



Dokumentation
zur
Modellierung der Geoinformationen
des amtlichen Vermessungswesens
(GeoInfoDok)

Hauptdokument

Version 5.1
Stand: 31.03.2006

Arbeitsgemeinschaft der Vermessungsverwaltungen
der Länder der Bundesrepublik Deutschland (AdV)

Inhaltsverzeichnis

AUFBAU, INHALT UND ZIEL	5
1.1 AUSGANGSSITUATION, MOTIVE UND ZIELVORSTELLUNG	5
1.2 GRUNDDATENBESTAND, OBJEKTARTENKATALOGE, VERSIONIERUNG UND MIGRATION	6
2 DAS AFIS-ALKIS-ATKIS-REFERENZMODELL	8
3 DAS KONZEPTUELLE MODELL	12
3.1 GRUNDSÄTZE DER MODELLIERUNG	12
3.1.1 <i>Normen und Standards</i>	12
3.1.2 <i>Modellierungs- und Beschreibungssprache</i>	12
3.2 AUFGABE UND STRUKTUR	13
3.3 DAS AFIS-ALKIS-ATKIS-BASISSCHEMA	15
3.3.1 <i>Objektbildungsgrundsätze</i>	17
3.3.2 <i>Attribute</i>	20
3.3.3 <i>Beziehungen</i>	20
3.3.4 <i>Raumbezug, Geometrie</i>	22
3.3.4.1 Grundsätze	22
3.3.4.2 Objekte mit einfacher Topologie	25
3.3.4.3 Objekte gemeinsam genutzter Geometrie	26
3.3.4.4 Objekte mit unabhängiger Geometrie	27
3.3.4.5 Raumbezugssystem, Koordinaten	28
3.3.5 <i>Signaturierung, Präsentationsobjekte</i>	29
3.3.6 <i>Kartengeometrieobjekte</i>	34
3.3.7 <i>Punktmengenobjekte</i>	35
3.3.8 <i>Externe Code-Listen</i>	35
3.3.9 <i>Identifikatoren, Verknüpfungen</i>	36
3.4 HISTORIE, VERSIONSKONZEPT	38
3.5 QUALITÄTS- UND METADATEN	44
3.6 OBJEKTARTENKATALOG	45
3.7 PROZESSE, VORGÄNGE UND AKTIVITÄTEN	46
3.7.1 <i>Grundsätze</i>	46
3.7.2 <i>Vorgang und Aktivität</i>	48
3.7.3 <i>Prozesse des AFIS-ALKIS-ATKIS-Anwendungsschemas</i>	49
3.7.3.1 Erhebungsprozess	49
3.7.3.2 Qualifizierungsprozess	49
3.7.3.3 Führungsprozess	49
3.7.3.4 Benutzungsprozess	50
3.7.3.5 Transferprozess	51
3.8 PROJEKTSTEUERUNG	51
3.8.1 <i>Antrag</i>	51
3.8.2 <i>Projektsteuerungskatalog</i>	54
3.8.3 <i>Meilenstein</i>	54
3.9 GRUNDSÄTZE FÜR DIE VERWENDUNG DES FILTERENCODINGS	55
4 KATALOGE DES BASISSCHEMAS UND DES VERSIONIERUNGSSCHEMAS	64
5 FACHSPEZIFISCHE ANWENDUNG DES BASISSCHEMAS	65
5.1 DAS GEMEINSAME AFIS-ALKIS-ATKIS-FACHSCHEMA	65
5.2 ERLÄUTERUNGEN ZU AFIS	65
5.3 ERLÄUTERUNGEN ZU ALKIS	65
5.4 ERLÄUTERUNGEN ZU ATKIS-DIGITALE LANDSCHAFTSMODELLE	65
5.5 ERLÄUTERUNGEN ZU ATKIS-DIGITALE GELÄNDEMDELLE	65
6 AFIS-KATALOGWERKE	66
6.1 AFIS-OBJEKTARTENKATALOG	66
6.2 PROZESSE IN AFIS	66

6.3	AFIS-SIGNATURENKATALOG	66
7	ALKIS-KATALOGWERKE	67
7.1	ALKIS-OBJEKTARTENKATALOG	67
7.2	PROZESSE IN ALKIS	67
7.3	ALKIS-SIGNATURENKATALOG	67
8	ATKIS-KATALOGWERKE	68
8.1	ATKIS-OBJEKTARTENKATALOGE	68
8.1.1	<i>ATKIS-Objektartenkatalog für das Digitale Basis Landschaftsmodell (ATKIS-OK Basis-DLM)</i> 68	
8.1.2	<i>ATKIS-Objektartenkatalog für das Digitale Landschaftsmodell 50 (ATKIS-OK50)</i>	68
8.1.3	<i>ATKIS-Objektartenkatalog für das Digitale Landschaftsmodell 250 (ATKIS-OK250)</i>	68
8.1.4	<i>ATKIS-Objektartenkatalog für das Digitale Landschaftsmodell 1000 (ATKIS-OK1000)</i>	68
8.2	ATKIS-OBJEKTARTENKATALOG FÜR DAS DIGITALE GELÄNDEMOMDELL	68
8.2.1	<i>ATKIS-Objektartenkatalog für das Digitale Geländemodell 2 (ATKIS-OK DGM2)</i>	68
8.2.2	<i>ATKIS-Objektartenkatalog für das Digitale Geländemodell 5 (ATKIS-OK DGM5)</i>	69
8.2.3	<i>ATKIS-Objektartenkatalog für das Digitale Geländemodell 25 (ATKIS-OK DGM25)</i>	69
8.2.4	<i>ATKIS-Objektartenkatalog für das Digitale Geländemodell 50 (ATKIS-OK DGM50)</i>	69
8.3	ATKIS-SIGNATURENKATALOGE	69
9	METADATENKATALOG	70
10	DAS EXTERNE MODELL, DATENAUSTAUSCH	71
10.1	NORMBASIERTER AUSTAUSCHSCHNITTSTELLE (NAS)	71
10.1.1	<i>Normen und Standards</i>	71
10.1.2	<i>Kodierungsprozess</i>	75
10.1.3	<i>NAS Encoding Rules</i>	77
10.1.3.1	Voraussetzungen	78
10.1.3.2	Eingangsdatenstruktur	81
10.1.3.3	Ausgabedatenstruktur	82
10.1.3.4	Schema-Abbildungsregeln	82
10.1.3.5	Abbildungsregeln für Instanzen	102
10.1.4	<i>XML-Schemata für Normteile und das externe Basisschema</i>	103
10.1.5	<i>XML-Schema für AFIS-ALKIS-ATKIS</i>	103
10.1.6	<i>GML_Profil für die NAS</i>	103
10.2	NAS-OPERATIONEN	104
10.2.1	<i>Funktionsumfang</i>	105
10.2.2	<i>Codierung der Selektionskriterien</i>	106
10.2.3	<i>Einrichten und Fortführen von Primärnachweisen</i>	110
10.2.4	<i>Anfordern von Ausgaben</i>	114
10.2.5	<i>Ausgabe von Benutzungsdaten</i>	115
10.2.6	<i>Führung von Sekundärnachweisen</i>	115
10.2.7	<i>Sperren und Entsperren von Objekten</i>	116
10.2.8	<i>Reservierungen</i>	116
10.2.9	<i>Übermittlung von Protokollinformationen</i>	116
10.2.10	<i>Ermitteln der Eigenschaften einer Bestandsdatenhaltung</i>	116
10.3	AUSZUTAUSCHENDE EINHEITEN	116
10.4	IMPLIZITE FUNKTIONALITÄT	117
10.4.1	<i>Implizite Funktionalität eines Systems für den Primärnachweis</i>	117
10.4.1.1	Geometriebehandlung	119
10.4.2	<i>Implizite Funktionalität eines Systems für den Sekundärnachweis</i>	121
10.5	NUTZERBEZOGENE BESTANDSDATENAKTUALISIERUNG (NBA)	121
10.5.1	<i>Fachliche Anforderungen</i>	121
10.5.2	<i>Modellierung</i>	122
10.5.2.1	Abgabe von Änderungsdaten	124
10.5.2.2	Abgabe von Differenzdaten	125
10.5.2.3	Portionierung von NBA-Daten	125
10.5.2.4	Quittierung von NBA-Lieferungen	129
11	KOORDINATENREFERENZSYSTEME UND MABEINHEITEN	130

12	QUALITÄTSSICHERUNG	131
12.1	ADV-QUALITÄTSSICHERUNGSSYSTEM	131
12.2	QUALITÄTSSICHERUNGSMODELL.....	131
12.3	SYSTEMATIK UND DOKUMENTATION DER QUALITÄTSSICHERUNG	133
13	GLOSSAR, ABKÜRZUNGEN UND ABBILDUNGSVERZEICHNIS	134
13.1	FACHBEGRIFFE UND IHRE ENGLISCHE ÜBERSETZUNG	134
13.2	ABKÜRZUNGSVERZEICHNIS.....	147
13.3	ABBILDUNGSVERZEICHNIS	148
	ANLAGE: GEOMETRISCHE UMSETZUNG DER AAA-REO MIT GML	150

1 Aufbau, Inhalt und Ziel

Die GeoInfoDok besteht aus einem Hauptdokument (dieses Dokument), dem AAA-Anwendungsschema und den daraus abgeleiteten oder im Zusammenhang stehenden Dokumenten (z.B. Objektartenkataloge, NAS, Koordinatenreferenzsysteme). Die Gliederung dieses Dokuments entspricht dabei der Kapitelnummerierung der abgeleiteten Kataloge. Im Folgenden werden nur Hinweise auf die separaten Dokumente gegeben. Die gesamte GeoInfoDok wird in der aktuellen und den versionierten Ständen unter www.adv-online.de veröffentlicht und gepflegt. Einige Kapitel werden nicht als separate Dokumente veröffentlicht und werden nur hier im Hauptdokument behandelt. Das Hauptdokument enthält neben vielen Erläuterungen und Modellierungsgrundsätzen wichtige Hinweise zur Spezifikation der NAS (Kapitel 10). Die Festlegungen dieses Dokuments haben daher **normativen Charakter** und ergänzen die Festlegungen im AAA-Anwendungsschema.

1.1 Ausgangssituation, Motive und Zielvorstellung

Die Vermessungs- und Katasterverwaltungen der Bundesländer haben die Aufgabe, raumbezogene Basisdaten (Geobasisdaten) für Verwaltung, Wirtschaft und private Nutzer zu liefern, und zwar zunehmend in digitaler Form. Hierauf wurde bereits sehr früh reagiert und begonnen, die Daten des Liegenschaftskatasters in den Projekten ALK (Automatisierte Liegenschaftskarte) und ALB (Automatisiertes Liegenschaftsbuch) sowie die Daten der Topographischen Landesaufnahme im Projekt ATKIS (Amtliches Topographisch-Kartographisches Informationssystem) deutschlandweit einheitlich digital zu erfassen und zur Verfügung zu stellen. In den meisten Bundesländern ist durch Kabinettsbeschluss geregelt, dass die ALK- und ATKIS-Daten als Basis für andere Fachinformationssysteme (FIS) zu verwenden sind.

Die Konzepte, nach denen ALB, ALK und ATKIS aufgebaut worden sind, stammen aus den 70er bzw. 80er Jahren des 20. Jahrhunderts. Sie dienen noch heute als Plattform für den Aufbau der entsprechenden Geobasisdatenbestände. Außerdem wurden daneben weitere umfangreiche digitale Datenbestände nach jeweils eigenen Konzepten der Bundesländer aufgebaut, z. B. Digitale Orthophotos, Rasterdaten der topographischen Landeskartenwerke und Digitale Höhenmodelle.

Vor dem Hintergrund der sich schnell entwickelnden Technik, der inzwischen umfangreichen Erfahrungen der Hersteller bei der Datenerfassung und den sich aus der Datennutzung ergebenden veränderten Anforderungen seitens der Anwender war es erforderlich, diese Konzepte zu überprüfen und weiterzuentwickeln.

Die bisherigen Informationssysteme ALK und ALB werden deshalb zukünftig integriert im Informationssystem **ALKIS** (Amtliches Liegenschaftskatasterinformationssystem) geführt. Darüber hinaus wurde eine formelle, inhaltliche und semantische Harmonisierung mit ATKIS vorgenommen.

Die **Digitalen Geländemodelle** (DGM) werden in ATKIS nicht wie bisher im Objektbereich Relief einem spezifischen Digitalen Landschaftsmodell (DLM) zugeordnet, sondern als eigenständiger Bestandteil unter den objektstrukturierten Daten ausgewiesen. Hiermit wird, ähnlich wie den Festpunkten der Grundlagenvermessung, die universelle Verwendbarkeit des DGM als eigenständiger Datenbestand verdeutlicht und die Möglichkeit zur Erzeugung von kombinierten Datenbeständen oder Erzeugnissen unter Verwendung von Daten aus anderen Produktgruppen besser herausgestellt.

Zu den Geoinformationen des amtlichen Vermessungswesens gehören auch die Informationen zu den Festpunkten. Da die Festpunkte weder originär zur ALK noch zu ATKIS gehören, wird deren Modellierung nunmehr in einem eigenen Informationssystem **Amtliches Festpunktinformationssystem** (AFIS) durch einen eigenen Objektartenkatalog vorgenommen.

Unter der Überschrift *Dokumentation zur Modellierung der Geoinformationen des amtlichen Vermessungswesens* werden die AdV-Projekte AFIS, ALKIS und ATKIS mit ihren länderübergreifend festgelegten Eigenschaften in durchgängiger Form gemeinsam beschrieben. Sie werden in einem **gemeinsamen Referenzmodell** miteinander in Beziehung gebracht und im Rahmen dieser Dokumentation in den weiteren Kapiteln als **gemeinsames Anwendungsschema für AFIS, ALKIS und ATKIS** beschrieben.

Das gemeinsame Anwendungsschema sieht die Erfassung und Führung von **Metadaten und Qualitätsdaten** gemäß der ISO-Spezifikationen vor.

1.2 Grunddatenbestand, Objektartenkataloge, Versionierung und Migration

Überregionale Nutzer und GIS-Industrie fordern im Hinblick auf die Inhalte und die Strukturierung der Geobasisdaten sowie aus Gründen der Wirtschaftlichkeit die Festlegung eines bundesweit einheitlichen Grunddatenbestandes. Aus der ganzheitlichen Sicht auf das amtliche Vermessungswesen sollen die Grunddatenbestände von AFIS, ALKIS und ATKIS zu diesem Grunddatenbestand der Geodaten des amtlichen Vermessungswesens zusammengeführt werden. Der **Grunddatenbestand** ist der von allen Vermessungsverwaltungen der Länder der Bundesrepublik Deutschland in AFIS, ALKIS und ATKIS bundeseinheitlich zu führende und dem Nutzer länderübergreifend zur Verfügung stehende Datenbestand. Dazu gehören auch die entsprechenden Metadaten. Der Grunddatenbestand

ist aus fachlicher Sicht festzulegen und wird im AAA-Fachschemata gekennzeichnet. Eine spätere Erweiterung des Grunddatenbestandes ist zu erwarten.

Die **Objektartenkataloge** des Liegenschaftskatasters und der Landesvermessung wurden im Interesse einer möglichst einheitlichen Realweltmodellierung semantisch harmonisiert. Die Harmonisierung hat sowohl Vorteile für die interne Nutzung als auch im externen Bereich. Sie orientiert sich an den bisherigen Katalogen (Muster-OBAK, Verzeichnis der Nutzungsarten, ATKIS-OK).

Im Zusammenhang mit der Beschreibung des Verfahrens zur **Nutzerbezogenen Bestandsdatenaktualisierung** (NBA) wird ein Versionskonzept eingeführt. Länder, die eine Historienverwaltung im Sinne der von der AdV beschlossenen Stufenlösung für ALKIS einsetzen, legen ihrer Modellierung und den darauf basierenden Funktionalitäten einer Historienverwaltung genau dieses um das Versionskonzept erweiterte Anwendungsschema zu Grunde. Hinsichtlich der **Historienverwaltung** wird für ATKIS die stichtagsbezogene Speicherung der jeweiligen Datenbestände für ausreichend erachtet.

Für die **Migration** aus den bisherigen Nachweisen ist ein grundsätzliches Vorgehen in Form eines Stufenkonzeptes vorgesehen. Die Detailausarbeitung von Migrationskonzepten ist länderspezifisch auszuführen. Eine Rückmigration in die Schnittstellen der bisherigen Systeme für eine übergangsweise Versorgung der Nutzer mit Daten ist noch für einen längeren Übergangszeitraum denkbar. Das Migrationskonzept besitzt nur temporäre Bedeutung und wird deshalb nicht in die Gesamtdokumentation aufgenommen.

2 Das AFIS-ALKIS-ATKIS-Referenzmodell

Das AFIS-ALKIS-ATKIS-Referenzmodell hat die Aufgabe, die nach dieser Dokumentation beschriebenen Datenbestände mit ihren Beziehungen im Kontext darzustellen. Ziel dabei ist es,

- Komponenten zu identifizieren,
- die Modularisierung zu erleichtern,
- die Verbindung zu den Normen aufzuzeigen und
- Doppelarbeit innerhalb der Komponenten zu vermeiden.

AFIS ist das Amtliche Festpunktinformationssystem und enthält beschreibende und darstellende Daten zu folgenden Produktgruppen:

- AFIS-Bestandsdaten,
- digitale AFIS-Auszüge sowie
- analoge AFIS-Auszüge.

ALKIS ist das Amtliche Liegenschaftskatasterinformationssystem und enthält liegenschaftsbeschreibende und -darstellende Daten in folgenden Produktgruppen:

- ALKIS-Bestandsdaten,
- digitale ALKIS-Auszüge sowie
- analoge ALKIS-Auszüge.

ATKIS ist das Amtliche Topographisch-Kartographische Informationssystem der deutschen Landesvermessung. ATKIS beschreibt die Landschaft mit unterschiedlichen Anwendungszielen in folgenden Produktgruppen:

- digitale Landschaftsmodelle (ATKIS-DLM und Zusatzdaten) einschließlich digitaler Geländemodelle (DGM),
- digitale topographische Karten (DTK)
- analoge Auszüge aus den DTK (Topographische Karten).

Die Inhalte, Strukturen und Herstellungsvorschriften der Produkte des Referenzmodells werden auf der **Regelungsebene** durch die Objektartenkataloge (OK) und Signaturenkataloge (SK) definiert. Diese umfassen:

- Vorschriften zur Abbildung der Informationen der Festpunkte, des Liegenschaftskatasters und der Topographie,
- Vorschriften zur Bildung von Präsentations- und Kartengeometrieobjekten (Zusatzdaten),

- Vorschriften zur Darstellung und kartographischen Gestaltung der Objekte.
- Vorschriften zur Ausgestaltung von analogen Auszügen

Die Erfassungsvorlagen in der **Produktionsebene** sind untergliedert in Landschaft, digitale Bildmodelle (digitale Orthophotos) sowie Karten und andere Unterlagen. Die Landschaft als Quelle der Originalinformation wird insbesondere im Rahmen der Fortführung als Erfassungsquelle herangezogen werden. Durch den digitalen Datenfluss fließen im Felde registrierte Daten ohne den Umweg über analoge Medien direkt oder nach Strukturierung und Klassifizierung in die Bestandsdaten von AFIS, ALKIS und ATKIS ein. Die aufgebauten Geobasisdatenbestände können zugleich wieder als Erfassungsquelle für abgeleitete Datenbestände dienen, z.B. sind Teile der ALKIS-Bestandsdaten, insbesondere die Gebäudedaten, Grundlage zur Ableitung entsprechender Daten für das ATKIS-DLM. Der Erfassungsvorgang umfasst auch die Bildung von Präsentations- und Kartengeometrieobjekten (s. Punkte 3.3.5 und 3.3.6) und schließt damit auch den Vorgang der kartographischen Generalisierung ein.

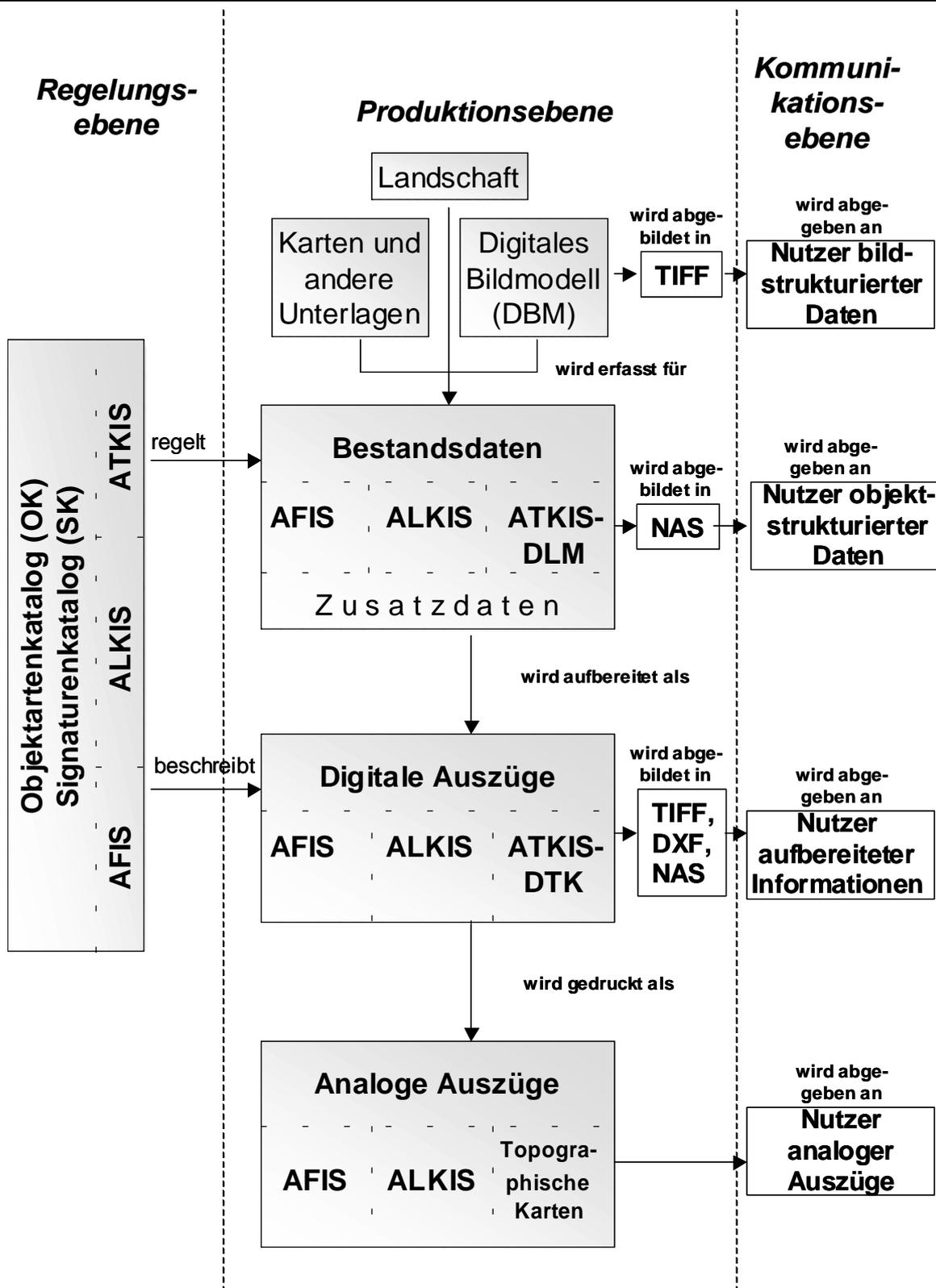


Abbildung 2-1: Gemeinsames AFIS-ALKIS-ATKIS-Referenzmodell

Die Bestandsdaten unterscheiden sich durch den Abstraktionsgrad, mit dem sie die Erdoberfläche und damit in Beziehung stehende Sachverhalte modellieren. Sie weisen Eigenschaften wie Objektstrukturierung und Geokodierung auf. Sie enthalten neben den

Fachobjekten mit ihren semantischen und geometrischen Informationen auch die zur Präsentation benötigten **Zusatzdaten**:

- nämlich die **Präsentationsobjekte** für Text und Signaturen (siehe 3.3.5)
- sowie die mit den topographischen Objekten durch eine einseitige Relation verknüpften **Kartengeometrieobjekte** mit der jeweiligen Kartengeometrie für einen bestimmten Kartenmaßstab (siehe 3.3.6).

Die Bestandsdaten enthalten die vollständige Beschreibung von Fachobjekten einschließlich der Daten zu ihrer kartographischen oder textlichen Darstellung in einem oder mehreren Zielmaßstäben. Damit sind die Bestandsdaten so modelliert, dass sie bei der Präsentation vollständig automatisch, d.h. ohne weiteren interaktiven Eingriff, in der vorgesehenen Ausgabeform dargestellt werden können.

An den Nutzer werden auf der **Kommunikationsebene** objekt- oder bildstrukturierte Daten, aufbereitete Informationen oder analoge Auszüge abgegeben, die den kompletten Dateninhalt, beliebige Auszüge nach Inhalt und Gebiet sowie Fortführungsdaten beliebiger Zeiträume umfassen können.

3 Das konzeptuelle Modell

3.1 Grundsätze der Modellierung

3.1.1 Normen und Standards

Internationale Normungs- bzw. Standardisierungsaktivitäten im Bereich von Geoinformationen erfolgen zurzeit in folgenden Gremien:

- ISO/TC 211 Geographic Information/Geomatics
- Open Geospatial Consortium (OGC).

Ziel ist die Schaffung von Grundlagen für die gemeinsame, ganzheitliche und fachübergreifende Nutzung von Geodaten an verschiedenen Orten durch Personen, Anwendungen und Systeme auf der Grundlage einer einheitlichen Beschreibung der Inhalte vorhandener oder geplanter Datenbestände, der Funktionalitäten der Datenbearbeitung und der Kommunikation. Der Modellierung liegen die Ergebnisse des ISO/TC 211 in Form der Normfamilie 19100 im aktuellen Bearbeitungsstand zu Grunde. Im Bereich der Datenaustauschnittstelle (s. Kap. 10) werden darüber hinaus auch Teile der Spezifikationen des OGC verwendet.

3.1.2 Modellierungs- und Beschreibungssprache

Zur Beschreibung des Anwendungsschemas und der Objektartenkataloge hat sich die AdV entschieden, die Datenmodellierungssprache *Unified Modeling Language (UML)* zu verwenden. Sie wird auch von ISO/TC 211 im Bereich der Normung von Geoinformationen eingesetzt.

UML wurde von der *Object Management Group (OMG)* zur Beschreibung von Anwendungsschemata entwickelt. Semantik und Notation von UML sind im *UML Notation Guide* beschrieben. Um eine einheitliche Nutzung von UML im Bereich der Normfamilie 19100 zu gewährleisten, ist deren Anwendung in der ISO-Norm 19103 *Conceptual schema language* festgelegt. Der Zweck liegt in der vollständigen und unzweifelhaft interpretierbaren, formalen Beschreibung von Inhalt und Struktur von Datenbeständen. Die Beschreibung ist unabhängig von der Art der Implementierung und der verwendeten Programmiersprache. Mit formalen Sprachen ist eine einheitliche Beschreibung aller Geodaten erreichbar. Die so beschriebenen Anwendungsschemata können von geeigneten Programmen automatisch interpretiert und in interne Datenstrukturen bzw. Datenbankstrukturen übersetzt werden.

Ein universelles und systemunabhängiges Datenaustausch- bzw. Dateiformat ergibt sich daraus automatisch in Verbindung mit sogenannten Kodierungsregeln (*Encoding Rules*). Diese Kodierungsregeln werden entsprechend der ISO-Norm 19118 *Encoding* und der GML-Spezifikation des Open Geospatial Consortiums (OGC) erstellt. Als Format wird die Auszeichnungssprache XML (Extensible Markup Language) des World-Wide-Web Consortiums (W3C) verwendet (siehe Kap. 10).

3.2 Aufgabe und Struktur

Ein Anwendungsschema liefert die formale Beschreibung für Datenstrukturen und Dateninhalte in einer oder mehreren Anwendungen. Es enthält die vollständige Beschreibung eines Datenbestandes und kann neben den geographischen auch weitere dazugehörige Daten enthalten. Ein grundsätzliches Konzept, die reale Welt zu abstrahieren, ist die Einführung des Fachobjekts und von Regeln, wie es erfasst und fortgeführt wird. Die Klassifizierung der Fachobjekte erfolgt nach Typen. Auf der Typenebene beschreibt das Anwendungsschema die Objektarten der realen Welt. Daten selbst existieren auf der Instanzebene. Sie stellen einzelne Exemplare einer Objektart in der realen Welt dar und können durch das Anwendungsschema interpretiert werden, siehe hierzu auch ISO 19101 *Reference model* und 19109 *Rules for application schema*.

Der Zweck eines Anwendungsschemas ist es, ein gemeinsames und einheitliches Verständnis der Daten zu erreichen und die Dateninhalte für eine bestimmte Anwendungsumgebung so zu dokumentieren, dass eindeutige Informationen über die Daten erhalten werden.

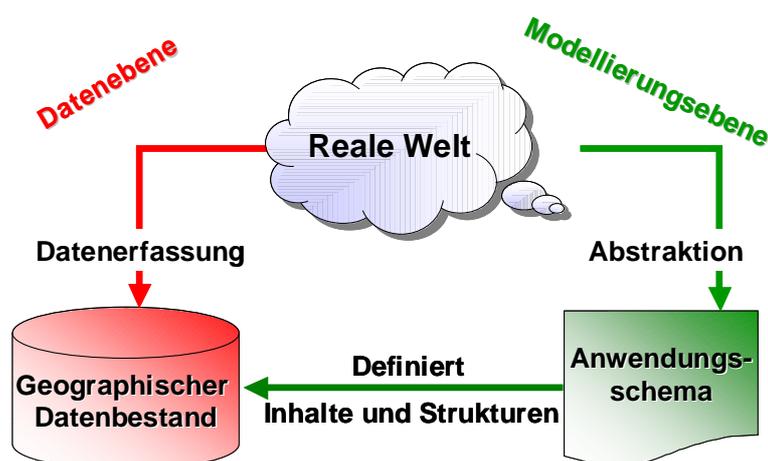


Abbildung 3-1: Die Rolle des Anwendungsschemas

Das gemeinsame AFIS-ALKIS-ATKIS-Anwendungsschema bietet einen einheitlichen und objektorientierten Modellansatz für AFIS, ALKIS und ATKIS, der möglichst mit den

marktüblichen und dem Stand der Technik entsprechenden GIS - Programmen abgebildet und geführt werden soll.

Ein Anwendungsschema kann Festlegungen aus verschiedenen Subschemata verwenden. Im Fall des AFIS-ALKIS-ATKIS-Anwendungsschemas werden hauptsächlich Subschemata aus der Normfamilie ISO 19100 verwendet. In Bereichen, in denen ISO bisher keine Festlegungen getroffen hat, werden zusätzlich Schemata von OGC verwendet.



Abbildung 3-2: Abhängigkeit des AFIS-ALKIS-ATKIS-Anwendungsschemas von den genormten Strukturen aus ISO 19100

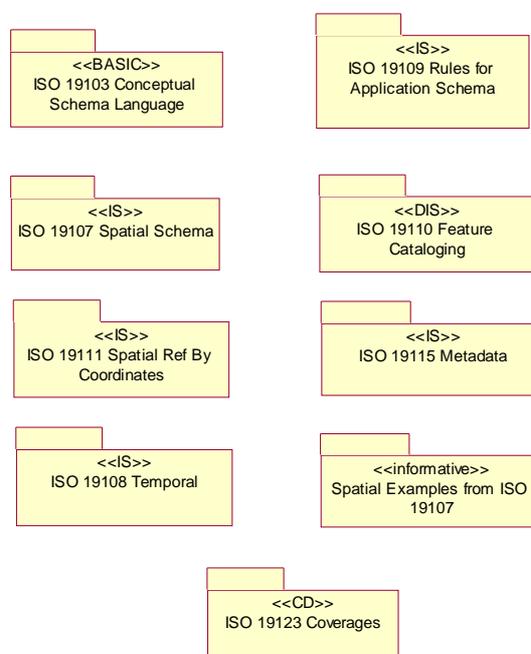


Abbildung 3-3: Verwendete Teile aus der Normfamilie ISO 19100

Das AFIS-ALKIS-ATKIS-Anwendungsschema gliedert sich in das Basisschema (Abschnitt 3.3), das Versionierungsschema (Abschnitt 3.4), das AFIS-ALKIS-ATKIS-Fachschemata (Abschnitt 5 bis 8), die NAS-Operationen (Abschnitt 10.2) und den AFIS-ALKIS-ATKIS-Ausgabekatalog. Daneben gibt es eine Reihe von weiteren Paketen, die zur Spezifikation des Datenaustausches erforderlich sind (Abschnitt 10).

Das Basisschema ist die Grundlage für die Modellierung der Fachobjekte in den Fachschemata. Das Versionierungsschema zeigt das Konzept zur Historisierung von

Fachobjekten auf. Ein internes Schema ist nicht Bestandteil der gemeinsamen Modellierung; es entsteht durch Abbildung des konzeptuellen Anwendungsschemas in spezifische GIS-Systeme im Zuge der Implementierung. Auf Basis des Anwendungsschemas sind schließlich Operationen für den Datenaustausch und fachliche Festlegungen für Datenausgaben definiert.

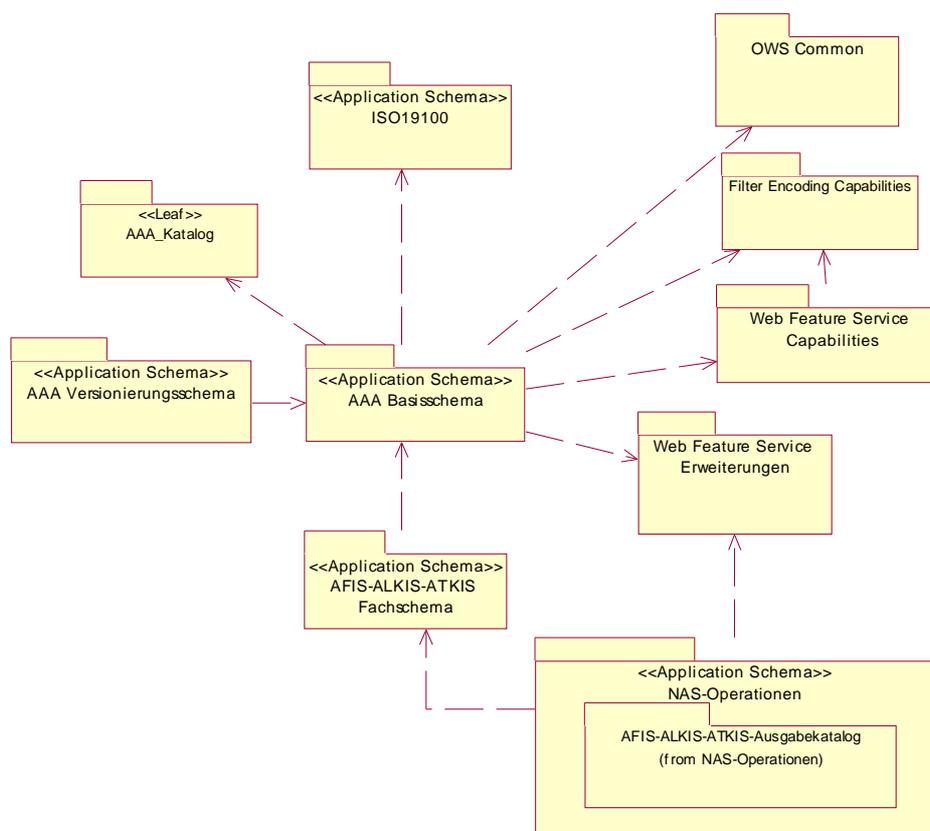


Abbildung 3-4: Die Bestandteile des AFIS-ALKIS-ATKIS-Anwendungsschemas

3.3 Das AFIS-ALKIS-ATKIS-Basisschema

Das AFIS-ALKIS-ATKIS-Basisschema (AAA-Basisschema) bildet die Grundlage der fachlichen Modellierung der AFIS-, ALKIS- und ATKIS-Objekte und für den Datenaustausch. Auf seiner Basis werden die Fachschemata erstellt. Seine Anwendung ist nicht auf AFIS, ALKIS und ATKIS beschränkt. Andere Fachinformationssysteme können die im Basisschema definierten Klassen zur Modellierung ihres Fachschemas ebenfalls nutzen.

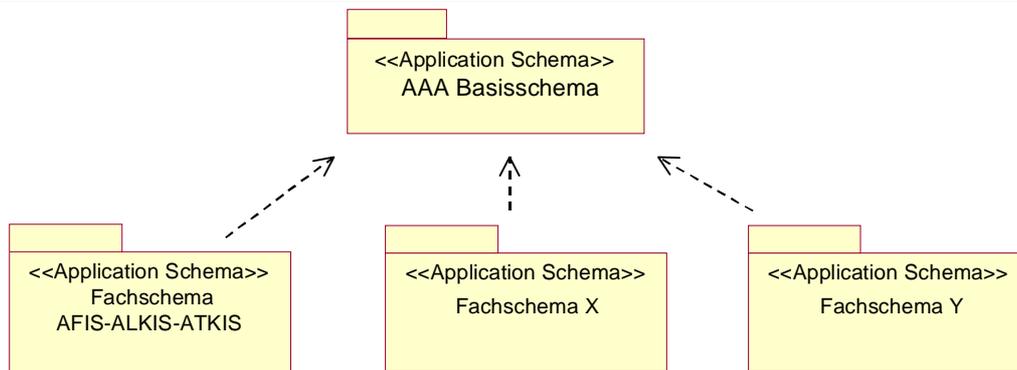


Abbildung 3-5: Das Basisschema als Grundlage der Modellierung von anwendungsspezifischen Fachschemata (z.B. AFIS, ALKIS und ATKIS)

Das Basisschema gliedert sich in die elf Pakete "AAA_Basisklassen", "AAA_Katalog", "AAA_SpatialSchema", "AAA_GemeinsameGeometrie", "AAA_UnabhaengigeGeometrie", "AAA_ExternalCodeLists", "AAA_Praesentationsobjekte", AAA_Punktmengenobjekte, AAA_Projektsteuerung, AAA_Nutzerprofile und AAA_Operationen.

