syntax definiert und beschrieben werden können. Daher war es naheliegend, für die Ermittlung von Arbeitssichten einen Ablauf zu definieren, in dem zunächst das Arbeitsschema ermittelt wird, das im Weiteren dann als Basis für die Verschneidung der Instanzmengen dient. Das Arbeitsschema sollte im Idealfall dem Schema des Arbeitsbereichs entsprechen, da nur so sichergestellt werden kann, dass die semantischen Voraussetzungen zur Erfüllung der gestellten Aufgaben gegeben sind. Die Verschneidung der Instanzmengen folgt in dem dargestellten Konzept einem deklarativen Ansatz. Dabei werden nicht die Mengen selbst, sondern die beschreibenden Anfragen miteinander verschnitten und ausgewertet.
In diesem Kapitel wurde damit eine Strategie für die Zusammenführung und gegenseitige Ergänzung von thematischen und rechtlichen Organisationsstrukturen in GIS eingeführt.

Kapitel 9

Architekturentwurf und Implementierung

Die in den bisherigen Kapiteln dargestellten Eigenschaften objektorientierter, raumbezogener Datenmodelle sowie die entwickelten Konzepte zur Modellierung geeigneter Zugriffs- und Berechtigungsstrategien für solche Daten, bilden die Grundlage für den Entwurf einer Zugriffsarchitektur. Diese stellt eine vereinfachte und abstrahierte Sicht eines Implementierungsansatzes dar. Das in den folgenden Abschnitten dargestellte Klassen- und Komponentendesign wurde im Rahmen der Weiterentwicklung von SupportGIS realisiert sowie an Realdaten und -szenarien getestet.

9.1 Die Modellierung von Zugriffsrechten

Eine wesentliche Eigenschaft der in den vergangenen Abschnitten entworfenen Strategie zur Beschreibung und Auswertung von Zugriffsrechten für Geodaten war die Integration der Objektorientiertheit von Daten und Benutzern in die Rechtevergabe. Dabei sollten Daten und Benutzer eines Systems in gleicher Weise Repräsentanten fachlicher Klassen darstellen, die über Methoden gesteuert werden und über Benachrichtigungsmechanismen Informationen austauschen. Die Position einer Klasse innerhalb einer Vererbungshierarchie war dabei das charakterisierende Merkmal für den Besitz von Privilegien bzw. die Sicherheitsrelevanz ihrer Instanzen. Mit den für einen Benutzer sichtbaren Klassen und deren Beziehungen wird ein Benutzerschema definiert, das einen Sprach- und Semantikraum für den Benutzer darstellt. Eingeschränkt auf diesen Semantikraum wird der Zugriff auf einzelne Instanzen der sichtbaren Klassen durch beschreibende Merkmale der betroffenen Objekte definiert. Diese

Merkmale können räumliche Ausschnitte festlegen, bestimmte Attributwerte verlangen oder komplexe, kontextabhängige Objekteigenschaften formulieren. Benutzer und Benutzergruppen sowie deren Semantikräume und Zugriffsrechte für Objekte müssen in einem geeigneten Systementwurf beschreibbar und sowohl effizient, als auch zuverlässig (im Sinne der Sicherheit) auswertbar sein. Die zentralen Ideen und Ansätze eines solchen Entwurfes sind in Abbildung 9.1 dargestellt. In den folgenden Abschnitten werden die einzelnen Komponenten des Entwurfs näher erläutert.

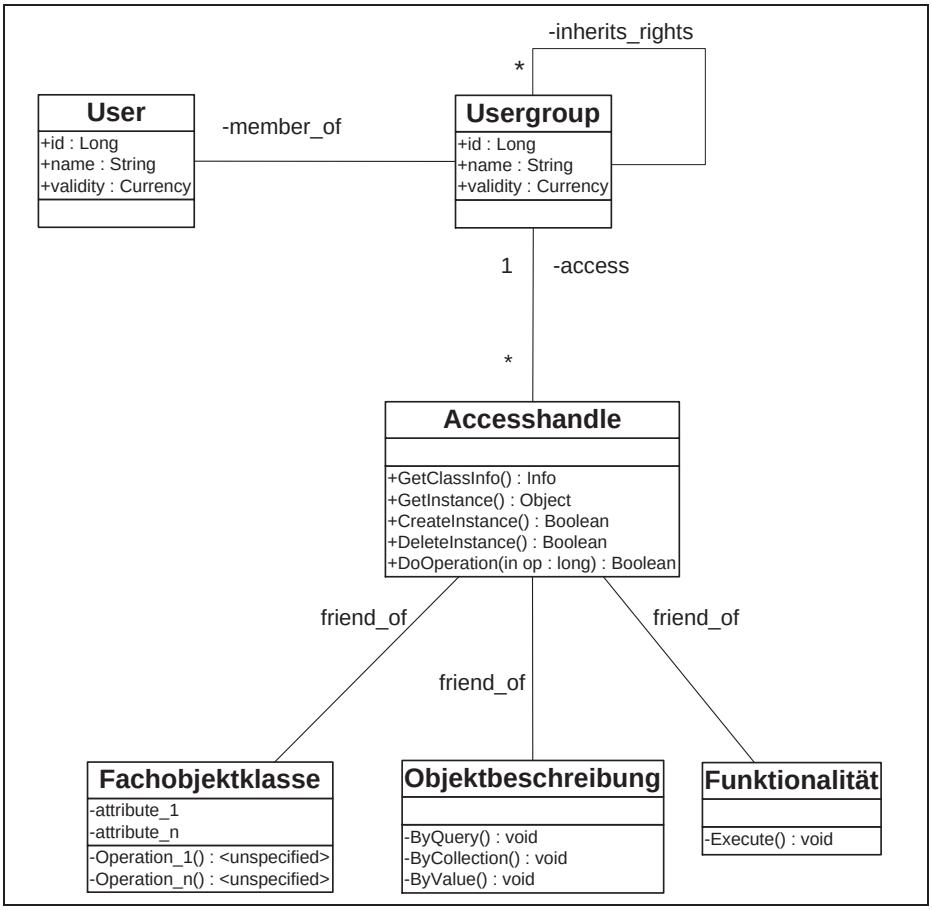

Abbildung 9.1: *Modellierung von Zugriffsrechten.*

9.1.1 Benutzer

Mit der Klasse `Benutzer` werden die relevanten und identifizierenden Eigenschaften von Anwendern des Systems modelliert. Die Instanzen dieser Klasse repräsentieren die einzelnen Systemkonten (*accounts*) natürlicher Personen.

Neben dem eindeutigen Identifikator (`id`) und einem Namen (`name`) werden Benutzer durch ein Gültigkeitsattribut (`gueltigkeit`) charakterisiert. Dieses beschreibt die zeitliche Beschränkung der Zugangsberechtigung - zum Beispiel durch die Angabe eines Ablaufdatums.

9.1.2 Benutzergruppe

Benutzer, die über die gleichen Berechtigungen verfügen sollen, werden zu fachlich motivierten Gruppen zusammengefasst. Diese Gruppen sind Instanzen der generischen Klasse `Benutzergruppe`. So wie schon einzelne Benutzer besitzen auch Benutzergruppen jeweils einen Identifikator, einen Namen sowie eine zeitlich begrenzte Gültigkeit. Durch die Einführung der selbstbezüglichen Relation „erbt_Rechte" kann eine Vererbungshierarchie im Sinne einer Halbordnung zwischen den Benutzergruppen beschrieben werden. Die Zugehörigkeit eines Benutzers zu einer Benutzergruppe wird durch das Setzen einer Beziehung „gehört_zu" modelliert. Die Instanzen der Klasse Benutzergruppe sind aus fachlicher Sicht somit Benutzerklassen, deren Instanzen einzelne Benutzer des Systems sind. Die zeitlichen Gültigkeiten eines Benutzers und der Benutzergruppe, der er angehört, steht in keinem direkten inhaltlichen Zusammenhang. Ein Benutzer-*Account* kann ablaufen, während die Gültigkeit seiner Benutzergruppe weiterhin erhalten bleibt. Umgekehrt gilt das gleiche.

9.1.3 Zugriffsklasse

Die Mitglieder von Benutzergruppen haben Beziehungen des Typs „hat_zugriff" zu ausgewählten Instanzen einer `Zugriffsklasse`. Über diese Instanzen erhalten die Benutzer Zugang zu ihrem eingeschränkten Datenmodell. Die Instanzen der Zugriffsklasse, die in Beziehung zu einer Benutzergruppe stehen, kapseln jeweils eine Klasse des zugriffsbeschränkten Datenschemas dieser Benutzergruppe sowie die zugreifbaren Exemplare dieser Klasse, oder sie regulieren den Zugriff von Mitgliedern dieser Gruppe auf spezielle Funktionalitäten. Die Zugriffsklasse beschreibt somit für jede Benutzergruppe Zugriffsfilter für Schemainformationen, Daten und Funktionen. Darüber hinaus kann mithilfe von Zugriffsklassen gewährleistet werden, dass für Benutzer nur solche Objekte zugreifbar sind, deren Klasse im Datenschema des Benutzers sichtbar ist.

9.1.4 Fachobjektklasse

Die Instanzen der Klasse `Fachobjektklasse` beschreiben das Datenschema der jeweiligen Anwendung. Das fachliche Schema einer Benutzersicht wird somit durch Instanzen des Implementierungsmodells beschrieben. Dabei werden

fachliche und strukturelle Beziehungen zwischen Klassen entsprechend den Beschreibungen im Ausgangsschema übernommen und Abhängigkeiten berücksichtigt. Der Zugriff auf Fachobjektklassen und deren Extensionen wird mithilfe von *friend*-Deklarationen über Zugriffsklassen gekapselt (Abschnitt 6.3). Fachobjektklassen besitzen nur private Eigenschaften und Methoden und liegen für keine Benutzergruppe im Namensraum. Somit haben ausschließlich die jeweiligen Zugriffsklassen die Möglichkeit Methoden von Fachobjektklassen auszuführen oder Eigenschaften von einzelnen Instanzen abzufragen.

9.1.5 Objektbeschreibung

Bezüglich jeder Fachobjektklasse, die über eine Zugriffsklasse referenziert wird, kann eine Beschränkung der zugreifbaren Fachobjekte dieser Fachobjektklasse vorgenommen werden. Mit dieser Spezifizierung von Fachobjekten durch in Beziehung gesetzte Instanzen der Klasse `Objektbeschreibung` werden die Objekte festgelegt, die über die Zugriffsklasse und damit für die betroffenen Benutzergruppen verfügbar sind. Zugriffsbeschränkungen für Fachobjekte können prinzipiell durch drei Methoden definiert werden: Durch die Festlegung eines Wertebereichs (`ByValue`), durch die Zuweisung einzelner Objekte zu einer ausgewiesenen Menge (`ByCollection`) und durch die Formulierung komplexer Anfragen (`ByQuery`). Dabei stellt eine geeignete Anfragelogik, mit der der Umfang der beiden anderen Formen abgedeckt werden kann, das allgemeingültigste Werkzeug zur Beschreibung von Objekten dar. An dieser Stelle wird von der Existenz einer Anfragesprache im Sinne der Beschreibungen aus Kapitel 8 ausgegangen.

9.1.6 Funktionalität

Die Klasse `Funktionalität` beschreibt spezielle Operationen und Prozesse, die nicht als Methoden einzelner Fachobjektklassen implementiert sind, möglicherweise aber in Bezug auf die Instanzen der referenzierten Fachobjektklasse durch einen Benutzer ausgelöst werden sollen. Derartige Funktionen dienen im allgemeinen der Modellierung von Geschäftsprozessen. Innerhalb von ALKIS® sind dies zum Beispiel Prozessobjekte wie: „Fortführungsauftrag", „Einrichtungsauftrag", „Reservierungsauftrag" oder „Benutzungsauftrag". In gleicher Weise wie für Fachobjektklassen, kann die Berechtigung von Benutzern zum Zugriff auf bestimmte Funktionalitäten durch *friend*-Deklaration der entsprechenden Zugriffsklasse gesteuert werden.

9.2 Klassenentwurf einer Zugriffsstruktur

Nach den Darstellung der Komponenten eines allgemeingültigen Klassendesigns sollen zum Abschluss der Untersuchungen dieser Arbeit noch Aspekte der konkreten Implementierung innerhalb der SupportGIS-Entwicklung betrachtet werden. Dabei stehen weniger die Algorithmen zur Auswertung von Zugriffsrechten und zur Ermittlung der sichtbaren Objekte im Vordergrund, als vielmehr der Entwurf einer effizienten Zugriffsstruktur, die sich in einem möglichst einfachen, am Datenmodel angelehnten Klassendesign manifestiert und eine nach außen sichtbare, einheitliche Zugriffsschicht in Form einer Anwendungsschnittstelle (*API*) anbietet, die die internen Methoden des selektiven und eingeschränkten Zugriffs auf die Objekte vor dem Benutzer verbirgt. Die Grundlage einer exemplarischen Implementierung bildet die bereits vorgestellte generische Kernel-Architektur aus Abbildung 3.8 und der Grobentwurf einer Zugriffsarchitektur in Abschnitt 9.1.
Der erste Schritt des Klassendesigns (Abbildung 9.2) basiert auf der Frage nach dem grundsätzlichen Zugang von Benutzern zu den Geo-Objekten einer Datenquelle. Innerhalb der zentralen Datenverwaltung ist der Zugriff auf die internen und fachlichen Strukturen sowie auf die Instanzen der beschriebenen Klassen über einen einheitlichen und persistenten Einstiegsknoten (`Root`) organisiert. Relativ zu diesem sind thematische Fachobjektklassen und geometrische Fachbedeutungsklassen in der beschriebenen Weise als Instanzen der generischen Klassen `DescClass` und `DescGeo` hinterlegt. Über diese generischen Klassen definiert sich das fachliche Datenschema. Mit der Modellierung von 1:n-Beziehungen zu den Instanzen der Klassen `InstClass` und `GeoObject` werden die Fach- und Geo-Objekte der Datenbank in Relation zu ihren beschreibenden Klassen gesetzt. Mittels dieser Beziehungen, die intern als Listen von Zeigern auf Fachobjekte oder entsprechende Objekt-IDs realisiert sind, ist über die Klassen des Datenschemas der Zugriff auf sämtliche Sachdaten der Datenverwaltung möglich.
Benutzer, die über eine Applikationsebene mit einer fachlichen Sicht der Daten arbeiten wollen, sollen im Sinne von Abschnitt 8.2.3 immer nur Zugriff auf thematische Ausschnitte des Datenmodells erhalten, die durch die Zugriffsrechte des jeweiligen Benutzers abgedeckt sind. Zu diesem Zweck wird zunächst ein weiterer Einstiegsknoten als statische Instanz der Klasse `UsrRoot` in die Datenbank eingeführt, über den sämtliche Zugriffe durch Benutzer des Systems aus Anwendungen heraus erfolgen.

Unterhalb dieser (einzigen) Instanz des *UsrRoot* befindet sich die Menge der definierten Benutzergruppen des GI-Systems als persistente Instanzen der generischen Klasse `UserGroup`. Die Ableitungsbeziehungen zwischen den einzelnen Benutzergruppen, wie in Abschnitt 6.2 dargestellt, werden durch $n : m$-

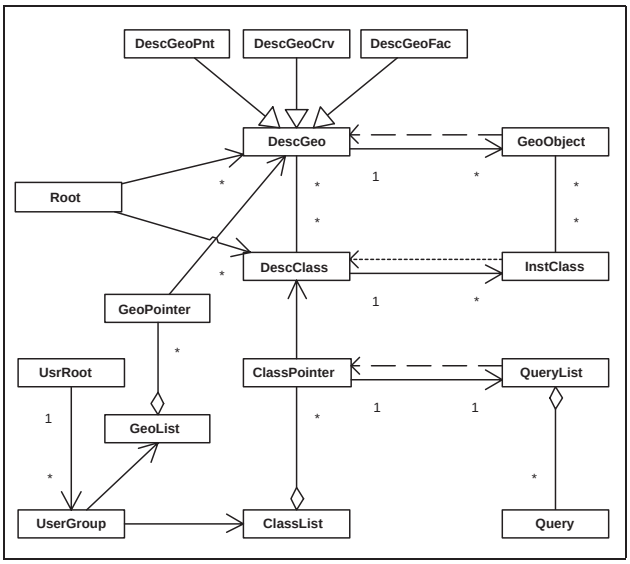

Abbildung 9.2: *Klassenentwurf 1 - Benutzerklasse (*`UserGroup`) mit Listen von Klassenzeigern und Anfragelisten für jede Klasse.

Beziehungen zwischen den Instanzen von `UserGroup` modelliert, sind aber in Abbildung 9.2 nicht dargestellt, da sie für die Grundzüge der Zugriffsarchitektur nicht maßgeblich sind. Zu jeder Benutzergruppe existiert jeweils eine Instanz der Klasse `GeoList` und eine Instanz der Klasse `ClassList`. Die Instanz der *GeoList* speichert für jede Benutzergruppe eine Liste von Zeigern auf die Geometrie-Fachbedeutungen der unterschiedlichen Geometrietypen, die für Mitglieder dieser Benutzergruppe sichtbar sein sollen. Die Instanz der Klasse `ClassList` beschreibt das für die Benutzer der Gruppe zugängliche Fachobjekte-Schema durch eine Liste von Zeigern auf Fachobjektklassen. Diese werden durch Instanzen der Klasse `ClassPointer` repräsentiert. Die Funktion eines `ClassPointer`s geht über die Verwaltung eines internen Zeigers auf eine Fachobjektklasse hinaus: Zusätzlich existiert relativ zu diesem jeweils eine Liste von Anfragen (`QueryList`), mit denen für jede Benutzergruppe die Daten der Benutzersicht dieser Klasse beschrieben werden. Mit der Verwendung von Zeigerklassen, deren Instanzen Schemainformationen referenzieren, werden zwei Ziele erreicht:

1. Das für eine Benutzergruppe definierte Teilschema bezieht seine Informationen bei jedem Zugriff direkt aus den originären Schemaklassen. Jede Änderung einer Schemainformation ist somit sofort allen betroffenen Benutzerklassen bekannt. Die Schemadefinitionen müssen nicht mehrfach gepflegt werden.

2. Gleichzeitig kann von den Mitgliedern einer Benutzergruppe immer nur auf die Schemainformationen zugegriffen werden, die in den entsprechenden Listen der Zeiger auf Geometrie-Fachbedeutungen und Fachobjektklassen vorgesehen sind.

Da jede sichtbare Fachobjektklasse einer Benutzergruppe durch eine Instanz der Klasse `ClassPointer` repräsentiert wird, kann mit der Zuweisung von Zeigern auf die sichtendefinierenden Anfragen dieser Klasse erreicht werden, dass über das Teilschema einer Benutzerklasse in effizienter Weise auf die sichtbaren Instanzen der jeweiligen Klasse zugegriffen werden kann. Gleichzeitig ist die Definition einer Sicht immer in einem konsistenten Zustand, da sich die Beschreibung zugreifbarer Instanzen einer Benutzerklasse immer nur auf die Instanzen von Klassen beziehen kann, die durch das Teilschema der Benutzerklasse abgedeckt sind.

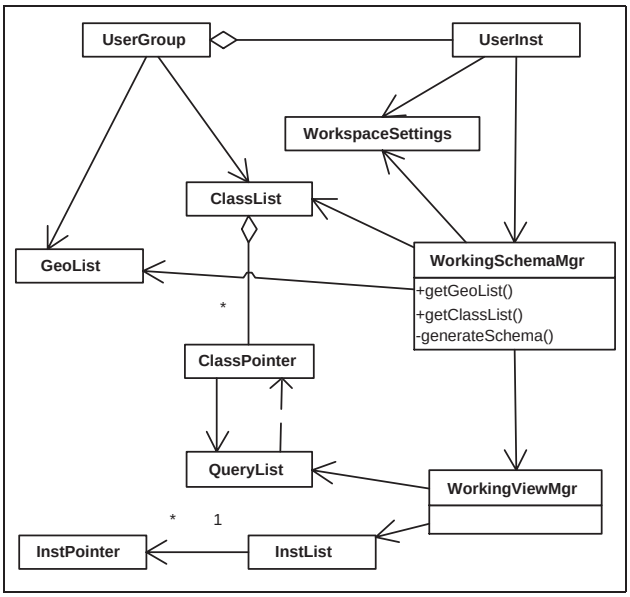

Abbildung 9.3: *Klassenentwurf 2 - Benutzer als Instanzen von Benutzergruppen haben Zugriff auf Fachobjekte über einen Schemamanager und einen Sichtenmanager.*

Im zweiten Schritt des Entwurfs (Abbildung 9.3) sollen die Benutzer selber als Repräsentanten von Benutzergruppen in die Betrachtungen einbezogen und deren Zugang zu den Daten spezifiziert werden. Benutzergruppen aggregieren sich im wesentlichen aus ihren Mitgliedern, ergänzt um weitere gruppenspezifische Eigenschaften. Die Benutzer sind als Instanzen der Klasse `UserInst` modelliert. Meldet sich ein Benutzer bei der Datenverwaltung an, so tut er dies mit seinen individuellen, identifizierenden Merkmalen. Bei erfolgreicher Anmeldung erhält er für die Dauer der Sitzung eine Verbindung zu dem Objekt, mit

dem seine Zugangseigenschaften und Privilegien persistent beschrieben sind: Die Repräsentation des Benutzers als Instanz von `UserInst`. Die gesamte Kommunikation eines einzelnen Benutzers mit den Daten des GIS wird von zwei Komponenten gekapselt und organisiert: Einem Schemamanager und einen Sichtenmanager. Der Schemamanager erscheint im Klassenentwurf als Instanz einer Klasse `WorkingSchemaMgr`. Dieser hat Zugriff auf die Fachbedeutungs- und Klassenlisten der Benutzergruppe, deren Mitglied der angemeldete Benutzer ist. Darüber hinaus referenziert er die Einstellungen des aktuell gewählten Themenbereichs aus einer Instanz der Klasse `WorkspaceSettings`. Die zentrale Aufgabe des Schemamanagers besteht darin, die Zugriffsrechte der jeweiligen Benutzergruppe hinsichtlich der verfügbaren Schemainformationen unter Berücksichtigung aller Vererbungs- und Abhängigkeitsbeziehungen zu ermitteln, diese im Kontext eines gewählten Themenbereichs auszuwerten und ein Datenschema für den aktuellen Arbeitsbereich zu generieren. Dieses Schema wird nun den Applikationen und damit dem angemeldeten Benutzer zur Verfügung gestellt.