Die Pakete AAA_Nutzerprofile und AAA_Operationen dienen lediglich der Verankerung einer Nutzerverwaltung bzw. einer Operationsmodellierung im Basisschema. Sie enthalten nur leere, abstrakte Klassen, die von den jeweiligen Fachschemata weiter ausgefüllt werden müssen. Sie werden im Folgenden deshalb nicht weiter erläutert.

Zur eindeutigen Benennung der definierten Klassen wird von folgender Systematik ausgegangen:

1. Genormte Klassen behalten das jeweilige genormte Präfix im Klassennamen (z.B. FC für "Feature Catalogue", MD für "Metadata").
2. Klassen als AFIS-ALKIS-ATKIS-spezifische Ergänzungen am genormten *Feature Catalogue* erhalten das Präfix AC.
3. Klassen mit grundsätzlicher Bedeutung für AFIS, ALKIS und ATKIS erhalten das Präfix AA.
4. Klassen, die aus den ISO TS_*Component - Klassen ("simple topology") abgeleitet wurden, erhalten das Präfix TA; ebenso die sinngemäß gebildete Klasse für topologische Flächen mit multipler räumlich getrennter Geometrie (TA_MultiSurfaceComponent).
5. Klassen mit gemeinsam genutzter Geometrie erhalten das Präfix AG.
6. Klassen der unabhängigen Geometrie erhalten das Präfix AU.
7. Klassen der Präsentationsobjekte erhalten das Präfix AP.

8. Klassen für die Modellierung von Punktmengenobjekten erhalten das Präfix AD.

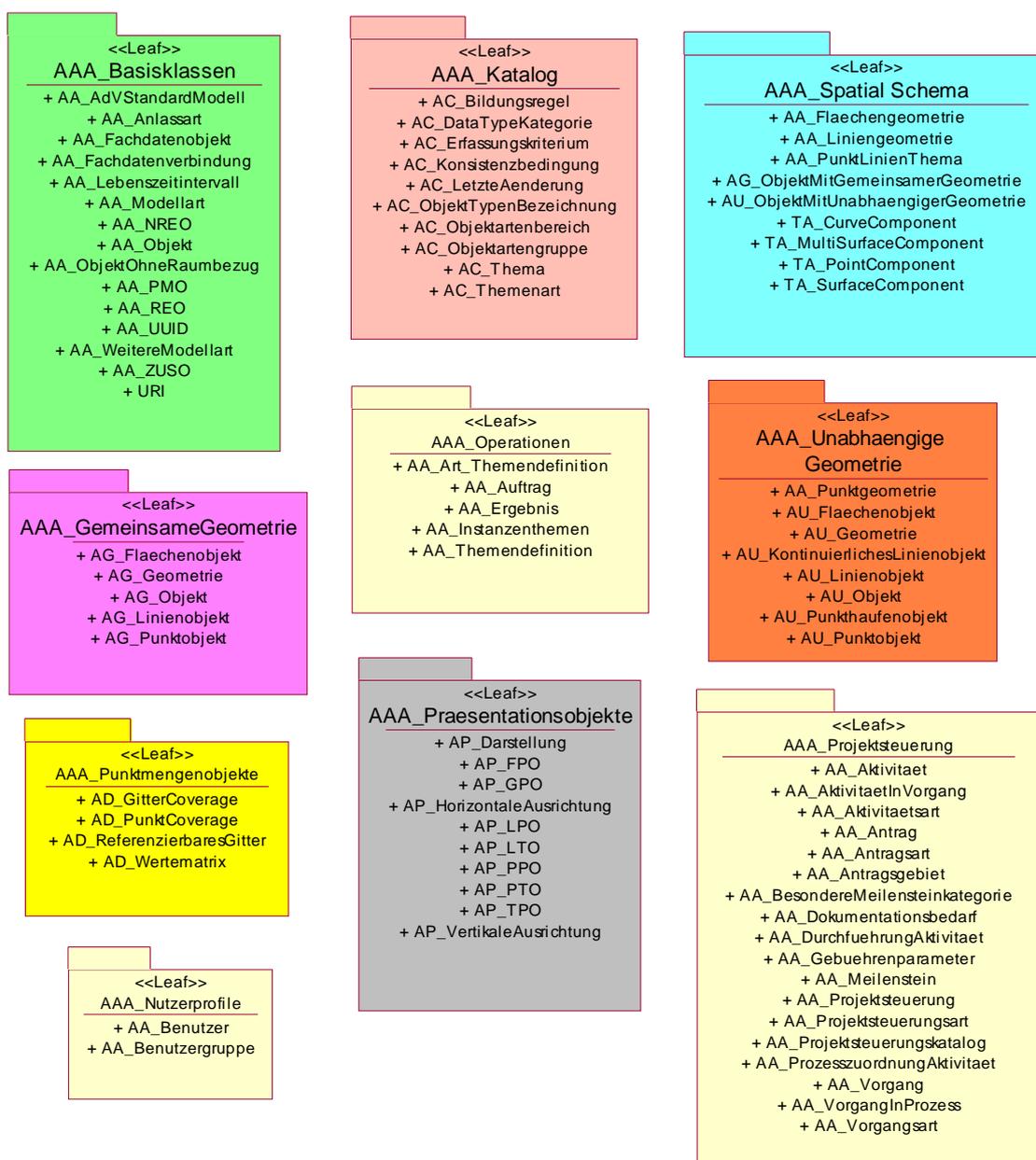


Abbildung 3-6: Bestandteile des Basisschemas

3.3.1 Objektbildungsgrundsätze

Die Regeln zur Erstellung von Anwendungsschemata werden durch die Norm 19109 "Rules for Application Schema" des ISO/TC 211 vorgegeben. In dieser Norm ist auch das allgemeine Modell zur Beschreibung und Bildung von Fachobjekten (*General Feature Model*) enthalten. Das gemeinsame Basisschema wird an das *General Feature Model* von

ISO 19100 angeschlossen und dieses um die Metaklasse "AA_ObjektOhneRaumbezug" erweitert, um Objektklassen bilden zu können, für die kein Raumbezug zulässig ist.

Die Bildung eigenständiger Objekte ergibt sich aus der fachlichen Objektsicht. **Objekte mit geometrischer Ausprägung** können punkt-, linien- oder flächenförmige Beschreibungen führen oder vom Typ Punktmengenobjekt sein. **Objekte ohne Raumbezug** (z. B. Personen) tragen keine Geometrie und lassen sich nicht auf einen bestimmten Ort festlegen. Sie können aber mit anderen raumbezogenen und nicht-raumbezogenen Objekten in Beziehung stehen, z. B. mit Flurstücken, Gebäuden oder Adressen.

Zur Systematisierung und zur Unterstützung bei der Erstellung der Fachschemata werden im gemeinsamen AAA-Basisschema 4 generelle Arten von Objektausprägungen vordefiniert:

- **Raumbezogene Elementarobjekte (AA_REO)**
Raumbezogene Elementarobjekte sind zu bilden, wenn zusätzlich zu fachlichen Eigenschaften auch geometrische oder topologische Eigenschaften nachgewiesen werden sollen.
- **Nicht raumbezogene Elementarobjekte (AA_NREO)**
Nicht raumbezogene Elementarobjekte sind zu bilden, wenn nur fachliche, aber keine geometrischen oder topologischen Eigenschaften nachgewiesen werden sollen.
- **Zusammengesetzte Objekte (AA_ZUSO)**
Zusammengesetzte Objekte werden gebildet, um den Zusammenhang einer beliebigen Zahl und Mischung semantisch zusammengehörender raumbezogener Elementarobjekte, nicht raumbezogener Elementarobjekte oder zusammengesetzter Objekte herzustellen. Ein zusammengesetztes Objekt muss aber mindestens ein Objekt als Bestandteil besitzen.
- **Punktmengenobjekte (AA_PMO)**
Für bestimmte Fachobjekttypen, die aus einer großen Anzahl geometrischer Orte mit jeweils gleichen Attributarten bestehen (z.B. Digitale Geländemodelle, Temperatur- und Luftdruckverteilungen), ist es günstiger, statt einzelner REOs ein alle Angaben klammerndes Objekt – ein sogenanntes Punktmengenobjekt – zu nutzen. Ein Punktmengenobjekt ist eine Abbildung einer Menge von Geometrien auf die zugehörigen Attributwerte.

Elementarobjekte sind die kleinste mögliche fachliche Einheit. Die Bildung von Objektteilen oder von Linien als Objektbestandteile mit fachlicher Information wie in den bisherigen Systemen ALK und ATKIS entfällt.

Die Führung der **Historie von Objekten** (siehe Kapitel 3.4) wird unterstützt. Ebenso werden **Integrations- und Verknüpfungsmöglichkeiten der Fachobjekte mit den Fachdaten** anderer Fachbereiche unterstützt (siehe Kapitel 3.3.7).

Alle instanziierten fachlichen Objektklassen sind in den anwendungsbezogenen Subschemas aus den folgenden Objektklassen des Basisschemas (siehe 3.3.4.1 bis 3.3.4.4) durch Vererbung abzuleiten:

- AA_ZUSO
- AA_NREO
- TA_PointComponent
- TA_CurveComponent
- TA_SurfaceComponent
- TA_MultiSurfaceComponent
- AG_Objekt
- AG_Punktobjekt
- AG_Linienobjekt
- AG_Flaechenobjekt
- AU_Objekt
- AU_Punktobjekt
- AU_Punkthaufenobjekt
- AU_Linienobjekt
- AU_KontinuierlichesLinienobjekt
- AU_Flaechenobjekt
- AD_PunktCoverage
- AD_GitterCoverage

Für Präsentationsobjekte (s. Punkt 3.3.5) können folgende Objektklassen des Basisschemas direkt verwendet bzw. instanziiert werden:

- AP_PPO
- AP_PTO
- AP_LTO
- AP_LPO
- AP_FPO
- AP_Darstellung

Alternativ ist zugelassen, dass aus diesen Objektklassen des Basisschemas durch Vererbung weitere instanziiierbare fachliche Objektklassen abgeleitet werden.

3.3.2 Attribute

Die in den Fachschemata zu beschreibenden Objekte können selbstbezogene Eigenschaften (Attribute) besitzen. Attribute sind die Träger der statischen Informationen der Objekte. Attribute werden immer über einen Namen und eine Wertart definiert. Wertarten können sowohl Basisdatentypen (Zahlen, Zeichenketten, Datums- und Zeitangaben) als auch komplexe Datentypen wie Geometrien oder Qualitätsmerkmale sein. **Attribute** können grundsätzlich multipel und Zeichenketten beliebig lang sein.

Attribute vom Typ Datums- und/oder Zeitangabe ("DateTime") werden entsprechend den Festlegungen von ISO 8601, Kapitel 5.4.1 in Verbindung mit 5.3.3 modelliert. Hierbei wird die Variante mit Trennzeichen gewählt. Zeitgenauigkeit ist die volle Sekunde, Zeitzone ist immer UTC (Universal Time Coordinated, Greenwich Mean Time, Abkürzung: Z). Beispiel: 2004-02-29T10:15:30Z

3.3.3 Beziehungen

Die in den Fachschemata zu beschreibenden Objekte können fremdbezogene Eigenschaften (Beziehungen bzw. Relationen) besitzen. In den Fachschemata können verschiedene Arten von Beziehungen verwendet werden:

- Nach dem *General Feature Model* von ISO können Fachobjekte beliebige Beziehungen zueinander eingehen. Diese werden in den fachspezifischen Subschemas definiert.
- Daneben sind im gemeinsamen Basisschema bereits einige Beziehungen zwischen Objekten fest vorgegeben:
 - Relation zur Bildung von Zusammengesetzten Objekten (ZUSO)
Ein ZUSO setzt sich aus mindestens einem Objekt zusammen. Die Klammer um diese Objekte bildet die Assoziation *Zusammensetzung* zwischen "AA_ZUSO" und "AA_Objekt".

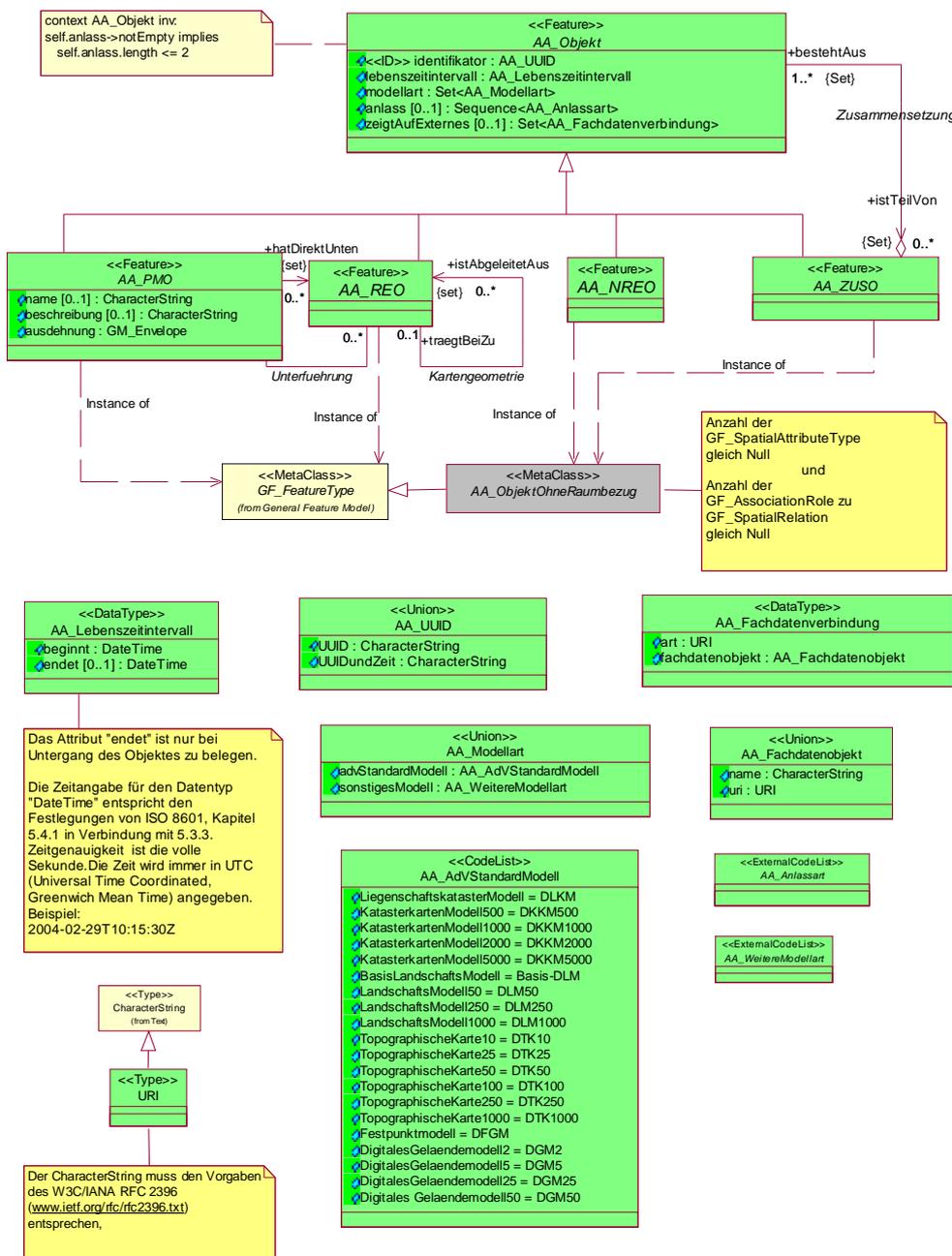


Abbildung 3-7: Modellierung der Basisklassen

– Unterfuehrungsrelation

Unterfuehrungsrelationen (*hatDirektUnten*) werden verwendet, um eine relative vertikale Lage einzelner Objekte im Verhältnis zu anderen Objekten abzubilden. Die Angabe einer absoluten "Höhenstufe" ist durch Verwendung von Überfuehrungs- bzw. Unterfuehrungsrelationen nicht möglich, da derartige

Relationen immer nur die Zweierbeziehung zwischen den beteiligten Objekten beinhalten.

– Kartengeometrie

Die Relation von Kartengeometrieobjekten (=generalisierte Geometrie, s. Punkt 3.3.6) zu den zugehörigen Basisobjekten (*istAbgeleitetAus*) gibt an, aus welchen Objekten die Kartengeometrieobjekte abgeleitet sind.

– Fachdatenverbindung

Soll ein AFIS-, ALKIS- oder ATKIS-Objekt auf ein Fachdatenobjekt zeigen, das in einem fremden Fachdatensystem geführt wird, so kann dies optional durch das Attribut *zeigtAufExternes* beschrieben werden.

– Darstellungsrelation

Präsentationsobjekte dienen der Darstellung von Objekten der Bestandsdaten. Dieser Zusammenhang wird durch den Verweis *dientZurDarstellungVon* zwischen dem Präsentationsobjekt und anderen Objekten nachgewiesen.

3.3.4 Raumbezug, Geometrie

3.3.4.1 Grundsätze

Die ISO - Norm 19107 *Spatial schema* stellt Raumbezugsgrundformen für die Verwendung in Anwendungsschemata zur Verfügung; für AFIS, ALKIS und ATKIS werden davon zur Verringerung der Komplexität ausschließlich folgende verwendet:

Geometrische Objekte (GM_Object)			Topologische Objekte (TP_Object)	
Geometrische Primitive	Geometrische Komplexe	Geometrische Aggregate	Topologische Primitive	Topologische Komplexe
GM_Point GM_Curve GM_PolyhedralSurface	GM_CompositeCurve GM_CompositeSurface	GM_MultiPoint GM_MultiCurve GM_MultiSurface	TS_PointComponent TS_CurveComponent TS_SurfaceComponent TS_Face	TP_Complex

Die geometrischen und topologischen Objekte sind als UML - Klassen beschrieben. Die Norm enthält auch räumliche Operationen, die die geometrischen und topologischen Objekte (*GM_Object* bzw. *TP_Object*) als Parameter benutzen (Erstellen, Löschen, Ändern, räumliche Auswertungen ...). Die definierten Klassen finden keine direkte Verwendung, d.h. sie sind nicht instanzierbar. Ihre Nutzung in speziellen Anwendungsschemata wird mittels Vererbung erreicht; soweit die Klassen des *Spatial Schema* für

AFIS, ALKIS und ATKIS nicht um spezielle Eigenschaften ergänzt werden, werden sie jedoch in diesem Anwendungsbereich zur Vereinfachung unmittelbar verwendet.

Die Raumbezugsgrundformen werden in der Regel als selbstbezogene Eigenschaften (Attribute) der Objekte geführt; dies bedeutet jedoch nicht, dass der Nachweis der Geometrie grundsätzlich redundant erfolgt. Das gemeinsame AFIS-ALKIS-ATKIS-Anwendungsschema verfügt hinsichtlich der Anbindung des Raumbezuges über folgende Möglichkeiten:

- Bildung knotenförmiger, kantenförmiger und maschenförmiger Objekte mit "einfacher Topologie". Zusätzlich maschenförmige Objekte mit „einfacher Topologie“, die aus zwei oder mehr räumlich getrennt liegenden Maschen bestehen (wird zur Modellierung von Überhakenflurstücken benötigt).
Es wird das ISO-Schema "Simple Topology" verwendet, das topologische Eigenschaften durch geometrische Eigenschaften ausdrückt, aber topologische Funktionalität bietet. (s. 3.3.4.2)
- Bildung punktförmiger, linienförmiger und flächenförmiger Objekte, die sich gegenseitig Linien und Punkte teilen. (s. 3.3.4.3)
- Bildung punktförmiger, linienförmiger und flächenförmiger Objekte mit voneinander "unabhängiger" Geometrie. (s. 3.3.4.4)
- Bildung von topologischen und geometrischen "Themen", die es erlauben, selektiv Objektarten zu sogenannten *Komplexen* zusammenzufassen, um geometrische Identitäten und/oder topologische Zusammenhänge auszudrücken.

Jedes raumbezogene AFIS-ALKIS-ATKIS-Fachobjekt (AA_REO) verweist auf maximal eine Geometrie. Besteht die Notwendigkeit, zu einem Realwelt-Objekt mehrere Geometrien vorzuhalten (z.B. Generalisierung, unterschiedliche Koordinatenreferenzsysteme, Punkt- und Flächengeometrie), so ist jeweils ein eigenständiges Fachobjekt (ggf. als Kartengeometrieobjekt) zu bilden.

Die erforderlichen Erweiterungen und Einschränkungen des *Spatial Schema* von ISO sind in den folgenden Abbildungen zusammenfassend dargestellt:

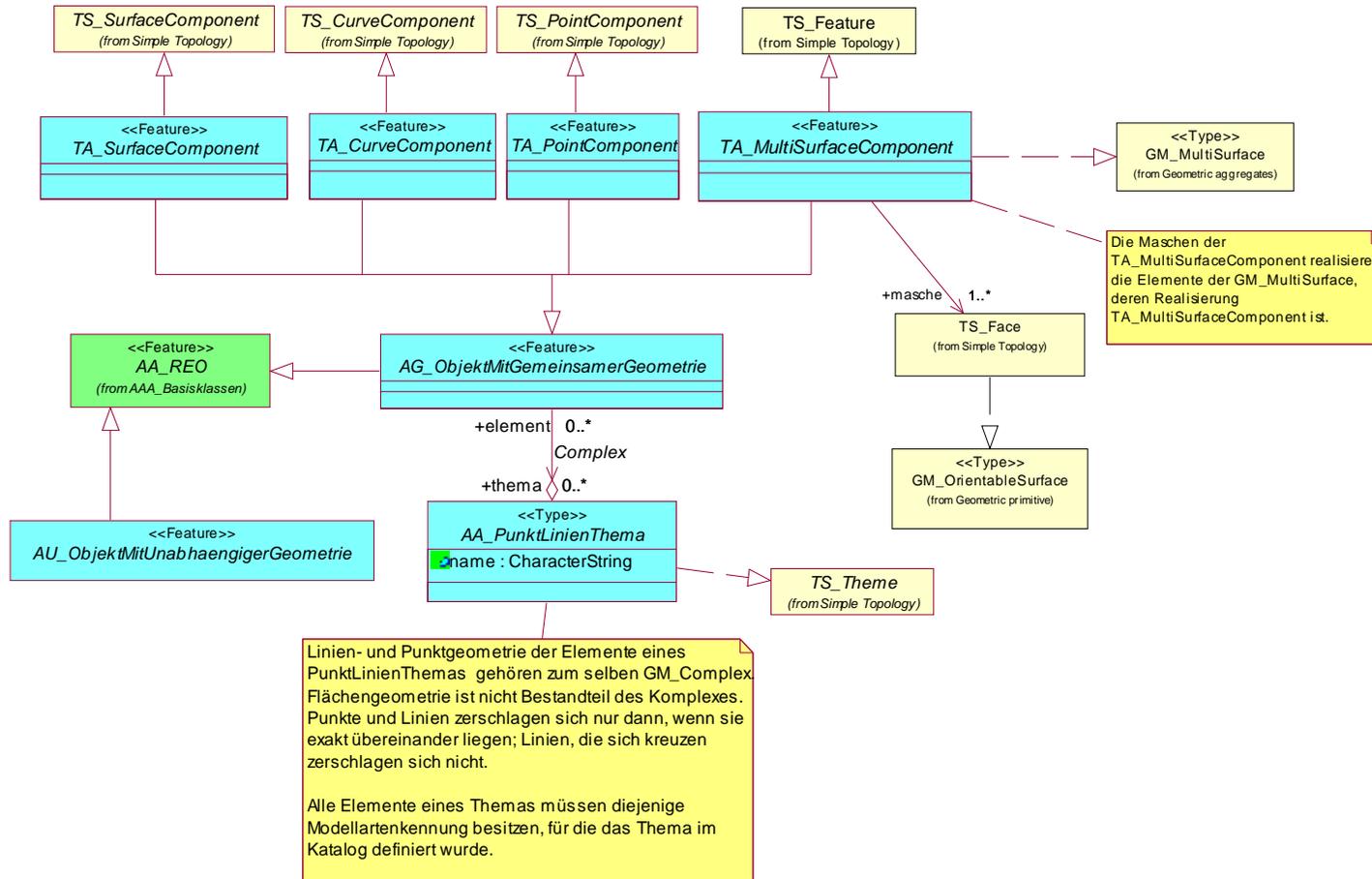


Abbildung 3-8: Zusammenfassende Darstellung der für AFIS-ALKIS-ATKIS erforderlichen Ergänzungen am genormten Spatial Schema

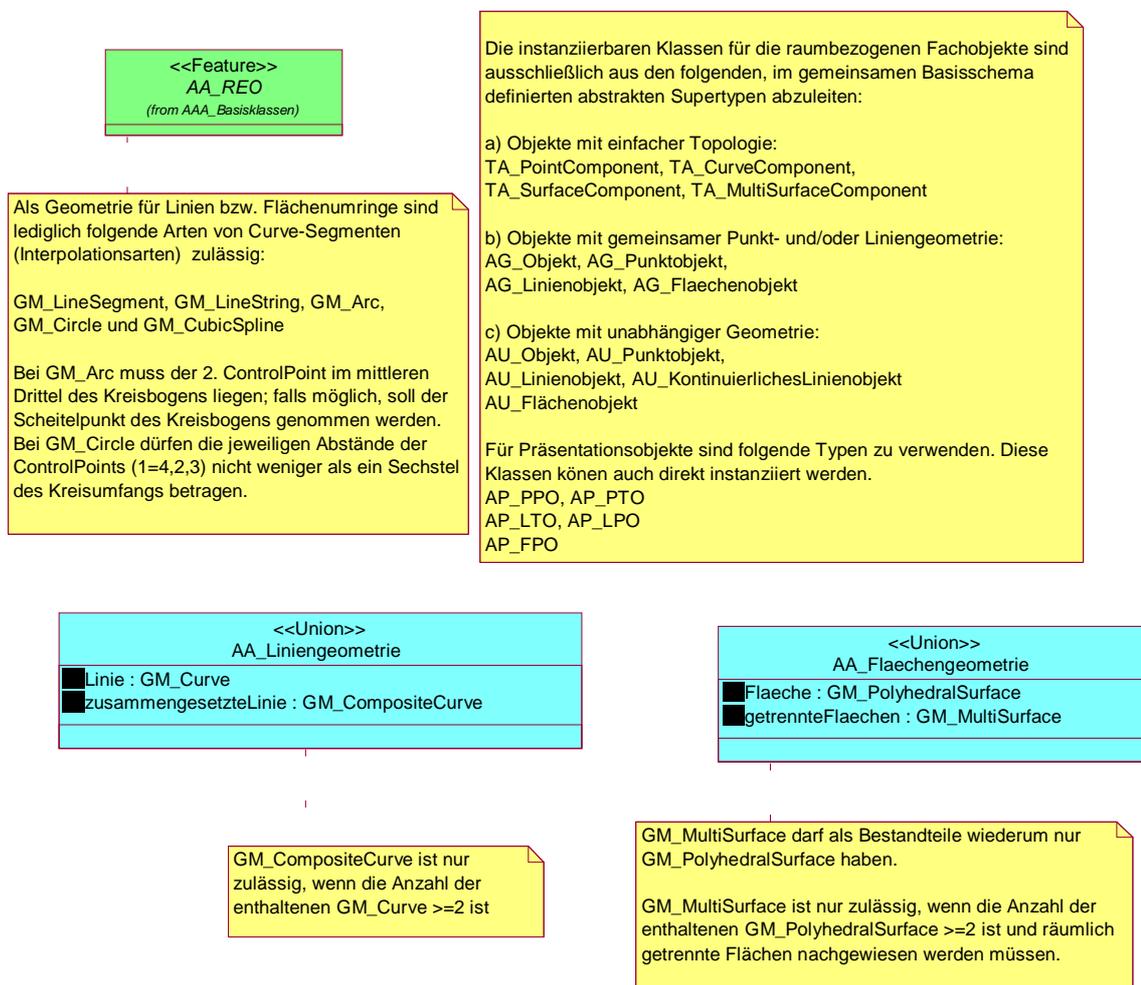


Abbildung 3-9: Restriktionen bezüglich der Geometrie und instanziiierbare Klassen

Für die Linien und Flächenumringe werden die zulässigen Interpolationsarten eingeschränkt auf: GM_LineSegment, GM_LineString, GM_Arc, GM_Circle und GM_CubicSpline. Diese gelten grundsätzlich für alle auf dem Basisschema aufbauenden Fachschemata, außer es werden dort weitere Einschränkungen vorgenommen. Im ALKIS-Fachschemata sind nach entsprechenden Beschlüssen der AdV-Gremien keine GM_CubicSplines erlaubt.

3.3.4.2 Objekte mit einfacher Topologie

ISO 19107 *Spatial schema* bietet als Modul für ein Anwendungsschema unmittelbar das Schema *Simple topology* an. Auf dieser Basis werden Objekte bereitgestellt, die topologische Eigenschaften durch geometrische Eigenschaften ausdrücken. Als

Anwendung dieses Moduls stellt das Basisschema die Klassen TA_*Component zur Verfügung. Diese Klassen bieten zusätzlich zu den entsprechenden Klassen des *Spatial Schema* die allgemeinen Eigenschaften aller AFIS-ALKIS-ATKIS-Objekte (Identifikator, Lebenszeitintervall, Anlass) sowie die Möglichkeit, verschiedene Objektarten über das Konstrukt des "PunktLinienThemas" topologisch und geometrisch zu verknüpfen. Die TA - Klassen können gleichzeitig einem topologischen Thema und einem oder mehreren PunktLinienThemen angehören. Die Klasse TA_MultiSurfaceComponent wurde abweichend zur Klasse TA_SurfaceComponent definiert, um zu erreichen, dass die referenzierten Maschen (*TS_Face*) auch Realisierungen getrennt liegender Flächen (*GM_OrientableSurface*) sein können. Damit ist auch die topologische Modellierung von Exklaven möglich. Exklaven sollen deshalb nicht über Relationen von Fachobjekt zu Fachobjekt (Relation *Composite* [*composite* > *component*] zwischen TS_Feature und TS_Feature) modelliert werden.

3.3.4.3 Objekte gemeinsam genutzter Geometrie

Das Paket "AAA_GemeinsameGeometrie" stellt Basisklassen für Fachobjekte (Features) zur Verfügung, deren Geometrie aus Punkten, Linien und Flächen bestehen, die sich jeweils ihre Geometrie teilen. Dazu werden die Eigenschaften des erweiterten "AAA-SpatialSchema" genutzt, das zusätzlich das Konstrukt des "PunktLinienThemas" zur Verfügung stellt. Außerdem wird die Geometrie durch die gemäß ISO 19107 und 19109 für die gemeinsame Nutzung von Geometrie vorgesehenen Raumbezugsgrundformen (*GM_PointRef* und *GM_CompositeCurve*) ausgedrückt. Damit sind die geometrietragenden Primitive (*GM_Point* und *GM_Curve*) relational mit den Fachobjekten verbunden und können so von mehreren Fachobjekten gemeinsam genutzt werden. Die gemeinsame Nutzung von Geometrie bezieht sich nur auf Punkte und Linien, nicht auf Flächengeometrien. Linien werden vereinigt und einer gemeinsamen Nutzung zugeführt, wenn sie exakt in allen Stützpunkten gleich sind und gleiche Linieninterpolationen aufweisen; Linien, die sich kreuzen, zerschlagen sich nicht. Die Basisklassen "AG_Objekt", "AG_Punktobjekt", "AG_Linienobjekt" und "AG_Flaechenobjekt" sollen für die Definition von raumbezogenen Objektarten mit gemeinsamer Geometrie verwendet werden.

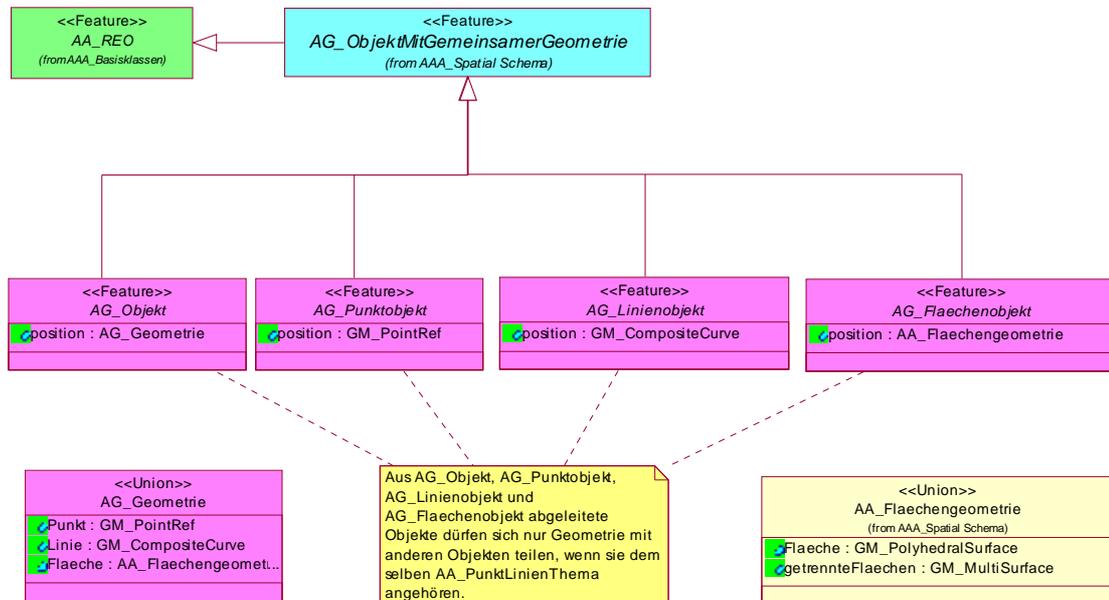


Abbildung 3-10: Objekte mit gemeinsamer Geometrie

3.3.4.4 Objekte mit unabhängiger Geometrie

Das Paket "AAA_UnabhaengigeGeometrie" stellt fünf Basisklassen für Fachobjekte zur Verfügung, deren Geometrie aus voneinander unabhängigen Punkten, Linien und Flächen bestehen. Diese Basisklassen sollen als Basis raumbezogener Objektarten mit unabhängiger Geometrie verwendet werden (z.B. Präsentationsobjekte).

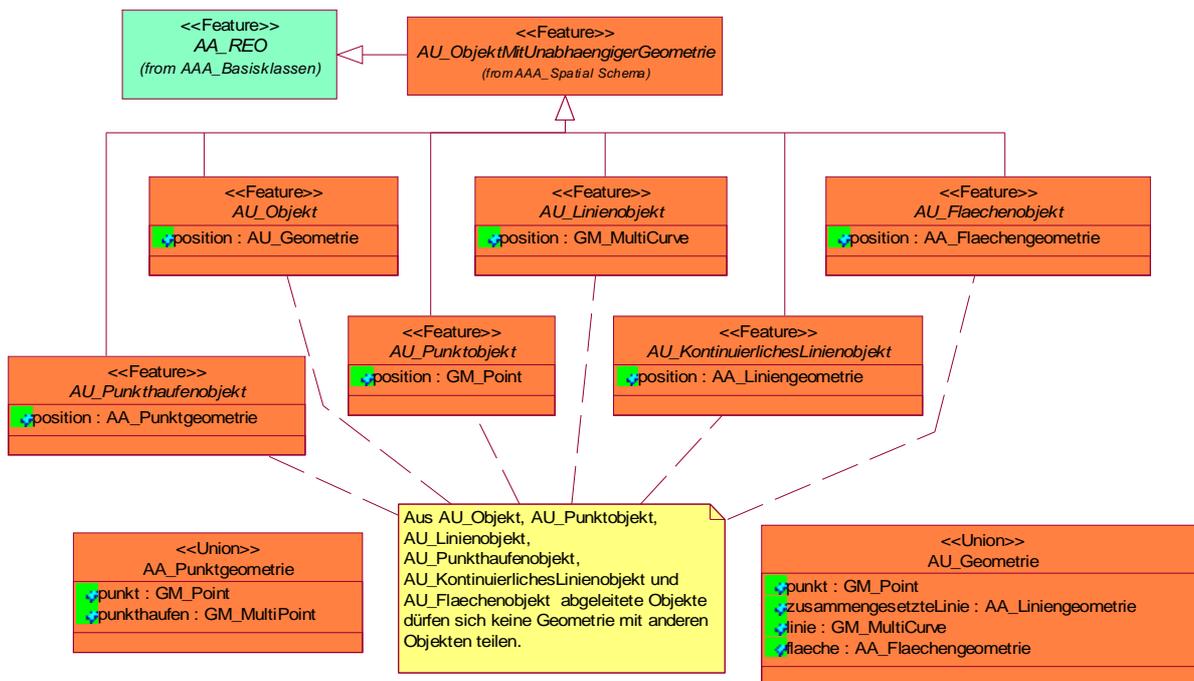
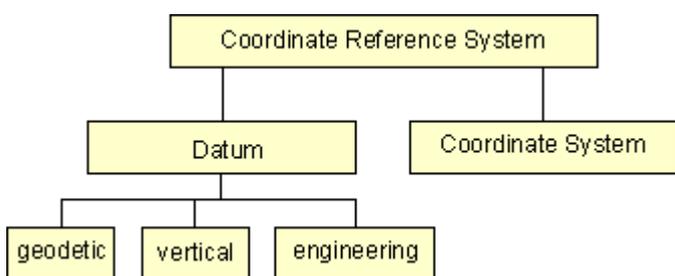


Abbildung 3-11: Objekte mit unabhängiger Geometrie

3.3.4.5 Raumbezugssystem, Koordinaten

In AFIS-ALKIS-ATKIS kann für jede Geometrie das zugehörige **Koordinatenreferenzsystem (CRS)** angegeben bzw. gespeichert werden.

Nach der Norm ISO 19111 (*Spatial Referencing by Coordinates*) besteht ein Koordinaten-



referenzsystem aus zwei Komponenten, dem „Datum“ und dem „Koordinatensystem“ (siehe Skizze).

Das **Datum** ist der physikalische Teil eines CRS, das per Definition des Nullpunkts, der Orientierung der Koordinatenachsen und des

Maßstabs den Bezug zur Erde festlegt. Ein Datum kann ein geodätisches Datum, ein vertikales Datum oder ein ingenieurtechnisches bzw. lokales Datum sein. Beispiele für ein geodätisches Datum sind das Deutsche Hauptdreiecksnetz (DHDN), auch “Potsdam-Datum“ genannt, oder das Europäische Terrestrische Referenzsystem 1989 (ETRS89).

Das **Koordinatensystem** ist der mathematische Teil eines CRS der durch Regeln festgelegt, wie einer Geometrie, z. B. einem Festpunkt, Koordinaten zugewiesen werden. Die Koordinaten einer Geometrie können z. B. als kartesische Koordinaten (X, Y, Z),

ellipsoidische Koordinaten (Breite, Länge und ggf. ellipsoidische Höhe) oder projizierte Koordinaten (Gauß-Krüger-Abbildung, UTM-Abbildung) angegeben werden.

Neben den CRS für 2-D-Lageangaben und 3-D-Positionsangaben sind für die Führung von Höhenangaben bzw. -koordinaten (z.B. NN-Höhen) eigene Koordinatenreferenzsysteme definiert. Die in Deutschland gebräuchlichen Koordinatenreferenzsysteme für Lage, Position und Höhe sind im Kapitel 11, [Koordinatenreferenzsysteme und Maßeinheiten für AFIS-ALKIS-ATKIS](#) mit ihren Bezeichnungen und Kurznamen aufgelistet.

Die Art des Koordinatenreferenzsystems bestimmt die Anzahl der vorhandenen Koordinatenwerte (z.B. Rechtswert, Hochwert oder Rechtswert, Hochwert, Höhe). Grundsätzlich können nach ISO 19111 auch zusammengesetzte CRS eingeführt werden. Bei Objekten der Objektart "Punktort" sind in AFIS-ALKIS-ATKIS gemäß der Definition der Objektart Punktort zusammengesetzte Koordinatenreferenzsysteme jedoch **nicht** zugelassen.

Da die heutigen ALK- und ATKIS-Systeme keine Angaben über das für den Grundriss relevante CRS führen, ist es Aufgabe des Migrationskonzepts, die entsprechenden Festlegungen zu treffen.

3.3.5 Signaturierung, Präsentationsobjekte

Die Präsentationsobjekte sind wegen den allgemeingültigen Eigenschaften im AAA Basisschema beschrieben.

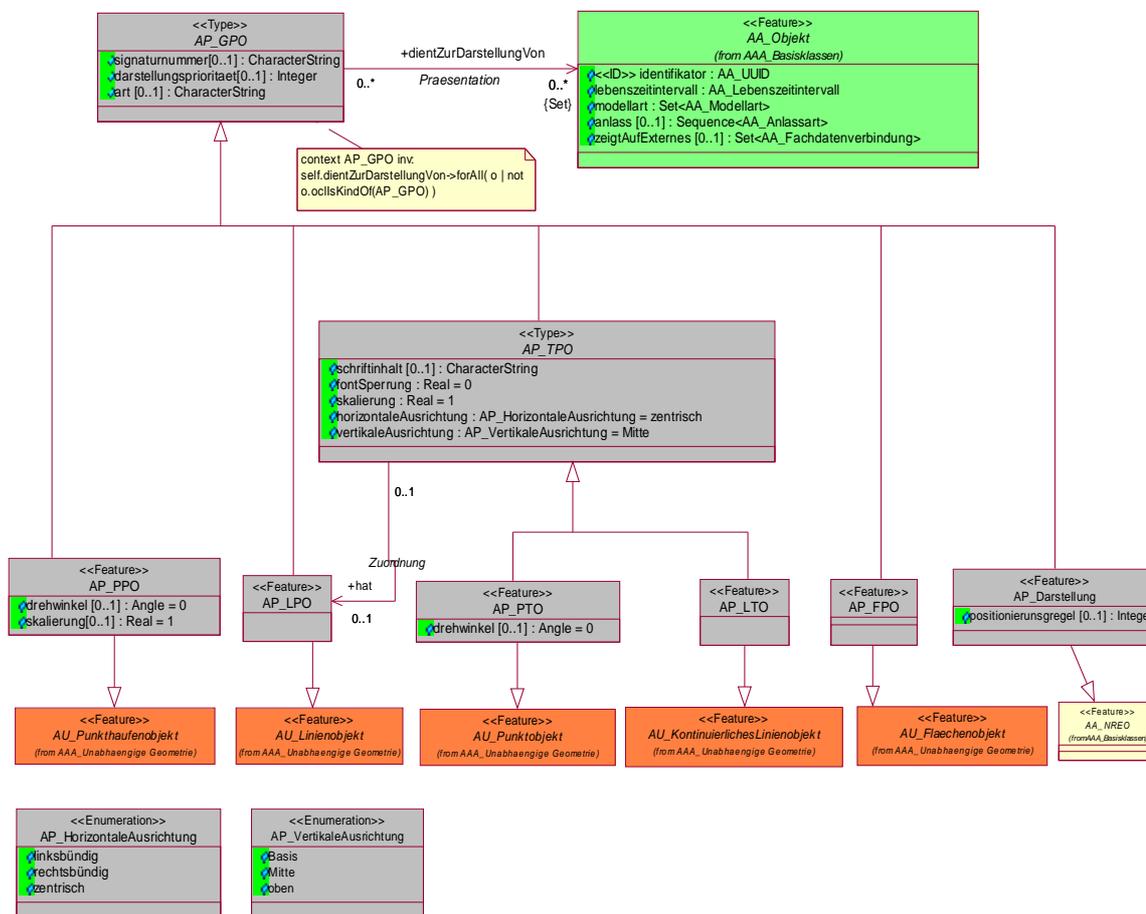


Abbildung 3-12: Präsentationsobjekte

Die Präsentationsobjekte enthalten die Signaturnummer und weitere Eigenschaften zur Steuerung der Präsentation, wie z. B. Darstellungspriorität und Art. Präsentationsobjekte **müssen in ALKIS** mit den entsprechenden Fachobjekten durch eine Relation "dientZurDarstellungVon" verbunden sein. In ATKIS gibt es keine derartige Regel, d.h. es dürfen „freie Präsentationsobjekte“. Die Präsentation von Objekten in graphischen sowie nicht graphischen Ausgaben erfolgt gemäß nachstehender Abbildung in folgender Weise:

Präsentation in der Karte

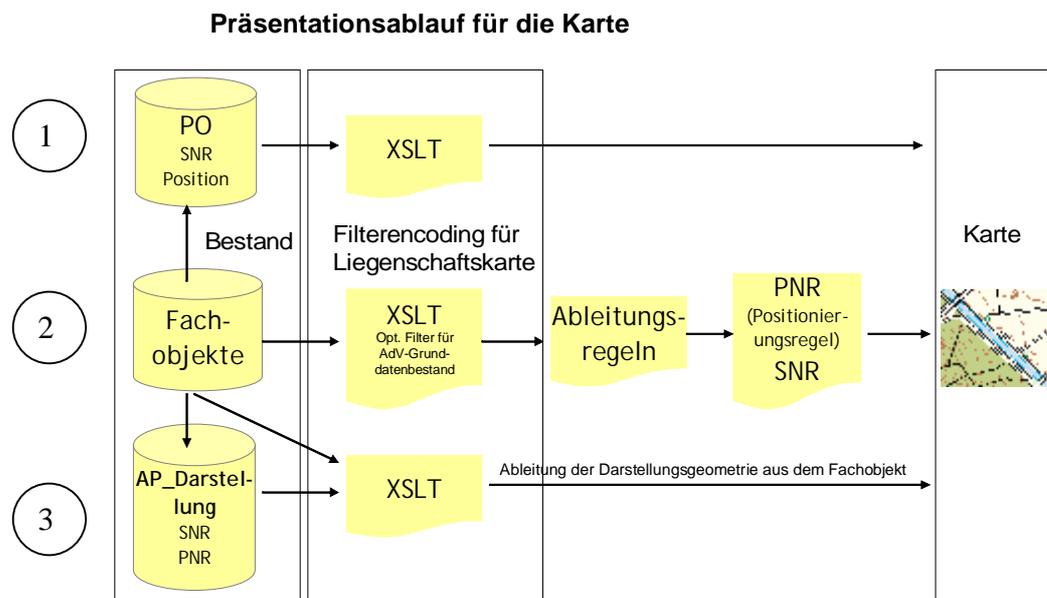


Abbildung 5.3 - 1. : Präsentationsablauf für die Karte

1. Präsentationsobjekte im Bestand

Präsentationsobjekte werden für alle Signaturen in Form von Schrift, Symbol, Linie, Fläche angelegt, die nicht vollautomatisch für einen bestimmten Zielmaßstab erzeugt und platziert werden können. Die konkrete Signaturnummer, die eine Ableitungsregel repräsentiert, sowie die Positionierungsnummer, die für eine bestimmte Positionierungsregel steht, kann optional im Präsentationsobjekt gespeichert werden. Präsentationsobjekte sind auch dann zu bilden, wenn bei der Ausgabe von der im Signaturenkatalog festgelegten Standarddarstellung abgewichen werden soll (z.B. abweichende Schriftgröße der Flurstücksnummer).

2. Präsentation mittels Ableitungs- und Positionierungsregel

Signaturen eines Fachobjektes in Form von Schrift, Symbol, Linie, Fläche werden an einer definierten Stelle (Standardposition) unter Anwendung des Filterencodings und einer konkreten Ableitungsregel, die zu einer bestimmten Signaturnummer führt und den Positionierungsregeln, die eine bestimmte Positionierungsnummer aktiviert, platziert. In diesem Falle wird ein Präsentationsobjekt in den Bestandsdaten nicht angelegt. Die darzustellende fachliche Information wird aus der angegebenen Attributart der Fachobjektart ermittelt. Dieser Weg wird als die

Standardvariante betrachtet, der aber durchaus aus Gründen der Performance nicht immer effizient ist.

3. Präsentation mittels gespeicherter Ableitungs- und Positionierungsregel

Um die Performance der Präsentation für die Standardvariante zu erhöhen, wird zu einem bestimmten Zeitpunkt (Ersteinrichtung, Fortführung) die konkrete Signaturnummer sowie die Positionierungsnummer mit der ein Fachobjekt zur Darstellung gebracht werden soll, unter dem zugeordneten Präsentationsobjekt AP_Darstellung als NREO gespeichert. Der Vorteil gegenüber der Bildung von Präsentationsobjekten (Variante 1) ist die Vermeidung von Redundanzen der Geometrie, da bei AP-Darstellung die Darstellungsgeometrie aus dem jeweiligen Fachobjekt abgeleitet wird. Im Zeitpunkt der Präsentation wird durch Anwendung des Filterencodings in Verbindung mit der Ableitung der Darstellungsgeometrie aus dem Fachobjekt und den gespeicherten Regeln, sprich Signaturnummer, Positionierungsnummer, die Darstellung schnellstmöglich herbeigeführt. IN ALKIS wird die Objektart AP_Darstellung zur Zeit ausschließlich dazu verwendet, um folgende Veränderungen in einer Liegenschaftskarte herbeizuführen:

- Unterdrückung einer Darstellung in der Liegenschaftskarte
- Herbeiführung einer bestimmten Bemusterung in der Liegenschaftskarte, wie z. B. flächenhafte Bemusterung.

Präsentation der Liegenschaftsbeschreibung

Die Präsentation der Angaben für eine Liegenschaftsbeschreibung, wie z. B. Flurstücksnachweis / Eigentümnachweis, erfolgt ausschließlich zur Laufzeit über die Anwendung des Filterencodings, womit die entsprechenden Ausgabedaten erzeugt werden. Diese Ausgabedaten werden in Verbindung mit einer konkreten Ableitungsregel präsentiert. Die entsprechenden Textpositionen können aus dem entsprechenden Muster (siehe Kapitel 7.3) entnommen werden. Die Objektart AP_Darstellung als NREO findet hierbei keine Anwendung. Ebenso werden keine Präsentationsobjekte im Bestand angelegt.

Präsentationsablauf für die Liegenschaftsbeschreibung

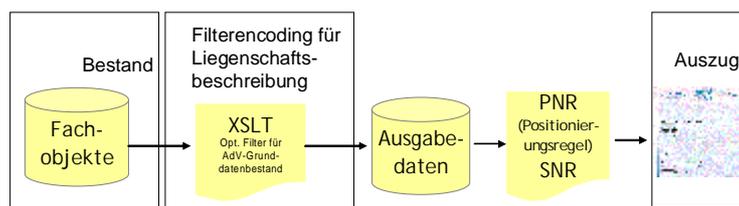


Abbildung 5.3 - 2. : Präsentationsablauf für die Liegenschaftsbeschreibung

Erzeugung der Präsentationsobjekte und AP_Darstellung für den Bestand

Um eine effiziente Präsentation der Fachobjekte in einer Ausgabe zu gewährleisten, müssen bereits zum Zeitpunkt der Erhebung / Fortführung geeignete Präsentationsvorgaben festgelegt werden. Es werden dabei, gemäß der nachfolgenden Abbildung drei Fälle unterschieden:

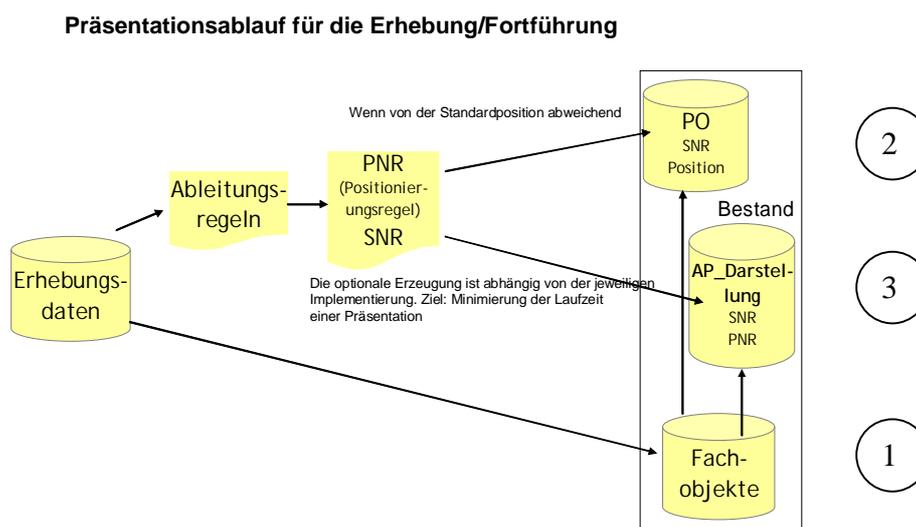


Abbildung 5.3 - 3. : Präsentationsablauf in der Erhebung / Fortführung

1. Keine Festlegung von Präsentationsvorgaben

Die in der Erhebung / Fortführung erzeugten ALKIS- strukturierten Erhebungsdaten brauchen für eine schnelle Präsentation in einer Ausgabe keine vordefinierten

Festlegungen in Form der Zuweisung einer konkreten Signaturnummer, Positionierungsnummer. Die für eine Präsentation benötigten Angaben können direkt während der Laufzeit für eine Darstellung aus dem 3A- Datenmodell und dem ALKIS-SK generiert werden.

2. Speicherung von Präsentationsobjekten im Bestand

In der Erhebung / Fortführung wird zur Darstellung von konkreten Signaturen eines Fachobjektes ein Präsentationsobjekt angelegt, da die Signaturen z. B. nicht vollautomatisch für einen bestimmten Zielmaßstab erzeugt und platziert werden können. Hierbei werden die Angaben über die Geometrie, optional eine Signaturnummer und / oder optional eine Positionierungsnummer im Objekt gespeichert.

3. Festlegung von Präsentationsvorgaben

Zur Minimierung der Laufzeit einer Präsentation kann in der Erhebung / Fortführung für ein Fachobjekt die Objektart AP_Darstellung als NREO angelegt werden, in der eine konkrete Signaturnummer, Positionierungsnummer gespeichert wird, so z. B. die Bemusterung einer Fläche. Die Geometrie für die Präsentation wird zur Laufzeit aus dem Fachobjekt mit geeigneten Methoden abgeleitet.

3.3.6 Kartengeometrieobjekte

Als **Kartengeometrieobjekte** werden diejenigen Fachobjekte definiert, die bei der Ableitung für einen bestimmten Kartenmaßstab aus Gründen der kartographischen Generalisierung ihre geometrische Form und/oder Lage verändert haben. Ein Kartengeometrieobjekt muss folgende eigenständige Informationen enthalten: Den Identifikator, die Angabe des Kartenmodells, z. B. DTK10, zu dem es gehört, die einseitige Relation *ist_abgeleitet_aus* auf das zugrundeliegende AFIS-ALKIS-ATKIS-Objekt sowie die eigentliche Geometrie. Darüber hinaus muss es die Attribute des zugrundeliegenden AFIS-ALKIS-ATKIS-Objekts enthalten, um in den Ableitungsregeln des Signaturenkatalogs für die Präsentation ausgewertet werden zu können. Damit sind diese Fachobjekte Instanzen derselben Objektarten und können identische selbstbezogene und fremdbezogene Eigenschaften haben wie die DLM Objekte auf die sie über die Relation *ist_abgeleitet_aus* verweisen. Sie unterscheiden sich jedoch durch die Verwendung anderer Modellartenkennungen (z.B. DTK10, DTK25, DTK50, DTK100). Bei der Plausibilisierung der Kartengeometrie-Objekte muss jedoch klar definiert sein, welche selbstbezogenen und fremdbezogenen Eigenschaften für diese Objekte zugelassen sind. Hierfür werden künftig entweder die Modellartenkennungen im UML Modell der AdV ergänzt oder klare Zuordnungen getroffen, welche Modellartenkennungen (z.B. Basis-DLM) welche anderen Modellartenkennungen (z.B. DTK10 und DTK25) mit einschliessen.

Jedoch erst mit der Einführung der DTK in das Modell (nach der Veröffentlichung der Version 5.1) können die Kartengeometrieobjekte eine DTK-Modellartenkennung erhalten und eindeutig beschrieben werden.

3.3.7 Punktmengenobjekte

Als Punktmengenobjekte (PMO) werden Fachobjekte dann definiert, wenn einer großen Anzahl geometrischer Orte Attributwerte jeweils gleicher Attributarten zugeordnet werden soll. Dies ist im AAA-Anwendungskontext insbesondere bei Digitalen Geländemodellen, die i.d.R. Höheninformationen in einer Gitterstruktur vorhalten, der Fall. Da aber auch häufig unregelmäßig verteilte, gleichartige Informationen vorgehalten werden sollen, z.B. Höhenmesspunkte, wurde außer Gittervariante der PMO (AD_GitterCoverage) auch eine Variante für eine beliebige Punktverteilung zugelassen (AD_PunktCoverage). Die Modellierung realisiert die Klassen aus ISO 19123 Coverages. Sie wird in der Weise eingeschränkt, dass für die Sequenz der Attributwerte (CV_SequenceType) nur „linear“ zulässig ist.

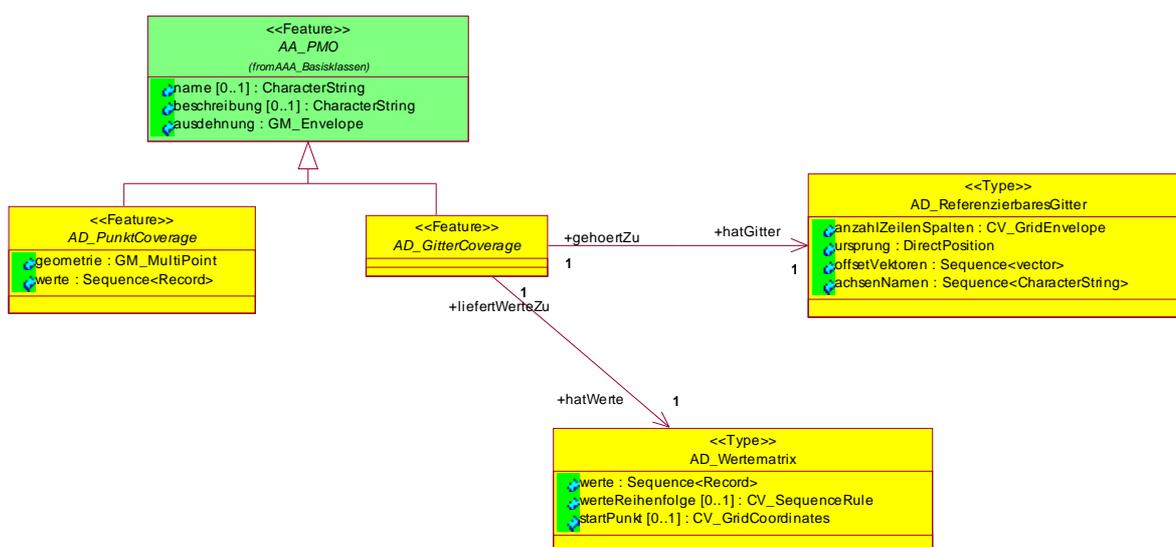


Abbildung 3-13: Modellierung der Punktmengenobjekte

3.3.8 Externe Code-Listen

Im Basisschema gebrauchte Code-Listen, die von ihrem Charakter her a) von den anwendungsspezifischen Subschemas gefüllt werden müssen und b) zur Integration unterschiedlicher Anwendungen erweiterbar sein müssen, werden im Basisschema als leere Klassen definiert und mit dem Stereotype `<<externalCodeList>>` versehen. Die

fachspezifischen Subschemas definieren entsprechende Klassen, die aus den Klassen des Basisschemas erben und die konkreten Codes festlegen. Erweiterungen und Änderungen dieser Listen führen nicht zu einer neuen Version der Austauschchnittstelle (s. Kap. 10). Sie werden in Form sog. *dictionaries* an zentraler Stelle mit der Möglichkeit des online-Zugriffs geführt und gepflegt.

3.3.9 Identifikatoren, Verknüpfungen

Identifikatoren stehen stellvertretend für das Objekt, das sie repräsentieren. Die wesentlichen Eigenschaften eines Identifikators sind:

- Er ist systemweit eineindeutig, wobei durch die entsprechende Definition von "systemweit" die Forderung nach bundesweiter und fachübergreifender Eindeutigkeit erfüllt werden kann.
- Sein Entstehen zeigt an, dass ein Objekt entstanden ist.
- Er bleibt während der Lebensdauer eines Objekts unverändert.
- Sein Untergehen zeigt an, dass ein Objekt nicht mehr existiert.

Damit ist der Lebenszyklus von Identifikatoren identisch mit dem Lebenszyklus der Objekte, deren Repräsentanten sie sind. Die Frage, wann Identifikatoren geändert werden dürfen und wann nicht, darf somit nicht aus dv-technischer Sicht beantwortet werden, sondern es müssen fachliche Kriterien benannt werden,

- wann ein Objekt entsteht,
- welche Änderungen es ohne Identitätsverlust verkraftet und
- wann es untergeht.

Für alle Fachobjekte wird eine eindeutige Bezeichnung als Objektidentifikator verwendet. Der Identifikator hat folgenden Aufbau:

	Anteile	Bedeutung	Festlegung																																										
1	Weltweit eindeutige Kennung (2 Zeichen)	Nationalität	"DE" für Deutschland																																										
2	Präfix (6 Zeichen)	Kennung für die den Identifikator erzeugende Implementierung oder Datenbank sowie für vorläufige Identifikatoren.	<p>Die Zeichen beginnen linksbündig mit den in der Norm ISO 3166-2 "Country Subdivision Code" (ISO, 15. Dezember 1998) genormten Abkürzungen der Bundesländer. Für Bundesdienststellen ist die Abkürzung "BU" vorgesehen bzw. im Falle des Bundesamtes für Kartographie und Geodäsie "BKG"; die weiteren Stellen werden durch das jeweilige Bundesland bzw. die Bundesdienststelle oder das BKG festgelegt. Soweit im Verarbeitungsprozess über die Verwendung von vollständigen Identifikatoren hinaus vorläufige Identifikatoren benötigt werden, beginnen diese linksbündig mit "*". Damit ergibt sich folgende Tabelle.</p> <table border="0"> <tr><td>Baden-Württemberg</td><td>"BW"</td></tr> <tr><td>Bayern</td><td>"BY"</td></tr> <tr><td>Berlin</td><td>"BE"</td></tr> <tr><td>Brandenburg</td><td>"BB"</td></tr> <tr><td>Bremen</td><td>"HB"</td></tr> <tr><td>Hamburg</td><td>"HH"</td></tr> <tr><td>Hessen</td><td>"HE"</td></tr> <tr><td>Mecklenburg-Vorpommern</td><td>"MV"</td></tr> <tr><td>Niedersachsen</td><td>"NI"</td></tr> <tr><td>Nordrhein-Westfalen</td><td>"NW"</td></tr> <tr><td>Rheinland-Pfalz</td><td>"RP"</td></tr> <tr><td>Sachsen</td><td>"SN"</td></tr> <tr><td>Sachsen-Anhalt</td><td>"ST"</td></tr> <tr><td>Saarland</td><td>"SL"</td></tr> <tr><td>Schleswig-Holstein</td><td>"SH"</td></tr> <tr><td>Thüringen</td><td>"TH"</td></tr> <tr><td> </td><td></td></tr> <tr><td>Bundesdienststellen</td><td>"BU"</td></tr> <tr><td>Bundesamt für Kartographie und Geodäsie</td><td>"BKG"</td></tr> <tr><td> </td><td></td></tr> <tr><td>Vorläufiger Identifikator</td><td>"_"</td></tr> </table> <p>Zulässige Zeichen sind: A-Z, a-z, 0-9, _ ohne Umlaute und ohne ß</p>	Baden-Württemberg	"BW"	Bayern	"BY"	Berlin	"BE"	Brandenburg	"BB"	Bremen	"HB"	Hamburg	"HH"	Hessen	"HE"	Mecklenburg-Vorpommern	"MV"	Niedersachsen	"NI"	Nordrhein-Westfalen	"NW"	Rheinland-Pfalz	"RP"	Sachsen	"SN"	Sachsen-Anhalt	"ST"	Saarland	"SL"	Schleswig-Holstein	"SH"	Thüringen	"TH"	 		Bundesdienststellen	"BU"	Bundesamt für Kartographie und Geodäsie	"BKG"	 		Vorläufiger Identifikator	"_"
Baden-Württemberg	"BW"																																												
Bayern	"BY"																																												
Berlin	"BE"																																												
Brandenburg	"BB"																																												
Bremen	"HB"																																												
Hamburg	"HH"																																												
Hessen	"HE"																																												
Mecklenburg-Vorpommern	"MV"																																												
Niedersachsen	"NI"																																												
Nordrhein-Westfalen	"NW"																																												
Rheinland-Pfalz	"RP"																																												
Sachsen	"SN"																																												
Sachsen-Anhalt	"ST"																																												
Saarland	"SL"																																												
Schleswig-Holstein	"SH"																																												
Thüringen	"TH"																																												
Bundesdienststellen	"BU"																																												
Bundesamt für Kartographie und Geodäsie	"BKG"																																												
Vorläufiger Identifikator	"_"																																												
3	Suffix (8 Zeichen)	Laufende Nummer	<p>Zulässige Zeichen sind: A-Z, a-z, 0-9 ohne Umlaute und ohne ß</p>																																										

Beispiele für Identifikatoren sind:

"DENW123412345678" (endgültiger Identifikator)

"DE_0000000000001" (vorläufiger Identifikator)

Zur Realisierung einer Geodateninfrastruktur im Sinne und unter Nutzung der Schnittstellendefinitionen des Open Geospatial-Consortiums (OGC) müssen alle beteiligten Stellen eine Systematik für die Vergabe der Identifikatoren und ein Service-Interface definieren, sodass sichergestellt ist, dass Objekte über ihren Identifikator ohne weiteres Wissen gefunden werden können. Hier bietet sich im Sinne einer bundesweiten Lösung ein gemeinsamer Service an; die Systematik der Vergabe und Verteilung kann unberührt davon länderspezifisch festgelegt werden.

Um Relationen zwischen den Fachobjekten im Datenaustausch aufzubauen werden Identifikatoren auch als Referenzen auf Fachobjekte geführt.

Identifikatoren sind unter anderem auch erforderlich, um bei der Formulierung von Fortführungen angeben zu können, welche Objekte gelöscht und welche Objekte überschrieben werden sollen. Da die Objekte dabei in ihrer konkret vorliegenden Version angesprochen werden müssen, wird der o.a. Identifikator in diesen Fällen um die Angabe von Entstehungsdatum und -zeit der angesprochenen Objektversion ergänzt. (Vgl. 10.1.3).

Eine wichtige Voraussetzung für die gemeinsame Führung von Datenbeständen unterschiedlicher Herkunft ist, dass die Integrationssituation in Form von Referenzen zwischen den Daten der Vermessungsverwaltung und den Fachdaten abgebildet ist (**Verknüpfung**). Diese Verknüpfung kann einseitig in den raumbezogenen Basisinformationssystemen der Vermessungsverwaltung oder im Fachinformationssystem (einseitige Verknüpfung) oder gegenseitig in beiden Informationssystemen (gegenseitige Verknüpfung) erfolgen. Als Verknüpfungsmerkmale sind eindeutige Bezeichnungen zu definieren und zu führen. Diese können aus den o.a. Identifikatoren und / oder aus Fachkennzeichen der jeweiligen Datenbestände bestehen.

3.4 Historie, Versionskonzept

Bei den AFIS-ALKIS-ATKIS-Daten besteht teilweise die Anforderung, Versionen und historische Daten zu führen. Der Umfang der Nutzung hängt vom Informationssystem und seiner Anwendung in den Ländern ab. Eine wesentliche Anwendung des Versionskonzeptes stellt das Verfahren zur Nutzerbezogenen Bestandsdatenaktualisierung (NBA) dar.

Das Versionskonzept wurde unter Berücksichtigung folgender Modellierungsgrundsätze erarbeitet:

- Im Anwendungsschema wird nicht zwischen aktuellen und historischen Daten unterschieden, d.h. bei der Vollhistorie werden keine eigenen historischen Objektarten gebildet.
- Zu jedem Objekt sind neben den aktuellen auch die historischen Informationen gespeichert (Versionen).
- Die zum Teil redundante Speicherung von Attributen eines Objekts in mehreren Versionen wird zugunsten eines schnelleren Datenzugriffs auf die entsprechende Version in Kauf genommen.

Das Versionskonzept geht davon aus, dass jedes Fachobjekt einen Identifikator, Attribute und Relationen sowie ein Lebenszeitintervall führt (Entstehungs- und Untergangsdatum). Mit dem Eintrag eines Objekts in die Bestandsdaten wird die erste Version des Objekts erzeugt und in einen Objektbehälter eingetragen. Ändert sich aufgrund einer Fortführung eine nicht objektbildende Eigenschaft, so wird eine neue Version des Objekts erzeugt, die historisch gewordene erste Version bleibt jedoch innerhalb des Objektbehälters bestehen, d.h. der Identifikator wird nicht geändert. Die neue Version erhält ein Entstehungsdatum, das gleichzeitig das Untergangsdatum der vorhergehenden Version ist. Die einzelnen Versionen eines Objekts können anhand des Lebenszeitintervalls eindeutig unterschieden werden. Durch Auswertungen der verschiedenen Versionen eines Objekts lassen sich alle Veränderungen bezogen auf einen beliebigen Zeitraum ermitteln.

Werden bei einer Fortführung objektbildende Eigenschaften geändert, führt dies aus fachlicher Sicht zum Untergang eines Objekts. Das Objekt wird historisiert, indem der letzten Version ein Untergangsdatum zugewiesen wird. Das Objekt bleibt weiterhin im Datenbestand erhalten. Zu einem beliebigen Zeitpunkt hat eine Version alle zu diesem Zeitpunkt gültigen Attribute und Relationen. Durch „Klammerung“ der Versionen innerhalb eines Objektbehälters bleibt die fachliche Objektsicht stets erhalten.

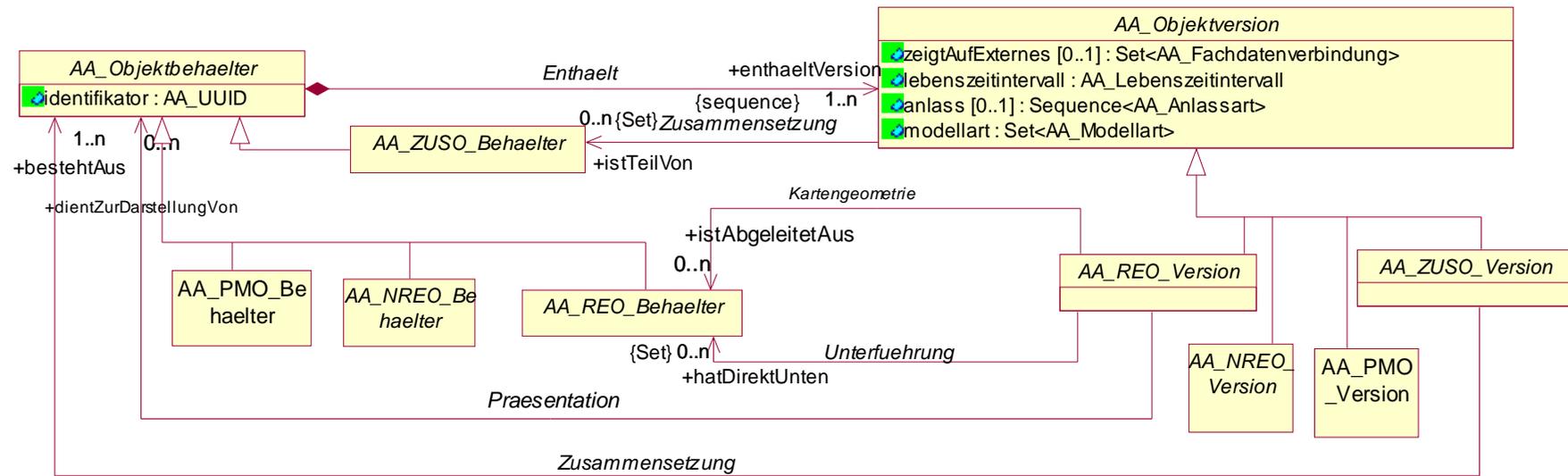


Abbildung 3-14: Versionierungsschema

Festlegung objektbildender Eigenschaften im AAA-Modell in UML

Im AAA-Modell in UML sind bei Attributen und Relationen Aussagen untergebracht, die festlegen, wie sich darauf bezogene Fortführungen auswirken und dann zum Untergang bzw. zur Historisierung von Objekten führt. Diese finden sich im jeweils zum/zur Attribut/Relation gehörigen Reiter "AAA" im Feld "objektbildend" in Form der Belegung mit dem Wert True oder False. Alle Eintragungen im AAA-Anwendungsschema wurden durch die AdV vorgenommen und haben daher normativen Charakter. Die folgende Tabelle enthält Erläuterungen zu den getroffenen Festlegungen:

Festlegung im UML-Modell (Reiter AAA / <i>objektbildend</i> zum/zur jew. Attribut/Relation)	Status der Festlegung im Reiter AAA / objektbildend
True	Unabänderliche AdV-Festlegung
False	AdV-seitig vorgegebener Rahmen, d.h. es kann länderspezifisch True oder False gesetzt werden*.

(*Diese Möglichkeit der Unterbringung länderspezifischer Festsetzungen ist im AAA-Modell die absolute Ausnahme).

Festlegung im UML-Modell (Reiter AAA / <i>objektbildend</i> zum/zur jew. Attribut/Relation)	Auswirkungen der Festlegung von Spalte 1 bei Veränderung betroffener Attribute/Relationen in einer Erhebungskomponente
True	Neues Objekt entsteht (Delete + Insert)
False	Neue Objektversion entsteht (Replace)

Länderspezifisch können nur „Verschärfungen“ vorgenommen werden, d.h. im AAA-Anwendungsschema optionale und nicht zum Grunddatenbestand gehörende Elemente können zusätzlich zum AdV-Grunddatenbestand dem länderspezifischen Grunddatenbestand zugeordnet werden.

Beispiel zum Versionskonzept

Änderung von Attributen

Frau Hilde Huber wird zum Zeitpunkt t_1 in ALKIS eingetragen, d.h. es wird ein neues Objekt der Objektart *Person* gebildet:

	Identifikator	Zeitintervall		Name	Vorname	hat_Anschrift
		Beginn	Ende			
Version 1	DEBU5t44dFzb70Lg	t_1	t_∞	Huber	Hilde	DEBUf88FFgVc761s

Die Zeitangabe 't_∞' bedeutet, dass der fachliche Untergang des Objekts bzw. der Version in der Zukunft liegt. Zum Zeitpunkt t₂ ändert Frau Huber ihren Namen und heißt nun Meier, d.h. vom Objekt „DEBU5t44dFzb70Lg“ der Objektart *Person* wird aufgrund der Änderung des Attributs *Name* eine neue Version angelegt:

	Identifikator	Zeitintervall		Name	Vorname	hat_Anschrift
		Beginn	Ende			
Version 1	DEBU5t44dFzb70Lg	t ₁	t ₂	Huber	Hilde	DEBUf88FFgVc761s
Version 2	DEBU5t44dFzb70Lg	t ₂	t _∞	Meier	Hilde	DEBUf88FFgVc761s

Der Zeitpunkt des Untergangs der Version 1 ist identisch mit dem Entstehungsdatum der Version 2 des Objekts. Zum Zeitpunkt t_x verkauft Frau Meier ihr einziges Grundstück. Da sie sonst keine weitere Rolle in ALKIS innehat, geht das Objekt aus fachlicher Sicht unter:

	Identifikator	Zeitintervall		Name	Vorname	hat_Anschrift
		Beginn	Ende			
Version 1	DEBU5t44dFzb70Lg	t ₁	t ₂	Huber	Hilde	DEBUf88FFgVc761s
Version 2	DEBU5t44dFzb70Lg	t ₂	t ₃	Meier	Hilde	DEBUf88FFgVc761s

Die Version 2 und damit das gesamte Objekt werden historisiert, nicht gelöscht.

Jede neue Version eines Objektes erhält eigene Relationen, die von ihr ausgehen. Relationen gehen stets von einer bestimmten Version des Objektes aus, d.h. eine Relation von einer Version zu einem anderen Objekt ist nur für diese eine Version gültig. Auf diese Weise werden sämtliche im Objektartenkatalog spezifizierten Kardinalitäten eingehalten.