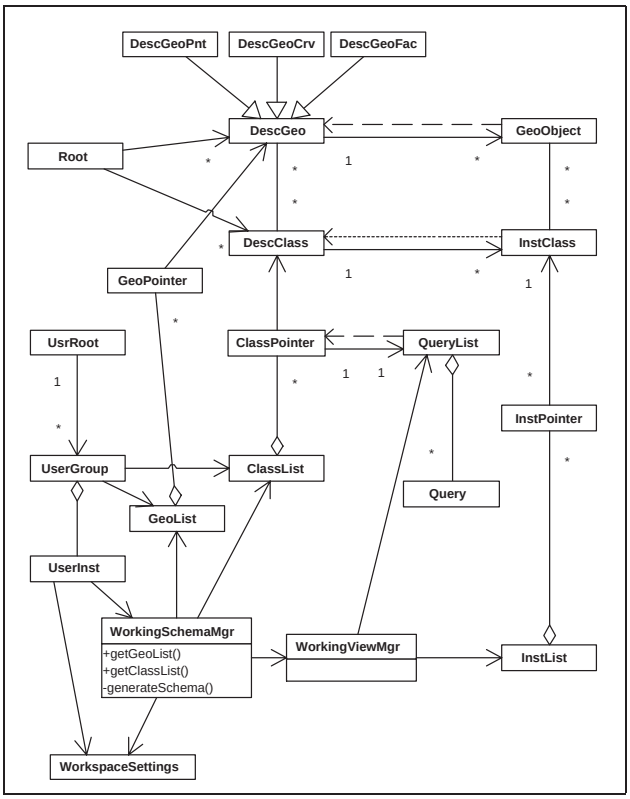

Abbildung 9.4: *Klassenentwurf 3 - vollständige Benutzer-Zugriffsstruktur für die Schema- und Instanzebene über einen Schema-Manager und einen Sichten-Manager.*

In Abhängigkeit vom ermittelten Datenschema des Arbeitsbereichs generiert der Schemamanager einen Sichtenmanager als Instanz von `WorkingViewMgr`. Dieser führt die Verschmelzung der sichtendefinierenden Anfragen durch, wertet diese aus und legt die Ergebnisse als Listen von Objektreferenzen in einem Verwaltungsbereich ab, der der Kontrolle des *Workspaces'* unterliegt und damit durch den Benutzer zugreifbar ist. Im Weiteren ist der generierte *Workspace* über die Applikationen zugreifbar. Die zuverlässige Implementierung der Methoden von Schemamanager und Sichtenmanager ist bei diesem Vorgang entscheidend für die Sicherheit des Gesamtsystems, da sämtliche Benutzerzugriffe über diese Komponenten ablaufen. Die Systemarchitektur unterstützt die Sicherheit und Zuverlässigkeit insbesondere in folgenden Aspekten: Auf der Ebene der Generierung des aktuellen Arbeitsschemas kann der Schemamanager nur auf die Referenzen von Schemaklassen zugreifen, die der jeweiligen Benutzergruppe zugewiesen sind. Da das Benutzerschema die Informationsbasis für den Zugriff auf themenspezifische Klassenbeschreibungen bildet und damit Obergrenze der verfügbaren Schemadaten darstellt, können keine Schemainformationen an einen Benutzer oder eine Applikation weitergereicht werden, die nicht innerhalb der Zugriffsrechte des Benutzers liegen. Die Algorithmen der Sichtengenerierung basieren auf der Auswertung von Datenbankanfragen. Die Strukturen von Schemamanager und Sichtenmanager garantieren eine Beschränkung der Sichten eines Arbeitsbereichs auf Instanzen der Klassen, die vom Schemamanager ermittelt wurden und somit innerhalb der Zugriffsrechte eines angemeldeten Benutzers liegen. Um sicherheitsrelevante Fehler in der Auswertung von Anfragen zu vermeiden, sollten Basiskriterien der Objektauswertung - wie zum Beispiel die Bereichsprüfung von Attributwerten oder die Auswertung von Vergleichsoperatoren - weitestgehend auf verifizierte und praxiserprobte Methoden des gewählten Datenbanksystems abgebildet werden. Vor diesem Hintergrund bietet das vorgestellte Klassendesign eine generische Basis für die Implementierung einer effizienten und zuverlässigen Zugriffsstrategie für objektorientierte Geoinformationssysteme.

9.3 Implementation der Zugriffsarchitektur

Die Umsetzbarkeit des vorgestellten Berechtigungskonzeptes im Sinne der entworfenen Zugriffsarchitektur wurde mit einer Implementation auf der Basis des Programmsystems SupportGIS demonstriert ([40]). Dabei wurden die Strukturen zur Beschreibung von Benutzerprofilen sowie die Algorithmen zur Filterung der Objekt- und Methodenzugriffe in den generischen System-Kernel integriert. Gegenüber externen Sichten sind die Benutzerstrukturen über eine anwendungs- und benutzerneutrale Programmschnittstelle gekapselt. Somit

ist für den Anwender oder Anwendungsentwickler nur das generische, objektorientierte Fachdatenmodell über die bereits gefilterten Fachobjektklassen und deren Extension zugreifbar. Eine Überprüfung der Benutzerrechte bei jedem Zugriff entfällt somit.
Für die GIS-Anwendungen sind Zugriffsrechte und Benutzerprofile in zwei Bereichen relevant:

1. Die Beschreibung von Zugriffsprofilen durch einen Administrator oder Teiladministrator in Bezug auf einen fachlichen Arbeitsbereich.

2. Die Generierung von Sichten und die Präsentation der sichtbaren Objekte durch die Auswertung der Zugriffsrechte eines Benutzers im Rahmen eines gewählten Arbeitsbereichs.

9.3.1 Beschreibung von Zugriffsprofilen

Administratoren oder Teiladministratoren können im Rahmen ihrer Zuständigkeiten Zugriffsprofile definieren und beschreiben. Die Zuständigkeiten eines Teiladministrators wird dabei durch die Zuweisung von Arbeitsbereichen zu einem Benutzer mit Administratorrechten festgelegt. In der Semantik relationaler Datenbanken entspricht die administrative Zuständigkeit eines Benutzers der *Grant*-Eigenschaft *„With Grant Option"*, die das Recht zur Weitergabe von Zugriffsrechten definiert.
Zugriffsprofile basieren auf der Beschreibung von Arbeitsbereichen, die mit einem Administrationswerkzeug bearbeitet werden können (Abbildung 9.5). Die Administrationsanwendung kann dabei grundsätzlich nur Datenbanken bearbeiten, für die der angemeldete Benutzer (Teil-)Administratorenrechte besitzt. Die zur Verfügung gestellten Daten sind jeweils auf die Zuständigkeiten des Administrators eingeschränkt. Der Administrator definiert nun fachlich und räumlich motivierte Arbeitsbereiche, die einen eindeutigen Namen und eine textuelle Beschreibung erhalten. Für jeden Arbeitsbereich kann ein räumlicher Ausschnitt festgelegt werden, innerhalb dessen die fachlichen Definitionen interpretiert werden sollen. Die fachliche Beschreibung eines Arbeitsbereichs unterteilt sich in die rein geometrischen Fachbedeutungsklassen und die komplexeren Fachobjektklassen. Diese können einzeln einem Arbeitsbereich hinzugefügt oder von diesem entfernt werden. Der *Kernel* des Berechtigungssystems überprüft die fachlichen und strukturellen Abhängigkeiten und entscheidet, ob die gewünschte Auswahl der Klassen zulässig ist. Im Fall einer Ablehnung wird in einem interaktiven Prozess festgelgt, ob auf die Aufnahme bzw. Entfernung einer Klasse zum bzw. vom Arbeitsbereich verzichtet wird oder ob die abhängigen Strukturen im Arbeitsbereich nachgebildet bzw. aufgelöst werden sollen.

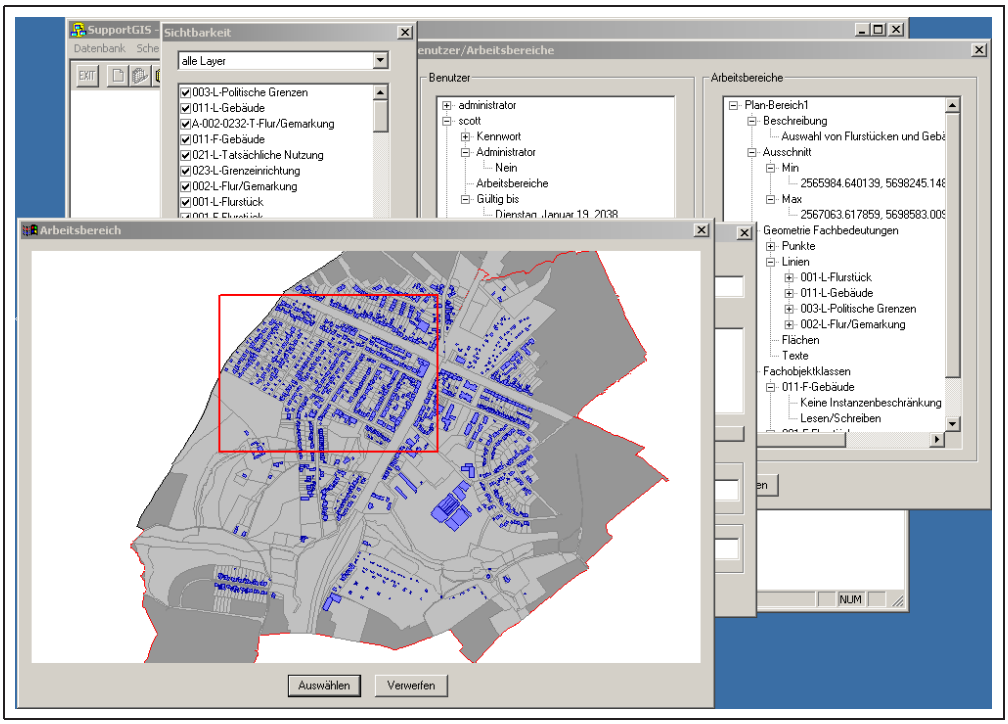

Abbildung 9.5: *Oberfläche zur Definition von Arbeitsbereichen und Berechtigungen.*

Für jede ausgewählte Klasse kann eine deklarative Anfrage formuliert werden, die den Extent der Klasse im jeweiligen Arbeitsbereich definiert. Wird auf eine Anfrage verzichtet, so wird der Extent der Klasse im Sinne des räumlichen Geltungsbereichs interpretiert.
Derart definierte Arbeitsbereiche können nun Benutzerprofilen zugewiesen werden. Damit werden implizit die Zugriffsrechte der Mitglieder einer Benutzergruppe festgelegt. Diese beschreiben als Benutzerklasse jeweils einzelne Benutzer oder Gruppen von Benutzern mit gleichen Berechtigungen. Neben den zugewiesenen Arbeitsbereichen und der Festlegung der Administratoreigenschaft, enthalten Benutzerprofile insbesondere eine Festlegung der zeitlichen Gültigkeit des Profils. Nach Ablauf der angegebenen Frist kann die Gültigkeit des Profils nur durch einen Administrator verlängert werden.

9.3.2 Auswertung von Zugriffsprofilen

Die schematische Auswertung eines Zugriffsprofils erfolgt zum Zeitpunkt der Anmeldung eines Benutzers bei einem Arbeitsbereich der gewählten Datenbank. Dabei wird zunächst das Anwendungsschema des Benutzers ermittelt.

Dies legt neben den Klassenstrukturen und -abhängigkeiten auch die verfügbaren Klassenmethoden der Sicht fest. Aus der ermittelten Struktur - insbesondere aus der ermittelten Vererbungshierarchie - ergibt sich nicht zuletzt die Präsentationsform der sichtbaren Objekte. Dies ist für grafische Ansichten des Datenbestandes besonders relevant (Abbildung 9.6).