In der Abbildung 3-14 wird das erläutert. Frau Hilde Huber, Anschrift Ottostraße 17 in München, wird zum Zeitpunkt t₁ in ALKIS eingetragen, d.h. es werden ein Objekt der Objektart *Person* und ein Objekt der Objektart *Anschrift* gebildet. Zum Zeitpunkt t₂ ändert Frau Huber ihren Namen und heißt fortan Meier. Es wird eine neue Version des Objektes *Person* angelegt.

In der Abbildung repräsentieren die Pfeile eine Relation. Die Richtung des Pfeils gibt gleichzeitig die Richtung der Relation an. Die neue Version des Objektes *Person* erhält wiederum eine Relation zum entsprechenden Objekt *Anschrift*. Das Objekt *Anschrift* selbst wird allerdings nicht versioniert, da die Relation zum Objekt *Person* unverändert bleibt. Ebenso würde eine neue Version des Objektes *Anschrift*, z. B. durch Berichtigung nach einem Schreibfehler, keine Änderung des Objektes *Person* bewirken.

An diesem Beispiel ist auch erkennbar, dass eine Relation stets von der Version über den Identifikator auf den Objektbehälter zeigt und nicht auf eine Version. Der Objektbehälter bildet somit eine Art Klammer um seine verschiedenen Versionen.

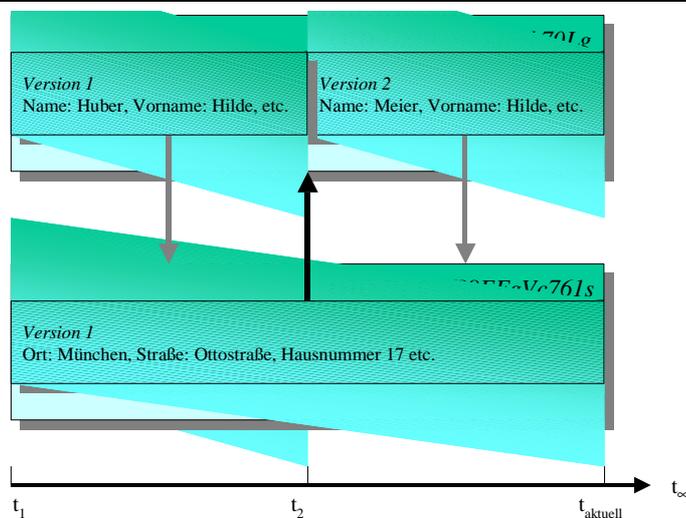


Abbildung 3-15: Beispiel zur Versionierung nach Änderung von Attributen

Mit dieser Technik können nur Relationen abgebildet werden, die sich auf die jeweils aktuelle Version der beteiligten Objekte beziehen. Sollte dies im konkreten Fall nicht ausreichen, so kann eine Version ausnahmsweise auch unmittelbar referenziert werden; hierzu ist der Identifikator in der Referenz um den Zeitstempel der Version zu ergänzen.

Änderung von Relationen

Änderungen bei Relationen führen ebenso zur Versionierung von Objekten wie Attributänderungen. Relationen ändern sich immer dann, wenn das Objekt, auf das die Relation zeigt, neu entsteht, ausgetauscht wird oder wegfällt.

In einem modifizierten Beispiel zur Abbildung 3-14 wird dies erläutert. Frau Hilde Huber zieht zum Zeitpunkt t_3 um von der Ottostraße 17 in München zur Platanenallee 34a in Berlin. Das Objekt *Anschrift* mit der OID "DEBUf88FFgVc761s", auf welches die Relation *hat_Anschrift* vom Objekt *Person* ausgehend zeigt, wird ausgetauscht (neue OID "DEBUk41233THjbkO"). Damit ändert sich die betreffende Relation beim Objekt *Person* und das Objekt *Person* muss versioniert werden.

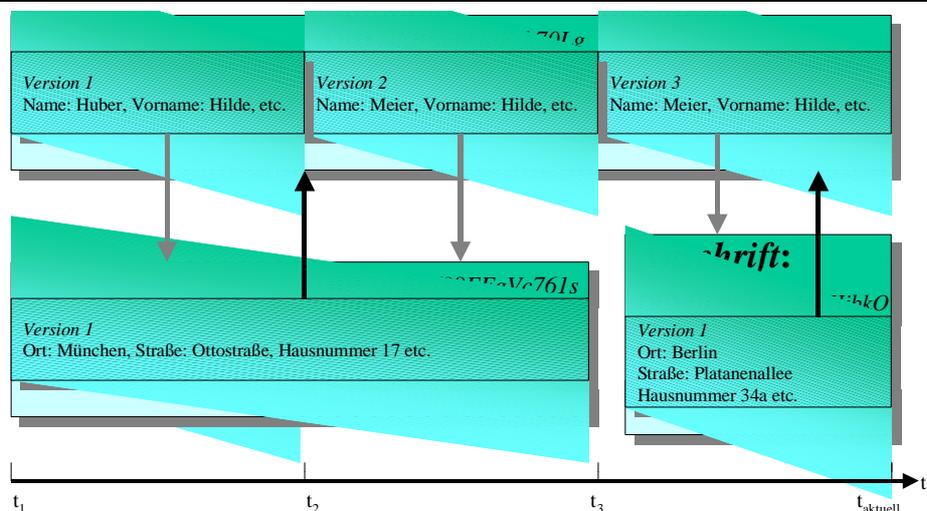


Abbildung 3-16: Beispiel zur Versionierung nach Änderung von Relationen

Tabellarisch ergibt sich folgendes Bild:

	Identifikator	Zeitintervall		Name	Vorname	hat_Anschrift
		Beginn	Ende			
Version 1	DEBU5t44dFzb70Lg	t ₁	t ₂	Huber	Hilde	DEBUf88FFgVc761s
Version 2	DEBU5t44dFzb70Lg	t ₂	t ₃	Meier	Hilde	DEBUf88FFgVc761s
Version 3	DEBU5t44dFzb70Lg	t ₃	t _∞	Meier	Hilde	DEBUk41233THjkbO

3.5 Qualitäts- und Metadaten

Das gemeinsame AFIS-ALKIS-ATKIS-Datenmodell sieht die Erfassung und Führung von Qualitäts- und Metadaten auf der Grundlage der ISO-Normen

- ISO 19109 Geographic Information – Rules for Application Schema
- ISO 19113 Geographic Information – Quality Principles,
- ISO 19114 Geographic Information – Quality Evaluation Procedures und
- ISO 19115 Geographic Information – Metadata

vor.

Die **Qualitätsdaten** werden dabei nach nicht quantifizierbaren Überblicksinformationen (Zweck, Verwendung und Historie) und quantifizierbaren Informationen (den Datenqualitäts - Elementen *Vollständigkeit*, *logische Konsistenz*, *geometrische*, *inhaltliche* und *zeitliche Genauigkeit*) unterschieden.

Die Angabe der Qualitätsinformationen erfolgt als Metadaten gemäß der Norm ISO 19115 und darüber hinaus für quantitative, aggregierte Qualitätsangaben bei Bedarf in Form von detaillierten Qualitäts - Bewertungsprotokollen gemäß Norm ISO 19114.

Metadaten sind „Daten über Daten“ und dienen der Beschreibung der Geodaten hinsichtlich nutzerrelevanter Aspekte zur Bewertung der Eignung der Daten und des Zugriffs auf dieselben. ISO unterscheidet etwa 400 optionale, obligatorische und bedingt obligatorische Metadatenelemente, gegliedert in inhaltliche Einheiten (entities) sowie in die folgenden Abschnitte (sections):

- Identifikation,
- Datenqualität,
- Fortführung,
- Raumbezogene Eigenschaften,
- Referenzsystem,
- Ausdehnung,
- Inhalt,
- Anwendungsschema,
- Signaturenkatalog,
- Vertrieb,
- Nutzungsbedingungen.

Qualitäts- und Metadaten können gemäß ISO für einen Datenbestand (Sammlung von logisch zusammengehörigen Objekten), für Berichtsgruppen (Teilmengen eines Datenbestandes) und für einzelne Objekte angegeben werden.

Der gemeinsame AFIS-ALKIS-ATKIS-Metadatenkatalog ist in Kap. 9 beschrieben.

3.6 Objektartenkatalog

Die Struktur der Objektartenkataloge ist durch die ISO - Norm 19110 *Feature Cataloguing Methodology* vorgegeben. Aufgrund der Objektorientierung ist es auch möglich, die Methoden im Objektartenkatalog zu beschreiben. Das gemeinsame Anwendungsschema erweitert diese Strukturen im Paket AAA - *Katalog* um einige Inhalte, die für die Anwendungen AFIS, ALKIS und ATKIS zusätzlich benötigt werden.

Kataloge werden zur einfacheren Implementierung ausschließlich vollständig versioniert und ausgetauscht.

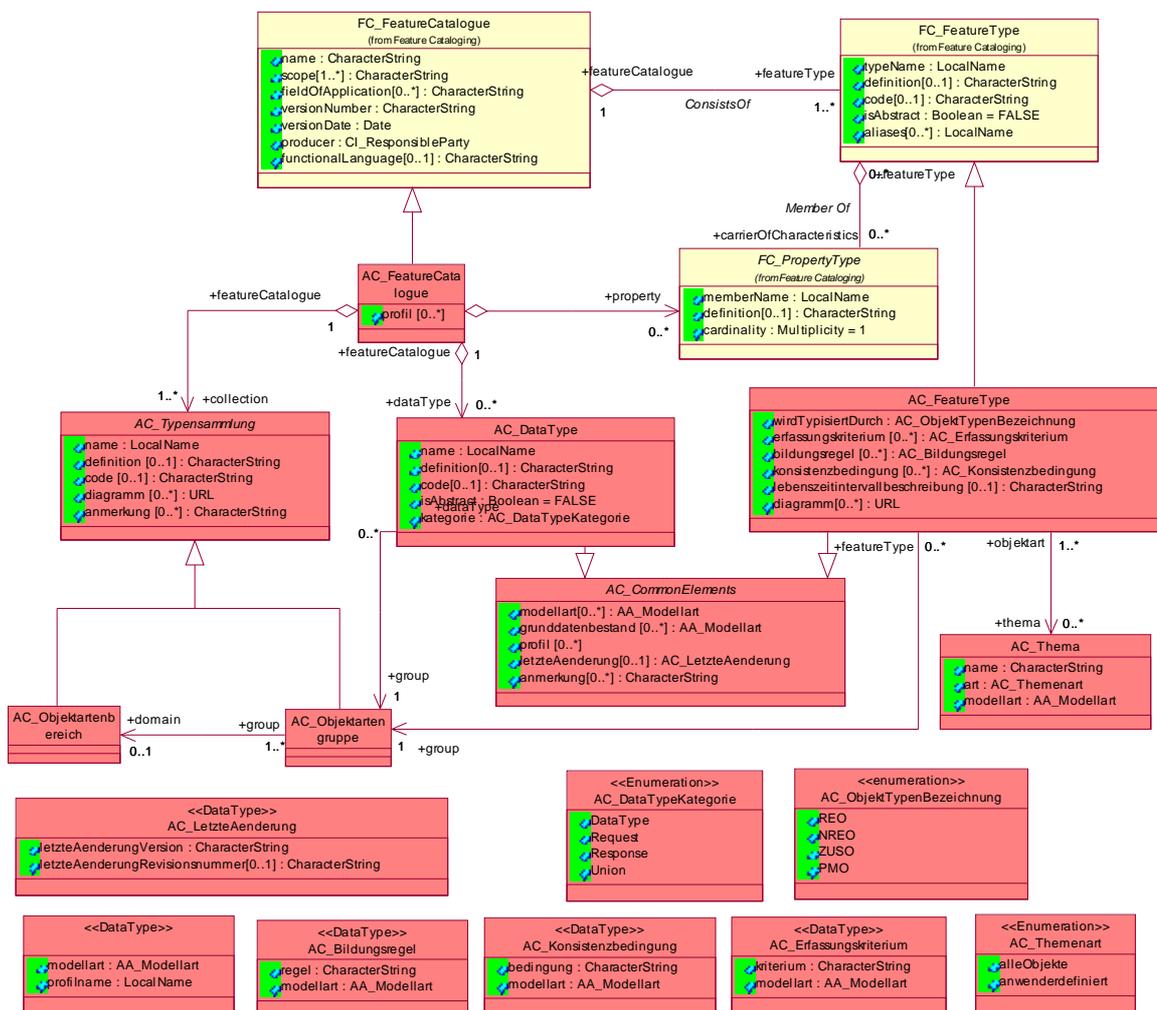


Abbildung 3-17: Erweiterungen der genormten Struktur der Objektartenkataloge

Die Elemente mit dem Präfix „FC_“ stammen aus der ISO-Norm, die Elemente mit dem Präfix „AC_“ sind AdV-spezifische Erweiterungen.

3.7 Prozesse, Vorgänge und Aktivitäten

3.7.1 Grundsätze

Im Rahmen der Zuständigkeit des amtlichen Vermessungswesens sind die Aufgaben Erhebung, Qualifizierung, Führung (Ersteinrichtung, Fortführung), Benutzung und Übertragung von Daten auszuführen. Jede dieser Aufgaben äußert sich in einem oder mehreren Prozessen. Es gibt Erhebungs-, Qualifizierungs-, Führungs-, Benutzungs- und Transferprozesse.

Die Geoinformationen des amtlichen Vermessungswesens bestehen aus den originären Bestandsdaten und den temporären Datenbeständen der Erhebungsdaten, Fortführungsdaten, Ausgabedaten und Transferdaten.

Die Projektsteuerung im AAA-Basisschema steuert den Ablauf aller Prozesse in Form von Vorgängen und Aktivitäten, womit vollständige Geschäftsprozesse beschrieben werden können. Sie stellt lediglich eine optionale Rahmenvorgabe dar, die inhaltlich durch die länderspezifischen Geschäftsprozesse zu untersetzen ist. In der Abbildung 3-18 werden die Prozesse und Daten der Geoinformationen des amtlichen Vermessungswesens dargestellt. Die im Rahmen des AdV-Projektes "Modellierung der Geoinformationen des amtlichen Vermessungswesens" fachlich zu modellierenden Bestandteile werden von einer gestrichelten Linie umrahmt.

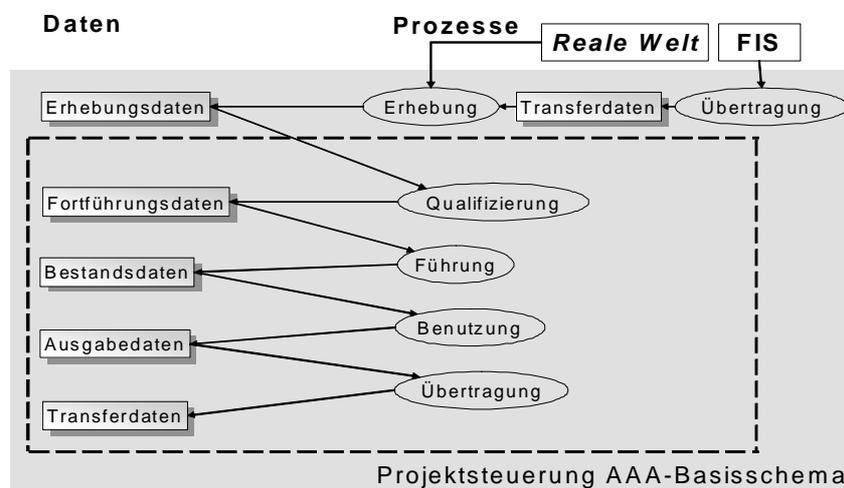


Abbildung 3-18: Prozesse und Daten der Geoinformationen des amtlichen Vermessungswesens

Zu einem Prozess gehören mehrere aufeinander aufbauende Aktivitäten, die zu Vorgängen zusammengefasst und fachlich gegliedert werden können. Zur Beschreibung der Prozesse (Vorgänge und Aktivitäten) werden folgende Sprachmittel verwendet:

- Aktivitäten als Bestandteil der UML-Klassen
- Textliche Beschreibung der Bearbeitungsschritte
- Sequenzdiagramme

Die graphische Darstellung der Vorgänge erfolgt entsprechend der UML-Notation in Sequenzdiagrammen

- Filterencoding-Ausdrücke (s. Abschn. 10.2.2) zur Beschreibung der Selektions- und Auswertefunktionalität bei der Erstellung von Standardausgaben (Benutzungsprozess).

3.7.2 Vorgang und Aktivität

Für eine vollständige Anwendungsbeschreibung sind Vorgänge und Aktivitäten zu definieren, die Daten in funktionelle Abhängigkeiten setzen und das dynamische Verhalten der Anwendung definieren. Vorgänge sind den einzelnen Prozessen im AAA-Anwendungsschema zugeordnet. Dies kann aus der folgenden Abbildung entnommen werden.

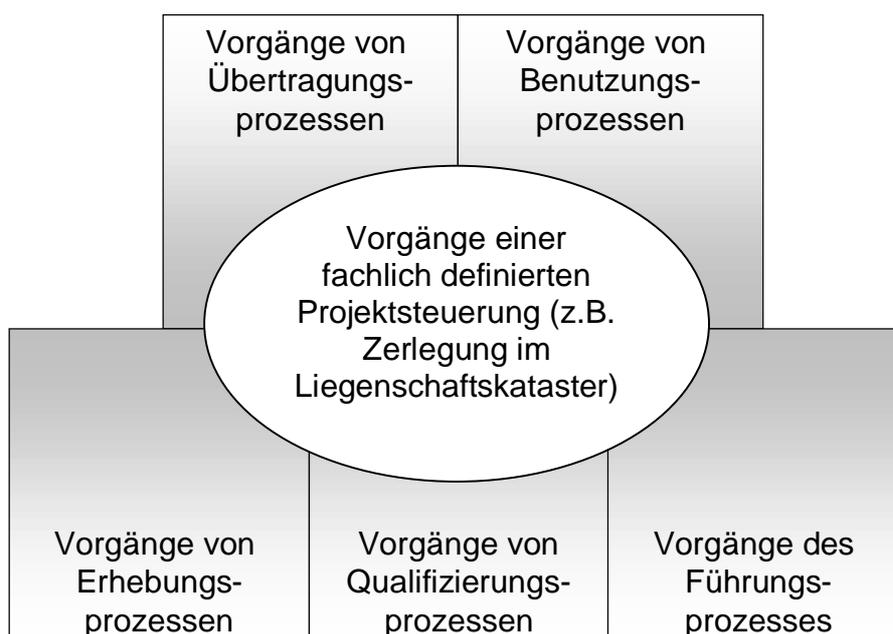


Abbildung 3-19: Vorgänge im AAA-Anwendungsschema

Ein Vorgang beinhaltet die Darstellung von Bearbeitungsschritten der Prozesse Qualifizierung, Führung, Benutzung und Übertragung, in denen auf verschiedene Aktivitäten verwiesen wird.

Eine Aktivität beschreibt das Verhalten eines Objekts und besteht aus einer Sequenz von Anweisungen. Den Anstoß dazu erhält ein Objekt durch eine Nachricht, die durch Eingaben des Nutzers oder durch Aktivitäten anderer Objekte ausgelöst werden (Eingabeparameter). Das Ergebnis der Aktivität wird in Form von Ausgabeparametern bereitgestellt. Aktivitäten werden objektbezogen definiert und sind im UML-Modell Bestandteil einer Klasse.

3.7.3 Prozesse des AFIS-ALKIS-ATKIS-Anwendungsschemas

Mit einem Prozess wird ein Quelldatenbestand in einen Zieldatenbestand überführt. Der Benutzungsprozess beispielsweise überführt die Bestandsdaten in temporäre Ausgabedaten.

Zur Steuerung der verschiedenen Prozesse werden spezielle Klassen gebildet (siehe auch Abschnitt 10.2), die Steuerungsparameter für den Ablauf von Prozessen beinhalten, wie z.B. "AX_Benutzungsauftrag" im Benutzungsprozess des ALKIS-Anwendungsschemas.

3.7.3.1 Erhebungsprozess

Quelldaten werden mit den bekannten geodätischen Mess- und Erkundungsmethoden in der realen Welt erhoben oder aus kartographischen Darstellungen und anderen Unterlagen erfasst. Die Zieldaten des Erhebungsprozesses sind die objektstrukturierten Erhebungsdaten, die eine Grundlage zur Fortführung der amtlichen Geoinformationen bilden.

3.7.3.2 Qualifizierungsprozess

Im Qualifizierungsprozess werden die digitalen, objektstrukturierten Erhebungsdaten nach einer Qualifizierung in Fortführungsdaten überführt. Er dient der Qualitätssicherung und stellt sicher, dass die Fortführungsdaten den Qualitätsanforderungen entsprechen.

Die Zieldaten des Qualifizierungsprozesses sind die Fortführungsdaten.

3.7.3.3 Führungsprozess

Im Führungsprozess sind Ersteinrichtung und Fortführung der Geoinformationen zusammengefasst, wobei die Ersteinrichtung als Sonderfall der Fortführung betrachtet werden kann. Beim Führungsprozess werden die Fortführungsdaten (Daten und Metadaten) durch Anwendung geeigneter Methoden in den Bestand überführt.

Die Zieldaten des Führungsprozesses sind die Bestandsdaten.

Die für die Einrichtung und Fortführung notwendigen Funktionalitäten sind im Rahmen der Austauschchnittstelle in 10.2, darüber hinaus gehende implizite Funktionen eines Führungssystems in Abschnitt 10.4 beschrieben. Das konzeptuelle Fachmodell für die Fortführung von ALKIS sowie die exakten Abläufe bei der Fortführungsverarbeitung sind in der Dokumentation zur Objektart „AX_Fortfuehrungsauftrag“ enthalten. Ferner zeigt ein Sequenzdiagramm eine beispielhafte Illustration der Beschreibung zum „AX_Fortfuehrungsauftrag“.

3.7.3.4 Benutzungsprozess

Benutzungsprozesse überführen Bestandsdaten in Ausgabedaten entsprechend den fachlichen Vorgaben

- in Form von Bestandsdatensätzen zur universellen Weiterverarbeitung beim Nutzer,
- als aufbereitete Bestandsdaten mit festgelegtem Inhalt in einem einheitlichen Erscheinungsbild des amtlichen Vermessungs- und Katasterwesens (Präsentationsausgaben, Auswertungen etc.) sowie
- als Änderungsdaten nach der Fortführung (Nutzerbezogene Bestandsdatenaktualisierung - NBA).

Eine Ausgabe kann Bestandsobjekte sowie temporär erzeugte Objekte beinhalten.

Zur Strukturierung der Ausgabedaten und für Elementangaben, die nicht aus den Attributarten des Bestandes entnommen werden können, die aber für eine Ausgabe notwendig sind, werden temporäre Objektarten gebildet (z.B. AX_Flurstueck_Grundbuch). Temporäre Objektarten sind keine AA_Objekte, sondern Datentypen. Sie besitzen keinen Identifikator und kein Lebenszeitintervall. Sie werden daher nicht im Bestand geführt.

Die temporäre Prozessobjektart „AX_Benutzungsauftrag“ des Anwendungsschemas enthält wesentliche Angaben zur Steuerung des Benutzungsprozesses, wie Umfang der Ausgabe, Antragsnummer, Anlassart, Benutzungsparameter, Ausgabenname usw. und wird zu Beginn des Benutzungsprozesses erzeugt. Durch die Attributart „Benutzungsparameter“ werden die erforderlichen Parameter für die Kosten- und Gebührenberechnung, die außerhalb der Geoinformationen des amtlichen Vermessungswesens vorgenommen wird, bereitgestellt. Die übrigen für eine Ausgabe notwendigen temporären Objektarten entstehen durch Methoden innerhalb des Benutzungsprozesses aus den Bestandsobjektarten.

Die temporären Objektarten, insbesondere die temporären Ausgabe-Objektarten, werden so modelliert, dass Relationen innerhalb einer Ausgabe vermieden werden.

Ausgabeobjektarten können je nach Anforderung auch unter Beachtung des Signaturenkatalogs präsentiert werden (siehe Kapitel 3.3.5). Danach werden z. B. für eine Präsentation ausgabe die Objektarten der Bestandsdaten an Hand einer definierten Abfrageabfolge (XSLT-Skripte) aufbereitet zu einer temporären Ausgabeobjektart entwickelt. Anschließend erfolgt unter Berücksichtigung der erforderlichen Signaturierung und des Präsentationslayouts eine Ausgabe am Bildschirm bzw. als Druckausgabe. Möglich ist aber auch die Abgabe von nicht aufbereiteten Ausgabedatensätzen, die von Nutzern unter Verwendung eigener Layoutvorgaben aufbereitet werden können.

3.7.3.5 Transferprozess

Übertragungsprozesse treten bei der Übernahme von Daten Dritter in Form von Fortführungsdaten und bei der Abgabe von Ausgabedaten an Kunden auf. Übertragungsprozesse zur Datenübernahme empfangen Ausgaben der Systeme Dritter einschließlich Transferfunktionen in Form von Transferdaten. Übertragungsprozesse zur Datenabgabe ergänzen Ausgabedaten um Transferfunktionen und erzeugen aus ihnen Transferdaten für Systeme Dritter.

3.8 Projektsteuerung

Die im Paket "AAA_Projektsteuerung" definierten Klassen beschreiben einen Strukturrahmen zur Beschreibung einer Projektsteuerung. Die Klassendiagramme "AA_Antrag", "AA_Projektsteuerungskatalog" und "AA_Meilenstein" zeigen das Konzept der modellierten Projektsteuerung.

3.8.1 Antrag

Dreh- und Angelpunkt der Projektsteuerung ist die Objektart AA_Antrag. Diese Objektart realisiert eine "Mini-Antragsverwaltung", d.h. eine Schnittstelle zur externen Antragsverwaltung. Dadurch wird es möglich, bei einem Eintrag in der externen Antragsverwaltung (Geschäftsbuch) direkt einen Bezug zu diesem Antrag (mit Raumbezug) zu generieren.

Das Antragsobjekt verwaltet außerdem die Wiedervorlage des Antrags und unterstützt die Überwachung der Projektsteuerungs-Objekte. Mit dem Raumbezug kann nach bestehenden Prozessen gesucht werden, um konkurrierende Anträge zu ermitteln oder um andere benachbarte Anträge bei der Bearbeitung zu berücksichtigen. Die fachliche Reihenfolge konkurrierender Anträge ist durch den Sachbearbeiter festzulegen. Das Antrags-Objekt wird mit dem Projektsteuerungs-Objekt (AA_Projektsteuerung) verbunden, um die Zuordnung des Antrags zu einem oder mehreren Projektsteuerungs-Objekten festzulegen und um die nicht zulässigen Kombinationen zu überwachen. Weiterhin steuert und überwacht das Projektsteuerungs-Objekt die korrekte Abwicklung der Vorgänge im Teilprozess "fachtechnische Qualifizierung". Die Fortführungsanlässe werden beim Projektsteuerungs-Objekt geführt.

Der Vorgang ist Teil einer Projektsteuerung und setzt sich aus einzelnen Aktivitäten zusammen. Die Vorgänge stellen in sich abgeschlossene Arbeitsschritte dar. Ein vorzugebender Arbeitsablauf ("Workflow") legt die Reihenfolge und Abhängigkeiten der Vorgänge und deren Arbeitsschritte fest. Die Vorgänge werden in Gruppen zusammenge-

fasst und in einer bestimmten Reihenfolge nacheinander bzw. nebeneinander bearbeitet. Die Entscheidung über den Abschluss des einzelnen Vorganges wird im Status (Meilenstein) dokumentiert.

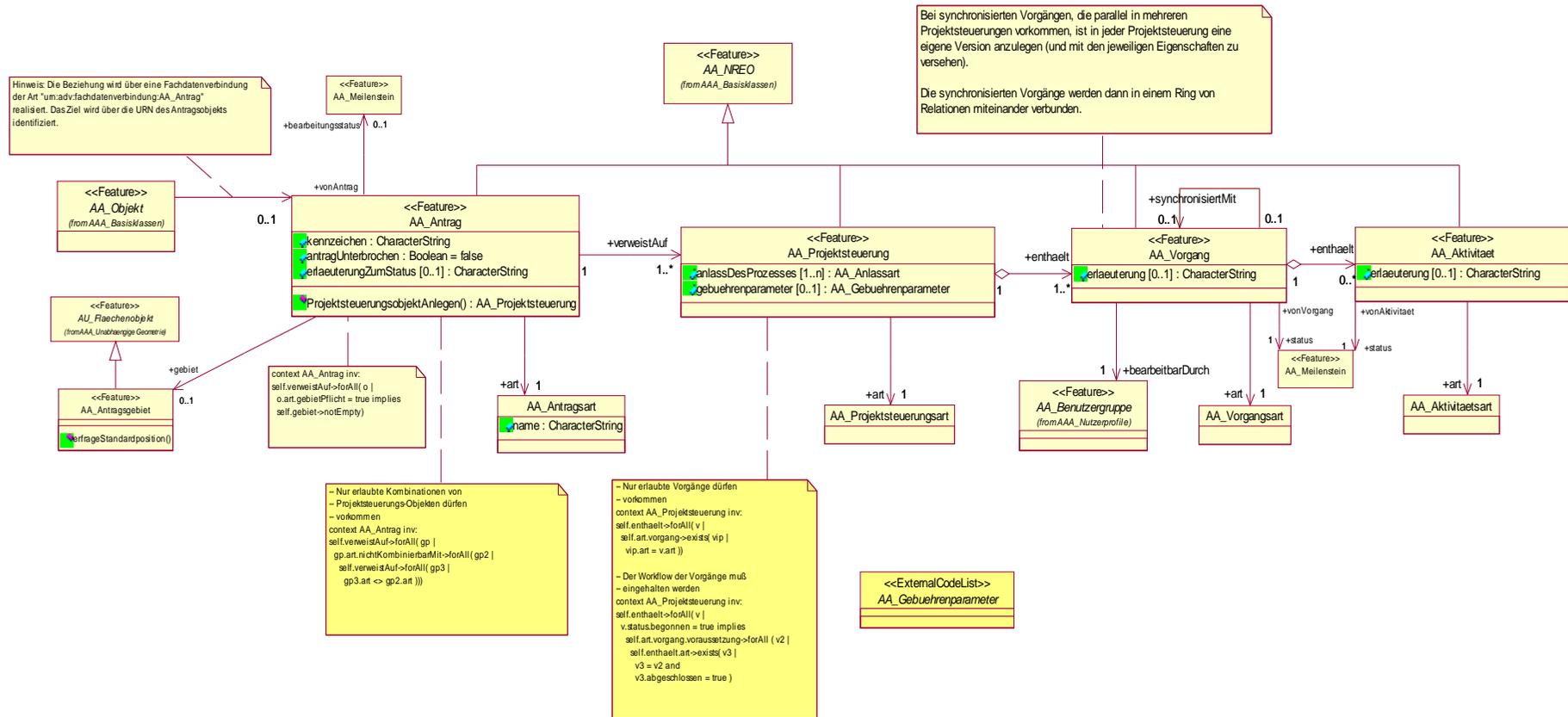


Abbildung 3-20: Klassendiagramm "AA_Antrag"

3.8.2 Projektsteuerungskatalog

Der Projektsteuerungskatalog definiert die innerhalb eines Projektsteuerungs-Objektes dieser Art erlaubten Fortführungsanlässe. Er beinhaltet die Projektsteuerungs- und Vorgangsarten. Die Projektsteuerungsart bündelt Projektsteuerungs-Objekte, die eine gemeinsame Charakteristik aufweisen. Analoges gilt für die Vorgangsart.

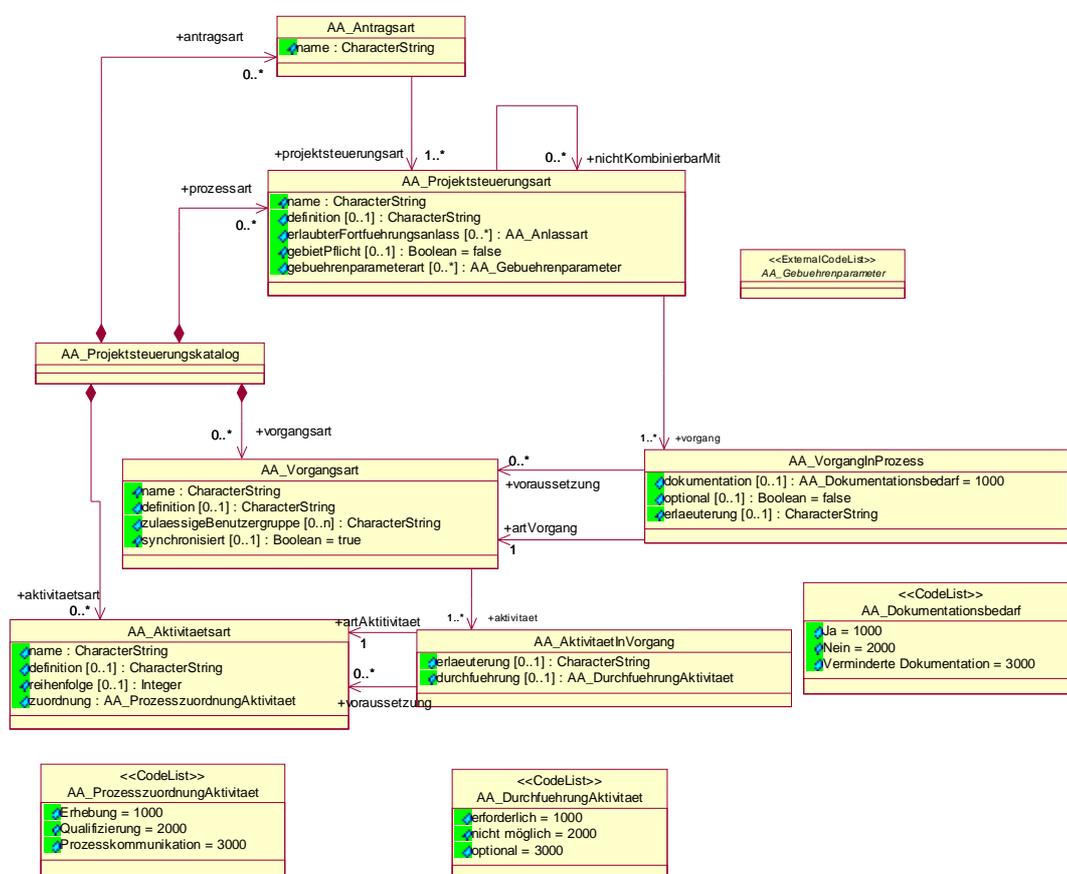


Abbildung 3-21: Klassendiagramm "AA_Projektsteuerungskatalog"

3.8.3 Meilenstein

Hierbei handelt es sich um einen Datentyp, der zu einem Vorgang usw. den aktuellen Zustand und die Verantwortlichkeiten vermerkt.

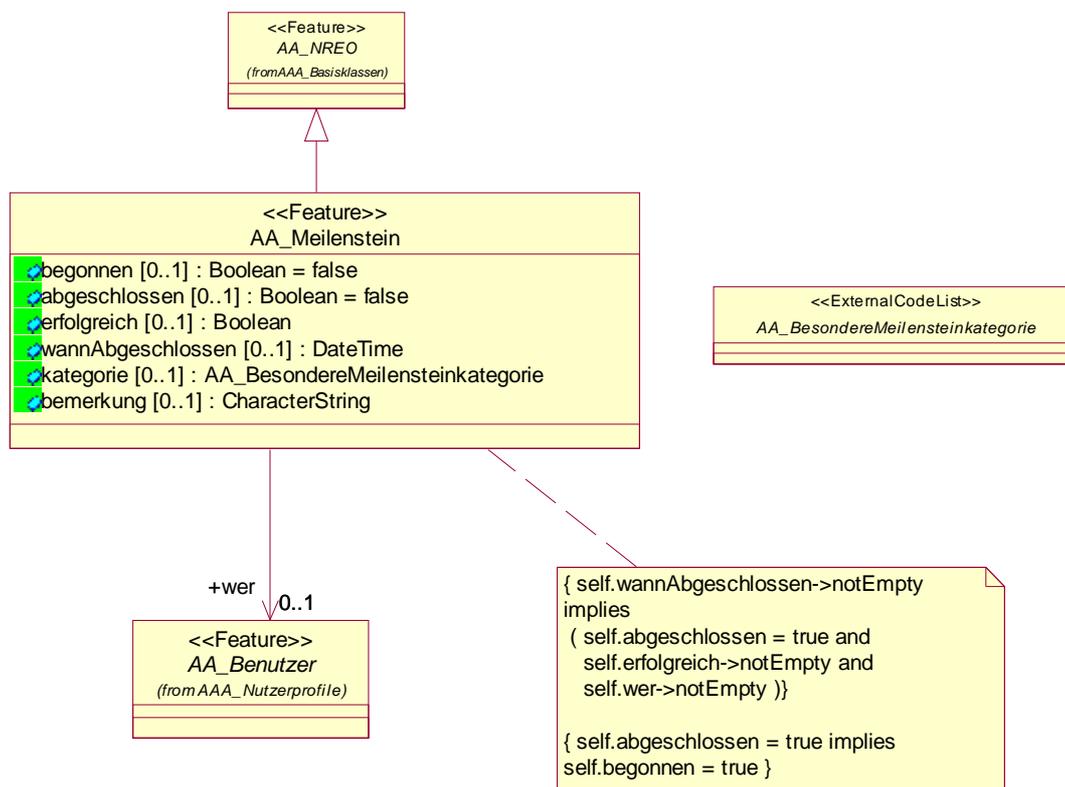


Abbildung 3-22: Klassendiagramm "AA_Meilenstein"

Eine anwendungsbezogene Erläuterung der Projektsteuerung des AAA-Basisschemas für das Fachschema ALKIS befindet sich in der GeoInfoDok, Kapitel 5.3 „Erläuterungen zu ALKIS“. Weitere Einzelheiten und Zusammenhänge können hieraus entnommen werden.

3.9 Grundsätze für die Verwendung des Filterencodings

Das Ziel der Verwendung von Filterencoding liegt in der Beschreibung der **formalen Regeln** bei der Ableitung der Standardausgaben - nicht in der Vorgabe einer algorithmischen Umsetzung dieser. Es ist selbstverständlich möglich und manchmal auch notwendig, entsprechende Optimierungen vorzunehmen, damit die Ausgaben zur Laufzeit erzeugt werden können. Entscheidend ist die logische und performante Füllung der Ausgabeobjektarten, nicht die direkte Implementierung der Filterausdrücke.