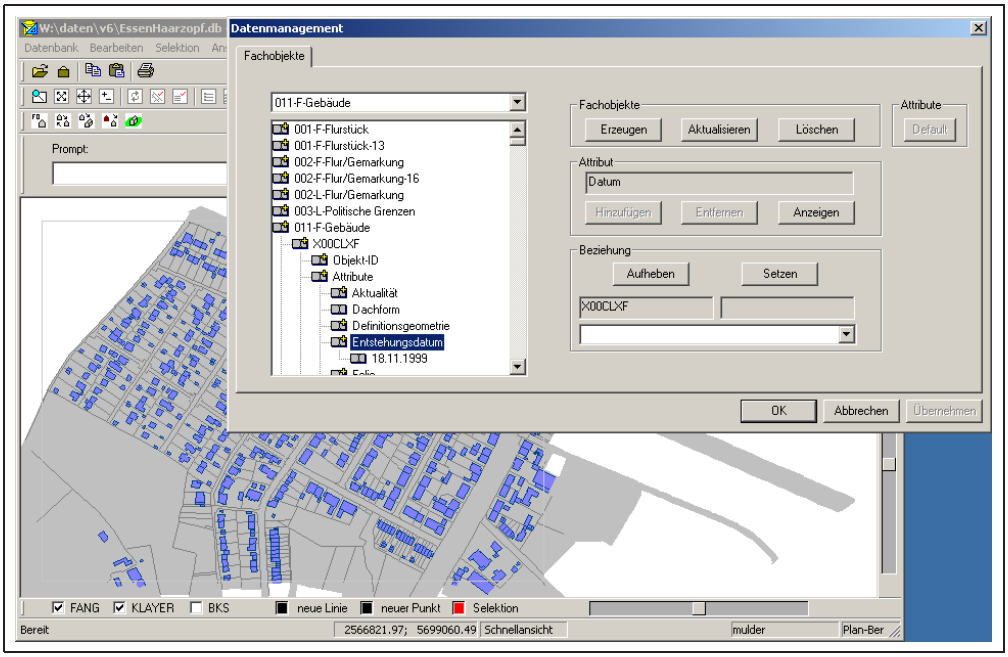

Abbildung 9.6: *Sicht eines Benutzers auf die Informationen eines zulässigen Arbeitsbereichs.*

Bei jedem erstmaligen Zugriff des Benutzers auf den Extent einer Klasse wird die entsprechende Instanzenbeschränkung, im Sinne räumlicher Geltungsbereiche und definierender Anfragen, interpretiert und das Ergebnis in die sichteigene Klassenextension übernommen. Dabei werden keine Objektkopien angelegt, sondern Referenzen der Originalobjekte als Extent der entsprechenden Zugriffsklasse verwaltet. Neben der Extension der ausgewählten Klasse werden Abhängigkeiten innerhalb des Sichtenschemas analysiert und bei Bedarf weitere Klassenextensionen berechnet. Die Verwaltungsstrukturen einer Sicht werden in einem Bereich gehalten, der datenbankseitig vollständig gesperrt und gezielt freigegeben werden kann. Im Fall des ObjectStore DBMS bot sich hierfür die Segment-Struktur an, die einer sicheren Zugriffskontrolle durch das DBMS unterliegt.
Aufgrund der Objektreferenzen sind die Instanzen einer Sicht immer im aktuellen Zustand. Externe Modifikationen von Sicht-Objekten durch andere Benutzer müssen nicht zu gesonderten Benachrichtigungen führen. Die Notwen-

digkeit zur Aktualisierung einer Extension innerhalb einer Sicht wird mittels eines Actor-/Reaktor-Konzeptes vom System erkannt und kann entsprechend umgesetzt werden. Die Kommunikation eines angemeldeten Benutzers mit den Objekten einer Datenbank erfolgt ausschließlich über die Strukturen der generiertenn Sicht. Dazu werden sämtliche Zugriffe eines Benutzers auf die Klassen eines Datenmodells bzw. auf die Extension einer Klasse intern auf entsprechenden Zugriffsklassen umgeleitet.

9.4 Zusammenfassung

In diesem Kapitel wurde als Abschluss der Untersuchungen ein Systementwurf vorgestellt, der als Basis für die im letzten Abschnitt skizzierte Implementation der entworfenen Zugriffsstrategie diente. Zur Gewährleistung der Allgemeingültigkeit war dabei ein generisches Modell von Fach- und Geo-Objekten zentraler Bestandteil des Entwurfs. Die Sichtendefinition und Zugriffskontrolle wird im wesentlichen durch vier Konzepte erreicht, die erst als Gesamtkonzept eine wirkungsvolle Beschränkung der Benutzerrechte und Überwachung sämtlicher Zugriffe ermöglichen:

1. Die Kapselung der Klassen des fachlichen Datenmodells durch entsprechende Zugriffsklassen sowie die Beschränkung sämtlicher Benutzerzugriffe auf die Methoden dieser Zugriffsklassen.

2. Die Einschränkung der Semantikräume durch konsistente Auswahl fachlich abgeschlossener Subschemata.

3. Die Definition einer geeigneten objektorientierten, räumlichen Anfragesprache zur Beschreibung von Teilmengen der Klassenextensionen.

4. Eine Hierarchische Strukturierbarkeit von Benutzerklassen und der damit verbundenen Vererbung von Benutzerrechten.

Die dargestellte Architektur ermöglicht eine an der Semantik eines fachlichen Datenmodells angelehnte Beschreibung und Auswertung von Zugriffsrechten. Dabei werden räumliche und fachliche Aspekte gleichermaßen berücksichtigt. Teilmodelle werden derart generiert, dass die Ausführung der verfügbaren Methoden die Konsistenz des Gesamtmodells nicht gefährdet. Mit einer konkreten Implementierung wurde schließlich die Realisierbarkeit der Strategien und Konzepte demonstriert.

Kapitel 10

Zusammenfassung

In der vorliegenden Arbeit wurde ein Konzept entwickelt, mit dem Zugriffsrechte für raumbezogene Daten in Anlehnung an die Semantik der modellierten Realweltmodelle in einem thematischen Kontext beschrieben und ausgewertet werden. Auf der Grundlage dieser Berechtigungsstrukturen lassen sich aus einem gegebenen intendierten Datenmodell Sichten auf Teilmodelle nach einfachen oder komplexen räumlichen und fachlichen Kriterien abgeleiten. Der Zugriff auf die Daten selbst wird dabei entsprechend den objektorientierten Paradigmen über Klassenmethoden gekapselt, deren Verfügbarkeit in den jeweiligen Benutzerprofilen festgelegt wird.
Ausgangspunkt der Untersuchungen war eine Analyse des Bedarfs an geeigneten Zugriffs- und Berechtigungsstrukturen für GIS in Kapitel 2. Dabei wurden einige Anwendungsfelder von GIS - sowohl im öffentlichen, als auch im privaten Sektor - exemplarisch betrachtet. Insbesondere war das Vorkommen und der Umgang der jeweiligen Anwendungen mit sicherheitskritischen Daten und Prozessen von Interesse.
GIS dienen der strukturierten Beschreibung und Auswertung von raumbezogenen Fachobjekten und deren gegenseitigen Beziehungen. Dabei werden mitunter Daten und Zusammenhänge verwaltet, die nur für ausgewählte Benutzer und spezielle Anwendungen vorgesehen und für die Sicherheit des Systems oder den Schutz der beschriebenen Realweltobjekte oder -personen relevant sind. Derartige Daten finden sich zum Beispiel in polizeilichen Anwendungen, in Systemen zum Katastrophenschutz aber auch in der öffentlichen Verwaltung. Der Begriff der Sicherheit bezieht sich sowohl auf thematisch motivierte Bestrebungen, wie den Schutz von Persönlichkeitsrechten und den Schutz vor wirtschaftlichem Schaden, als auch auf strukturelle Anforderungen, wie den Erhalt von Konsistenz und Integrität der modellierten Objekte.
In aktuellen Entwicklungen im GIS-Umfeld ist der allgemeine Trend festzu-

stellen, raumbezogene Daten vermehrt über geeignete netzbasierte Architekturen im Intra- und Internet den Anwendern und Bearbeitern zur Verfügung zu stellen. Dies soll unter anderem dazu beitragen, den Wert und das Marktpotenzial von Geodaten optimal auszuschöpfen und eine redundante Verwaltung von Daten und Diensten zu vermeiden. Neben den offensichtlichen Vorteilen netzbasierter GIS wächst mit der Anzahl der Benutzer eines Systems auch die potenzielle Gefahr für die Sicherheit der verfügbaren Daten und Dienste. Wenn Benutzer mit vielseitigen Interessenslagen und unterschiedlichen Zuständigkeiten die Dienste eines GIS in Anspruch nehmen wollen, so muss und darf diesen dabei kein Vollzugriff auf alle Komponenten und Daten des Systems gewährt werden. Vielmehr soll der Zugriff eines Benutzers ausschließlich auf die Daten und Funktionen eingeschränkt werden, die er zur Erfüllung einer konkreten Aufgabe benötigt. Um den Zugriff auf moderne GIS-Architekturen kontrollierbar zu machen, bedarf es geeigneter und zuverlässiger Strategien und Konzepte, die insbesondere den Anforderungen raumbezogener Informationssysteme gerecht werden.

Diese Anforderungen an die Zugriffsstrukturen für GIS, die zur effizienten Beschreibung thematisch orientierter Berechtigungsprofile geeignet sind, werden in Kapitel 3 mit einer Untersuchung raumbezogener Datenmodelle verdeutlicht. Dabei erlangen die Konzepte der objektorientierten Modellierung eine ständig wachsende Bedeutung. Anders als zum Beispiel relationale Modelle ermöglichen sie eine realweltnahe, an der Semantik angelehnte Beschreibung von raumbezogenen Daten und Prozessen. Insbesondere wird bei der objektorientierten Modellierung die Trennung zwischen Daten und Methoden aufgehoben. Die objektorientierte Abbildung eines Realweltausschnitts in die Modellstrukturen eines GIS unterstützt den Erhalt der wesentlichen Objekteigenschaften hinsichtlich einer fachlichen Interessenslage und Aufgabenstellung. Dabei können objektorientierte Datenmodelle komplexe Beziehungen und Abhängigkeiten zwischen den modellierten Objekten beschreiben.

Die Ausdrucksstärke und Flexibilität objektorientierter Geo-Datenmodelle lässt sich insbesondere auf die Verwendung der Konzepte Vererbung, Aggregation und Assoziation zurück führen. Eine zentrale Anforderung an ein Zugriffs- und Berechtigungskonzept für raumbezogene Daten besteht darin, die fachlichen und strukturellen Zusammenhänge und Abhängigkeiten zwischen den Modellobjekten zu erkennen und diese in die definierten Berechtigungsprofile zu integrieren. Durch geeignete Methoden muss vermieden werden, dass mit Berechtigungsprofilen inkonsistente oder nicht abgeschlossene Teilmodelle definiert werden.