Beispiel:

```

<wfs:Query typeName="AX_Flurstueck">
  <ogc:Filter>

```

```

      <ogc:PropertyIsEqualTo>
        <ogc:PropertyName>flurstueckskennzeichen</ogc:PropertyName>
        <ogc:Literal><xsl:value-of select="$p3"/></ogc:Literal>
      </ogc:PropertyIsEqualTo>
    </ogc:Filter>
  </wfs:Query>
  <!-- Besorge ausserdem alle Objekte der weiteren Objektarten. -->
  <wfs:Query typeName="AX_Person"/>
  <wfs:Query typeName="AX_Anschrift"/>
  <wfs:Query typeName="AX_Buchungsblatt"/>
  <wfs:Query typeName="AX_Buchungsstelle"/>
  <wfs:Query typeName="AX_Namensnummer"/>
  <wfs:Query typeName="AX_Gemarkung"/>
  <wfs:Query typeName="AX_Flurstueck"/>

```

Diese Selektion könnte beispielsweise optimiert werden, indem gezielt der Katalogeintrag für die Gemarkung selektiert wird, in der das Flurstück liegt (z.B. über das entsprechende Flurstückskennzeichen) und nicht sämtliche Katalogeinträge.

Zum Codierung einer Selektion wird in der NAS das <wfs:Query>-Element aus der „Web Feature Service“-Spezifikation (WFS) des Open Geospatial Consortiums in der Version 1.0.0 verwendet. In einer Selektion können mehrere Queries vorkommen, wobei sich jede Query auf eine instanziierbare Objektart bezieht. Die unterschiedlichen Queries wirken ergänzend.

Die aktuelle WFS-Spezifikation unterstützt hierbei nur die Angabe der konkreten, instanziierten Objektarten, d.h. die im AAA-Anwendungsschema modellierte Vererbungshierarchie wird nicht unterstützt. Es ist also z.B. nicht möglich, eine einzige Query für „AX_TatsaechlicheNutzung“ abzusetzen, um alle TN-Objekte zu erfragen, sondern es muss ein <wfs:Query>-Element pro Objektart angegeben werden. Eine Erweiterung zur Unterstützung der Vererbungshierarchie wäre grundsätzlich möglich, aber erst einmal AdV-spezifisch.

In eine <wfs:Query> eingebettet ist u.a. ein <ogc:Filter>-Element zur Filterung der Objekte aus dem Gesamtumfang der Objektart. Ein <ogc:Filter>-Ausdruck besteht aus einem Prädikat, das für jedes Objekt der Objektart in der Datenbasis, auf der die Suche ausgeführt werden soll, angewendet wird. Erfüllt das Objekt das Prädikat ist es Teil der Selektion, ansonsten nicht. Die Prädikate sind entsprechend so zu verstehen, dass sie grundsätzlich auf den XML-Instanzen wirken, die diese Objekte repräsentieren.

Entsprechend besteht das Prädikat aus einem booleschen Ausdruck, der aus beliebig vielen atomaren Operatoren besteht, die über

- die logischen Operatoren

```
<ogc:And>
```

```
<ogc:Or>
```

<ogc:Not>

verbunden werden.

Bei den atomaren Operatoren werden

- räumliche Operatoren

<ogc:Equals>

<ogc:Disjoint>

<ogc:Touches>

<ogc:Within>

<ogc:Overlaps>

<ogc:Crosses>

<ogc:Intersects>

<ogc:Contains>

<ogc:DWithin>

<ogc:Beyond>

<ogc:BBOX>

und

- Vergleichsoperatoren

<ogc:PropertyIsEqualTo> (=)

<ogc:PropertyIsNotEqualTo> (<>)

<ogc:PropertyIsLessThan> (<)

<ogc:PropertyIsGreaterThan> (>)

<ogc:PropertyIsLessThanOrEqualTo> (<=)

<ogc:PropertyIsGreaterThanOrEqualTo> (>=)

<ogc:PropertyIsLike> (Textvergleich mit „Wildcards“ für ein oder mehrere Zeichen)

<ogc:PropertyIsNull> (Prüfung auf fehlenden Wert)

<ogc:PropertyIsBetween> (Kombination von >= und <=)

unterstützt. Die Bedeutung der logischen Operatoren und der Vergleichsoperatoren ergibt sich aus der in SQL verwendeten bzw. der direkt mit dem Namen ausgedrückten Bedeutung.

Die Bedeutung der räumlichen Operatoren ist i.d.R. in der OGC Simple Features Spezifikation definiert und in das Filter Encoding übernommen worden. Vermutlich ist `<ogc:Intersects>` der wichtigste Operator, der „true“ ergibt, wenn zwei Geometrien nicht überschneidungsfrei sind. `<ogc:BBOX>` ist eine vereinfachte Form, die als Testgeometrie nur eine Bounding Box erlaubt. `<ogc:Disjoint>` ist die Umkehrung zu `<ogc:Intersects>`. `<ogc:Contains>` oder `<ogc:Within>` sind zu verwenden, wenn es nicht um Überlappung geht, sondern um echtes Enthaltensein. Für weitergehende Fragen, siehe die OGC Spezifikationen Filter Encoding und Simple Features for SQL.

Bei räumlichen Operatoren und den Vergleichsoperatoren wird i.d.R. eine Eigenschaft des Objekts angegeben, für die der Vergleich durchgeführt werden soll.

Dies geschieht unter Verwendung von Xpath, dabei beschränkt man sich auf die Kurzschreibweise. Dies bedeutet:

- Ein Attribut „att“ der Query-Objektart wird wie folgt referenziert:

```
<ogc:PropertyName>att</ogc:PropertyName>
```

Oder mit einem konkreten Beispiel aus dem AAA-Anwendungsschema:

```
<ogc:PropertyName>flurstueckskennzeichen</ogc:PropertyName>
```

- Sofern „att“ ein Attribut der Query-Objektart ist und der Wert des Attributs vom Datentyp „AX_DT“ ist und darin das Attribut „att2“ referenziert werden soll, dann geschieht dies wie folgt:

```
<ogc:PropertyName>att/AX_DT/att2</ogc:PropertyName>
```

Oder mit einem konkreten Beispiel aus dem AAA-Anwendungsschema:

```
<ogc:PropertyName>lebenszeitintervall/AA_Lebenszeitintervall/endet</ogc:PropertyName>
```

- Eine Relation (genauer gesagt die Rolle in der Definitionsrichtung einer Relation) „rel“ der Query-Objektart wird wie folgt referenziert

```
<ogc:PropertyName>rel</ogc:PropertyName>
```

Handelt es sich dabei um die Objektart „AX_OA“ als Relationspartner und besitzt dieses ein Attribut „att3“, dann wird dieses wie folgt referenziert:

```
<ogc:PropertyName>rel/AX_OA/att3</ogc:PropertyName>
```

Oder mit einem konkreten Beispiel aus dem AAA-Anwendungsschema (über zwei Relationen):

```
<ogc:PropertyName>
  istGebucht/AX_Buchungsstelle/zu/AX_Buchungsstelle/laufendeNummer
</ogc:PropertyName>
```

- Für den Fall, dass ein XML-Attribut konkret referenziert und ausgewertet werden muss (z.B. „xlink:href“, „uom“ oder „srsName“), so geschieht dies wie folgt:

```
<ogc:PropertyName>att/@xmlatt</ogc:PropertyName>
```

Oder mit zwei konkreten Beispielen aus dem AAA-Anwendungsschema:

```
<ogc:PropertyName>amtlicheFlaeche/@uom</ogc:PropertyName>
<ogc:PropertyName>istGebucht/@xlink:href</ogc:PropertyName>
```

Hierbei wird davon ausgegangen, dass der Default-Namespace des XML-Dokuments "<http://www.adv-online.de/namespaces/adv>" ist. Ansonsten sind alle Bezeichner durch das Namespacekürzel zu qualifizieren (wie dies im Beispiel von xlink:href oben bereits erfolgt ist), also in der Regel

```
<ogc:PropertyName>adv:att</ogc:PropertyName>
```

statt

```
<ogc:PropertyName>att</ogc:PropertyName>
```

Im Fall von einfachen Attributen wird i.d.R. der Vergleichsoperator den Attributwert mit einem festen Wert vergleichen (Element `<ogc:Literal>`), z.B.

```
<ogc:PropertyIsEqualTo>
  <ogc:PropertyName>stellenart</ogc:PropertyName>
  <ogc:Literal>1100</ogc:Literal>
</ogc:PropertyIsEqualTo>
```

was für alle Objekte in der Datenbasis erfüllt ist, bei denen das Stellenart-Attribut einen entsprechenden Wert (Wertart 1100) aufweist, oder

```
<ogc:PropertyIsGreaterThanOrEqualTo>
  <ogc:PropertyName>
    lebenszeitintervall/AA_Lebenszeitintervall/beginnt
  </ogc:PropertyName>
  <ogc:Literal>2003-05-20T00:00:00</ogc:Literal>
</ogc:PropertyIsGreaterThanOrEqualTo>
```

oder

```
<ogc:PropertyIsLessThan>
  <ogc:PropertyName>
    lebenszeitintervall/AA_Lebenszeitintervall/endet
  </ogc:PropertyName>
  <ogc:Literal>2003-05-20T00:00:00</ogc:Literal>
</ogc:PropertyIsGreaterThanOrEqualTo>
```

oder im Fall eines Vergleichs auf den NULL-Wert

```
<ogc:PropertyIsNull>
  <ogc:PropertyName>
    lebenszeitintervall/AA_Lebenszeitintervall/endet
  </ogc:PropertyName>
</ogc:PropertyIsNull>
```

Für die Prüfung auf Werte in einem Bereich, z.B. für die Prüfung ob die Stellenart ein Wert im 1xxx-Bereich ist, würde folgender Vergleich verwendet:

```
<ogc:PropertyIsBetween>
  <ogc:PropertyName>stellenart</ogc:PropertyName>
  <ogc:LowerBoundary>
    <ogc:Literal>1000</ogc:Literal>
  </ogc:LowerBoundary>
  <ogc:UpperBoundary>
    <ogc:Literal>1999</ogc:Literal>
</ogc:PropertyIsBetween>
```

```

    </ogc:UpperBoundary>
  </ogc:PropertyIsBetween>

```

Analog ein Prädikat für Flurstücke mit einer amtlichen Fläche von mindestens 1000qm aber maximal 2000qm:

```

  <ogc:PropertyIsBetween>
    <ogc:PropertyName>amtlicheFlaeche</ogc:PropertyName>
    <ogc:LowerBoundary>
      <ogc:Literal>1000</ogc:Literal>
    </ogc:LowerBoundary>
    <ogc:UpperBoundary>
      <ogc:Literal>2000</ogc:Literal>
    </ogc:UpperBoundary>
  </ogc:PropertyIsBetween>

```

Der LIKE-Vergleich ist für flexible Textvergleiche hilfreich. So filtert das folgende Prädikat alle Anschriften heraus, deren Telefonnummer mit 0228 beginnt

```

  <ogc:PropertyIsLike wildCard="*" singleChar="?" escape="\">
    <ogc:PropertyName>telefon</ogc:PropertyName>
    <ogc:Literal>0228*</ogc:Literal>
  </ogc:PropertyIsLike>

```

während das folgende Prädikat die Personen filtert, bei denen der Geburtsname gesetzt ist, mit einem „M“ beginnt und als dritten und vierten Buchstaben ein „t“ hat:

```

  <ogc:PropertyIsLike wildCard="*" singleChar="?" escape="\">
    <ogc:PropertyName>geburtsname</ogc:PropertyName>
    <ogc:Literal>M?tt*</ogc:Literal>
  </ogc:PropertyIsLike>

```

Bei räumlichen Operatoren erfolgt ein Vergleich einer Eigenschaft (der Name der geometrischen Attributart) mit einer festen Geometrie analog zu den Vergleichen einer textlichen oder numerischen Eigenschaft mit einem festen Wert. Bei den räumlichen Operatoren wird der feste Wert statt durch ein `<ogc:Literal>`-Element durch das jeweilige GML-Geometrieelement ausgedrückt, zum Beispiel

```

  <ogc:Intersects>
    <ogc:PropertyName>position</ogc:PropertyName>
    <gml:Polygon>
      <gml:exterior>
        <gml:Ring>
          <!-- hier steht der Umring der Suchfläche -->
        </gml:Ring>
      </gml:exterior>
    </gml:Polygon>
  </ogc:Intersects>

```

Sofern der Gesamtschlüssel eines Katalogeintrags bekannt ist, kann der entsprechende Katalogeintrag z.B. mit einer Query der folgenden Art erfragt werden (hier die Gemarkung mit der Kennung „071234“)¹:

```
<wfs:Query typeName="AX_Gemarkung">
  <ogc:Filter>
    <ogc:PropertyIsEqualTo>
      <ogc:PropertyName>schluesselGesamt</ogc:PropertyName>
      <ogc:Literal>071234</ogc:Literal>
    </ogc:PropertyIsEqualTo>
  </ogc:Filter>
</wfs:Query>
```

Sollen alle Katalogeinträge mit einem bestimmten Teilschlüssel erfragt werden, dann kann entweder mit `<ogc:PropertyIsLike>` oder mit Vergleichsoperatoren für die einzelnen Attribute des Schlüssel-Datentyps gesucht werden. Alle Gemarkungen im Land findet man z.B. mit:

```
<wfs:Query typeName="AX_Gemarkung">
  <ogc:Filter>
    <ogc:PropertyIsLike wildCard="*" singleChar="?" escape="\ ">
      <ogc:PropertyName>schluesselGesamt</ogc:PropertyName>
      <ogc:Literal>07*</ogc:Literal>
    </ogc:PropertyIsLike>
  </ogc:Filter>
</wfs:Query>
```

oder

```
<wfs:Query typeName="AX_Gemarkung">
  <ogc:Filter>
    <ogc:PropertyIsEqualTo>
      <ogc:PropertyName>
        schluessel/AX_Gemarkung_Schluessel/land
      </ogc:PropertyName>
      <ogc:Literal>07</ogc:Literal>
    </ogc:PropertyIsEqualTo>
  </ogc:Filter>
</wfs:Query>
```

Neben der Filter-Bedingung können in das `<wfs:Query>`-Element noch weitere Elemente eingebettet sein. Die beiden Elemente `<wfs:XlinkPropertyName>` und `<wfs:XlinkPropertyPath>` können dazu genutzt werden, mit einer Query auf einen Schlag auch noch weitere Objekte in die Ergebnismenge aufzunehmen. Auf diese Weise kann die Anzahl der Queries – und damit auch der Benutzungsaufträge – häufig deutlich reduziert werden.

Das Element

Statt

```
$1 = concat( ausBuchungsblattbezirk/land, ausBuchungsblattbezirk/bezirk )
```

müsste es zum Beispiel

```
$1 = concat( ausBuchungsblattbezirk/AX_Buchungsblattbezirk/land,
  ausBuchungsblattbezirk/ AX_Buchungsblattbezirk/bezirk )
```

lauten.

```
<wfs:XlinkPropertyPath>
  istGebucht/AX_Buchungsstelle/istBestandteilVon/AX_Buchungsblatt
</wfs:XlinkPropertyPath>
```

führt dazu, dass genau die Objekte entlang des Pfades in die Ergebnismenge aufgenommen werden. Hierbei kann mit dem Attribut „leafOnly“ geregelt werden, ob alle Objekte längs des Pfades oder nur das Ziel des Pfades selektiert werden. Default ist das bis zur GeoInfoDok Version 4.0 benutzte Standardverhalten (alle Objekte entlang eines Pfades).

Beispiel:

```
<wfs:XlinkPropertyPath leafOnly="true">
  istGebucht/AX_Buchungsstelle/istBestandteilVon/AX_Buchungsblatt
</wfs:XlinkPropertyPath>
```

Sofern nur einzelne, ganz bestimmte nachgeordnete Objekte benötigt werden (in dem Beispiel nur für wenige Flurstücke die Buchungsblätter), dann bietet es sich i.d.R. an, die Selektion in zwei Abfragen aufzuteilen. Die erste Abfrage zur Selektion der Flurstücke und anschließend die Selektion der Buchungsblätter.

In der NAS werden alle Relationen nur in einer, der im UML-Modell als navigierbar ausgezeichneten Richtung repräsentiert. Die folgende Query erfragt alle Flurstücke und die Buchungsstellen unter denen diese gebucht sind:

```
<wfs:Query typeName="AX_Flurstueck">
  <XlinkPropertyPath>istGebucht/AX_Buchungsstelle</XlinkPropertyPath>
</wfs:Query>
```

Oder im Fall, dass ein Flurstück bekannt ist, dann kann aus dem <istGebucht>-Element der Identifikator der Buchungsstelle extrahiert werden (der String nach dem „urn:adv:oid:“-Präfix, in diesem Beispiel „DEBY123412345678“) und die Buchungsstelle wie folgt erfragt werden:

```
<wfs:Query typeName="AX_Buchungsstelle">
  <ogc:Filter>
    <ogc:FeatureId fid="DEBY123412345678"/>
  </ogc:Filter>
</wfs:Query>
```

In umgekehrter Richtung, d.h. von der Buchungsstelle zum Flurstück ist die Relation zwar auch benannt („grundstueckBestehtAus“), aber nicht in der NAS repräsentiert. Sollen nun die Flurstücke ermittelt werden, die über „istGebucht“ einer bestimmten Buchungsstelle (im Beispiel wird wieder die ID „DEBY123412345678“ verwendet) zugeordnet sind, so kann dies über die Prüfung der Relation erfolgen:

```
<wfs:Query typeName="AX_Flurstueck">
  <ogc:Filter>
    <ogc:PropertyIsEqualTo>
      <ogc:PropertyName>istGebucht/@xlink:href</ogc:PropertyName>
      <ogc:Literal>urn:adv:oid:DEBY123412345678</ogc:Literal>
    </ogc:PropertyIsEqualTo>
  </ogc:Filter>
```

```
</wfs:Query>
```

Eine äquivalente Abfrage (sofern sich Flurstück und Buchungsstelle in derselben lokalen Datenbasis befinden) ist

```
<wfs:Query typeName="AX_Flurstueck">
  <ogc:Filter>
    <ogc:PropertyIsEqualTo>
      <ogc:PropertyName>
        istGebucht/AX_Buchungsstelle/@gml:id
      </ogc:PropertyName>
      <ogc:Literal>DEBY123412345678</ogc:Literal>
    </ogc:PropertyIsEqualTo>
  </ogc:Filter>
</wfs:Query>
```

Eine Möglichkeit so etwas wie „XlinkPropertyPath“ auch in inverser Richtung, also das gleichzeitige Selektieren bestimmter Buchungsstellen und aller Flurstücke, die auf diese gebucht sind, besteht durch die Möglichkeit der Verwendung inverser Relationen in Filterausdrücken. Hier kann die Selektion stattdessen natürlich auch in zwei Schritten, d.h. über zwei Queries, erfolgen.

In aller Regel werden mehrere Queries erforderlich sein, um sich die Objekte aus dem Datenbestand zu besorgen, die für komplexere Abfragen benötigt werden. Hierbei wird aus den Ergebnissen der vorigen Query die neue Query formuliert. Sehr häufig werden hierbei Zugriffe auf die Katalogeinträge zum Entschlüsseln von Schlüsselwerten erforderlich sein.

Verweise:

- OGC Implementation Specification Filter Encoding 1.0.0
(<http://www.opengis.org/techno/specs/02-059.pdf>)
- OGC Implementation Specification Web Feature Service 1.0.0
(<http://www.opengis.org/techno/specs/02-058.pdf>)
- OGC Implementation Specification Simple Features for SQL 1.1
(<http://www.opengis.org/techno/specs/99-049.pdf>)
- GeoInfoDok 5.1 – Abschnitt 7.2 Prozesse in ALKIS – Teil B ([http://www.adv-online.de/veroeffentlichungen/afis-alkis-atkis/Dokumente/Prozesse in ALKIS.pdf](http://www.adv-online.de/veroeffentlichungen/afis-alkis-atkis/Dokumente/Prozesse%20in%20ALKIS.pdf))
- GeoInfoDok 5.1 – Abschnitt 10
- Xpath (<http://www.w3.org/TR/xpath>).

4 Kataloge des Basisschemas und des Versionierungsschemas

Der Katalog des Basisschemas und des Versionierungsschemas enthält die Dokumentation zu den in UML beschriebenen Klassen der Schemata als lesbares Textdokument.

Der Katalog des Basisschemas ist als HTML-Dokument und als PDF-Datei verfügbar.

Der Katalog des Versionierungsschemas ist ebenfalls als HTML-Dokument und als PDF-Datei verfügbar.

5 Fachspezifische Anwendung des Basisschemas

5.1 Das gemeinsame AFIS-ALKIS-ATKIS-Fachschemas

Die fachspezifischen Anforderungen aus AFIS, ALKIS und ATKIS führten zur Entwicklung des gemeinsamen AFIS-ALKIS-ATKIS-Fachschemas. Dieses gemeinsame Schema ist zusammen mit dem Basisschema sowie den verwendeten ISO-Normen als Rational-Rose-Datei verfügbar. Die notwendigen Dateien sind dem entsprechenden ZIP-Archiv zu entnehmen.

5.2 Erläuterungen zu AFIS

In diesem Kapitel werden die fachspezifischen Anwendungen des AAA-Basisschemas aus der Sicht von AFIS erläutert. Das Dokument ist als PDF-Datei verfügbar.

5.3 Erläuterungen zu ALKIS

Diese Erläuterungen orientieren sich am Aufbau des ALKIS-Objektartenkatalogs und legen die Modellierungsgrundsätze für sämtliche Objektarten dar. Das Zusammenwirken der Bestandsdaten mit den Prozessen in ALKIS werden ebenso erläutert, wie ein Beispiel zur NAS. Ferner werden umfangreiche Beispiele zur Modellierung aufgeführt. Das Dokument ist als PDF-Datei verfügbar.

5.4 Erläuterungen zu ATKIS-Digitale Landschaftsmodelle

Die Erläuterungen beschreiben die Grundsätze der Modellierung der ATKIS-DLM und zeigen Besonderheiten bei einzelnen Objektarten mit Beispielen auf. Die Erläuterungen befinden sich in einem eigenen Dokument.

5.5 Erläuterungen zu ATKIS-Digitale Geländemodelle

Die Erläuterungen beschreiben die Grundsätze der Modellierung der ATKIS-Geländemodelle und zeigen Besonderheiten bei einzelnen Objektarten mit Beispielen auf. Die Erläuterungen befinden sich in einem eigenen Dokument.

6 AFIS-Katalogwerke

6.1 AFIS-Objektartenkatalog

Der aktuelle Stand des AFIS-Objektartenkataloges ist als PDF-Datei und als HTML-Dokument verfügbar.

6.2 Prozesse in AFIS

Der AFIS-Ausgabekatalog besteht aus folgenden Teilen:

- Dem aus dem Rational-Rose-Modell abgeleiteten Ausgabekatalog als PDF-Datei oder als HTML-Dokument.
- Den Filterausdrücken für die AFIS-Standardausgaben als Word-Dateien.

Die Filterausdrücke für die AFIS-Standardausgaben als XSLT-Skripte werden zu einem späteren Zeitpunkt veröffentlicht.

6.3 AFIS-Signaturenkatalog

Der AFIS-Signaturenkatalog wird analog zum ALKIS-Signaturenkatalog aufgestellt wird als PDF-Datei veröffentlicht.

7 ALKIS-Katalogwerke

7.1 ALKIS-Objektartenkatalog

Der ALKIS-Objektartenkatalog besteht aus dem Titelblatt, dem Inhaltsverzeichnis und den folgenden Teilen:

- Den Vorbemerkungen (allgemeine Angaben) – Teil A
- Dem aus dem Rational-Rose-Modell abgeleiteten Objektartenkatalog (OK) – Teil B als PDF-Datei oder als HTML-Dokument. Die HTML-Version enthält keine Vorbemerkungen.

7.2 Prozesse in ALKIS

Der ALKIS-Ausgabekatalog besteht aus dem Titelblatt, dem Inhaltsverzeichnis und den folgenden Teilen:

- Ableitung der Ausgaben mit Filterencoding – Teil A.
- Dem aus dem Rational-Rose-Modell abgeleiteten Ausgabekatalog – Teil B – als PDF-Datei oder als HTML-Dokument.
- Den Filterausdrücken für die ALKIS-Standardausgaben als XSLT-Skripte sowie für die sonstigen Ausgaben als Word-Dateien.

Das gesamte Dokument ist als ZIP-Datei gepackt.

7.3 ALKIS-Signaturenkatalog

Der ALKIS-Signaturenkatalog besteht aus:

- dem Titelblatt, dem Inhaltsverzeichnis und den Vorbemerkungen – Teil A
- dem tabellarischen Signaturenkatalog – Teil B
- Beispielsammlung
- Positionierungsregeln.

8 ATKIS-Katalogwerke

8.1 ATKIS-Objektartenkataloge

8.1.1 ATKIS-Objektartenkatalog für das Digitale Basis Landschaftsmodell (ATKIS-OK Basis-DLM)

Der ATKIS-Objektartenkatalog besteht aus dem Titelblatt, dem Inhaltsverzeichnis und den folgenden Teilen:

- Den Vorbemerkungen (allgemeine Angaben) – Teil A.
- Dem aus dem Rational-Rose-Modell abgeleiteten Objektartenkatalog (OK) – Teil B als PDF-Datei oder als HTML-Dokument.

8.1.2 ATKIS-Objektartenkatalog für das Digitale Landschaftsmodell 50 (ATKIS-OK50)

Der ATKIS-OK50 ist als PDF bzw. HTML-Datei verfügbar.

8.1.3 ATKIS-Objektartenkatalog für das Digitale Landschaftsmodell 250 (ATKIS-OK250)

Der ATKIS-OK250 ist als PDF bzw. HTML-Datei verfügbar.

8.1.4 ATKIS-Objektartenkatalog für das Digitale Landschaftsmodell 1000 (ATKIS-OK1000)

Der ATKIS-OK1000 ist als PDF bzw. HTML-Datei verfügbar.

8.2 ATKIS-Objektartenkatalog für das Digitale Geländemodell

8.2.1 ATKIS-Objektartenkatalog für das Digitale Geländemodell 2 (ATKIS-OK DGM2)

Der ATKIS-OK DGM2 ist als PDF bzw. HTML-Datei verfügbar.

8.2.2 ATKIS-Objektartenkatalog für das Digitale Geländemodell 5 (ATKIS-OK DGM5)

Der ATKIS-OK DGM5 ist als PDF bzw. HTML-Datei verfügbar.

8.2.3 ATKIS-Objektartenkatalog für das Digitale Geländemodell 25 (ATKIS-OK DGM25)

Der ATKIS-OK DGM25 ist als PDF bzw. HTML-Datei verfügbar.

8.2.4 ATKIS-Objektartenkatalog für das Digitale Geländemodell 50 (ATKIS-OK DGM50)

Der ATKIS-OK DGM50 ist als PDF bzw. HTML-Datei verfügbar.

8.3 ATKIS-Signaturenkataloge

Derzeit liegen noch keine Signaturenkataloge für das ATKIS-Fachschemata vor. Geplant sind folgende Kataloge:

1. ATKIS-Signaturenkatalog für die Digitale Topographische Karte 1 : 10 000
(ATKIS-SK10)
2. ATKIS-Signaturenkatalog für die Digitale Topographische Karte 1 : 25 000
(ATKIS-SK25)
3. ATKIS-Signaturenkatalog für die Digitale Topographische Karte 1 : 50 000
(ATKIS-SK50)
4. ATKIS-Signaturenkatalog für die Digitale Topographische Karte 1 : 100 000
(ATKIS-SK100)
5. ATKIS-Signaturenkatalog für die Digitale Topographische Karte 1 : 250 000
(ATKIS-SK250)
6. ATKIS-Signaturenkatalog für die Digitale Topographische Karte 1 : 1 000 000
(ATKIS-SK1000)

9 Metadatenkatalog

Der AFIS-ALKIS-ATKIS-Metadatenkatalog besteht aus dem Titelblatt, dem Inhaltsverzeichnis und den folgenden Teilen:

- Eine Tabelle mit den Metadatenelementen des AAA-Profiles in Anlehnung an die ISO 19115
- Der Beispielsammlung.

10 Das externe Modell, Datenaustausch

In Kapitel 2 sind die Grundlagen und Zusammenhänge für die mit dieser Dokumentation zu beschreibenden Geoinformationen erläutert. Das dort festgelegte Referenzmodell stellt unter anderem auch den Bedarf für den Datenaustausch dar. Soweit es erforderlich ist, den Datenaustausch als AdV-Standard einheitlich zu definieren, enthält dieses Kapitel die Festlegungen zu den zu verwendenden Austauschchnittstellen. Die in der Folge aufgeführten XML-Schemadateien sind dem [ZIP-Archiv](#) zu entnehmen.

10.1 Normbasierte Austauschchnittstelle (NAS)

Die Normbasierte Austauschchnittstelle (NAS) wird verwendet, wenn Geoinformationen ausgetauscht werden sollen, die im gemeinsamen AFIS-ALKIS-ATKIS-Anwendungsschemas modelliert wurden. Dabei kann es sich sowohl um Informationen handeln, die in ihrer Struktur den gespeicherten Datenbeständen, einschließlich der Zusatzdaten (Präsentationsobjekte, Kartengeometrieobjekte, vgl. Kapitel 2) entsprechen, oder um Informationen aus daraus abgeleiteten Sichten auf diese Datenbestände (z.B. Ausgabeobjektarten, s. Abschnitt 7.2), nicht jedoch um Datenbestände, bei denen der Objektbezug völlig verloren geht (z.B. rein graphisch strukturierte Daten), oder Daten, die nach einem anderen Basisschema zu definieren sind (z.B. DXF - Daten).

Entsprechend wird sie dort eingesetzt, wo der Anwendungsschwerpunkt nach Anforderung des Nutzers auf

- der Originalität der Daten,
- der vollen Auswertbarkeit und
- der differenzierten Fortführbarkeit

liegt.

10.1.1 Normen und Standards

Die Standards AFIS, ALKIS und ATKIS der AdV sind in dieser Dokumentation in konzeptueller Form auf der Grundlage der Norm ISO 19109 *Rules for Application Schema* beschrieben. Dies bedeutet insbesondere:

- Modellierung in UML mit dem Softwarewerkzeug *Rational Rose*
- Einhaltung der Regelungen von ISO/TS 19103 für die Verwendung von UML
- Verwendung von ISO 19107 (und damit implizit auch ISO 19111), ISO 19115, ISO 19123

- Automatisierte Ableitung und Darstellung der Objektartenkataloge gemäß ISO 19110

Die automatisierte Ableitung der Schnittstelle für den Austausch von AFIS-, ALKIS- und ATKIS-Objekten, die NAS, vervollständigt dieses Bild.

ISO 19118 *Encoding* definiert zu diesem Zweck u.a. ein Rahmenwerk für die Erstellung von sogenannten *Encoding Rules* zur Ableitung von Schnittstellen-Definitionen für den Datenaustausch aus einem UML-Anwendungsschema. Darüber hinaus beschreibt die Norm in einem informativen Anhang spezielle *Encoding Rules* für die Erzeugung von XML-Schemadefinitionen. Die zugelassene Variabilität bei der Abbildung von UML nach XML-Schema führt allerdings dazu, dass die ISO-19100-Basisnormen auf unterschiedliche Weise umgesetzt werden können. Eine Festlegung durch die AdV würde zu AdV-spezifischen Schnittstellen führen. Damit wäre der Zweck der Normen, nämlich Interoperabilität zu erreichen, nicht erfüllt. Genormte XML-Schemata für die Basisnormen existieren zurzeit nicht.

Neben der offiziellen Norm zur Definition von Schnittstellendefinitionen gibt es mit der *Geography Markup Language (GML)* des *Open Geospatial Consortiums (OGC)* einen de-facto Standard für die Verschlüsselung von Geoinformationen, der darüber hinaus auch zur Normung in der ISO 19100er Serie als ISO 19136 vorgesehen ist. GML ist - vereinfacht gesagt - ein Regelwerk zur Modellierung von anwendungsspezifischen Objektarten mit ihren Eigenschaften in XML Schema. Für die Beschreibung der Objektarten und deren numerischer, textlicher, geometrischer, zeitlicher und anderer Eigenschaften kann hierbei auf eine Vielzahl häufig verwendeter, standardisierter Komponenten, z.B. eben Geometrietypen, zurückgegriffen werden. Auf diese Weise wird in einem GML-Anwendungsschema eine anwendungsspezifische XML-Sprache definiert. Eine Marktakzeptanz von GML zeichnet sich ab.

Da die NAS neben der Codierung von Fachobjekten auch die im Anwendungsschema modellierten Operationen auf einem System zur Haltung von Bestandsdaten umfasst (Fortführen, Einrichten, Sperren/Entsperren von Objekten, Reservieren von Fachkennzeichen, Erfragen von Ausgabeprodukten einschließlich der Nutzerbezogenen Bestandsdatenaktualisierung), werden die GML-Objektarten unter Verwendung von Elementen der zu GML komplementären OGC-Spezifikationen Web Feature Service (WFS) und Filter Encoding (FES) in entsprechende, grundsätzlich Web-Service-fähige, Operationen eingebettet. In diesem Sinne ist eine AFIS-ALKIS-ATKIS-Datenhaltung mit einem gekapselten Web Feature Server zu vergleichen, der zusätzlich AFIS-ALKIS-ATKIS-spezifische Anforderungen berücksichtigt.

Gemäß der in Kapitel 3.1 genannten Grundlagen verfolgt die AdV mit der Neuentwicklung von AFIS, ALKIS und ATKIS das Ziel, Grundlagen für die gemeinsame, ganzheitliche und fachübergreifende Nutzung von Geodaten zu schaffen. In diesem Sinne soll soweit wie möglich auf bestehende oder absehbare Standardfunktionalitäten von Anwendungssoftware zurückgegriffen werden. Ein Beispiel ist die in diesem Kapitel beschriebene NAS. Auf die Spezifikation von AdV-spezifischen Lösungen wird soweit wie möglich verzichtet. Dies ist allerdings aufgrund des aktuellen Status in der internationalen Standardisierung im Bereich der Metadaten und der Operationen zur Fortführung und Abfrage von GML-Datenbeständen zurzeit nur mit Einschränkungen und AFIS-ALKIS-ATKIS-spezifischen Erweiterungen möglich. Aus diesen Gründen hat sich die AdV für folgendes Vorgehen entschieden:

- Das in ISO 19118 Kapitel 8 definierte Rahmenwerk für *Encoding Rules* wird für die NAS angewendet (Level-1-Konformität mit ISO 19118).
- Es werden "NAS Encoding Rules" konform zu ISO 19118 Kapitel 8 definiert und dokumentiert. Diese "NAS Encoding Rules" bilden das konzeptuelle AFIS-ALKIS-ATKIS-Fach- und Basisschema auf ein GML-3.0-Anwendungsschema ab, das in der NAS durch die in der Schemadatei AAA-Fachschemaxsd (einschließlich der zugehörigen weiteren Schemadateien, siehe 10.1.3.4) enthaltenen Schemakomponenten spezifiziert ist.

Das konzeptuelle Schema der NAS-Operationen wird unter Verwendung der „NAS Encoding Rules“ und der WFS/FES-XML-Schemata in XML-Schemakomponenten abgebildet und in der Schemadatei NAS-Operationen.xsd spezifiziert.

- Die "NAS Encoding Rules" werden konform zu GML 3.0 und darüber hinaus so einfach und so konform zu den ISO 19118 "*XML Encoding Rules*" wie möglich festgelegt. Das bedeutet, die NAS verwendet GML in Übereinstimmung mit den Vorgaben der GML-Spezifikation.
- Eine automatische Ableitung der NAS wird unterstützt, so dass anhand der "NAS Encoding Rules", dem UML-Anwendungsschema und von in Form von Steuerparametern formal beschriebenen Zusatzregeln die XML-Schema-Definitionen der NAS abgeleitet werden können.

Es ist wichtig festzuhalten, dass das stabile konzeptuelle Modell im UML-Anwendungsschema vollständig beschrieben ist. Bei der Abbildung auf spezifische Implementierungsmodelle (wie z.B. XML-Repräsentierungen) werden auch zukünftig Anpassungen an den IT/GI-Mainstream erforderlich werden.