Formale Zugriffsmodelle aus dem Umfeld der IT-Sicherheit liefern eine wissenschaftliche Basis für den Entwurf eines zuverlässigen Berechtigungskonzeptes für GIS. Die wesentlichen Modelle zu skizzieren und einen Überblick über das Thema IT-Sicherheit zu vermitteln ist das Anliegen von Kapitel 4. Einer der

zentralen Ansätze zur Zugriffsbeschränkung wurde mit dem Bell/LaPadula-Modell definiert. Obwohl dieses Modell vor dem Hintergrund der Sicherheit von Betriebssystemen entstanden ist, enthält es Ideen wie die stufenweise Klassifizierung von Objekten und die Vergleichbarkeit von Benutzern, die auch in den Ansätzen eines objektorientierten Zugriffskonzeptes für raumbezogene Daten nützlich sind: Sowohl Benutzer als auch die zu schützenden Objekte werden dabei den Knoten eines gerichteten zyklenfreien Graphen zugeordnet. Die Rechte eines Benutzers in Bezug auf ein Objekt können somit aus der Existenz eines Pfades im Graphen vom Benutzerknoten zum Objektknoten abgeleitet werden.
Auf der Grundlage objektorientierter Modellierungsstrategien für raumbezogene Daten wird in Kapitel 5 die Erteilung von Zugriffsrechten und deren Folgen näher untersucht. Dabei werden Zugriffsrechte für Geodaten hinsichtlich der impliziten Auswirkungen auf die Sichtbarkeit und Zugreifbarkeit anderer Daten und Funktionen näher betrachtet. In Anlehnung an die objektorientierten Konzepte zur Definition und Vererbung von Klassenbeschreibungen sowie der Verwendung der Beziehungstypen Aggregation und Assoziation entstehen bei diesen Untersuchungen Regeln, die die Zusammenhänge der Zugriffe auf objektorientierte Geodaten sowie die Durchsetzbarkeit von Zugriffsbeschränkungen für diese Daten aufdecken und formalisieren.
Der Entwurf einer effizienten Zugriffs- und Berechtigungsstrategie für raumbezogene Daten muss diesen Zugriffsregeln Rechnung tragen und eine konsistente Vergabe oder Einschränkung von Zugriffsrechten unterstützen.
Der hierzu in Kapitel 6 präsentierte Entwurf basiert auf der Idee, Benutzergruppen und Benutzer - in Analogie zu den Geodaten - als Klassen und deren Instanzen vollständig in das Datenmodell zu integrieren. Werden nun sämtliche Zugriffe auf die Daten eines Modells über die Methoden der definierenden Klassen gekapselt, so können Zugriffsrechte für Benutzer durch die Vererbung dieser Klassen an die entsprechenden Benutzerklassen erteilt werden. Art und Umfang des Zugriffs wird somit ausschließlich über die verfügbaren Methoden definiert und muss konzeptionell nicht nach lesendem und schreibendem Zugriff unterschieden werden. Darüber hinaus können Zugriffshierarchien zwischen Benutzern durch Vererbungsbäume festgelegt werden.
Klassenvererbungen implizieren eine Spezialisierung von Objektbeschreibungen. Mit Aggregationen von Klassen können Teilmengenbeziehungen zwischen Objekten einer Vererbungsebene deklariert werden. Diese beiden grundsätzlichen Eigenschaften führen dazu, dass mit der Erteilung von Zugriffsrechten für Objekte über die Klassen- und Methodendeklarationen einer ausgewählten Vererbungsebene die Tiefe der verfügbaren Objektdetails sowie die Möglichkeiten zur Bearbeitung dieser Informationen festgelegt werden kann. Angewendet auf Geo-Objekte zeigen die Ausführungen in Kapitel 7, wie die fachlichen Repräsentationen und grafischen Darstellungen der sichtbaren Objekte in Abhängigkeit vom jeweiligen Benutzer kontrolliert werden können: Während

ein privilegierter Benutzer innerhalb eines räumlichen Ausschnitts nach öffentlichen und privaten Gebäuden und deren spezifischen Nutzungen unterscheiden kann, soll ein anderer Benutzer lediglich erkennen, dass es sich bei den abgebildeten Objekten um Gebäude handelt. Die Repräsentationen bestimmender Objekte, zum Beispiel der definierenden Begrenzungslinien von Flächen einer bestimmten Fachbedeutung, ergeben sich aus der Vererbungsebene der Klassen, die Bestandteile der beschreibenden Assoziationen oder Aggregationen sind.

Für eine Beschreibung von fachlich motivierten Zugriffsprofilen bietet sich die Einführung von Arbeitsbereichen an, mit denen Sichten als thematische Einheit definiert werden. Eine solche Strategie wird mit den Entwürfen in Kapitel 8 verfolgt. Die definierten Arbeitsbereiche aggregieren thematisch zusammengehörige Zugriffsrechte nach fachlichen und räumlichen Kriterien. Dabei werden sowohl die verfügbaren Klassen innerhalb eines Arbeitsbereichs eingeschränkt, als auch die Menge der zugreifbaren Instanzen der sichtbaren Klassen beschränkt. Die Auswahl bestimmter Instanzen einer Klasse basiert auf Objektbeschreibungen mit den Methoden einer für raumbezogene Daten geeigneten Anfragesprache. Die Methodik hierzu wird am Beispiel der objektorientierte Anfragesprache des Systems SupportGIS erläutert. Deren Komponenten erlaubt eine am fachlichen Datenschema orientierte Beschreibung von Objekteigenschaften, unter Einbeziehung des Raumbezugs sowie der fachlichen Beziehungen zwischen Objekten. Mit der Verknüpfung dieser Komponenten über logische Kanten entsteht die Möglichkeit, Objektmengen über komplexe Eigenschaften modellnah und flexibel zu definieren. Benutzern bzw. deren Benutzergruppen können nun Arbeitsbereiche in komfortabler Weise zugewiesen werden. Diese Zuweisung erfolgt wiederum durch die Methode der Vererbung. Über die Arbeitsbereiche erhalten die Benutzer Zugriff auf die Daten einer oder mehrerer Datenquellen eines Themenbereichs in einem vorgegebenen räumlichen Ausschnitt.

Mit dem Entwurf einer objektorientierten Zugriffsarchitektur wird in Kapitel 9 die grundsätzliche Klassenstruktur für die Realisierung des entwickelten Berechtigungskonzeptes definiert. Dabei wird die Funktionalität und Arbeitsweise der zugriffsrelevanten Komponenten näher erläutert. Eine wesentliche Eigenschaft des Entwurfs besteht darin, dass sämtliche Benutzerzugriffe auf Klassen und Instanzen eines Datenmodells jeweils über entsprechende Zugriffsobjekte gekapselt werden. Diese enthalten die Beschreibungen der für das angemeldete Benutzerprofil zugreifbaren Schemainformationen und Fachobjekte. Für den Benutzer verbergen diese Objekte das Verhalten der aufgerufenen Methoden und präsentieren diesem eine gefilterte, seinem Benutzerprofil entsprechende Sicht des Ausgangsmodells. Die Realisierbarkeit der dargestellten Konzepte und die Anwendbarkeit der Zugriffsarchitektur wurde mit einer konkreten Implementierung im Rahmen der Weiterentwicklung von SupportGIS gezeigt.

Dabei wurde die vorgeschlagene Zugriffsarchitektur erfolgreich in den vorhandenen Systementwurf integriert. Die relevanten Ausschnitte des Systemkernels von SupportGIS sind schließlich am Ende dieser Arbeit dargestellt.
Mit dem vorgestellten Konzept einer Zugriffs- und Berechtigungsstruktur für raumbezogene Datenmodelle können fachlich motivierte Berechtigungsprofile hierarchisch organisierter Benutzer beschrieben werden. Die Definition der Berechtigungsprofile basiert darauf, aus den vorgegebenen Datenschemata Teilaspekte in konsistenter Weise zu extrahieren und für die so beschriebenen Klassen einer Benutzersicht Instanzbeschreibungen nach fachlichen und räumlichen Kriterien festzulegen. Bei der Definition der Profile werden die vom Datenmodell implizierten Abhängigkeiten ausgewertet. Durch das beschriebene Vorgehen können auch komplexe Berechtigungsprofile komfortabel und flexibel definiert und konsistenzerhaltend ausgewertet werden.

Literaturverzeichnis

[1] C.Averdung. *Lösungsmodell zur Unterstützung raumbezogener Planungen durch wissensbasierte Informationsverarbeitung*. Heft 21 der Schriftenreihe des Insituts für Kartographie und Topographie der Rheinischen Friedrich-Wilhelms-Universität Bonn, 1993.

[2] C.Averdung. *GIS im Kontext der Steuerung von Geschäftsprozessen*. Habilitationsschrift. Heft 27 der Schriftenreihe des Insituts für Kartographie und Geoinformation der Rheinischen Friedrich-Wilhelms-Universität Bonn, 2000.

[3] ANSI/X3/SPARC Study Group on Data Base Management Systems. *Interim Report*. FDT ACM SIGMOD bulletin7, No.2, 1975.

[4] N.Bartelme. Geoinformatik - Modelle, Strukturen, Funktionen. Springer Verlag, Hamburg, 1995.

[5] D.E.Bell, L.J.LaPadula. *Secure Computer System: Unified Exposition and MULTICS Interpretation*. Revision 1, US Air Force ESD-TR-75-306, MITRE Report MTR-997, 1976.

[6] A.Beutelspacher. *Geheimsprachen - Geschichte und Techniken*. C.H.Beck, München, 1997 Addison-Wesley Co., Inc., 1995.

[7] Bundesamt für Sicherheit in der Informationstechnik. *IT-Grundschutzhandbuch*. Bundesanzeiger-Verlag. Bonn, 2000.

[8] Bundesamt für Sicherheit in der Informationstechnik. Kriterien für die Bewertung der Sicherheit von Systemen der Informationstechnik. Vorlufige Form der harmonisierten Kriterien, Version 1.2., hrsg. v.d. Europischen Union, Bundesanzeiger-Verlag Kln (1991), ISBN 92-826-3003-X (deutsche Fassung), Juni 1991.

[9] Bundesamt für Sicherheit in der Informationstechnik - E.Brauen, E.v.Essen. *Common Criteria - Gemeinsame Kriterien fr die Prüfung und*

Bewertung der Sicherheit von Informationstechnik. KES, Bundesanzeiger-Verlag Köln, 1996.

[10] Bundesamt für Sicherheit in der Informationstechnik. *E-Government Handbuch - Sicherer Internetauftritt im E-Government*. BSI, Bonn, 2001.

[11] F.G.Sthler(Hrsg).*Datenschutzgesetz Nordrhein-Westfalen, Kommentar*. Dtsch. Gem.-Vlg. Kln, 1988.

[12] Der Bundesbeauftragte fr den Datenschutz (Hrsg.). *Bundesdatenschutzgesetz - Text und Erluterung*. Bonn, 1991.

[13] BVerfGE. *Entscheidungen der Bundesverfassungsgerichts*. Band 65,1. Karlsruhe, 1983.

[14] S.Castano, M.G.Fugini, G.Martella, P.Samarati. *Database Security*. Addison-Wesley/ACMPress,Reading,MA,1995.

[15] D.Chamberlin*A Complete Guide to DB2 Universal Database*. 2nd revised edition, Morgan Kaufmann, August 1998.

[16] O.Costich, Myong Kang, J.N.Froscher. *The SINTRA Data Model: Structure and Operations*. Database Security, Elsevier Science B.V., 1994.

[17] A.B.Cremers, Griefhahn, Hinze. *Deduktive Datenbanken*. Vieweg Verlag, 1994.

[18] C.J.Date. *An introduction to database systems*. Addison-Wesley Co., Inc., 1995.

[19] D.E.Denning, T.F.Lunt. *The SeaView Security Model*. Symposium on Security an Privacy 88, IEEE Computer Society Press, pp 218-233, 1988.

[20] J.Fitzke,C.Rinner,D.Schmidt. *GIS-Anwendungen im Internet*. GIS Geo-Informations-Systeme 10(6), 25-31, 1997.

[21] O.Kyas, M.Campo.*IT Crackdown.Sicherheit im Internet*. mitp-Verlag, Bonn, 2000.

[22] M.Kofler. *R-trees for visualizing and organizing large 3d GIS databases*. Dissertation, TU Graz, 1998.

[23] Kommision der Europäischen Gemeinschaften. *Vorschlag für eine Richtlinie des Europäischen Parlaments und des Rates über die Verarbeitung personenbezogener Daten und den Schutz der Privatsphäre in der elektronischen Kommunikation*. KOM(2000)385, 2000/0189, Brüssel, Juli 2000.

[24] S.Jajodia,B.Kogan,R.Sandhu. *A multilevel-secure object-oriented data model.* Readings in Object-Oriented Systems and Applications, D. Rine, ed., IEEE Computer Society Press, pages 206-215, 1995.

[25] K.Loney, G.Koch.*Oracle8i - Die umfassende Referenz.* Carl Hanser Verlag, München, 2001.

[26] T.F.Lunt. *A Near-Term Design for the SeaView Multilevel Database Systems.* SRI International, Computer Science Laboratory,IEEE, 1988.

[27] B.Oestereich. *Objektorientierte Softwareentwicklung. Analyse und Design mit der Unified Modeling Language.* Oldenbourg Verlag, Mnchen, 1998.

[28] F.Rabitti,E.Bertino,W.Kim,D.Woelk. *A Model of Authorization for Next-Generation Database Systems.* ACM Trans. on Database Systems, Vol. 16, N. 1, pp. 88-131, 1991.

[29] Object Design, Inc. *ObjectStore Management.* ObjectStore Release 6.0, Excelon Corp., Burlington, MA, 2000.

[30] C.Robie, D.Bartels. *A comparison between relational and object oriented databases for object-oriented application development.* POET Software Corporation, 1994.

[31] K.Pommering. *Datenschutz und Datensicherheit.* BI-Wissenschaftsverlag, ISBN 3-411-15171-4, Mannheim, 1991.

[32] G.Saake, C.Türker, I.Schmitt. *Objektdatenbanken - Konzepte, Sprachen, Architekturen.* International Thomson Publishing, Bonn, 1997.

[33] R.Schneider. *Geo-Datenbanksysteme.Eine Speicher- und Zugriffsarchitektur.* Wissenschaftsverlag, 1993.

[34] B.Schneier. *Secrets & Lies. IT-Sicherheit in einer vernetzten Welt.* dpunkt.Verlag, Heidelberg, 2001.

[35] S.Singh. *Geheime Botschaften. Die Kunst der Verschlüsselung von der Antike bis in die Zeiten des Internet.* dtv, München, 2001.

[36] A.Spalka. *A study of the extensibility of logic-based databases with confidentiality-capabilities.* Institut für Informatik, Abt.III, Universität Bonn, 1996.

[37] J.D.Ullman. *Principles of database and knowledge-based systems.* Vol.I,II, Computer Science Press., 1988.

[38] G.Vossen, K.U.Witt. *Das SQL/DS-Handbuch*. Addison-Wesley Co., Bonn, 1988.

[39] A.W.Wood, S.R.Lewis, S.R.Wiseman. *The SWORD Multilevel Secure DBMS*. Defence Research Agency Report 9005, 1992.

[40] SupportGIS. *www.supportgis.de*. Internetportal des Geo-Informationssystems SupportGIS. Informationen zu Entwicklung und Funktionsumfang des Produkts. CPA Geo-Information. Siegburg, 2003.

Anhang A

Grafische Notationen zur Datenbeschreibung

Zur effizienten Modellierung und Dokumentation von Phänomenen der realen Welt und zum Entwurf von Softwarekonzepten bietet sich die Verwendug von grafisch-notierten Beschreibungskonzepten an. Die weiteste Verbreitung und Akzeptanz haben dabei *Entity-Relationship-Diagramme*sowie die *Unified Modeling Language* erlangt. Die wesentlichen Konzepte und Bestandteile werden im Folgenden erläutert.

A.1 Entity-Relationship-Diagramme

Durch *ER-Diagramme* werden *Entity-Relationship-Modelle* dargestellt. Im Mittelpunkt der Betrachtung stehen Entitäten bzw. Entitätsmengen und die Beziehungen zwischen Entitäten. Eine Entität ist ein abgrenzbares identifizierbares Objekt der realen Welt. Es ist entweder physisch existent oder eine gedankliche Abstraktion. In einer Entitätsmenge werden Objekte mit gleichen oder ähnlichen Eigenschaften zusammengefasst. Eine Entitätsmenge stellt eine Abstrahierung einzelner Entitäten dar. Jede Entitätsmenge ist über einen Namen identifizierbar. Der Entitätstyp trägt einen gewählten Namen als Strukturbezeichner einer Entitätsmenge. Dem Entitätstyp werden die relevanten Eigenschaften der Entitäten einer Entitätsmenge zugeordnet. Im ER-Diagramm werden Entitätsmengen durch Rechtecke dargestellt, die den Namen der Entitätsmenge enthalten. Attribute stellen die Merkmale bzw. die Eigenschaften einzelner Entitäten dar. Jedes Merkmal hat spezifische Werte als Ausprägung. Attribute werden mit Kreisen bzw. Ellipsen und dem Attributbezeichner dar-

gestellt. Zwischen Entitäten oder Entitätsmengen bestehen Beziehungen, die wesentliche Eigenschaften dieser Objekten in der realen Welt darstellen.

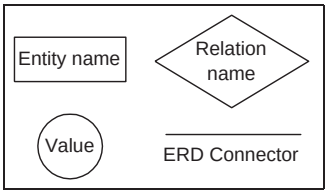

Abbildung A.1: *Basissymbole der ERD-Modellierung*

Beziehungen können unterschiedliche Kardinalitäten haben:

- 1:1-Beziehung: Genau eine Entität steht in Beziehung zu genau einer anderen Entität. Obwohl dieser Beziehungstyp sehr einfach ist, kommt er sehr selten vor, da sich die meissten potenziellen 1:1-Beziehungen auch über direkte Attributierung modellieren lassen. Ein Beispiel hierfür ist: Jedes Kartenblatt DGK5 hat genau eine Blattnummer.
- 1:n-Beziehung: Genau eine Entität steht in Beziehung zu unbestimmt vielen anderen Entitäten, von denen jede in Umkehrbeziehung zu genau der einen Entität steht. Beispiel: Eine Flur besteht aus n Flurstücken - Jedes dieser Flurstücke gehört zu genau einer Flur.
- n:m-Beziehung: Die Beziehung und die Umkehrbeziehung zwischen Entitäten haben eine unbestimmte Kardinalität. Beispiel: Ein Flurstück hat n Nachbar-Flurstücke - Jedes dieser benachbarten Flurstücke hat seinerseits m Nachbarflurstücke.

Ebenso können Beziehungen unterschiedliche Charakteristik haben:

- Aggregation (*„is part of"*): Mittels der Aggregation werden über Beziehungen Entitäten als Summe seiner Einzelteile modelliert. Auf diese Art können komplexe Objekte beschrieben werden. Beispiel: Ein Flurstück ist Teil einer (*„is part of"*) Flur. Die Aggregation besitzt eine hierarchische Charakteristik und ist insofern transitiv. D.h. Wenn (A *„is part of"* B) und (B *„is part of"* C) dann gilt auch (A *„is part of"* C).
- Generalisierung und Spezialisierung (*„is a"*): Werden aus Mengen Teilmengen gebildet so spricht man von Spezialisierung. Werden umgekehrt Mengen zu Obermengen zusammengefasst, so spricht man von Generalisierung. Ebenso wie die Aggregation sind Generalisierung und Spezialisierung hierarchische und transitive Beziehungen. In der übergeordneten

Menge von Entitäten kommen alle Eigenschaften vor, die jedes Objekt der Teilmenge besitzt. Die Entitäten der Teilmenge übernehmen (erben) die Eigenschaften der übergeordneten Menge. Beispiel: Jedes öffentliche Gebäude ist ein (*„is a"*) Gebäude.

Beziehungen zwischen Entitäten werden durch Kanten und Rauten dargestellt.

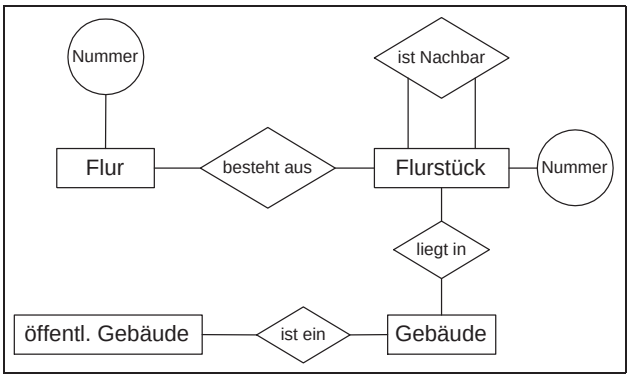

Abbildung A.2: *Beispiel eines ER-Diagramms*

Abbildung A.2 zeigt ein einfaches Beispiel eines Entity-Relationship-Diagramms mit Entitätstypen `Flur`, `Flurstück`, `Gebäude` und `öffentliches Gebäude`. Der Typ `Flur` hat das Attribut `Flurnummer`. Der Typ `Flurstück` hat das Attribut `Flst.Nummer`. Das Diagramm zeigt eine Aggregation: Eine `Flur` besteht aus mehreren `Flurstücken`. Zwischen `Gebäude` und `öffentlichem Gebäude` ist eine Generalisierung bzw Spezialisierung definiert. Außerdem sind die beiden topologischen Beziehungen „`Gebäude` liegt in `Flurstück` " und „`Flurstück` ist Nachbar von `Flurstück` " beschrieben.

A.2 Die *Unified Modeling Language* – UML

Die *Unified Modeling Language* (UML) versteht sich als Weiterentwicklung der bis dahin gängigen *Object Modeling Technique* (OMT). Sie ist, wie der Name schon zum Ausdruck bringt, eine vereinheitlichte Modellierungssprache, die sich an den Paradigmen objektorientierter Modellierung und Programmierung orientiert. Sie unterstützt die Modellbildung anhand einer graphischen Notation unter Berücksichtigung der objektorientierten Grundkonzepte. UML ist daher in besonderer Weise zur Entwicklung objektorientierter Datenmodelle geeignet. Die Generierung von Quellcode einer konkreten objektorientierten Programmiersprache kommt einem automatisierbaren übersetzungsvorgang gleich,

der von UML-unterstützenden CASE-Werkzeugen geleistet werden kann. Damit wird der kreative Teil der Softwareentwicklung von der Programmierung zur Modellierung verschoben. Die folgenden Abbildungen zeigen nur einen kleinen Ausschnitt der Mächtigkeit von UML unter Berücksichtigung der Konzepte, die in der vorliegenden Arbeit zur Darstellung objektorientierter Modelle verwendet wurden.

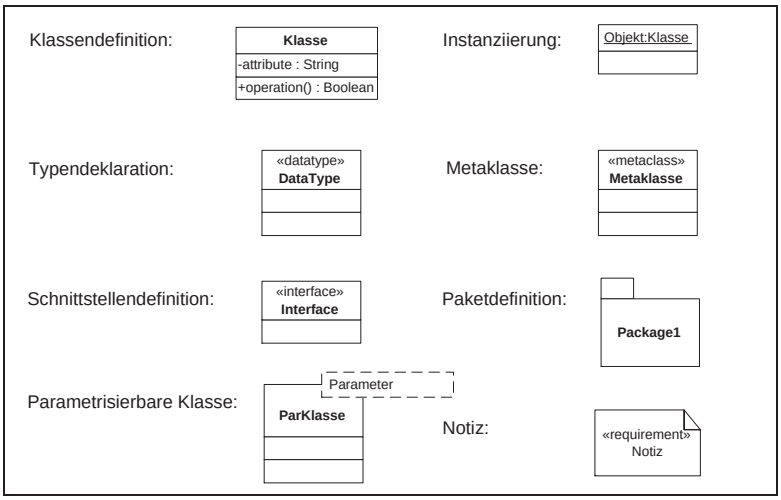

Abbildung A.3: *UML Syntax - Klassen und Objekte*

Abbildung A.3 zeigt die UML Notation mit der Klassen und Objekte mit Attributen und Methoden beschrieben werden.