Über die genannten Normen der Normfamilie ISO 19100 hinaus werden zur Definition der NAS folgende Dokumente herangezogen:

- UML 1.3:1999, *Unified Modeling Language (UML)*, Object Management Group (OMG), <http://www.omg.org/>
- XML 1.0:1998, *Extensible Markup Language (XML)*, W3C Recommendation, 6 Oktober 2000, <http://www.w3.org/TR/2000/REC-xml-20001006>
- XML Schema Part 1: *Structure* - W3C Recommendation, 2. Mai 2001, <http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/>
- XML Schema Part 2: *Datatypes* - W3C Recommendation, 2. Mai 2001, <http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/>
- XLink *XML Linking Language (XLink)* Version 1.0, W3C Recommendation 27 June 2001, <http://www.w3.org/TR/2000/REC-xlink-20010627/>
- Geography Markup Language (GML) 3.0
Open Geospatial Consortium, 2003
[Online](#)
- Web Feature Service (WFS) 1.0
Open Geospatial Consortium, 2002
[Online](#)
- Filter Encoding (FES) 1.0
Open Geospatial Consortium, 2002
[Online](#)
- OWS Common Implementation Specification 1.0
Open Geospatial Consortium, 2005
[Online](#)

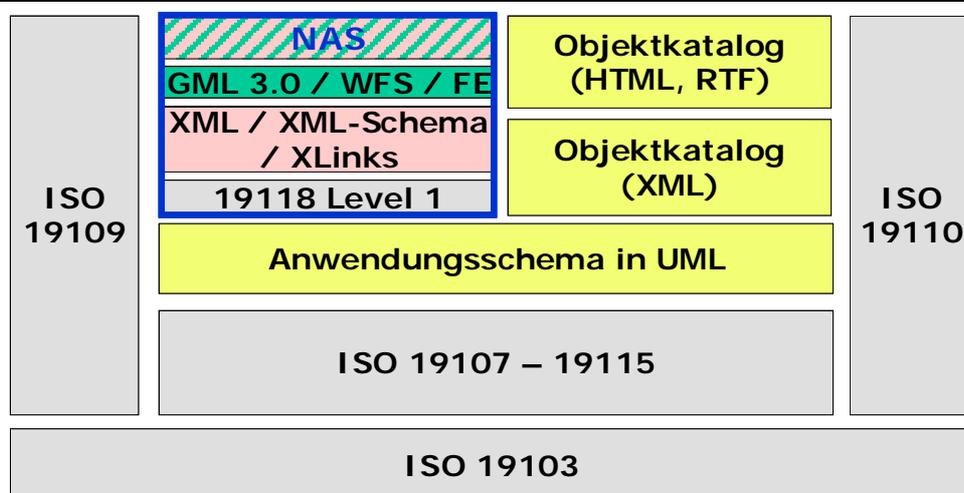


Abbildung 10-1: Einbettung der NAS in Normen und Standards

Dadurch, dass GML-Anwendungsschemata auf standardisierte XML-Komponenten, z.B. für Geometrietypen, zurückgreifen und es in GML Regeln gibt, wie XML Schema bei der Definition eines Anwendungsschemas zu verwenden ist, kann auch generische GML-Software - sofern sie die verwendeten XML-Komponenten implementiert hat – durch Analyse des GML-Anwendungsschemas der NAS AFIS-ALKIS-ATKIS-Objekte verarbeiten und syntaktisch interpretieren. Dies gilt auch dann, wenn die Software zuvor kein Wissen über die NAS und AFIS-ALKIS-ATKIS besessen hat.

Mit dem für die NAS definierten GML-Profil (siehe 10.1.6) werden aus diesem Grunde auch Anforderungen an die Fähigkeiten von GML-Software spezifiziert und dokumentiert. Bei der Festlegung des Profils wurde auch die Zielsetzung berücksichtigt, dass dieses Profil auch über AFIS, ALKIS und ATKIS hinaus Anwendungsanforderungen abdecken soll und sich von einer AdV-internen Festlegung zu einer breiter akzeptierten Festlegung entwickelt.

Durch die Spezifikation der NAS als Operationen und nicht als reines „Datenformat“ sind die GML-Objekte in der NAS i.d.R. in die XML-Elemente der Operationsaufrufe und -ergebnisse eingebettet. Im Fall des Bestandsdatenauszugs zum Beispiel ist die Menge der GML-Objekte, d.h. das GML-Dokument, in das NAS-Ergebnisdokument eingebettet und kann auf einfache Weise erkannt und extrahiert werden.

10.1.2 Kodierungsprozess

Die Norm ISO 19118 beschreibt den durchzuführenden Kodierungs- und Dekodierungsprozess in allgemeiner Form folgendermaßen:

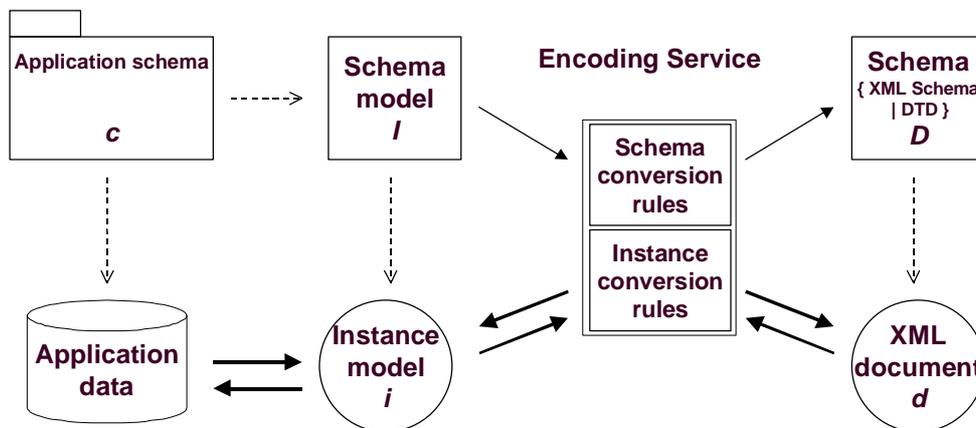


Abbildung 10-2: XML - basierende Kodierungsregeln gemäß ISO 19118

Der Prozess geht dabei von folgenden Rahmenbedingungen aus:

- Es existiert ein formal (z.B. in UML) beschriebenes Anwendungsschema.
- Auf der Basis von Umwandlungsregeln (*Schema Conversion Rules*) und ggf. Steueranweisungen werden die Informationen des UML-Anwendungsschemas in eine XML-Schemadatei überführt.
- In gleicher Weise werden die auf dem Anwendungsschema beruhenden Anwendungsdaten (Objekt-Instanzen) mit Hilfe von Umwandlungsregeln (*Instance Conversion Rules*) in eine XML-Datei überführt, die in ihrem Aufbau den Definitionen der XML-Schema-Datei entspricht.

Im Kontext der NAS wird die Umwandlung des UML-Schemas in das XML-Schema von einem Software-Script innerhalb des Modellierungsprogramms *Rational Rose* übernommen. Eine herstellerunabhängige Lösung (Überführung des UML-Schemas in XMI o.Ä.) erscheint zurzeit nicht zufriedenstellend realisierbar. Die Definition der Steuerparameter wird ebenfalls in *Rational Rose* vorgenommen (s. 10.1.3.4). Daraus ergibt sich folgender Ablauf bei der automatischen Ableitung der NAS-Schemata:

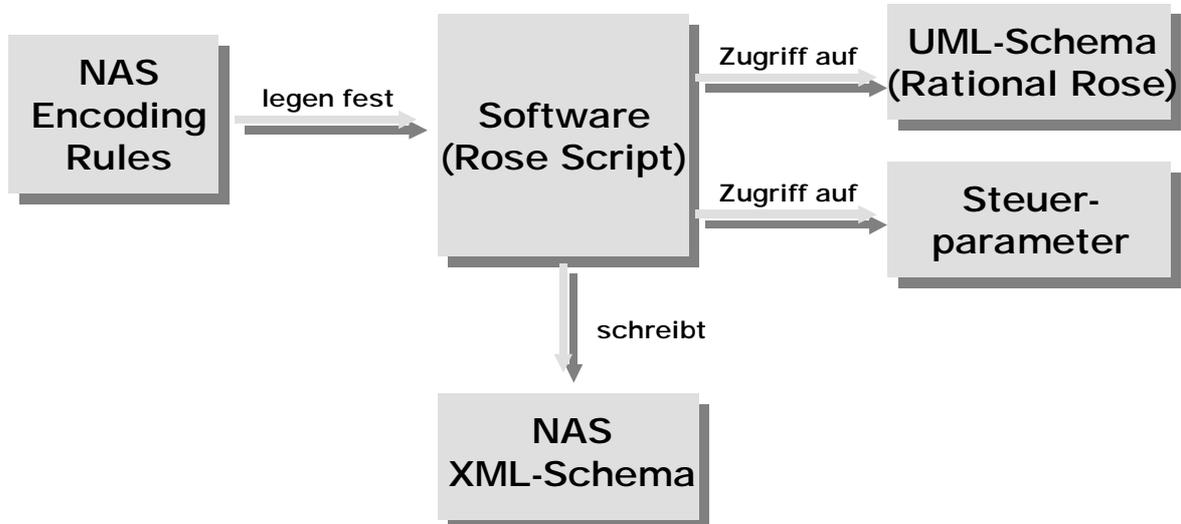


Abbildung 10-3: Automatische Ableitung der NAS-Schemata

Die Basis für die Struktur der NAS ist demgemäß das in Kapitel 3.3 beschriebene AAA-Basisschema mit dem anwendungsspezifischen Subschema für AFIS, ALKIS und ATKIS (s. Kap. 5) sowie entsprechende Steuerparameter.

10.1.3 NAS Encoding Rules

Im folgenden werden die "NAS Encoding Rules" beschrieben. Die Struktur erfüllt die Anforderungen aus ISO 19118 Kapitel 8 und richtet sich zur einfachen Vergleichbarkeit an ISO 19118 Annex A aus.

ISO 19118 legt in Kapitel 8 Anforderung an Encoding Rules fest. Eine Encoding Rule beschreibt Abbildungsregeln mit denen Daten aus einer Eingangsdatenstruktur (Instanzen gemäß dem AAA-Anwendungsschema in Rational Rose) in eine Ausgabedatenstruktur (XML-Datei gemäß NAS) überführt werden können. Eine Encoding Rule deckt folgende Themen ab:

- Voraussetzungen
 1. Anwendungsschema
 2. Zeichensatz und unterstützte Sprachen
 3. Austausch-Metadaten (*exchange metadata*)
 4. Identifikatoren
 5. Updatemechanismen
- Eingangsdatenstruktur
 1. Instanzenschema
 2. Beziehung des Instanzenschemas zum Anwendungsschema

- Ausgabedatenstruktur
- Abbildungsregeln
- Beispiele

10.1.3.1 Voraussetzungen

Anwendungsschema

Das AAA-Basisschema und das AAA-Fachschemata wurden auf der Basis der Regeln für Anwendungsschemata aus ISO 19103 und ISO 19109 entwickelt. Die entsprechenden Regeln aus ISO 19118 A.2.1 gelten auch für die "NAS Encoding Rules".

Beide Schemata, sowohl das AAA-Basisschema als auch das darauf aufbauende AAA-Fachschemata, werden mit den hier beschriebenen NAS Encoding Rules in getrennte XML-Schema-Definitionen überführt, wobei auch hier das Fachschema in XML-Schema auf das Basisschema in XML-Schema aufbaut. Das Basisschema muss in sich schlüssig sein und darf keine Verweise auf das Fachschema haben.

Die beiden Schemata müssen im UML Modell in separaten *Packages* geführt werden. Beide Packages müssen den *Stereotype* <<Application Schema>> tragen.

Packages mit dem *Stereotype* <<Application Schema>> werden über Steuerparameter qualifiziert. Der Steuerparameter „targetNamespace“ bezeichnet den XML-Namespaces des Anwendungsschemas, „xmlns“ die in XML-Schema-Dokumenten zu verwendende Kurzform; beide Steuerparameter müssen bei Packages mit dem Stereotype gesetzt werden. Der Steuerparameter „ISO19100“ wird auf „True“ gesetzt, wenn die Inhalte des Packages Teil des konzeptuellen Modells der Normenserie ISO 19100 sind. Sofern der Name der zugehörigen XML-Schema-Datei vom Namen des Packages abweichen soll, kann der Dateiname im Steuerparameter „Dateiname“ gesetzt werden (ohne Endung „.xsd“).

Weiterhin müssen die Abhängigkeiten zwischen Packages mit dem *Stereotype* <<Application Schema>> vollständig gesetzt sein. Der Einsatz von Vererbung ist nur bei Objekten mit Identität erlaubt (*Stereotypes* <<Feature>>, <<Type>> und ohne *Stereotype*) sowie bei im Modell abschließend definierten Aufzählungen (*Stereotypes* <<Enumeration>> und <<CodeList>>). Dies soll vor allem dazu dienen, die XML-Schema-Struktur der NAS möglichst einfach zu gestalten. Durch die Vermeidung von Vererbung bei *DataTypes* kann auf die Anwendung von ISO 19118 A.5.2.9 verzichtet werden. Bei enthaltenen Elementen - eben bei *DataTypes* - wird die Definition entsprechender *choice*-Elemente überflüssig und das XML Schema einfacher auswertbar.

Auch die mit der NAS durchführbaren Operationen werden in UML modelliert und automatisch in XML-Definitionen überführt. Zu diesem Zweck werden Auftragsobjekt-klassen mit dem *Stereotype* <<Request>> und Ergebnisobjekt-klassen mit dem *Stereotype* <<Response>> modelliert (s. Abschnitt 10.2). Die Objekt-klassen müssen in einem *Package* mit dem *Stereotype* <<Application Schema>> liegen. Jeder dieser Objekt-klassen wird bei der Abbildung in die NAS in ein XML-Root-Element in einer eigenen XSD-Datei überführt. Alle Klassen zu einer Operation, d.h. zu einem Auftrag oder einem Ergebnis, müssen mit diesem in einem Paket liegen. Bei der Modellierung der Operationen, d.h. von Aufträgen und Ergebnissen, ist auf Konsistenz mit ISO 19119 *Services* zu achten.

Alle Verweise auf Klassen aus den zugrundeliegenden ISO-Schemata müssen zur korrekten Abbildung in die NAS über Steuerparameter konkretisiert werden. Entweder muss die ISO-Klasse über den Steuerparameter "ISOTypBeruecksichtigen" als zu exportieren markiert sein oder das Attribut bzw. die Assoziation im AAA-Basischema oder -Fachschemata muss anderweitig abgebildet werden (über die Steuerparameter "Unterdruecken", "ImplementiertDurchTyp", "ExtendsType", "ErsetzendesElement" oder "ReferenziertesElement").

Die Beschreibung der NAS-Operationen werden an einer zentralen Stelle vorgehalten. Jeder Anbieter von NAS-Operationen, d.h. insbesondere jede Stelle, die eine Bestandsdatenbank führt, soll entsprechende *Service Metadata* entsprechend ISO 19119 *Services*, Abschnitt 6.6, bereitstellen. Der Name des *Service-Interfaces* (Eigenschaft "SV_ServiceIdentification.serviceType") ist "NAS" mit dem *nameSpace* "AdV".

Zeichensatz und unterstützte Sprachen

Wie in ISO 19118 A.2.3 spezifiziert, soll grundsätzlich der *Universal Character Set* (UCS) von "ISO-10646-1" als Zeichenvorrat verwendet werden. Dieser ist identisch mit dem *Unicode Character Repertoire*.

Als *Character Encoding* für NAS-Daten soll einheitlich "UTF-8" (*UTF = UCS Transformation Format*) verwendet werden. "UTF-8" ist auch der Standardwert in XML, falls eine Encoding-Angabe fehlt.

Sprache ist Deutsch ("de") oder Sorbisch (Niedersorbisch bzw. Obersorbisch).

Exchange Metadata

Im Zuge der Modellierung der Aufträge und Ergebnisse werden jeweils die erforderlichen Exchange Metadata modelliert und mit der automatischen Umsetzung nach XML Schema überführt.

Identifikatoren

Identifikatoren sind in der NAS nur auf der Ebene der Fachobjekte definiert, d.h. in allen XML-Elementen die Typen repräsentieren, welche eine Unterklasse von AA_Objekt sind. Bei diesen sind die Identifikatoren stets anzugeben (mit Ausnahme der weiter unten definieren Fälle). Identifikatoren an allen übrigen Elementen werden überlesen und nicht beachtet.

Die Identifikatoren an Fachobjekten sind stets im Sinne von UUIDs zu verstehen, d.h. sie sind innerhalb der "AFIS-ALKIS-ATKIS-"Application-Domain" eindeutig.

Der AAA-Identifikator besteht stets aus 16 Zeichen. Der Aufbau wird in Kapitel 3.3.7 beschrieben.

Verweise auf andere Objekte werden stets als XLink repräsentiert. Innerhalb der NAS sind Verweise auf andere AAA-Objekte ausnahmslos über [URNs](#) auszudrücken. *Uniform Resource Names* (URNs) dienen als persistente, Speicherort-unabhängige Identifikatoren, in denen die Angabe von Namensräumen (*namespaces*) möglich ist. Die AAA-Identifikatoren liegen im *URN-Namespace* "adv:oid".

Beispiel: "urn:adv:oid:DENWAL1212345678".

Der Identifikator wird in den folgenden Fällen um die Zeitangabe des Beginndatums ergänzt:

- Im Benutzungsergebnis, wenn mehrere Versionen eines Objektes in einer NAS-Datei enthalten sind, innerhalb des Attributes gml:id des Objekts (s. Abschn. 10.2.3).
- Im Fortführungsauftrag bei den Operationen Ersetzen (replace) und Löschen (delete) im Attribut fid des Filterausdrucks (s. Abschn. 10.2.3).

Bei Verweisen auf andere Objekte (z.B. in Assoziationen) wird hingegen stets der Identifikator ohne Zeitangabe verwendet.

Updatemechanismen

Ein Updatemechanismus im Sinne von ISO 19118 Kapitel 8 wird unterstützt (s. Punkt 10.1.6)

10.1.3.2 Eingangsdatenstruktur

Das Instanzenschema wird mit den nachfolgenden Änderungen aus ISO 19118 Annex A.3 übernommen.

Bezüglich der Angabe von Identifikatoren in IM_Object gilt aufgrund der obigen Festlegungen zum Identifikator, dass dieser stets in "id" anzugeben ist. Eine dazu redundante Wiederholung in "uuid" ist nicht erlaubt, dieses Attribut fehlt stets.

Tabelle: Festlegungen bei der Abbildung von UML-Konzepten in das Instanzenschema in der Übersicht

UML	Instanzenschema
<<Interface>>	siehe <u>ISO 19118 A.3.2</u>
<<BasicType>>	siehe <u>ISO 19118 A.3.2</u>
<<DataType>>	siehe <u>ISO 19118 A.3.2</u>
<<Request>>	IM_StructuredValue
<<Response>>	IM_StructuredValue
<<Union>>	siehe <u>ISO 19118 A.3.2</u>
<<Enumeration>>	siehe <u>ISO 19118 A.3.2</u>
<<CodeList>>	siehe <u>ISO 19118 A.3.2</u>
<<ExternalCodeList>>	wie <<CodeList>>
<<Feature>>	IM_Object
<<Type>> oder kein Stereotype	siehe <u>ISO 19118 A.3.2</u>
Abstrakte Klasse	siehe <u>ISO 19118 A.3.2</u>
Abgeleitete Klasse	siehe <u>ISO 19118 A.3.2</u>
Attribute	siehe <u>ISO 19118 A.3.2</u>
Assoziation	siehe <u>ISO 19118 A.3.2</u>
Aggregation	wie Assoziation
Komposition	siehe <u>ISO 19118 A.3.2</u>
Methoden	siehe <u>ISO 19118 A.3.2</u>
OCL-Constraints	siehe <u>ISO 19118 A.3.2</u>
<<Application Schema>> (Package)	XML-Schema-Datei

Abgebildet werden nur Eigenschaften mit globaler Sichtbarkeit, keine Attribute und Relationen, die als "*private*" oder "*protected*" gekennzeichnet sind.

Abgeleitete Attribute („*derived*“) werden standardmäßig in die NAS abgebildet, dasselbe gilt für Klassenattribute.

In allen übrigen Punkten gelten die Regeln aus ISO 19118 Annex A.3.2.

10.1.3.3 Ausgabedatenstruktur

Das Instanzenmodell wird durch die NAS Encoding Rules in Übereinstimmung mit ISO 19118 A.4 in ein XML-Dokument abgebildet. Das Ergebnis der Abbildung ist valide und bezieht sich auf das zugehörige, automatisch erzeugte XML Schema.

10.1.3.4 Schema-Abbildungsregeln

Packages mit Stereotype <<Application Schema>>, vordefinierte XML-Schema-Dokumente

Bei der Überführung der UML-Schemata wird ein mit dem *Stereotype* <<Application Schema>> versehenes Package in eine XML-Schema-Datei (XSD-Datei) abgebildet.

Dies gilt für beide AAA-Schemata, sowohl für das AAA-Basisschema als auch für das darauf aufbauende AAA-Fachschemata. Beide werden mit den hier beschriebenen NAS Encoding Rules in getrennte XSD-Dateien mit den Namen "[AAA-Basisschema.xsd](#)" und "[AAA-Fachschemata.xsd](#)" überführt, wobei das Fachschema auch hier auf das Basisschema aufbaut. Die in diesen Dateien beschriebenen Schemakomponenten beschreiben das GML-Anwendungsschema von AFIS-ALKIS-ATKIS.

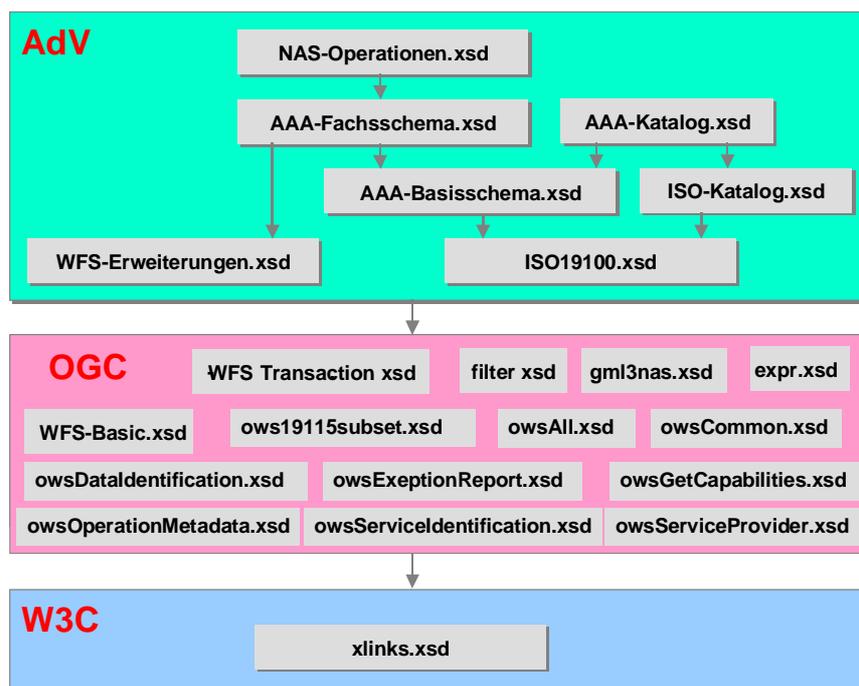


Abbildung 10-4: Grundstruktur der zentralen XML Schemata der NAS

Die Trennung zwischen Basisschema und Fachschema wird somit auch in die XML-Schema-Definitionen übernommen. Beide Schemata liegen im *XML-Namespace* "

<http://www.adv-online.de/namespaces/adv/gid/5.0>“. Ein wesentlicher Grund für diese Trennung ist das Ziel, anderen Fachbereichen außerhalb der Vermessungsverwaltung die Entwicklung eigener Fachschemata auf der Grundlage des AAA-Basischemas zu ermöglichen. Diese können eigene Fachschemata modellieren und anhand der NAS Encoding Rules nach XML Schema abbilden. Die Datei "AAA-Fachschemata.xsd" ist für diese Anwendungen durch ein entsprechendes anderes GML-Anwendungsschema zu ersetzen. Diese Anwendungsschemata müssen in einem anderen Namespace als dem oben genannten liegen, die Verwendung des AdV-Namespace ist der AdV vorbehalten.

Auf das AAA-Fachschemata aufbauend wird für die einzelnen, mit der NAS durchzuführenden Operationen, in der XSD-Datei "NAS-Operationen.xsd" die Beschreibung der Aufträge und Ergebnisse der jeweiligen Operation definiert (s. Punkt 10.2).

Grundsätzlich gelten die Aussagen aus ISO 19118 A.5.1.

Für die direkt aus den ISO-Schemata übernommenen Klassen wird eine XSD-Datei [ISO19100.xsd](#) erzeugt. Die Elemente und Typen liegen im *adv-namespace*, da es sich um eine AdV-spezifische Umsetzung handelt (ISO selbst hat bisher keine eigenen XML-Schema-Definitionen festgelegt, die verwendet werden könnten).

Die in 10.2 beschriebenen Erweiterungen an den WFS- und FES-XML-Schemata werden in einer XSD-Datei „WFS-Erweiterungen.xsd“ beschrieben.

Zum Austausch von Objektartenkatalogen über XML (außerhalb der NAS) wurden aus den AC-Klassen im AAA-Basischema und den FC-Klassen aus ISO 19110 die XSD-Dateien „ISO-Katalog.xsd“ und „AAA-Katalog.xsd“ abgeleitet.

Basisdatentypen

Die Abbildung der Basisdatentypen (<<BasicType>>) entsprechend ISO 19118 A.5.2.1 gilt unverändert auch für die NAS.

Für die zurzeit im UML-Modell verwendeten Basisdatentypen gilt:

Tabelle: Abbildung der Basisdatentypen (Auszug)

Basisdatentyp in UML	Datentyp in der NAS
Integer	xs:integer
Number	xs:double
Decimal	xs:decimal
Real	xs:double
CharacterString	xs:string
Date	xs:date

DateTime	xs:dateTime
Boolean	xs:boolean

Die Abbildung dieser Tabelle, d.h. vom Typ in UML zum zugehörigen XML-Schema-Basisdatentyp erfolgt durch den Steuerparameter "ImplementiertDurchTyp". Diese Konstruktion erlaubt auch Fachinformationssystem die Definition eigener Basisdatentypen, die als Typen mit "simple Content" umgesetzt werden sollen.

Datentypen

Datentypen (<<DataType>>) werden analog zu ISO 19118 A.5.2.2 umgesetzt. Dabei ist zu beachten, dass in der NAS Attribute und Assoziationen anders als in ISO 19118 A.5.3 beschrieben umgesetzt werden (siehe unten).

Enumeration

Enumerationen (<<Enumeration>>) werden grundsätzlich wie in ISO 19118 A.5.2.3 beschrieben umgesetzt.

Codelisten

Codelisten (<<CodeList>>) werden wie Enumerationen (ISO 19118 A.5.2.3) umgesetzt. Kodiert wird – sofern vorhanden - der Code, nicht der Wert.

Anmerkung: Dies bedeutet, dass auch Codelisten nur durch die AdV erweitert werden dürfen und eine Änderung zu einer neuen Version der NAS führt.

Als Unterstützung der Übersetzung von Codes in sprechende Beschreibungen werden die Codelisten eines Anwendungsschemas als GML-3.0-Dictionary umgesetzt.

Externe Codelisten

Externe Codelisten (<<ExternalCodeList>>) eines Anwendungsschemas werden als GML-3.0-Dictionary umgesetzt. Sie erscheinen demnach nicht im Ausgabe-Schema, sondern werden in Form eines *dictionaries* in einer "externen" XML-Datei geführt. Dies hat den Vorteil, dass Erweiterungen dieser Codelisten nicht zu einer neuen NAS-Version führen.

In der NAS-Datei wird jeweils <name> und nicht die gml:id oder <description> kodiert.

Beispiel:

```
<?xml version="1.0" encoding="UTF-8"?>
<Dictionary gml:id="AAAExternalCodeLists"
  xmlns="http://www.opengis.net/gml"
  xmlns:gml="http://www.opengis.net/gml">
```

```

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/gml
  gml/3.00/base/dictionary.xsd">
<description>Externe Codelisten von AFIS-ALKIS-ATKIS.</description>
<name>AAA Codelisten</name>
<dictionaryEntry>
  <DefinitionCollection gml:id="AA_Anlassart">
    <description>Stand: 2003-03-10</description>
    <name>AA_Anlassart</name>
    <definitionMember>
      <Definition gml:id="AA_Anlassart_1">
        <description>Nach Quellenlage nicht zu spezifizieren</description>
        <name codeSpace="adv:AA_Anlassart">9998</name>
      </Definition>
    </definitionMember>
    <definitionMember>
      <Definition gml:id="AA_Anlassart_2">
        <description>Sonstiges</description>
        <name codeSpace="adv:AA_Anlassart">9999</name>
      </Definition>
    </definitionMember>
    <definitionMember>
      <Definition gml:id="AA_Anlassart_3">
        <description>Flurstücksdaten fortführen</description>
        <name codeSpace="adv:AA_Anlassart">010000</name>
      </Definition>
    </definitionMember>
    <!-- ... -->
    <definitionMember>
      <Definition gml:id="AA_Anlassart_127">
        <description>Änderung der Anschrift</description>
        <name codeSpace="adv:AA_Anlassart">090300</name>
      </Definition>
    </definitionMember>
    <definitionMember>
      <Definition gml:id="AA_Anlassart_128">
        <description>Änderung der Personendaten</description>
        <name codeSpace="adv:AA_Anlassart">090400</name>
      </Definition>
    </definitionMember>
  </DefinitionCollection>
</dictionaryEntry>
<!-- ... -->
<dictionaryEntry>
  <DefinitionCollection gml:id="AX_CRS">
    <description>Stand: 2003-03-10</description>
    <name>AX_CRS</name>
  </DefinitionCollection>
</dictionaryEntry>
</Dictionary>

```

Union

Union-Datentypen (<<Union>>) werden wie in ISO 19118 A.5.2.5 beschrieben umgesetzt.

Objekte

Objektarten (<<Feature>> und <<Type>> sowie Klassen ohne *Stereotype*) werden konform zu den Regeln aus GML 3.0, Kapitel 8., als *complexType* umgesetzt; der Name ergibt sich aus dem Klassennamen plus das Suffix "Type". Die *complexTypes* "erben" über den "Extension-Mechanismus" von XML Schema entweder direkt oder indirekt von `gml:AbstractFeatureType` (und damit auch den Mechanismus, wie Identifikatoren beschrieben werden).

Der AAA-Identifikator wird bei AFIS-ALKIS-ATKIS-Objekten stets im "gml:id"-Attribut angegeben, z.B.:

```
<AX_Flurstueck gml:id="DEBY0000F0000001">
  ...
</AX_Flurstueck>
```

Set<T> und Sequence<T>

Eigenschaften, die diese Templateklassen als Typ verwenden, werden als GML-Array-Eigenschaften umgesetzt.

Vererbung / abstrakte Klassen

Auch abstrakte Klassen werden im Regelfall als *complexType* abgebildet. Ausnahmen sind typischerweise Mixin-Klassen, bei denen, da XML Schema keine Abbildung multipler Vererbung unterstützt, deren Eigenschaften statt dessen in die instanziiierbaren Unterklassen propagiert werden. Hierbei kommt der in ISO 19118 beschriebenen "copy down"-Mechanismus zur Anwendung.

Die abstrakten Klassen werden im Schema mit dem Attribut 'abstract="true"' versehen, vgl. ISO 19118 A.5.2.10. Die in die NAS nicht zu übernehmenden abstrakten Klassen sind für die automatische NAS-Ableitung über entsprechende Steuerparameter anzugeben ("AbstraktenTypBeruecksichtigen" = "false").

In abgeleiteten Klassen werden die Eigenschaften von im XML Schema repräsentierten abstrakten Oberklassen nicht über den "copy down"-Mechanismus übernommen, sondern diese Klassen werden über "XML-Schema-Extensions" erweitert. Diese Regel für die Umsetzung von Vererbung liegt im Zwischenbereich der beiden Alternativen aus ISO 19118 A.5.2.8 und verwendet Elemente aus beiden.

Beispiel:

```
<!-- aus AAA-Basisschema.XSD: -->

<xs:element name="AA_Objekt" type="adv:AA_ObjektType"
  substitutionGroup="gml:_Feature" abstract="true"/>

<xs:complexType name="AA_ObjektType" abstract="true">
  <xs:complexContent>
    <xs:extension base="gml:AbstractFeatureType">
      <xs:sequence>
        <xs:element name="lebenszeitintervall">
          <xs:complexType>
            <xs:sequence>
              <xs:element ref="adv:AA_Lebenszeitintervall" />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="modellart">
          <xs:complexType>
            <xs:sequence>
              <xs:element ref="adv:AA_Modellart" />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

```

        maxOccurs="unbounded" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="anlass" type="xs:string" minOccurs="0"
    maxOccurs="unbounded">
    <xs:annotation>
      <xs:appinfo>
        <adv:referenzierteCodeList>
          adv:AA_Anlassart
        </adv:referenzierteCodeList>
      </xs:appinfo>
    </xs:annotation>
  </xs:element>
  <xs:element name="zeigtAufExternes" minOccurs="0">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="adv:AA_Fachdatenverbindung"
          maxOccurs="unbounded" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="istTeilVon" type="gml:ReferenceType"
    minOccurs="0" maxOccurs="unbounded">
    <xs:annotation>
      <xs:appinfo>
        <adv:referenziertesElement>
          adv:AA_ZUSO
        </adv:referenziertesElement>
      </xs:appinfo>
    </xs:annotation>
  </xs:element>
  <xs:element name="inversZu_dientZurDarstellungVon"
    type="adv:InverseReferenceType"
    minOccurs="0" maxOccurs="unbounded">
    <xs:annotation>
      <xs:appinfo>
        <adv:referenziertesElement>
          adv:AP_Darstellung
        </adv:referenziertesElement>
        <adv:referenziertesElement>
          adv:AP_LTO
        </adv:referenziertesElement>
        <adv:referenziertesElement>
          adv:AP_FPO
        </adv:referenziertesElement>
        <adv:referenziertesElement>
          adv:AP_PTO
        </adv:referenziertesElement>
        <adv:referenziertesElement>
          adv:AP_LPO
        </adv:referenziertesElement>
        <adv:referenziertesElement>
          adv:AP_PPO
        </adv:referenziertesElement>
        <adv:inverseEigenschaft>
          adv:dientZurDarstellungVon
        </adv:inverseEigenschaft>
      </xs:appinfo>
    </xs:annotation>
  </xs:element>
</xs:sequence>
</xs:extension>
</xs:complexType>

<xs:element name="AA_ZUSO" type="adv:AA_ZUSOType"
  substitutionGroup="adv:AA_Objekt" abstract="true" />

<xs:complexType name="AA_ZUSOType" abstract="true">
  <xs:complexContent>
    <xs:extension base="adv:AA_ObjektType">
      <xs:sequence/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

```

<!-- aus AAA-Fachschema.XSD: -->

<xs:complexType name="AX_SicherungspunktType">
  <xs:complexContent>
    <xs:extension base="adv:AA_ZUSOType">
      <xs:sequence>
        <xs:element name="punktkennung" type="xs:string" minOccurs="0"/>
        <xs:element name="wirdVerwaltetVon" minOccurs="0">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="AX_Dienststelle_Schluessel"
                type="adv:AX_Dienststelle_SchluesselType"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="sonstigeEigenschaft" type="xs:string"
          minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="horizontfreiheit"
          type="adv:AX_Horizontfreiheit_NetzkpunktType" minOccurs="0"/>
        <xs:element name="relativeHoehe" type="gml:LengthType"
          minOccurs="0"/>
        <xs:element name="vermarktung_Marke" type="adv:AX_MarkeType"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

Attribute

Der Umgang mit den Kardinalitäten wird aus ISO 19118 A.5.3.1 übernommen. Alle Attribute werden als Elemente umgesetzt (nicht als XML-Attribute).

Die Angaben zur Kardinalität des Attributs der UML-Klasse wird unmittelbar in „minOccurs“- und „maxOccurs“-Attribute des Elements umgesetzt. Das bedeutet, dass optionale Attribute aus dem konzeptuellen Modell (Kardinalität „0..n“ mit n=1,2,3,...,*) als Elemente mit „minOccurs='0'“ umgesetzt werden. Ist das Attribut in einem Objekt nicht belegt, dann fehlt in der NAS das entsprechende Element in dem Objekt.

Bei der Codierung von Eigenschaften gelten die Regeln aus GML (siehe GML 3.0, Kapitel 7.2 und Kapitel 8).

Assoziationen

Assoziationen werden gemäß den Regeln für by-reference-wertige Eigenschaften in GML umgesetzt. Diese sind grundsätzlich im Einklang mit ISO 19118 A.5.3.2. Der wesentliche Unterschied liegt in der Verwendung von "gml:AssociationAttributeGroup" statt "IM_ObjectReference", für Verweise wird in der NAS stets der vordefinierte GML-Type „gml:ReferenceType“ verwendet. Der Zieltyp der Assoziation wird zur Auswertung durch NAS-Parser in einem oder mehreren appinfo-Annotationselementen "adv:referenziertesElement" angeben. Mehrere Elemente kommen immer dann vor, wenn die Assoziation auf eine abstrakte, nicht im XML-Schema repräsentierte Klasse verweist; in diesem Fall werden die direkt abgeleiteten instanzierbaren Klassen abgeleitet.

GML verwendet ausschließlich XLink für Beziehungen zwischen Objekten. Zur Abbildung der AAA-Schemata benötigt man nur einen kleinen Ausschnitt der Möglichkeiten von XLink, und zwar einfache Verweise auf ein anderes Objekt über einen URN. In der NAS sind nur xlink:href-Attribute erlaubt, sodass man funktional kompatibel mit der "uriref"-Variante von ISO 19118 bleibt.

Verweise auf andere Objekte erfolgen stets über den URN des Zielobjekts. Es ist stets der vollständige URN (im Namensraum "adv:oid") zu verwenden, nicht ein "#"-Verweis, auch wenn das Objekt im selben Dokument beschrieben ist.

In der NAS wird stets nur die im Anwendungsschema als bevorzugte Richtung gekennzeichnete Assoziation angegeben. Gegenrelationen sind, falls notwendig, von der Implementierung zu erzeugen und zu pflegen.

Zur Vereinfachung von Abfragen ist es in FilterEncoding-Ausdrücken erlaubt, Gegenrelationen in Queries zu verwenden. Zu diesem Zweck – und nur zu diesem Zweck – sind die Gegenrelationen unter Verwendung des Typs "adv:InverseReferenceType" im XML Schema enthalten. In den in XML codierten Objektinstanzen sind die Gegenrelationen nie enthalten. Näheres siehe bei 10.2.4 "Anfordern von Ausgaben".

Anmerkung: Die beschriebene Abbildungsregel für Assoziationen stellt nicht sicher, dass das Zielobjekt von der erwarteten Objektart ist bzw. es überhaupt ein gültiges Zielobjekt gibt. Diese Prüfung ist vom Zielsystem selbst durchzuführen. Zu diesem Zweck wurden entsprechende Annotationen eingeführt. Beispiel:

a) XML-Schema:

```
<xs:complexType name="AX_FlurstueckType">
  <!-- ... -->
  <xs:element name="istGebucht" type="gml:ReferenceType">
    <xs:annotation>
      <xs:appinfo>
        <adv:referenziertesElement>
          adv:AX_Buchungsstelle
        </adv:referenziertesElement>
      </xs:appinfo>
    </xs:annotation>
  </xs:element>
  <!-- ... -->
</xs:complexType>
```

b) XML-Datei:

```
<AX_Flurstueck gml:id="DEBY0000F0000002">
  ...
  <istGebucht xlink:type="simple"
    xlink:href="urn:adv:oid:DEBY0000B0000002">
```