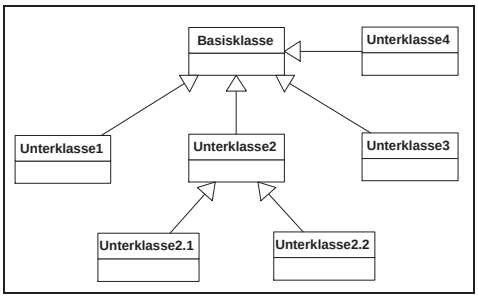

Abbildung A.4: *UML Syntax - Vererbung*

Vererbungen von Basisklassen auf abgeleitete Klassen werden nach der Notation in Abbildung A.4 durch gerichtete Graphen dargestellt, deren Knoten die Klassen repräsentieren und deren Kanten Richtung und Pfad der Vererbung angeben. Dabei „erbt" eine Klasse K_i die Eigenschaften und Methoden einer Klassen K_j, wenn es im Ableitungsgraphen einen Pfad von K_i nach K_j gibt.

In Abbildung A.5 sind Beziehungen bzw. Beziehungstypen zwischen Klassen

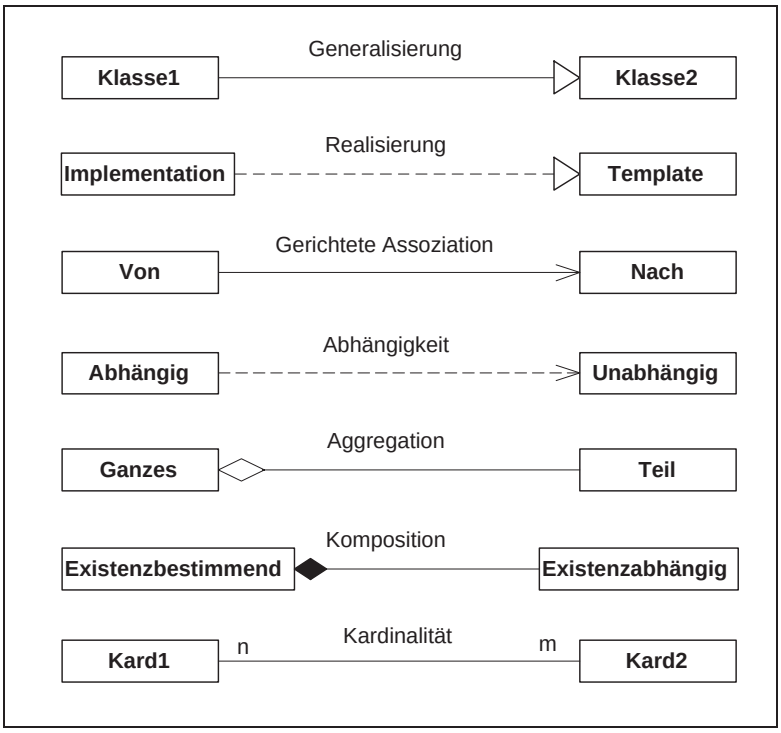

Abbildung A.5: *UML Syntax - Beziehungen*

und Objekten dargestellt. Die dargestellten Notationen beziehen sich auf die UML Version 1.4. Eine vollständige und ausführliche Beschreibung der Syntax und Semantik von UML kann in Ausarbeitungen von Bernd Oesterreich [27] nachgelesen werden. Darüber hinaus hat Bernd Oestereich umfangreiche Untersuchungen über die Mächtigkeit und Entwicklungsmöglichkeiten von UML und deren Werkzeugen angestellt. Die für diese Arbeit generierten UML Diagramme wurden mit den Werkzeugen *Together Control Center* der *TogetherSoft Corporation* und *Microsoft® Visio® 2000* erstellt.

Anhang B

Berechtigungskonzepte kommerzieller DBMS

An dieser Stelle werden die Sicherheitsstrategien und Sicherheitsfeatures gängiger kommerzieller DBMS kurz dargestellt. Die gewählten Datenbanksysteme repräsentieren zwei Klassen von Datenbanken, die zwei völlig unterschiedliche Ansätze in der Modellierungsphilosophie repräsentieren: Unter den weit verbreiteten relationalen Datenbanken wurde Oracle in den Versionen 8i bzw. 9i, sowie die DB2 Universal Database in der Version 7.1 als bedeutendste Vertreter ausgewählt. Für die Geodaten-Modellierung besonders geeignet, jedoch wenig verbreitet ist die Klasse der objektorientierten Datenbanken. Die Sicherheits- und Zugriffsstrategien des Produktes ObjectStore Version 6.0 der Firma eXcelon Corp. werden stellvertretend für die Gruppe der objektorientierten Datenbanksysteme vorgestellt.

B.1 Sicherheitsstrategien von Oracle8i/9i

Als erster Datenbankhersteller erhielt die Firma Oracle im September 1998 ein Zertifikat für evaluierte Sicherheit nach den international gültigen *Common Criteria* CC für das Datenbankprodukt Oracle7 (Version 7.2.2.4.13) auf der CC-Stufe EAL4. Die Sicherheitsmechanismen der Oracle Datenbank orientieren sich an den Leitbegriffen: Benutzerauthentisierung, Zugriffsrechte und Auditmechanismen.
In der Standardausstattung von Oracle8i basiert die Benutzerauthentisierung auf Passwortkontrollen. Um die Sicherheit der Benutzerpasswörter und damit die Sicherheit der Benutzerauthentität zu erhöhen, können Eigenschaften der

Passwörter wie minimale Länge, Gültigkeitsdauer und Passworthistorie konfiguriert werden.
Zugriffsrechte für Benutzer ergeben sich bei Oracle-Datenbanken aus der Vergabe von *Owner*-kontrollierten Privilegien, die mit einem `GRANT` Befehl definiert werden. Jede Datenbankresource (Tabellen, Views, Tablespace, Typen, Trigger, ...) kann eindeutig einem Besitzer (einem Oracle Benutzer) zugeordnet werden. Jeder Oracle Benutzer kann die Zugriffsrechte an den Ressourcen, die sich in seinem Besitz befinden, an andere Benutzer oder an Rollen vergeben. Eine Oracle-Rolle ist eine Ansammlung von Privilegien, die eine Art Benutzerprofil darstellt, das vordefiniert und mehreren Benutzern zugewiesen werden kann. Dadurch reduziert sich der Aufwand und das Risiko fehlerhafter Rechtevergaben, wenn die gleichen Rechte an eine Gruppe von Benutzern vergeben werden sollen. Der Besitzer der Ressourcen legt dabei das Recht fest, Methoden (SQL Kommandos) zum Einsehen (`SELECT`), Einfügen (`INSERT`), Ändern (`UPDATE`) und Löschen (`DELETE`) von Daten auf der Ressource auszuführen. Darüber hinaus berechtigen Systemprivilegien zum Anlegen neuer Ressourcen (`CREATE`) oder zur Weitergabe von Privilegien (`GRANT`). Im Grundzustand einer Oracle-Datenbank sind drei Rollen vordefiniert.

- **CONNECT**: Die CONNECT-Rolle vereint die minimalen Rechte, die ein Benutzer benötigt, um sich bei einer Datenbank anmelden zu können. Die Vergabe weiterer Rechte für Benutzer dieser Gruppe, wie das Auslesen von Tabelleninhalten, wird den Besitzern von Ressourcen überlassen. Die Rolle `CONNECT` wird Benutzern zugewiesen, die keine eigenen Ressourcen anlegen oder verändern sollen, sondern nur als Anwender auf Informationen zugreifen.
- **RESSOURCE**: Die RESSOURCE-Rolle enthält erweiterte Berechtigungen zum Anlegen eigener Datenbankressourcen wie Tabellen, Views oder Trigger.
- **DBA**: Mit der DBA-Rolle werden dem Benutzer uneingeschränkt alle Systemprivilegien zuerkannt . Über diese Rolle sollten in einer Datenbank nur sehr wenige Benutzer verfügen.

Einmal vergebene Berechtigungen können mit dem Kommando `REVOKE` wieder entzogen werden.
Über diese Grundmechanismen der Sicherheit hinaus unterstützt Oracle für gehobene Sicherheitsansprüche die sogenannten *Advanced Security Options*. Diese stellen erweiterte Sicherheitsmechanismen zur Verfügung:

- Datenverschlüsselung mit RC4 und DES.

- Kryptographische Prüfsummen mit MD5.
- Zertifizierte Benutzer-Authentisierung.
- Zertifizierte Client/Server-Authentisierung.
- SSL Verschlüsselung.
- Unterstützung von *Secure Electronic Transaction* - SET.

Weitere optionale Zusatzprodukte zur Oracle Datenbank erweitern die Authentisierungsmechanismen um *Besitz-und-Wissen-* Konzepte (z.B.*Smardcard*) und Biometrische Verfahren (z.B.*Fingerprint*) zur Personenidentifizierung. Ausführliche Details der Sicherheitsstrategien von Oracle wurden von Loney und Koch [25] zusammengestellt.

B.2 Sicherheitsstrategien DB2 Universal Database v.7.1

Die Sicherheitsstrategien der *DB2 Universal Database* von IBM [15] sind in Schichten organisiert, die jeweils eine Sammlung von Sicherheitsfunktionen repräsentieren. Diese muss jeder Benutzer zur Kommunikation mit der Datenbank und deren Daten passieren.
Die äußerste Schicht der Sicherheit wird mit der Authentisierung der Benutzer realisiert. Diese ist vollständig ausgelagert und wird zum Beispiel durch Passwort-Kontrolle vom Betriebssystem oder dem Produkt eines anderen Drittanbieters übernommen. Die Benutzer-Authentisierung kann damit jederzeit aktuellen Entwicklungen in diesem Bereich angepasst werden, ohne dass das DBMS ausgetauscht werden muss. Durch die Konfiguration der *DB2 Authentication Parameter* kann die Authentisierung mit unterschiedlichen Optionen durchgeführt werden:

- **Server-Authentisierung:** Die Anmeldung eines Benutzers mit Name und Passwort bei einer DB2 Datenbank veranlasst das DB2 Managementsystem, eine Funktion des Server-Betriebssystems aufzurufen, die eine Validierung des Benutzers vornimmt. Die Server-Authentisierung kann zusätzlich mit der Option `SERVERENCRYPT` aufgerufen werden, wodurch die Benutzerinformationen vor der Übertragung zum Server verschlüsselt werden (56-Bit single DES).

- **Client Authentisierung:** Die Authentisierung des Benutzers wird vom Client übernommen. Mit den Server Optionen `TRUSTALLCLNTS` und `TRUSTCLNTAUTH` kann festgelegt werden, wie der Server sich bei Client-Anfragen - im Fall von Client-Authentisierung - verhalten soll. Dabei ist entscheidend, ob der Client auf einem vertrauenswürdigen (*trusted client*) oder einem nicht vertrauenswürdigen Betriebssystem (z.B. Windows 9x) läuft.

- **Kerberos-Authentisierung:** Mit der DB2 UDB v.7.1 wurde für Installationen unter Windows 2000 die Kerberos Authentisierung eingeführt. Eine spezielle Umgebung sorgt dafür, dass ein Client nach erstmaliger Anmeldung mit Kerberos Option alle DB2-Server der definierten Kerberos Umgebung anfragen kann, ohne sich erneut authentisieren zu müssen. Voraussetzung hierfür ist, dass der Client und alle Server unter dem Betriebssystem Windows 2000 (oder höher) laufen.

Nach erfolgreicher Authentisierung des Benutzers folgt die Schicht der Autorisierung, also der Zuweisung von speziellen Benutzerrechten. Die DB2 UDB unterscheidet dabei zwischen *Authorities* und *Privileges*. *Authorities* bezeichnen vordefinierte Mengen von Privilegien. Die DB2 *Authorities* sind vergleichbar mit den Rollen in ORACLE. In jeder DB2 UDB sind die *Authorities* `SYSADM` (System Administration), `DBADM` (Datenbank Administration), `SYSCTRL` (System Kontrolle) und `SYSMAINT` (System Pflege) vordefiniert. *Privileges* definieren jeweils einzelne Zugriffsrechte von Benutzern oder Benutzergruppen an Datenbankressourcen. Die Zugriffsrechte beschreiben dabei jeweils einzelne Datenbankoperationen, wie `CONNECT` (Verbindung zu einer Datenbank), `CREATETAB` (Anlegen von Tabellen), `SELECT` (Selektion) oder `UPDATE` (Daten ändern).
Auf der innersten Zugriffsschicht befinden sich die Methoden der Zugriffskontrolle, die für die Auswertung und Organisation der Zugriffsrechte verantwortlich sind. Die DB2 Methoden der Zugriffskontrolle bedienen sich verschiedener Datenbanktechniken:

- **Zugriffskontrolle mit Packages**: Eine Sammlung von einem oder meheren SQL-Kommandos werden als *Package* bezeichnet. *Packages* können also SQL-basierte Prozeduren definieren. Die Ausführung solcher Prozeduren kann als *Privilege* einem Benutzer erlaubt werden, auch wenn dieser für einzelne SQL-Kommandos der Prozedur keine Berechtigung besitzt. Nur der Benutzer, der ein *Package* definiert, muss für jedes einzelne Kommando der Prozedur die Berechtigung besitzen.

- **Zugriffskontrolle mit Views**: Durch die Definition von *Views* können Ausschnitte von Relationen nach Zeilen, Spalten oder auf der Grundlage

von Anfragen festgelegt werden. Um die Rechte eines Benutzers an einer Relation einzuschränken, können entsprechende *Views* definiert werden, für die der Benutzer dann Zugriffsrechte erhält.

- **Zugriffskontrolle mit Triggern** : Mit der Definition von Triggern können Sicherheitsfunktionen höherer Komplexität implementiert werden. Mit Triggern werden Aktionen festgelegt, die durch bestimmte Ereignisse ausgelöst werden und je nach Definition vor, während oder nach dem auslösenden Ereignis abgearbeitet werden. Die Verwendung von Triggern eröffnet damit auch die Möglichkeit, zusätzliche Sicherheitsüberprüfungen durchzuführen, bevor sicherheitskritische Transaktionen ausgeführt werden - beispielsweise eine erneute Eingabe des Benutzer-Passwortes vor der Annahme eines `UPDATE` Kommandos.

Als zusätzliche Sicherheitsstandards unterstützt DB2 UDB diverse *Audit* Funktionalitäten zur Protokollierung und Überwachung von Datenbankzuständen und Benutzeraktionen.

B.3 Sicherheitsstrategien von ObjectStore 6.0

Das Produkt ObjectStore der Firma eXcelon Corp. [29] ist eine vollständig objektorientierte Datenbank, die auf der Anwenderebene Schnittstellen verschiedener objektorientierter Programmiersprachen, wie C++ oder Java anbietet. Diese Schnittstelle integriert sich vollständig in die gewählte Programmiersprache, d.h.: Klassen und Instanzen werden in der vertrauten Syntax definiert und über Klassenhierarchien und überladene Operatoren als persistent in der Datenbank oder transient im Arbeitsspeicher abgelegt. Für den Anwender verhalten sich transiente und persistente Objekte identisch.
ObjectStore unterscheidet zwei Typen von Datenbanken:*File-Database* und *Rawfs -Database* (*Raw File System*). Die Unterscheidung ist relevant hinsichtlich der Sicherheitsstrategie. ObjectStore *File-Databases* werden auf Seiten des Datenbankservers physikalisch in das Dateisystem des Betriebssystems integriert und von diesem als Systemressource verwaltet. *Rawfs-Databases* werden vom ObjectStore Server in einem eigenen Dateisystem verwalten. Pysikalisch ist dies ein Bereich auf einem Speichermedium, der ausschliesslich der Kontrolle des ObjectStore Servers unterliegt. Aus Aspekten der Sicherheit haben *Rawfs-Databases* den Vorteil, dass der *Transaction Log* in der Speicherverwaltung von ObjectStore verborgen ist und damit nicht vom Betriebssystem aus zerstört oder manipuliert werden kann. Das *Raw File System* ist vom Betriebssystem aus weder sichtbar, noch zugreifbar.

Jede ObjectStore Datenbank (*File* oder *Rawfs*) ist in Segmente eingeteilt, deren Anzahl und Grösse grundsätzlich nicht beschränkt ist. Durch die Verwendung von Segmenten wird der Inhalt einer Datenbank in strukturelle oder fachliche Einheiten unterteilt und bei geeignetem Design, die Leistungsfähigkeit und Verfügbarkeit erhöht (*Paging*-Verfahren).
Die ObjectStore Zugriffsmethoden setzen hierarchisch auf drei Ebenen an: Verzeichnis, Datenbank und Segment. Für alle drei Zugriffsebenen unterscheidet die Sicherheitspolitik von ObjectStore die drei Benutzerkategorien *owner*, *primary group* und *default group*. Die Kategorie *owner* bezeichnet den Eigentümer eines Verzeichnisses, einer Datenbank oder eines Segments. Die Kategorie *primary group* ist eine vom *owner* ausgezeichnete Gruppe von Benutzern, denen besondere Rechte am Verzeichnis, an der Datenbank oder an einem Segment zugestanden und zugewiesen werden. In der Kategorie *default group* sind alle übrigen ObjectStore Benutzer zusammengefasst. Benutzer dieser Gruppe müssen in der Regel nur selten oder nie auf die betroffenen Daten zugreifen und werden daher mit minimalen Rechten ausgestattet. Der *owner* vergibt auf den drei Zugriffsebenen die Rechte `ostore::read` (lesendes Zugriffsrecht), `ostore::write` (schreibendes Zugriffsrecht) oder `ostore::noaccess` (kein Zugriffsrecht). Handelt es sich bei der Datenbank um eine *File*-Datenbank, so werden diese Rechte für die Ebenen Verzeichnis und Datenbank vom Betriebssystem verwaltet und dominiert. Die Hierarchie der Zugriffsebenen definiert die Abhängigkeiten: Lesender- bzw. schreibender- Zugriff auf einem Segment kann einem Benutzer nur dann erteilt werden, wenn dieser auch Lese- bzw. Schreibberechtigung auf der Datenbank hat, die das Segment enthält. Eine Applikation kann eine Datenbank nur dann zum Lesen bzw. Schreiben öffnen, wenn der ausführende Benutzer Lese- bzw. Schreibberechtigung an der Datenbank und an dem Verzeichnis hat, in der die Datenbank liegt. Befindet sich ein Segment in lesendem Zugriff durch einen Benutzer, so ist dieses Segment im Zustand *read lock* : Auf das Segment kann durch andere Benutzer weiterhin lesend zugegriffen werden. Schreibender Zugriff ist aber während des *read lock* nicht möglich. Bei schreibendem Zugriff auf ein Segment ist dieses im Zustand *write lock* : In diesem Zustand ist auch ein lesender Zugriff auf das Segment nicht mehr möglich. Durch diesen *Locking* Mechanismus werden Segmente als kleinste Verwaltungseinheit davor bewahrt, undefinierte oder unvereinbare Zustände anzunehmen.

Anhang C

Abkürzungen

AAA:	Authentifizierung, Autorisierung und Access
ALB:	Automatisiertes Liegenschaftsbuch
ALK:	Automatisierte Liegenschaftskarte
ALKIS:	Amtliches Liegenschaftskataster-Informationssystem
API:	Application Programing Inferface
ATKIS:	Amtliches Topographisch Kartographisches Informationssystem
ASP:	Active Server Pages
B2B:	Business-to-Business (E-Commerce)
B2B:	Business-to-Consumer (E-Commerce)
BSI:	Budesamt für Sicherheit in der Informationstechnik
CAD:	Computer Aided Design
CC:	Common Criteria
DAC:	Discretionary Access Control
DB:	Datenbank
DBMS:	Datenbank Management System
DGK5:	Deutsche Grundkarte 1:50000
DoD:	Department of Defense (US-Verteidigungsministerium)
DTD:	Document Type Definition
E-Commerce:	Electronic-Commerce
ER:	Entity Relationship
FTP:	File Transfer Protocol
GDI NRW:	Geodaten-Infrastruktur NRW
GI:	Geoinformation
GIS:	Geoinformationssysteme, Geografische Informationssysteme
GML:	Geography Markup Language
HTML:	HyperText Markup Language
HTTP:	HyperText Transfer Protocol
ITSEC:	Information Technology Security Evaluation Criteria
IVBB:	Informationsverbund Bonn-Berlin

LAN:	Local Area Network
MAC:	Mandatory Access Control
OO:	Objekt-orientiert
OGC:	OpenGIS Consortium
ODBC:	Open Database Connectivity
OODB:	Objektorientierte Datenbank(en)
OODBMS:	Objektorientiertes Datenbank-Managementsystem
OMT:	Object Modeling Technique
RDB:	Relationale Datenbank
RDBMS:	Relationales Datenbank-Managementsystem
SSL:	Secure Socket Layer
TCB:	Trusted Computing Base
TCSEC:	Trusted Computer Security Evaluation Criteria
UML:	Unified Modeling Language
URL:	Uniform Resource Locator
WAN:	Wide Area Network
WWW:	World Wide Web
XML:	Extended Markup Language
XSD:	XML Schema Definition

Lebenslauf

Persönliche Daten

Name:	René Thiele
Geburtstag:	30.06.1970
Geburtsort:	Lüchow (Wendland)
Familienstand:	verheiratet
	2 Kinder

Schulbildung

1977 - 1980:	Grundschule Donauwörth (Bayern)
1980 - 1981:	Grundschule Ahrweiler (Rheinland-Pfalz)
1981 - 1991:	Are-Gymnasium Bad Neuenahr (Rheinland-Pfalz)
	Abitur im Juni 1991

Wehrdienst

Jul. 1991 - Jun. 1992:	1. Fernmeldebataillon 330, Koblenz

Studium

Okt. 1992 - Mär. 1997	Studium der Informatik an der
	Rheinischen Friedrich-Wilhelms-Universität Bonn
	Abschluss: Diplom am 18.März 1997

Berufliche Stationen

Apr. 1997 - Dez. 1997	Wissenschaftlicher Mitarbeiter am
	Institut für Kartographie und Geoinformation,
	Universität Bonn
Seit Jan. 1998	Mitarbeiter der Firma
	CPA Geo-Information, Siegburg

Danksagung

Die vorliegende Dissertation entstand im Rahmen meiner wissenschaftlichen Tätigkeit am Institut für Kartographie und Geoinformation der Rheinischen Friedrich-Wilhelms-Universität zu Bonn. Unterstützung und Förderung erfuhr ich dabei durch die Firma CPA Geo-Information aus Siegburg.
Für die freundliche und unermüdliche Betreuung und Unterstützung während dieser Zeit danke ich ganz besonders Herrn Professor Dr.-Ing. Dieter Morgenstern, meinem akademischen Lehrer und Hauptreferenten dieser Dissertation. Desweiteren gilt mein Dank Herrn Professor Dr. rer. nat. Lutz Plümer für die freundliche Übernahme des Korreferats. Herr Plümer hat durch viele fachliche Gespräche und seine wertvollen Hinweise wesentlich zum Gelingen dieser Arbeit beigetragen.
Mein weiterer Dank richtet sich an Herrn PD Dr.-Ing Christoph Averdung für seine langjährige Förderung und Unterstützung sowie für die Schaffung eines überaus angenehmen und kreativen Arbeitsumfeldes.
Für die unermüdliche Hilfsbereitschaft in allen Angelegenheiten jenseits der wissenschaftlichen Fragestellungen danke ich ganz besonders Frau Helga Koch und Herrn Dieter Brandenburg.
Schließlich danke ich allen Mitarbeitern des Instituts für Kartographie und Geoinformation sowie den Mitarbeitern der Firma CPA Geo-Information, die immer für anregende Diskussionen zur Verfügung standen und damit wesentlich zum Gelingen dieser Dissertation beigetragen haben.