```
...  
</AX_Flurstueck>
```

Aggregationen

Zur Vereinfachung und zur Reduzierung der für die NAS notwendigen Konstrukte werden Aggregationen wie Assoziationen umgesetzt.

Kompositionen

Kompositionen werden gemäß den Regeln für by-value-wertige Eigenschaften in GML umgesetzt.

Unterstützung für mehrere Sprachen

Ein Bedarf dafür ist zurzeit nicht erkennbar.

Elementdefinitionen

Für jeden Auftrag (*Request*) und für jedes Ergebnis (*Response*) wird jeweils ein *Root-Element* in einer eigenen XSD-Datei mit dem Namen der Objektklasse definiert (s. Punkt 10.1.6).

Für jede Objektart wird ein globales Element definiert (in der 'substitutionGroup' des Elements, das als Typ den Typ trägt, aus dem der Typ der Objektart abgeleitet wird). Name des Elements ist der Klassenname.

GML - und somit auch die NAS - verwendet das "substitutionGroup"-Attribut, um zu ermöglichen, dass sich ein XML-Element, welches ein Objekt repräsentiert, auch wie ein Feature "verhalten" kann; d.h. dort stehen kann wo ein XML-Element "_Feature" erwartet wird. Beispiel:

```
<xs:element  
  name="AX_Flurstueck"  
  type="adv:AX_FlurstueckType"  
  substitutionGroup="adv:TA_MultiSurfaceComponent" />
```

In allen anderen Fällen werden standardmäßig lokale Elemente definiert.

Vordefinierte Typen und Elemente

Die in GML vordefinierten Elemente und Typen können entsprechend den Regelungen von GML verwendet werden. Für Operationen (Aufträge und Ergebnisse) können außerdem die Elemente und Typen der [Web Feature Server Spezifikation](#) verwendet werden (s. Punkt 10.2). Die Verwendung dieser Typen und Elemente in der NAS wird durch Steuerparameter (siehe unten) geregelt.

Auf die Abbildung der Geometriemodellierung im AAA-Basisschema auf GML im Rahmen der NAS wird nachfolgend eingegangen.

Weiterhin können direkt aus dem ISO-Schema abgeleitete Elemente verwendet werden (ISO19100.xsd). Diese liegen zurzeit ausschließlich im Bereich der Metadaten.

Abbildung der Geometrieigenschaften des Basisschemas in der NAS

Es ist für die NAS festzulegen, wie die verwendeten Klassen aus den ISO-19100-Normen auf die vordefinierten Typen abgebildet werden sollen. Dies erfolgt über Steuerparameter (siehe unten). Die Abbildung muss die im ISO-Modell enthaltenen und für AFIS-ALKIS-ATKIS relevanten Informationen abbilden.

Auf die Kodierung der Orientierung von Linien (Curves) wird verzichtet. Da die Richtung einer Linie aber fallweise (z.B. Fließrichtung von Gewässern) eine Bedeutung hat, muss a) die Erfassung in positiver Richtung erfolgen und b) sichergestellt werden, dass diese Richtung im Zuge der Verarbeitung und Speicherung unverändert bleibt. Damit kann davon ausgegangen werden, dass die Linienorientierung in der NAS immer positiv ist und es einer gesonderten Kennzeichnung nicht bedarf.

Bei Flächenumringen liegt die begrenzte Fläche gemäß ISO 19107 immer zur Linken der in positiver Richtung orientierten begrenzenden Linien.

Um die NAS möglichst einfach zu gestalten, wird Geometrie ausschließlich redundant ausgetauscht. NAS-Daten aufnehmende Programmsysteme müssen Topologie bzw. gemeinsame Geometrienutzung selbst erkennen - sofern sie sich für diese Information interessieren. Die Einstiegshürde für die Nutzung von AFIS-ALKIS-ATKIS-Daten wird dadurch möglichst niedrig gehalten.

Das "Erkennen" von Geometrieteilung wird durch die folgenden Punkte - auf möglichst einfache Weise - definiert.

Topologische Objekte und solche mit gemeinsam genutzter Geometrie können Themen zugeordnet werden. Topologische Beziehungen und gemeinsame Geometrienutzung sind nur innerhalb eines Themas möglich. Ein Thema ist immer auf eine Modellart beschränkt.

Damit zwei Geometrien identisch sind, müssen sie identische Definitionen in einem <Point> bzw. einer <Curve> besitzen, ein identischer Geometrieverlauf allein ist bei Linien nicht ausreichend.

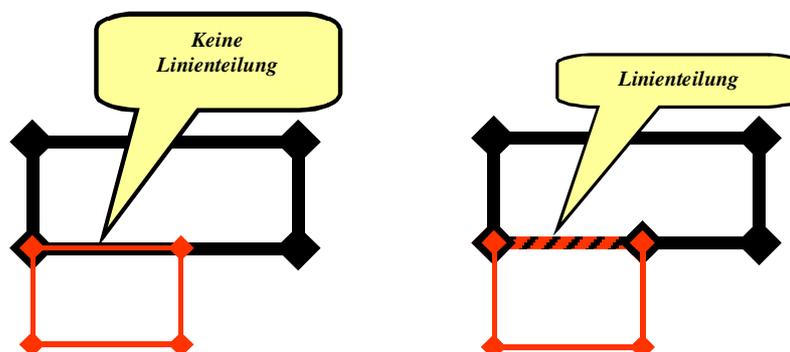


Abbildung 10-5: Erläuterung zur Linienteilung

Identität bei Linien wird stets auf der Ebene der "GM_Curve" untersucht. Sie ist gegeben, wenn alle Positionen der Geometriedefinition in Lage und Reihenfolge sowie verwendeter Interpolationsart identisch sind. Hierbei ist auch eine Umkehrung der Reihenfolge erlaubt.

Zwei Positionen sind identisch, wenn ihr Abstand kleiner als die vorzugebende Koordinatenauflösung ist. In AFIS-ALKIS-ATKIS ist diese für metrische Lagekoordinaten auf 3 Nachkommastellen (mm) festgelegt. Diese Festlegung gilt unabhängig von der tatsächlichen Genauigkeit der Koordinaten.

Zur Erläuterung werden die beiden Situation aus Abbildung 10-5 beispielhaft codiert. Zuerst die linke Situation (ohne Linienteilung):

```

...
<AX_Flurstueck gml:id="DEBY0000000000001">
  <!-- ... -->
  <position>
    <gml:Surface>
      <gml:patches>
        <gml:PolygonPatch>
          <gml:exterior>
            <gml:Ring>
              <gml:curveMember>
                <gml:Curve>
                  <gml:segments>
                    <gml:LineStringSegment>
                      <gml:pos>3601085.954 5943996.138</gml:pos>
                      <gml:pos>3601085.954 5943998.138</gml:pos>
                    </gml:LineStringSegment>
                  </gml:segments>
                </gml:Curve>
              </gml:curveMember>
              <gml:curveMember>
                <gml:Curve>
                  <gml:segments>
                    <gml:LineStringSegment>
                      <gml:pos>3601085.954 5943998.138</gml:pos>
                      <gml:pos>3601078.954 5943998.138</gml:pos>
                    </gml:LineStringSegment>
                  </gml:segments>
                </gml:Curve>
              </gml:curveMember>
            </gml:Ring>
          </gml:exterior>
        </gml:PolygonPatch>
      </gml:patches>
    </gml:Surface>
  </position>
</AX_Flurstueck>

```

```

    <gml:curveMember>
      <gml:Curve>
        <gml:segments>
          <gml:LineStringSegment>
            <gml:pos>3601078.954 5943998.138</gml:pos>
            <gml:pos>3601078.954 5943996.138</gml:pos>
          </gml:LineStringSegment>
        </gml:segments>
      </gml:Curve>
    </gml:curveMember>
    <gml:curveMember>
      <gml:Curve>
        <gml:segments>
          <gml:LineStringSegment>
            <gml:pos>3601078.954 5943996.138</gml:pos>
            <gml:pos>3601085.954 5943996.138</gml:pos>
          </gml:LineStringSegment>
        </gml:segments>
      </gml:Curve>
    </gml:curveMember>
  </gml:Ring>
</gml:exterior>
</gml:PolygonPatch>
</gml:patches>
</gml:Surface>
</position>
<!-- ... -->
</AX_Flurstueck>
...
<AX_Gebaeude gml:id="DEBY0000000000002">
  <!-- ... -->
  <position>
    <gml:Surface>
      <gml:patches>
        <gml:PolygonPatch>
          <gml:exterior>
            <gml:Ring>
              <gml:curveMember>
                <gml:Curve>
                  <gml:segments>
                    <gml:LineStringSegment>
                      <gml:pos>3601082.954 5943994.138</gml:pos>
                      <gml:pos>3601082.954 5943996.138</gml:pos>
                    </gml:LineStringSegment>
                  </gml:segments>
                </gml:Curve>
              </gml:curveMember>
              <gml:curveMember>
                <gml:Curve>
                  <gml:segments>
                    <gml:LineStringSegment>
                      <gml:pos>3601082.954 5943996.138</gml:pos>
                      <gml:pos>3601078.954 5943996.138</gml:pos>
                    </gml:LineStringSegment>
                  </gml:segments>
                </gml:Curve>
              </gml:curveMember>
              <gml:curveMember>
                <gml:Curve>
                  <gml:segments>
                    <gml:LineStringSegment>
                      <gml:pos>3601078.954 5943996.138</gml:pos>

```

```

        <gml:pos>3601078.954 5943994.138</gml:pos>
      </gml:LineStringSegment>
    </gml:segments>
  </gml:Curve>
</gml:curveMember>
<gml:curveMember>
  <gml:Curve>
    <gml:segments>
      <gml:LineStringSegment>
        <gml:pos>3601078.954 5943994.138</gml:pos>
        <gml:pos>3601082.954 5943994.138</gml:pos>
      </gml:LineStringSegment>
    </gml:segments>
  </gml:Curve>
</gml:curveMember>
</gml:Ring>
</gml:exterior>
</gml:PolygonPatch>
</gml:patches>
</gml:Surface>
</position>
<!-- ... -->
</AX_Gebaeude>
...

```

Und zum Vergleich die rechte Situation, bei der die untere Kante des Flurstücks zur Herstellung der Geometrieidentität im oben beschriebenen Sinne in zwei Kanten aufgetrennt wurde:

```

...
<verwendeteInstanzenthemen>Flurstücke und
Gebäude</verwendeteInstanzenthemen>
...
<AX_Flurstueck gml:id="DEBY000000000001">
  <!-- ... -->
  <position>
    <gml:Surface>
      <gml:patches>
        <gml:PolygonPatch>
          <gml:exterior>
            <gml:Ring>
              <gml:curveMember>
                <gml:Curve>
                  <gml:segments>
                    <gml:LineStringSegment>
                      <gml:pos>3601085.954 5943996.138</gml:pos>
                      <gml:pos>3601085.954 5943998.138</gml:pos>
                    </gml:LineStringSegment>
                  </gml:segments>
                </gml:Curve>
              </gml:curveMember>
            </gml:curveMember>
          </gml:Curve>
        </gml:patches>
      </gml:Surface>
    </position>
  </AX_Flurstueck>

```

```

    </gml:Curve>
  </gml:curveMember>
</gml:curveMember>
  <gml:Curve>
    <gml:segments>
      <gml:LineStringSegment>
        <gml:pos>3601078.954 5943998.138</gml:pos>
        <gml:pos>3601078.954 5943996.138</gml:pos>
      </gml:LineStringSegment>
    </gml:segments>
  </gml:Curve>
</gml:curveMember>
</gml:curveMember>
  <gml:Curve>
    <gml:segments>
      <gml:LineStringSegment>
        <gml:pos>3601078.954 5943996.138</gml:pos>
        <gml:pos>3601082.954 5943996.138</gml:pos>
      </gml:LineStringSegment>
    </gml:segments>
  </gml:Curve>
</gml:curveMember>
</gml:curveMember>
  <gml:Curve>
    <gml:segments>
      <gml:LineStringSegment>
        <gml:pos>3601082.954 5943996.138</gml:pos>
        <gml:pos>3601085.954 5943996.138</gml:pos>
      </gml:LineStringSegment>
    </gml:segments>
  </gml:Curve>
</gml:curveMember>
</gml:Ring>
</gml:exterior>
</gml:PolygonPatch>
</gml:patches>
</gml:Surface>
</position>
<!-- ... -->
</AX_Flurstueck>
...
<AX_Gebaeude gml:id="DEBY000000000002">
  <!-- ... -->
  <position>
    <gml:Surface>
      <gml:patches>
        <gml:PolygonPatch>
          <gml:exterior>
            <gml:Ring>
              <gml:curveMember>
                <gml:Curve>
                  <gml:segments>
                    <gml:LineStringSegment>
                      <gml:pos>3601082.954 5943994.138</gml:pos>
                      <gml:pos>3601082.954 5943996.138</gml:pos>
                    </gml:LineStringSegment>
                  </gml:segments>
                </gml:Curve>
              </gml:curveMember>
            </gml:Ring>
          </gml:exterior>
        </gml:PolygonPatch>
      </gml:patches>
    </gml:Surface>
  </position>
  <!-- ... -->
  <gml:Curve>
    <gml:segments>
      <gml:LineStringSegment>
        <gml:pos>3601082.954 5943994.138</gml:pos>
        <gml:pos>3601082.954 5943996.138</gml:pos>
      </gml:LineStringSegment>
    </gml:segments>
  </gml:Curve>
</gml:curveMember>
</gml:curveMember>
  <gml:Curve>
    <gml:segments>

```

```

        <gml:LineStringSegment>
            <gml:pos>3601082.954 5943996.138</gml:pos>
            <gml:pos>3601078.954 5943996.138</gml:pos>
        </gml:LineStringSegment>
    </gml:segments>
</gml:Curve>
</gml:curveMember>
<gml:curveMember>
    <gml:Curve>
        <gml:segments>
            <gml:LineStringSegment>
                <gml:pos>3601078.954 5943996.138</gml:pos>
                <gml:pos>3601078.954 5943994.138</gml:pos>
            </gml:LineStringSegment>
        </gml:segments>
    </gml:Curve>
</gml:curveMember>
<gml:curveMember>
    <gml:Curve>
        <gml:segments>
            <gml:LineStringSegment>
                <gml:pos>3601078.954 5943994.138</gml:pos>
                <gml:pos>3601082.954 5943994.138</gml:pos>
            </gml:LineStringSegment>
        </gml:segments>
    </gml:Curve>
</gml:curveMember>
</gml:Ring>
</gml:exterior>
</gml:PolygonPatch>
</gml:patches>
</gml:Surface>
</position>
<!-- ... -->
</AX_Gebaeude>
...

```

Die Themen werden in der NAS-Datei wie folgt abgebildet:

- Die Themen, und dies gilt sowohl für "TS_Theme" als auch das "PunktLinien-Thema", sind (implizite) Realisierungen von *GM_Complex* und sind letztlich eine Aggregation von geometrischen Elementen. Sie können im AFIS-ALKIS-ATKIS-Kontext in zwei Varianten vorkommen. Zum einen in einer vollständigen Form (Art der Themendeklaration = "alle Objekte", Klassenthemen) und in einer instanzenbezogenen Form (Art der Themendeklaration = "anwenderdefiniert", Instanzenthemen).
- Bei der vollständigen, klassenbezogenen Form liegen alle Objekte einer Objektart automatisch in diesem Thema. Es besteht keine "Wahlmöglichkeit". Die explizite Angabe des Themas ist deshalb nicht erforderlich.

Bei der instanzenbezogenen Form können objektbezogene geometrische Identitäten zum Ausdruck gebracht werden, z.B. zwischen einer Gebäudelinie und einer Flurstücksgrenze. Die Namen der in einer NAS-Datei verwendeten Instanzenthemen sind jeweils in den "Exchange Metadata" anzugeben, d.h. sie werden in den Auftrags- und Ergebnisobjekten

modelliert (siehe oben). Die Angabe muss nicht notwendigerweise auf die tatsächlich in der jeweiligen Datei vorkommenden Instanzen-Themen beschränkt werden. Neben der Deklaration der Themen entscheidet die Art der Geometrie (identische Punkte oder Linien, s.o.) über eine eventuelle Geometrieidentität mit redundanzfreier Datenspeicherung oder explizitem Identitätsverweis. Bei „Widersprüchen“ zwischen Geometrie und Themendeklaration „gewinnt“ letztere, d.h., werden identische Geometrien gefunden, aber es ist kein entsprechendes Thema definiert, so wird keine beabsichtigte Identität angenommen (die Geometrien werden redundant abgelegt).

In einer NAS-Datei kann das z.B. wie folgt aussehen:

```
<?xml version="1.0" encoding="UTF-8"?>
<AX_Fortfuehrungsauftrag
  xmlns:adv="http://www.adv-online.de/namespaces/adv/gid/5.0"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:ogc="http://www.opengis.net/ogc"
  xmlns:wfs="http://www.opengis.net/wfs"
  xmlns:gml="http://www.opengis.net/gml"
  xsi:schemaLocation="http://www.adv-online.de/namespaces/adv/gid/5.0 NAS-
  Operationen.xsd">

  <profilkennung>ii</profilkennung>
  <antragsnummer>4711</antragsnummer>
  <empfaenger>
    <AX_Empfaenger>
      <email>mailto:mustermann@foobar.de</email>
    </AX_Empfaenger>
  </empfaenger>
  <auftragsnummer>1174</auftragsnummer>
  <verarbeitungsart>1000</verarbeitungsart>

  <verwendeteInstanzenthemen>Flurstücke und Gebäude</verwendeteInstanzenthemen>
  <verwendeteInstanzenthemen>Flurstücke und öffentlich-rechtliche
  Festlegungen</verwendeteInstanzenthemen>
  <verwendeteInstanzenthemen>Flurstücke und tatsächliche
  Nutzung</verwendeteInstanzenthemen>

  ...

</AX_Fortfuehrungsauftrag>
```

Die Definitionen der Themen sind in einer [XML-Datei](#) zusammengefasst.

Codierung der Punktmengenobjekte in der NAS

Für die Codierung von Punktmengenobjekte (Coverages) in einer NAS-Datei wird auf die Implementierung der entsprechenden CV-Klassen aus ISO 19123 in GML zurückgegriffen.

- Die Klasse `AD_PunktCoverage` wird als Erweiterung von `gml:MultiPointCoverageType` definiert (als abstrakter XML Schema Type `adv:AD_PunktCoverage`).
- Die Klasse `AD_GitterCoverage` wird als Erweiterung von `gml:RectifiedGridCoverageType` definiert (als abstrakter XML Schema Type `adv:AD_GitterCoverage`).

Als Besonderheit der Abbildung ist zu beachten, dass diese abstrakten Klassen des Basisschemas nicht direkt `gml:AbstractFeatureType` erweitern, sondern die erwähnten GML-Typen. Diese Zuordnung erfolgt durch einen NAS-Steuerparameter („ExtendsType“, siehe unten).

Da die beiden vordefinierten GML-Typen mit ihren Eigenschaften `gml:domainSet` und `gml:rangeSet` die im konzeptuellen AA_PMO-Modell beschriebenen spezifischen Coverage-Eigenschaften bereits abdecken, werden diese Attribut- und Relationsarten in der NAS-Codierung unterdrückt und stattdessen die vordefinierten GML-Eigenschaften verwendet.

In dem in der NAS verwendeten GML-Profil werden darüber hinaus im Einklang mit der konzeptuellen Modellierung weitere Einschränkungen vorgenommen. So sind z.B. zur Darstellung des *range set* nur die Varianten *file* und *data block* erlaubt.

Die aus AA_PMO geerbten Eigenschaften werden auf bereits in den GML-Typen vordefinierte Eigenschaften abgebildet: (`gml:name`, `gml:description` und `gml:boundedBy`) und entsprechend bei der Ableitung unterdrückt.

Die aus AA_Objekt geerbten Eigenschaften werden in den Coverage-Typen über den oben beschriebenen *copy-down*-Mechanismus automatisch übernommen.

Codierung der Koordinatenreferenzsysteme in der NAS

Grundsätzlich muss jede Geometrieinheit in der NAS-Datei (Punkt, Linie, Fläche) auf ein Koordinatenreferenzsystem (CRS, s. 3.3.4.5) verweisen. Dies kann entweder implizit durch Angabe des CRS bei einer übergeordneten Geometrieinheit oder explizit bei der jeweiligen Geometrieinheit erfolgen. Der Verweis erfolgt durch Angabe eines [URI](#) (*Uniform Resource Identifier*). Um diese Angabe nicht immer bei jeder Objektgeometrie machen zu müssen, werden in den Exchange Metadata der NAS alle verwendeten Referenzsysteme angegeben, von denen eines als Standardreferenzsystem gekennzeichnet werden kann. Für Geometrien, die in diesem Standardreferenzsystem vorliegen, muss keine Angabe zum Koordinatenreferenzsystem mehr gemacht werden. Das dafür

vorgesehene Attribut bei GML-Geometrien "srsName" ist in diesen Fällen nicht vorhanden. Für alle Geometrien, die nicht im Standardreferenzsystem vorliegen, ist das Attribut zu belegen. Hierbei sind die in Kapitel 11 beschriebene Syntax und die dort definierten Bezeichnungen zu verwenden.

Bei NAS-Dokumenten, die Objekte in einer „FeatureCollection“ enthalten, ist das Standardreferenzsystem im Attribut „srsName“ des „gml:Envelope“ anzugeben.

Darüber hinaus dient die Deklaration der verwendeten Koordinatenreferenzsysteme in den Exchange Metadata der Angabe der für das Referenzsystem geltenden Koordinatenauflösung bzw. der Anzahl der relevanten Nachkommastellen. Diese kann von Referenzsystem zu Referenzsystem unterschiedlich sein und macht keine Aussage über die Genauigkeit der Koordinaten. In AFIS-ALKIS-ATKIS ist die Koordinatenauflösung für metrische Lagekoordinaten auf 3 Nachkommastellen (mm) festgelegt. Die Angabe der relevanten Nachkommastellen ist notwendig, da sowohl GML als auch ISO 19107 *Spatial Schema* keine Einschränkungen diesbezüglich machen und auch keine Möglichkeiten dazu vorsehen (Datentyp: *decimal* oder *double*). Folgende Definition wird in den NAS Schema-Dateien verwendet:

```
<xs:complexType name="AX_KoordinatenreferenzsystemangabenType">
  <xs:sequence>
    <xs:element name="crs" type="xs:anyURI"/>
    <xs:element name="anzahlDerNachkommastellen" type="xs:integer"/>
    <xs:element name="standard" type="xs:boolean"/>
  </xs:sequence>
</xs:complexType>
```

Codierung von Maßeinheiten in der NAS

Grundsätzlich muss jeder mit einer Maßeinheit versehene Wert in der NAS-Datei (z.B. Längen, Flächen, Winkel) auf eine Maßeinheit verweisen. Der Verweis erfolgt durch Angabe eines [URI](#) (*Uniform Resource Identifier*) im Attribut „uom“. Hierbei sind die in Kapitel 11 beschriebene Syntax und die dort definierten Bezeichnungen zu verwenden.

Wahlmöglichkeiten bei der Abbildung

Grundsätzlich bestehen bei der Festlegung der "NAS Encoding Rules" Freiheiten bezüglich der Codierung; im Hinblick auf eine eindeutige Schnittstellenbeschreibung werden diese jedoch in genau einer damit festgelegten Weise genutzt. Alle Wahlmöglichkeiten innerhalb der "NAS Encoding Rules" werden formal so beschrieben, dass die XML-Schemata der NAS in eindeutiger Weise automatisch erzeugt werden können. Dies entspricht der Funktion der Konfigurationstabelle aus den "XML Encoding Rules" (ISO 19118 A.5.7).

Der folgende Informationsbedarf für Zusatzregeln zur Steuerung der automatischen Erzeugung reicht aus, um die NAS automatisiert ableiten zu können:

- Eine nach den Standardregeln zu exportierende Objektklasse soll nicht in der NAS definiert werden. Beispiel: AC-Klassen.
- Eine nach den Standardregeln zu exportierende abstrakte Objektklasse soll nicht in der NAS definiert werden. Beispiele: AP_GPO, AX_Auftrag.
- Ein nach den Standardregeln zu exportierendes Attribut soll nicht in der NAS definiert werden. Beispiel: "identifikator" in AA_Objekt, da in der NAS die XML-Schema-eigenen Mechanismen verwendet werden.
- Ein nach den Standardregeln zu exportierende Rolle soll nicht in der NAS definiert werden. Beispiel: Überflüssige Rollen, die aus *GM_Object* geerbt werden.
- Ein Attribut oder eine Rolle soll durch ein spezielles, lokales Element abgebildet werden. Beispiel: Die Rolle "theme" in *TS_Feature* wird durch ein Element mit Namen "theme" und dem Typ "xs:string" abgebildet.
- In einem komplexen Fall (Geometrieeigenschaften bei AU_Objekt) ist die direkte Abbildung naheliegend. Hier ist das Attribut "position" sinnvollerweise direkt als lokales Property-Element mit einem *choice*-Element der erlaubten Geometrietypen ("gml:Point", "gml:Curve", "gml:CompositeCurve", "gml:MultiCurve" und "gml:Surface") einzubinden.

Eine Objektart soll statt gml:AbstractFeatureType einen anderen in GML vordefinierten *feature type* erweitern. Beispiel: Coverages.

Die Steuerinformationen werden ebenfalls im Anwendungsschema in Rational Rose gehalten, d.h. für Klassen, Attribute und Rollen werden sogenannte "*Tools*" (Datei „nas.pty“) definiert, in denen man die Steuerinformationen ablegt. Die Steuerparameter sind in der Datei [NASSteuerparameterReport.html](#) dokumentiert.

Vorteil der auf *Rational Rose* basierten Lösung ist, dass alle Informationen im Anwendungsschema selbst gehalten werden (und z.B. grundsätzlich auch im Objektartenkatalog usw. wiedergegeben werden könnten).

Es sind die folgenden Steuerparameter definiert:

Tabelle: Übersicht über die definierten Steuerparameter für Klassen

Name	Bedeutung
Unterdruecken	Gibt an, ob die Klasse bei der NAS-Definition unterdrückt werden soll, auch wenn sie nach den Standard-NAS-Encoding-Rules umgesetzt werden würde.
AbstraktenTypBeruecksichtigen	Gibt an, ob die Klasse bei der NAS-Definition berücksichtigt werden soll. Standardmäßig ist dies der Fall (TRUE).

ISOTypBeruecksichtigen	Dieser Steuerparameter wird nur bei Klassen in einem der ISO-19100-Pakete berücksichtigt, nicht bei Klassen in den AFIS-ALKIS-ATKIS-Paketen. Ist der Parameter TRUE, dann wird die Klasse (sowie alle abhängigen Klassen) in ISO19100.xsd aufgenommen.
Metadata	Dieser Steuerparameter wird nur bei Klassen in einem der ISO-19100-Pakete gesetzt, nicht bei Klassen in den AFIS-ALKIS-ATKIS-Paketen. Ist der Parameter TRUE, dann wird die Klasse in ISO19100.xsd berücksichtigt.
ImportSchema	Dieser Steuerparameter wird nur bei Klassen mit dem Stereotype "Request" oder "Response" berücksichtigt. Ist der Parameter gesetzt, dann wird diese Zeile vor der ersten Elementdefinition eingefügt. Es muss sich um ein <xs:import .../>-Element handeln.
Erlaeuterung	Optionale Erläuterung für die getroffenen Angaben, sodass die Entscheidung durch Dritte nachvollzogen werden können.
ExtendsType	Gibt an, dass der complexType über den xsd:extension-Mechanismus einen bestimmten vordefinierten XML-Schema-Type erweitern soll.
ImplementiertDurchTyp	Gibt an, durch welchen XML-Schema-Typ eine UML-Klasse implementiert werden soll.

Tabelle: Übersicht über die definierten Steuerparameter für Attribute

Name	Bedeutung
Unterdruecken	Gibt an, ob das Attribut bei der NAS-Definition unterdrückt werden soll, auch wenn sie nach den Standard-NAS-Encoding-Rules umgesetzt werden würde.
Metadata	Dieser Steuerparameter wird nur bei Attributen in einem der ISO-19100-Pakete gesetzt, nicht in den AFIS-ALKIS-ATKIS-Paketen. Ist der Parameter TRUE, dann wird das Attribut in ISO19100.xsd berücksichtigt.
ErsetzendesElement	Ist dieser Parameter angegeben, so werden die Standardregeln außer Kraft gesetzt und das in diesem Parameter definierte Element verwendet. Es muss sich um ein <xs:element .../>-Element handeln.
ReferenziertesElement	Ist dieser Parameter angegeben, so werden die Standardregeln ausser Kraft gesetzt und auf das in diesem Parameter definierte Element verwiesen. Es muss sich um einen Elementnamen ggf. mit Namespace-deklaration handeln.
Erlaeuterung	Optionale Erläuterung für die getroffenen Angaben, so dass die Entscheidung durch Dritte nachvollzogen werden können.

Tabelle: Übersicht über die definierten Steuerparameter für Rollen in Relationen

Name	Bedeutung
Unterdruecken	Gibt an, ob die Rolle bei der NAS-Definition unterdrückt werden soll, auch wenn sie nach den Standard-NAS-Encoding-Rules umgesetzt werden würde.
Metadata	Dieser Steuerparameter wird nur bei Rollen in einem der ISO-19100-Pakete gesetzt, nicht in den AFIS-ALKIS-ATKIS-Paketen. Ist der Parameter TRUE, dann wird die Rolle in ISO19100.xsd berücksichtigt.
ErsetzendesElement	Ist dieser Parameter angegeben, so werden die Standardregeln ausser Kraft gesetzt und das in diesem Parameter definierte Element verwendet. Es muss sich um ein <xs:element .../>-Element handeln.

ReferenziertesElement	Ist dieser Parameter angegeben, so werden die Standardregeln ausser Kraft gesetzt und auf das in diesem Parameter definierte Element verwiesen. Es muss sich um einen Elementnamen ggf. mit Namespace-deklaration handeln.
Erlaeuterung	Optionale Erläuterung für die getroffenen Angaben, sodass die Entscheidung durch Dritte nachvollzogen werden können.

Tabelle: Übersicht über die definierten Steuerparameter für Packages mit Stereotype <<Application Schema>>

Name	Bedeutung
targetNamespace	XML-Namespace, z.B. " http://www.adv-online.de/namespaces/adv/gid/4.0 "
xmlns	XML-Namespace-Abkürzung, z.B. "adv"
Dateiname	Name der XSD-Datei, sofern vom Namen des Packages abweichend
ISO19100	"True", wenn die Inhalte des Packages Teil von ISO 19100 sind

10.1.3.5 Abbildungsregeln für Instanzen

Dieses Kapitel beschreibt die Abbildung des Instanzenmodells in entsprechende XML-Elemente. Das Ergebnis der Abbildung ist ein valides XML-Dokument (NAS-Dokument). Entsprechend gezippte XML-Dokumente sind ebenfalls gültige NAS-Dokumente. Als Komprimierungsverfahren zugelassen sind „zip“ und „gzip“.

Die Datei enthält:

- Den *XML-Header*, der fest ist: "<?xml version="1.0" encoding="UTF-8" ?>". Die Verwendung von "UTF-8" wird für das Encoding vorgeschrieben.
- Das *Root-Element* aus einer Auftrags- oder Ergebnis-XSD-Datei mit einem Verweis auf den AdV-Namespace "<http://www.adv-online.de/namespaces/adv/gid/version>" und die XSD-Datei.
- Elemente in Übereinstimmung mit der referenzierten XSD-Datei.

Jedes Objekt im Instanzenschema wird in ein entsprechendes Element überführt. Das passende Element trägt denselben Namen wie die Klasse, zu der das Objekt gehört. Das "gml:id"-Attribut, das den Objektidentifikator trägt, wird gesetzt.

Jede Eigenschaft des Objekts, d.h. jedes Attribut und jede Rolle in einer Assoziation, wird entsprechend der in den Schema-Abbildungsregeln definierten Abbildung auf XML-Elemente abgebildet, i.d.R. in ein lokales Element mit dem Namen des Attributs oder der Rolle.

10.1.4 XML-Schemata für Normteile und das externe Basisschema

Im Folgenden sind die Verweise auf die vollständigen XML-Schema-Dateien für die benötigten Teile aus den Normen ISO 19100 und das Basisschema angegeben. Beide Dateien sind als eigenständige Dokumente verfügbar:

[ISO19100.xsd](#)

[AAA-Basisschema.xsd](#)

10.1.5 XML-Schema für AFIS-ALKIS-ATKIS

Das XML-Schema für das AFIS-ALKIS-ATKIS-Fachschemata befindet sich in folgender Datei:

[AAA-Fachschemata.xsd](#)

10.1.6 GML_Profil für die NAS

Als Bestandteil der NAS wird ein GML-Profil dokumentiert (gml3nas.xsd), das die GML-Elemente und -Typen auf den benötigten Umfang eingrenzt und in der aktuellen Version nicht benötigte Teile, wie die Topologie oder nicht unterstützte Objekteigenschaften, „ausblendet“.

Neben der Weglassung von fachlich nicht benötigten GML-Strukturen wurde auch eine Reihe von zusätzlichen Festlegungen zur Verwendung von GML in der NAS getroffen. Ziel ist die Beschränkung von Freiräumen der Codierung, sodass die Verarbeitung von NAS-Dokumenten erleichtert wird:

1. Bei GML-Objekten, die neben der Verwendung von normalen auch Array-Eigenschaften erlauben, wurde ein der Varianten, i.d.R. die Array-Eigenschaften, gestrichen.
2. Bei der Darstellung von GM_Curve in AFIS-, ALKIS- und ATKIS-Daten wird die Verwendung von gml:Curve mit genau einem gml:LineStringSegment vorgeschrieben, sofern die GM_Curve zwischen allen Stützpunkten linear interpoliert wird. (gml:LineString darf in diesen Fällen nicht verwendet werden und ist ausschließlich in Filterausdrücken erlaubt.)
3. Bei der Darstellung von GM_PolyhedralSurface in AFIS-, ALKIS- und ATKIS-Daten wird die Verwendung von gml:Surface mit genau einem

gml:PolygonPatch vorgeschrieben. (gml:Polygon darf in diesen Fällen nicht verwendet werden und ist ausschließlich in Filterausdrücken erlaubt.)

4. Da die Mehrzahl der unter Verwendung von GM_MultiSurface definierten Flächenobjekte (z.B. Flurstücke) aus lediglich einer einzigen Fläche bestehen, wird vorgeschrieben, dass bei einer einzigen Fläche stets gml:_Surface zu verwenden ist und die Verwendung von gml:MultiSurface nur bei mehreren getrennten Flächen erlaubt ist.
5. Bei der Darstellung von GM_Ring in AFIS-, ALKIS- und ATKIS-Daten wird die Verwendung von gml:Ring mit genau einer gml:Curve vorgeschrieben, sofern der GM_Ring durch eine einzige GM_Curve beschrieben wird. (gml:LinearRing darf in diesen Fällen nicht verwendet werden und ist ausschließlich in Filterausdrücken erlaubt.)
6. Die Verwendung von gml:coordinates wurde mit Blick auf die GML-Version 3.1 aus dem Profil entfernt und es muss stets gml:pos verwendet werden. In GML 3.1 wird gml:coordinates durch ein neues Element (gml:posList) „ersetzt“ und gml:coordinates nur noch übergangsweise unterstützt werden.
7. Derzeit wird gml:metaDataProperty nicht im GML-Anwendungsschema benutzt, um die objektbezogenen Metadaten abzulegen (wegen unangemessener Anforderungen an die Benutzung in GML 3.0), daher wurde diese Eigenschaft nicht in das GML-Profil der NAS übernommen (und in GML 3.2 auch durch einen anderen Mechanismus ersetzt).
8. Die Standardeigenschaften von GML-Objekten „gml:name“ und „gml:description“ dürfen nur in GML-Dictionaries verwendet werden, nicht in Eigenschaften von Objekten im Namespace der NAS.

In der **Anlage** zu diesem Dokument befindet sich eine vollständige Abbildung des Basisschemas in GML.

10.2 NAS-Operationen

Die NAS ist vom Grundsatz her zunächst für die Kommunikation nach "außen", d.h. für die Nutzer der AAA-Daten konzipiert. Darüber hinaus kann sie, je nach Implementierungskonzept, auch für die interne Kommunikation zwischen Erfassungs- bzw. Qualifizierungssystemen und Führungssystemen verwendet werden. In den folgenden Kapiteln sind die letztgenannten Funktionalitäten mit berücksichtigt. Eine Implementierung, die die interne Kommunikation mit systemspezifischen Funktionen ermöglicht, muss aus der

Palette der beschriebenen Operationen der NAS nur diejenigen bereitstellen, die für die Abgabe von Daten an Dritte relevant sind. Dazu gehören insbesondere die Abgabe von Benutzungsdaten und die Führung von Sekundärnachweisen. Im Zuge der Realisierung einer netzbasierten Geodateninfrastruktur kann es darüber hinaus notwendig werden, weitere Funktionen als NAS-Operationen zur Verfügung zu stellen.

Für die Verwendung in Fachinformationssystemen sind drei allgemeine Operationen zur Fortführung von Bestandsdaten, zur Abfrage von Auszügen dieser und zur generellen Auskunft über die Eigenschaften der Bestandsdatenhaltung spezifiziert.

10.2.1 Funktionsumfang

Die NAS soll verschiedene Operationen unterstützen. Folgender Bedarf wird z.Zt. gesehen:

- Einrichten und Fortführen von Primärnachweisen
- Anfordern von Ausgaben
 - Ausgabe von Benutzungsdaten (Auszüge)
 - Führen von Sekundärnachweisen (Erstausstattung und Fortführung)
- Sperren und Entsperrern von Objekten
- Reservieren (von Punktnummer u.a.)
- Übermittlung von Protokollinformationen (z.B. Verarbeitungsprotokolle, Fehlerprotokolle)
- Ermitteln der Eigenschaften einer Bestandsdatenhaltung

Zu jeder NAS-Operation gehören zwei XML-Schema-Definitionen, eine für den Aufruf der Operation (*Request*) und eine für das Ergebnis (*Response*):

1. Aufruf (*Request*)
z.B. Fortführungsauftrag, Benutzungsauftrag
2. Ergebnis (*Response*)
z.B. Fortführungsprotokoll, Benutzungsergebnis

Hierbei ist durchaus die Mehrfachverwendung einer XML-Schema-Definition für mehrere Operationen möglich. Soweit standardisierte XML-Schemata für die genannten Zwecke vorliegen, werden diese verwendet, im anderen Fall werden die Definitionen selbst erstellt. Die XML-Schema-Definitionen für NAS-Operationen werden, wie alle anderen Inhalte der NAS, automatisch aus UML-Modellen abgeleitet werden. Für die Anwendungsbereiche AFIS und ALKIS wurden die UML-Modelle dazu bereits erstellt (s. Abschnitte 6.2 und

7.4). Sollte es sich herausstellen, dass die dort vorgenommenen Definitionen auch für andere Anwendungen verwendet werden sollen, so sind jene an dieser Stelle aufzunehmen.

Alle XML-Schemata für die NAS-Operationen sind in der Datei [NAS-Operationen.xsd](#) zusammengefasst. Die allgemein verwendbaren Basisoperationen sind in AAA-Basschema.xsd enthalten.

Den Operationen liegt die OWS Common Implementation Specification 1.0 zugrunde, die einzuhalten ist. Insbesondere ist von jeder NAS-Implementierung die GetCapabilities-Operation zu unterstützen.

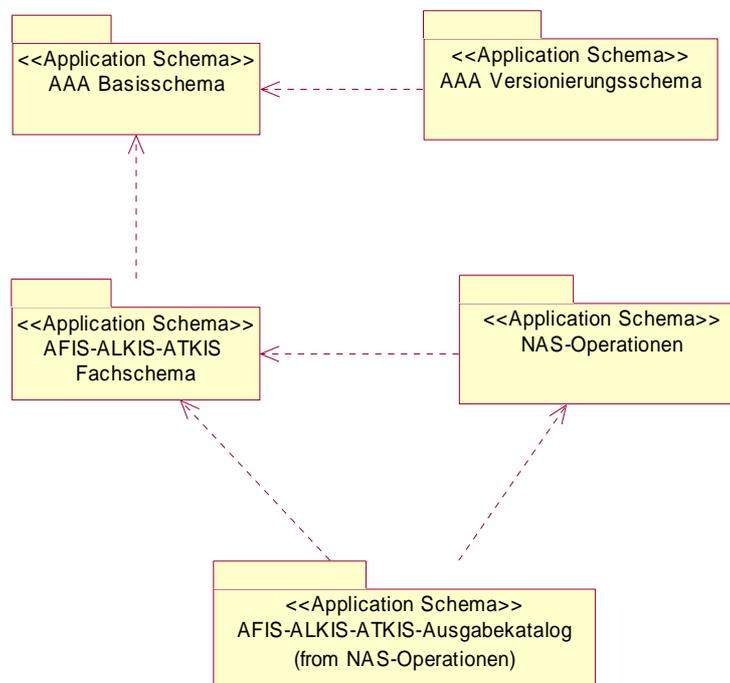


Abbildung 10-6: Das UML-Paket "NAS-Operationen" im Kontext der Bestandteile des Anwendungsschemas

10.2.2 Codierung der Selektionskriterien

Zur Codierung von Selektionskriterien wird in der NAS das `<wfs:Query>`-Element aus der Spezifikation "Web Feature Service, Version 1.0.0" in Verbindung mit den Festlegungen der Spezifikation "Filter Encoding, Version 1.0.0" des Open Geospatial Consortiums verwendet.

Die Anforderungen an die Selektions- bzw. Filterfunktionalität von AFIS-ALKIS-ATKIS gehen über die z.Zt. in den o.g. Spezifikationen beschriebenen Funktionalitäten hinaus. Deshalb werden zusätzlich die im Folgenden erläuterten Erweiterungen festgeschrieben, die derzeit in dieser Form **kein** Bestandteil der oben genannten Spezifikationen sind². Die fachlichen Anforderungen sind noch nicht abschließend überprüft; ggf. werden durch diese Überprüfung weitere Ergänzungen erforderlich. Folgende Erweiterungen werden z.Zt. für die NAS definiert:

- In GML können Assoziationen standardmäßig entweder über das Verschachteln des referenzierten Objekts oder über "Xlink:href"-Verweise oder ausgedrückt werden. Beide Darstellungen sind hierbei semantisch grundsätzlich vollkommen äquivalent³ :

Darstellung 1:

```
<AX_Flurstueck>
  <istGebucht>
    <AX_Buchungsstelle gml:id="DEXXXX00000001">
      <zu>
        <AX_Buchungsstelle gml:id="DEXXXX00000002">
          <laufendeNummer>1</laufendeNummer>
        </AX_Buchungsstelle>
      </zu>
    </AX_Buchungsstelle>
  </istGebucht>
</AX_Flurstueck>
```

Darstellung 2:

```
<AX_Flurstueck>
  <istGebucht xlink:href="#DEXXXX00000001"/>
</AX_Flurstueck>
<AX_Buchungsstelle gml:id="DEXXXX00000001">
  <zu xlink:href="#DEXXXX00000001"/>
</AX_Buchungsstelle>
<AX_Buchungsstelle gml:id="DEXXXX00000002">
  <laufendeNummer>1</laufendeNummer>
</AX_Buchungsstelle>
```

Für die erste Darstellung ist eine explizite Verfolgung der Objekt-Assoziationen durch den „/“-Operator von *Xpath* in einem Web Feature Service bereits explizit erlaubt. Da diese Darstellungen semantisch equivalent sind, wird für die NAS explizit erlaubt, den „/“-Operator auch auf `Xlink:href`-Verweise wirken zu lassen, wobei hier bis auf weiteres nur lokal auflösbare `Xlink:href`-Verweise unterstützt werden müssen. Das bedeutet, dass z.B. eine Abfrage über die Flurstücke, deren Buchungsstelle über die „zu“-Relation mit einer anderen Buchungsstelle mit der laufenden Nummer „1“ verbunden ist, wie folgt formuliert werden kann:

```
<wfs:Query typeName="AX_Flurstueck">
```

² Vgl. auch die entsprechenden Änderungsanträge (OGC-Dokumente 03-106, 05-039 und 05-040), die Dokumente sind derzeit nur OGC-Mitgliedern zugänglich

³ Zur einfacheren Interpretierbarkeit der NAS-Dateien ist die Verwendung der 2. Darstellung in der NAS explizit vorgeschrieben.

```

<ogc:Filter>
  <ogc:PropertyIsEqualTo>
    <ogc:PropertyName>
      istGebucht/AX_Buchungsstelle/zu/AX_Buchungsstelle/laufendeNummer
    </ogc:PropertyName>
    <ogc:Literal>1</ogc:Literal>
  </ogc:PropertyIsEqualTo>
</ogc:Filter>
</wfs:Query>

```

- Ein bekanntes Problem beim Filter Encoding ist, dass keine angemessene Unterstützung für Prädikate auf multiplen Eigenschaften besteht. Beispiel:

```

<wfs:Query typeName="AX_Gebaeude">
  <ogc:Filter>
    <ogc:PropertyIsEqualTo>
      <ogc:PropertyName>weitereGebaeudefunktion</ogc:PropertyName>
      <ogc:Literal>1100</ogc:Literal>
    </ogc:PropertyIsEqualTo>
  </ogc:Filter>
</wfs:Query>

```

Hier ist a priori nicht klar, welche Gebäude gefunden werden (sollen): Müssen zum Beispiel alle weitereGebaeudefunktion-Attribute diesen Wert besitzen oder muss mindestens ein solches Attribut gesetzt sein?

Für die Anwendung in der NAS wird bis auf weiteres explizit vereinbart, dass entsprechende Prädikate im Sinne von „mindestens ein Attributwert erfüllt die Bedingung“ anzuwenden ist.

- Verwendung von `<wfs:XlinkPropertyName>`⁴. Die [WFS-Erweiterungen.xsd](#) wird um folgende Definitionen ergänzt:

```

<xsd:complexType name="XlinkPropertyNameType">
  <xsd:complexContent mixed="true">
    <xsd:extension base="ogc:PropertyNameType">
      <xsd:attribute name="traverseXlinkDepth" type="xsd:string" use="required">
        <xsd:annotation>
          <xsd:documentation>
            Dieses Attribut gibt an, in welcher Tiefe Xlink:href-Verweise verfolgt und
            aufgelöst werden sollen. Ein Wert von "1" führt dazu, dass ein href-Verweis
            (auf ein lokales Objekt, bei entfernt liegenden Objekten muss die Auflösung
            nicht unterstützt werden) aufgelöst wird und das Zielobjekt mit in der
            Ergebnismenge zurückgeliefert wird. Hhref-Verweise aus diesem Zielobjekt
            werden wiederum nicht aufgelöst, da diese der Tiefe 2 zugerechnet werden.
            Ein Wert von "*" gibt an, dass alle (lokalen) href-Verweise aufgelöst werden
            sollen. Die erlaubten Werte sind positive, ganze Zahlen sowie der "*".
            Auch wenn ein Objekt durch mehrfache Verweise mehrfach aufgelöst wird,
            ist es nur ein einziges Mal in der Ergebnismenge repräsentiert.
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="XlinkPropertyName" type="wfs:XlinkPropertyNameType"
  substitutionGroup="ogc:PropertyName">
  <xsd:annotation>
    <xsd:documentation>

```

⁴ Vgl. OGC-Dokument 02-063, das Dokument ist derzeit nur OGC-Mitgliedern zugänglich.

```

    Dieses Element darf in der NAS ein ogc:PropertyName Element ersetzen.
  </xsd:documentation>
</xsd:annotation>
</xsd:element>

```

Die Verwendung ist innerhalb der Queries im Benutzungsauftrag sowie in den Nutzerprofilen erlaubt. Hierbei werden insbesondere die folgenden Regelungen festgehalten:

- Sofern das Anwendungsschema (wie im Fall der NAS) fordert, dass die Objekt-Assoziationen nicht inline, sondern stets über Xlink-Verweise angebunden sind, führt ein Xlink-Traversal dazu, dass das referenzierte Objekt in der Ergebnismenge (Feature Collection) enthalten sind.
- Die Auflösung von href-Verweisen unterstützt im Fall der NAS explizit die URN-Identifikatoren des AAA-Modells.
- Eine Auflösung von href-Verweisen erfolgt bis auf weiteres nur für lokal verfügbare Ressourcen. Eine Unterstützung für Remote-Xlink-Auflösungen wird bei Bedarf zu ggf. zu einem späteren Zeitpunkt ergänzt.
- Verwendung von `<wfs:XlinkPropertyPath>`. Die [WFS-Erweiterungen.xsd](#) wird um folgende Definitionen ergänzt:

```

<xs:complexType name="XlinkPropertyPathType" mixed="true">
  <xs:complexContent mixed="true">
    <xs:extension base="ogc:PropertyNameType">
      <xs:attribute name="leafOnly" type="xs:boolean" default="false">
        <xs:annotation>
          <xs:documentation>Das Attribute regelt, ob alle Objekte
längs des Pfades oder nur das Ziel des Pfades selektiert werden. Default ist
das bis zur GeoInfoDok 4.0 benutzte Standardverhalten (alle Objekte entlang
des Pfades).</xs:documentation>
        </xs:annotation>
      </xs:attribute>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:element name="XlinkPropertyPath" type="adv:XlinkPropertyPathType"
substitutionGroup="ogc:PropertyName">
  <xs:annotation>
    <xs:documentation>
      Dieses Element darf in der NAS ein ogc:PropertyName Element ersetzen,
      allerdings nie in einem ogc:Fliter Element, nur als direktes Kindelement
      eines wfs:Query Elements.
      Es erlaubt (im Unterschied zu wfs:XlinkPropertyName, das die generelle
      Expansion bis zu einer bestimmten Tiefe unterstützt) die gezielte Auflösung
      von Xlink:href-Verweisen entlang eines bestimmten „Eigenschaftspfades“.
      Als Wert wird eine Xpath-Pfadangabe verwendet, bei der am Ende ein
      Objekt steht, bei dem die Auflösung abbricht.
      Beispiel: Ein XlinkPropertyPath
      "istGebucht/AX_Buchungsstelle/istBestandteilVon/AX_Buchungsblatt"
      bei einer Query auf AX_Flurstueck führt dazu, dass die Buchungsstelle und das
      Buchungsblatt in der Ergebnismenge für jedes selektierte Flurstück direkt
      mit zurückgeliefert werden.
    </xs:documentation>
  </xs:annotation>
</xs:element>

```

- Verwendung von `<ogc:PropertyIsOfType>` zur Ermittlung des Typs einer Objekteigenschaft. Bei Eigenschaften mit `complexContent` ist dies der qualifizierte

Elementname des Kindelements, bei Eigenschaften mit simpleContent der qualifizierte Typname des Eigenschaftselements.

Die [WFS-Erweiterungen.xsd](#) wird hierfür um folgende Definitionen ergänzt:

```
<xsd:element name="PropertyIsOfType" type="ogc:BinaryComparisonOpType"
  substitutionGroup="ogc:comparisonOps">
  <xsd:annotation>
    <xsd:documentation>Der Vergleichsoperator ermittelt den Typ der im ersten
Operanden angegebenen Eigenschaft (ogc:PropertyName) und vergleicht ihn mit der Angabe
im zweiten Operanden (ogc:Literal). Beide Angaben müssen identisch
sein.</xsd:documentation>
  </xsd:annotation>
</xsd:element>
```

10.2.3 Einrichten und Fortführen von Primärnachweisen

Da GML selbst keine Elemente für Fortführungsoperationen anbietet, werden zu diesem Zweck die Festlegungen des *Web Feature Service* (WFS) von OGC verwendet. In der WFS-Spezifikation sind neben dem Transaktionsmechanismus 3 Änderungsfunktionen definiert: <Insert> (neues Objekt einfügen), <Replace> (Objekt ändern, überschreiben) und <Delete> (Objekt löschen). Welche Änderungen zu <Replace> oder zu <Delete> mit anschließendem <Insert> führen, ist in den Objektartenkatalogen fachlich festzulegen. Dies gilt sowohl für Änderungen in den Attributwerten und Relationen als auch für geometrische Änderungen. Bei letzteren muss ggf. der Bearbeiter im Erhebungsprozess entscheiden, welche Fortführungsart zu verwenden ist.

Beispiel:

```

<wfs:Transaction>
  <wfs:Insert>
    <AX_Flurstueck gml:id="DEBY0000F0000001">
      ...
    </AX_Flurstueck>
    <AX_Gebaeude gml:id="DEBY0000G0000001">
      ...
    </AX_Gebaeude>
  </wfs:Insert>

  <adv:Replace typeName="AX_Flurstueck">
    ...
    <Name>amtlicheFlaeche</wfs:Name>
    <Value>287.3</wfs:Value>
    ...
    <ogc:Filter>
      <ogc:FeatureId fid="DEBY0000F000000220010101T000000Z"/>
    </ogc:Filter>
  </adv:Replace>

  <wfs>Delete typeName="AX_Buchungsstelle">
    <ogc:Filter>
      <ogc:FeatureId fid="DEBY0000B000000320010101T000000Z"/>
      <ogc:FeatureId fid="DEBY0000B000000420010101T000000Z"/>
    </ogc:Filter>
  </wfs>Delete>
</wfs:Transaction>

```

Folgendes ist zu beachten:

- Die Filter-Ausdrücke bei <Delete>- und <Replace>-Operationen dürfen nur *FeatureId*-Elemente beinhalten. Komplexere Filterkriterien sind nicht erlaubt.
- Bei *FeatureId*-Bedingungen in "<Replace>"- und "<Delete>"-Operationen wird der Identifikator zusätzlich um Entstehungsdatum und -uhrzeit ergänzt, damit eine Prüfung auf Aktualität erfolgen kann. Datum und Uhrzeit werden ohne Trennzeichen kodiert, damit sie den Bedingungen eines gml:id genügen, also in der folgenden Form: CCYYMMDDThhmmssZ. Treten mehrere Versionen ein- und desselben Objektes in einem einzigen Einrichtungsauftrag auf, so wird der Objektidentifikator ebenso um Entstehungsdatum und -uhrzeit ohne Trennzeichen ergänzt, damit auftragsbezogen keine ID mehrfach vorkommt. Bezüglich der Ergänzung um das Entstehungsdatum gilt folgende Ausnahme: Innerhalb von Fortführungsaufträgen mit mehreren Fortführungsfällen kann bei <Replace> bzw. <Delete>-Anweisungen von mehrfach fortzuführenden Objekten ab dem 2. Fortführungsfall das Entstehungsdatum innerhalb der OID entfallen. Nur in diesem speziellen Fall erfolgt keine Prüfung der Aktualität und es wird die -soeben erzeugte- aktuelle Version genutzt.
- In <Delete>-Operationen dürfen nur mehrere AAA-Objekte behandelt werden, wenn sie dieselbe Objektart haben; innerhalb von <Insert>-Operationen können unterschied-

liche Objektarten vorkommen. <Replace>-Operationen behandeln immer nur ein Objekt.

- Bei <Replace>-Operationen sind stets alle Eigenschaften des AAA-Objekts als „Properties“ zu übergeben, also auch die unveränderten. Dies stellt eine Verschärfung der WFS-Spezifikation von OGC für die <Update>-Operation dar, in der gefordert wird, dass mindestens alle geänderten Eigenschaften übermittelt werden. Grund für diese Verschärfung war die Forderung, dass Datenhaltungskomponenten sich nicht merken müssen, welche Eigenschaften eines Objekts geändert wurden, sondern lediglich die Tatsache, dass ein Objekt geändert wurde.
- Alle durchgeführten Änderungen innerhalb eines Fortführungsfalls werden zeitgleich gültig. In das Attribut "lebenszeitintervall" der Objekte wird die Systemzeit (umgerechnet in UTC) zum Beginn der Transaktion eingetragen. Dabei ist fallweise das Beginn- oder das Endedatum zu belegen. Die angelieferten Angaben bei den einzelnen Fachobjekten sind unerheblich und werden überschrieben. Letzteres gilt nicht für die Ersteinrichtung eines Datenbestandes durch Übernahme von Objekten aus einem Vorgängerdatenbestand (Einrichtungsauftrag). Um hierbei ein Eintragen von historischen Informationen zu ermöglichen, wird dort das angelieferte Datum übernommen. Wird als Datum/Zeit "9999-01-01T00:00:00Z" (Dummy-Datum) angeliefert, so wird dies wie bei Fortführungsaufträgen mit der Systemzeit überschrieben. Zeitangaben werden immer in UTC-Zeit (Universal Time Coordinated, Greenwich Mean Time) gemacht. Die Zeiteinheit für die Einträge ins Lebenszeitintervall (Datentyp: *DateTime*) ist die volle Sekunde einschließlich der obligatorischen Kennung „Z“ für UTC (CCYY-MM-DDTHH:MM:SSZ). Das Führungssystem stellt bei der Übernahme sicher, dass nicht 2 Versionen desselben Objekts mit identischem Lebenszeitintervall entstehen. Dies kann dann auftreten, wenn ein Objekt innerhalb eines Fortführungsauftrags in mehreren Fortführungsfällen verändert wird und diese aufgrund der Systemgeschwindigkeit in der gleichen Sekunde abgearbeitet werden.

Die XML-Schemata für einen Fortführungsauftrag und sein Ergebnis sind wie alle anderen NAS-Operationen in der Datei [NAS-Operationen.xsd](#) enthalten. Einrichtungsaufträge und deren Ergebnisse sind Unterklassen von Fortführungsaufträgen.

Die Einrichtung und Fortführung von Metadatenbeständen ist noch nicht Bestandteil der Modellierung bzw. der Festlegungen. Sie wird nach Vorliegen der fachlichen Anforderungen ergänzt.

Die Funktionen zur Fortführung werden in Systemen mit vollständigem Nachweis der Historie und in Systemen ohne vollständige Historie unterschiedlich ausgeführt:

Systeme ohne vollständigen Historiennachweis

<Insert>:

Die übermittelten Fachobjekte werden als neue Informationen eingetragen.

<Replace>:

Die übermittelten Fachobjekte ersetzen die Fachobjekte, die denselben Identifikator haben. Zur eindeutigen Bezeichnung der zu überschreibenden bzw. zu versionierenden Version wird der Identifikator (XML - Attribut *fid*) des neuen Fachobjekts im Filterausdruck um die Angabe des Entstehungsdatums/zeit der zu überschreibenden Objektversion ergänzt. Damit sollen Fehler aufgedeckt werden, die durch Fortführungsaufträge entstehen könnten, die nicht zum gespeicherten Datenbestand passen. Im aufnehmenden System wird das Fachobjekt wieder mit dem originalen (nicht um Entstehungsdatum/zeit ergänzten) Identifikator gespeichert. Es ist nicht zulässig, die Operation <Replace> durch <Delete> und nachfolgendes <Insert> mit dem selben Identifikator zu ersetzen.

<Delete>:

Das Attribut *fid* des Filterausdrucks im WFS-<Delete>-Element bezeichnet das zu löschende Fachobjekt. Zur eindeutigen Bezeichnung der zu löschenden Version wird der Identifikator in der Austauschdatei um die Angabe des Entstehungsdatums/zeit der zu löschenden Version ergänzt. Damit sollen Fehler aufgedeckt werden, die durch Fortführungsaufträge entstehen könnten, die nicht zum gespeicherten Datenbestand passen. Das so bezeichnete Objekt wird im aufnehmenden System mit allen selbstbezogenen Eigenschaften und referenzierten Raumbezugsgrundformen gelöscht. Raumbezugsgrundformen werden nur dann gelöscht, wenn sie von keinem weiteren Objekt referenziert werden.

Diese Funktionalität wird vor allem von Datenhaltungssystemen genutzt, die Sekundärdatenbestände halten.

Systeme mit vollständigem Historiennachweis

Ist das aufnehmende System zur Führung eines vollständigen Historiennachweises konfiguriert, reagiert es auf

<Insert>

mit der Erzeugung einer neuen Instanz eines Objektbehälters und fügt in den Behälter eine erste Version des übermittelten Fachobjekts ein.

<Replace>

Die übermittelten Fachobjekte werden als neue Version in den durch den Identifikator

bezeichneten Objektbehälter eingetragen. Zur eindeutigen Bezeichnung der Vorgängerversion wird der Identifikator im Filterausdruck (XML - Attribut *fid*) des neuen Fachobjekts in der Austauschdatei um die Angabe des Entstehungsdatums/zeit der zu überschreibenden Objektversion ergänzt. Damit sollen Fehler aufgedeckt werden, die durch Fortführungsaufträge entstehen könnten, die nicht zum gespeicherten Datenbestand passen. Im aufnehmenden System bleibt das überschriebene Fachobjekt als historische Version bestehen.

<Delete>

Die im Filterausdruck durch den um Entstehungsdatum/zeit erweiterten Identifikator (XML-Attribut *fid*) bezeichnete Version des Fachobjekts wird mit dem aktuellen Untergangsdatum/zeit (aus der Systemzeit abgeleitet) versehen und dadurch historisiert. Das System stellt sicher, dass keine weiteren Versionen angelegt werden können.

Diese Funktionalität wird auch von den Datenhaltungssystemen genutzt, die die Versionierung nur befristet zur Bereitstellung von Fortführungsinformationen für Dritte im Rahmen des NBA-Verfahrens (s.u.) verwenden.

Das konzeptuelle Fachmodell für die Fortführung von ALKIS sowie die Abläufe bei dessen Fortführungsverarbeitung sind (vorläufig) in Abschnitt 5.3 beschrieben.

10.2.4 Anfordern von Ausgaben

Die aus einem Datenhaltungssystem auszugebenden Daten (Benutzungsdaten oder Daten zur Führung von Sekundärnachweisen) werden hinsichtlich des auszugebenden Informationsumfangs durch Kriterien zur Selektion und Filterung bestimmt. Ein Datenhaltungssystem muss deshalb in der Lage sein, komplexe Selektions - und Filterausdrücke auszuwerten und die sich damit qualifizierenden Daten auszugeben. Die **Selektion** erfolgt durch räumliche, fachliche (Objektart, Attribut, Relation) und zeitliche Kriterien. Diese Kriterien können auch geschachtelt und miteinander verbunden werden, so dass ganze Selektionsketten entstehen. Damit kann auch formuliert werden, welche Elemente weitere Elemente über Referenzen zur Ausgabe nachziehen. In der NAS wird stets nur die im Anwendungsschema als navigierbare Richtung gekennzeichnete Rolle der Assoziation angegeben. Gegenrelationen sind in der NAS nicht zulässig. Gleichwohl ist es zur Vereinfachung von Filterausdrücken möglich, Objekte anzufordern, die per Gegenrelation mit einem Objekt verbunden sind. Hierbei ist -soweit vorhanden- der explizit im Modell benannte Rollenname der Gegenrelation zu verwenden, in Ermangelung eines solchen der um "inversZu_" ergänzte Rollenname der als navigierbare Richtung gekennzeichneten Rolle.

Kriterien der **Filterung** bestimmen, welche Elemente der Selektionskette ausgegeben werden sollen und welche Attribute und Verweise mit diesen Elementen ausgegeben werden.

Die Selektions- und Filterungskriterien werden als Bestandteil der Benutzungsanforderung an das datenführende System übermittelt oder dort in Benutzerprofilen hinterlegt. Für die Definition einheitlicher Produkte der AdV werden einheitliche Selektions - und Filterkriterien definiert (z.B. in Kap 7.2).

Als formale Sprache zur Definition der Selektionsketten wird die *Filter Encoding Specification* von OGC verwendet.

Das XML-Schema für einen Benutzungsauftrag ist in der Datei [NAS-Operationen.xsd](#) enthalten. Es nutzt das Schema [filter.xsd](#) von OGC.

10.2.5 Ausgabe von Benutzungsdaten

Die Ausgabe von Benutzungsdaten ist eine Datenausgabe ohne explizite Angabe einer im aufnehmenden System auszuführenden Funktionalität. Eine spezielle Aufbereitung der Daten in Abhängigkeit von der Ausgabeanforderung (z.B. Herstellung der "flurstückszentrierten Sicht" in ALKIS) ist möglich, indem entsprechende temporäre Objekte ausgegeben werden (s. Abschnitt 7.2).

Für das Ergebnis einer Benutzung wird die FeatureCollection (WFS-basic.xsd) aus dem *Web Feature Service* von OGC verwendet und für AAA entsprechend um weitere Informationen ergänzt. Für jede Art der Ausgabe wird je nach Benutzungsauftrag eine eigene Schema-Datei verwendet.

10.2.6 Führung von Sekundärnachweisen

Die Führung von Sekundärnachweisen erfolgt über die Nutzerbezogene Bestandsdatenaktualisierung (Abschnitt 10.5) fortführungsfallbezogen (kontinuierlich) oder stichtagsbezogen (durch Zusammenfassung mehrerer Fortführungen am selben Objekt zu einer einzigen Fortführung). Die Operationen *<Insert>*, *<Replace>* und *<Delete>* werden sinngemäß wie bei der Führung von Primärnachweisen durch das aufnehmende System ausgeführt.

Für die [Schema-Datei](#) gilt das bei den Ausgaben (Punkt 10.2.4) Gesagte.

10.2.7 Sperren und Entsperrern von Objekten

Sperraufträge ermöglichen das Sperren von Objekten im Führungssystem gegen Fortführungen von Dritten durch Angabe einer Liste mit Objekt-Identifikatoren. Entsperraufträge heben die Sperrung wieder auf.

10.2.8 Reservierungen

Zur Reservierung von Kennungen (z.B. für Vermessungspunkte, Flurstückskennzeichen, etc.) können entsprechende Aufträge an ein Führungssystem formuliert werden. In der Ergebnis-Datei erhält man eine Liste mit den angeforderten Kennungen.

10.2.9 Übermittlung von Protokollinformationen

Da für jede Operation der NAS sowohl eine *Request*- als auch eine *Response*-Klasse definiert wurde, wird in letzterer definiert, welche Protokollinformation bei der jeweiligen Operation ausgegeben werden. Sie sind somit in den bei den einzelnen Operationen enthalten.

10.2.10 Ermitteln der Eigenschaften einer Bestandsdatenhaltung

Jede Software-Komponente, die eine NAS-Schnittstelle besitzt, muss die *GetCapabilities*-Operation unterstützen.

10.3 Auszutauschende Einheiten

Die kleinsten Einheiten des Datenaustauschs sind vollständige Fachobjekte. Dies gilt grundsätzlich auch für die Fortführung des Primärnachweises (AAA-Führungssystem). Unabhängig davon, ob sich Objekte durch eigene Eigenschaften zur Ausgabe qualifiziert haben oder über die Auswertung einer vorgegebenen Selektionskette zur Ausgabe qualifiziert wurden, sind sie hinsichtlich der Fortführungsfunktionalität grundsätzlich als eigene fortzuführende Einheit zu betrachten (Ausnahmen siehe Abschnitt 10.4).

Benutzungen, die nicht dem Zweck der Fortführung des Primärnachweises dienen, können, je nach Nutzerwunsch oder Nutzerprofil, unvollständige Fachobjekte (fehlende Attribute oder Relationen) oder durch spezielle Aufbereitung der Daten entstandene "temporäre Objekte" für den Datenaustausch erzeugen.

Der Datenaustausch erfolgt in der NAS unabhängig vom konzeptuellen Modell der Versionierung (Behälter mit Versionen) so, als ob alle Objektversionen unabhängige

Objekte wären. Auf diese Art und Weise ist es möglich, die Austauschchnittstelle für Stellen, die eine vollständige Historie führen und solche, die dies nicht tun, identisch zu definieren. Dabei sind jedoch folgende Rahmenbedingungen zu beachten:

- Damit die Zahl der entstehenden Versionen reduziert wird, müssen zweiseitige Relationen im Datenaustausch durch eine einzige einseitige Relation dargestellt werden. Es wird diejenige Relation im Datenaustausch codiert, die im UML - Schema als bevorzugte Navigationsrichtung definiert wurde. Zweiseitige Relationen in genormten Schemata werden mittels geeigneter Parametrisierung durch einseitige ersetzt.
- Um beim Datenaustausch die zu löschende oder zu überschreibende bzw. zu versionierende Version eines Objekts eindeutig identifizieren zu können, wird in der Austauschdatei der Identifikator in den XML-`<Delete>`- und `<Replace>`- Elementen um Entstehungsdatum und Entstehungszeit ergänzt. Die Ergänzung des Identifikators um den Zeitstempel ist nur im Datenaustausch erforderlich, um sicherzustellen, dass sich Fortführungen auch auf den aktuellen Datenbestand beziehen. Im Datenbestand selbst werden die zu referenzierenden Versionen durch Auswertung des Lebenszeitintervalls der Versionen auf attributiver Ebene gewonnen.

10.4 Implizite Funktionalität

Bei der Führung von Primär- und Sekundärnachweisen über die Schnittstelle NAS ist es erforderlich, dass das aufnehmende System neben der Ausführung der expliziten Funktionen `<Insert>`, `<Delete>` und `<Replace>` (siehe 10.2.1.1) auch über implizite Funktionen verfügt, die erst die komfortable Arbeitsweise mit dem System erlauben.

Der Umfang der zu realisierenden impliziten Funktionalität eines Datenhaltungssystems ist für ein System zum Primärnachweis und zum Sekundärnachweis unterschiedlich groß. Die von einem Sekundärnachweissystem beim Datennutzer zu fordernden Funktionen sollten grundsätzlich möglichst gering sein, damit eine einfache Implementierung ermöglicht wird. Demgegenüber kann ein Datenhaltungssystem für den Primärnachweis bei der originär zuständigen datenführenden Stelle über wesentlich mehr Funktionen verfügen.

10.4.1 Implizite Funktionalität eines Systems für den Primärnachweis

Bei der Nutzung der NAS für die Kommunikation zwischen einem Qualifizierungs- bzw. Erfassungssystem und einem Führungssystem sind folgende implizite Funktionen notwendig:

- Das aufnehmende System leitet beim Eintrag neuer Versionen das **Entstehungsdatum/zeit** aus der Systemzeit ab. Alle in einem Fortführungsfall eingetragenen (oder durch die Funktion *<Replace>* entstandenen) neuen Versionen erhalten dasselbe Entstehungsdatum/zeit. Dies ist i.d.R. die Zeit, an der die Transaktion begonnen wird (*commit*). Besteht ein Auftrag aus Teilaufträgen (Fortführungsfällen), werden diese in der Reihenfolge ihres Auftretens in der NAS-Datei abgearbeitet. Für jeden Teilauftrag wird ein eigenes Entstehungsdatum/-zeit vergeben.
- Referenzen werden beim Datenaustausch über die NAS nur einseitig in der bevorzugten Richtung der Referenz ausgetauscht. Das aufnehmende System baut die **Gegenreferenz** implizit auf. Durch den Aufbau der Gegenreferenz entsteht keine neue Version.
- Es gibt Fachobjekte, die nur dann eine Existenzberechtigung haben, wenn sie von anderen Objekten referenziert werden (z.B. Objekte vom Typ Lage). Weil Gegenreferenzen nicht über die NAS übermittelt werden, kann ein fortführendes System dann nicht wissen, ob ein Objekt, das durch die Fortführung nicht mehr referenziert wird auch gelöscht werden kann. Das **nicht mehr referenzierte Fachobjekt** muss durch die Datenbank **gelöscht** werden. Die Fachobjekte, die wegen fehlender Referenzierung gelöscht werden können, sind im Objektartenkatalog zu bezeichnen. Dieser Fortführungsfall findet Eingang in die Versionierung und Historisierung.
- Es gibt Fachobjekte, die Objekte referenzieren, die im Rahmen der Fortführung gelöscht werden sollen. Weil Gegenreferenzen nicht über die NAS übermittelt werden, kann ein fortführendes System nicht wissen, ob ein zu löschendes Objekt durch weitere Objekte referenziert wird. Dadurch kann es vorkommen, dass Referenzen nach der Fortführung nicht mehr befriedigt werden. Das Datenhaltungssystem muss solche **unbefriedigten Referenzen automatisch löschen**. Dieser Fortführungsfall findet Eingang in die Versionierung und Historisierung.
- Es gibt Fachobjekte, die nur dann eine Existenzberechtigung haben, wenn sie andere Fachobjekte referenzieren (z.B. Präsentationsobjekte). Werden im Rahmen einer Fortführung alle solchen Referenzen explizit oder implizit gelöscht, so **löscht** das Datenhaltungssystem automatisch das entsprechende **Fachobjekt, dem die notwendigen Referenzen fehlen**. Die Fachobjekte, die wegen fehlender notwendiger Referenzen gelöscht werden müssen, sind im Objektartenkatalog zu bezeichnen. Dieser Fortführungsfall findet Eingang in die Versionierung und Historisierung.

- Werden im Zuge einer Fortführung nur die fachlich geänderten Objekte angeliefert, muss die Datenbank ggf. die topologische und geometrische Konsistenz selbstständig herstellen (Geometriebehandlung, s. Abschnitt 10.4.1.1).
- Beim **Löschen von Geometrien** sind ggf. Zerschlagungen aus vorherigen Implizitprozessen nach folgender Regel wieder rückgängig zu machen. Eine Position wird aus der Geometrie aller Objekte entfernt, wenn sie in keinem Objekt, in dem sie verwendet wird zur geometrischen Definition dieses Objektes beiträgt; trägt sie auch nur in einem Objekt zur geometrischen Definition bei, bleibt sie in allen Objekten erhalten. Eine Position trägt dann zur geometrischen Definition eines Objekts bei, wenn das Objekt punktförmigen Raumbezug hat, oder wenn sie (bei linienhaftem oder flächenhaftem Raumbezug) nicht in einer Geraden mit der vorhergehenden und der folgenden Position liegt. Der Begriff "liegt in der Geraden" ist dabei in Abhängigkeit von der festgelegten Koordinatenauflösung (für metrische Lagekoordinaten in AFIS-ALKIS-ATKIS: Millimeter) zu definieren. Dieses Implizitverhalten führt im aufnehmenden System zu Fortführungen, die im auslösenden Fortführungsauftrag aus der NAS nicht explizit angegeben sind. Diese Fortführungen sind durch das aufnehmende System implizit zu veranlassen und führen zur Erzeugung neuer Versionen aller beteiligten Objekte.
- Werden zur Fortführung eines Primärnachweises Austauschelemente mit vorläufigen Identifikatoren (s. 3.3.7) angeliefert, **erzeugt** das aufnehmende System endgültige **eindeutige Identifikatoren**.
- Bei den Stellen, die keine vollständige Historie führen, **erzeugt** die Datenhaltung beim Löschen eines aktuellen Flurstücks automatisch das entsprechende Objekt "Historisches Flurstück".
- Weitere implizite Funktionen (z.B. Vergabe von Punktkennzeichen) sind **implementierungsspezifisch**.

10.4.1.1 Geometriebehandlung

Geometriebehandlung stellt eine Funktionalität der Datenbank (AAA-Führungskomponente) im Rahmen der Fortführungsverarbeitung dar. Dabei werden neue bzw. geänderte Geometrien so mit dem Altbestand verknüpft, dass bei geometrischen Identitäten zwischen Alt- und Neubestand in Abhängigkeit von der Themenzugehörigkeit der beteiligten Objekte redundanzfreie Geometrien entstehen.

Diese Funktionalität ist unabhängig von den Geometrie behandelnden bzw. Identitäten herstellenden Funktionen des Verarbeitungssystems (AAA-Verarbeitungskomponente) immer dann notwendig, wenn vom Verarbeitungssystem im Rahmen einer Fortführung nicht alle von geometrischen Operationen betroffenen Objekte an die AAA-

Führungskomponente geliefert werden. (z.B. bei einer Flurstückszerlegung nur das gelöschte und die neuen Flurstücke). Für die Geometriebehandlung gelten folgende Grundsätze:

- Die Funktionalität der Geometriebehandlung kann von AAA-Führungssystemen optional realisiert werden. AAA-Verarbeitungskomponenten können ggf. von einer vorhandenen Geometriebehandlung in der AAA-Führungskomponente Gebrauch machen. Sofern in der AAA-Führungskomponente keine Geometriebehandlung realisiert ist, müssen die AAA-Verarbeitungskomponenten vollständige Daten anliefern. Unberührt davon ist die Verpflichtung der AAA-Führungskomponente zur Prüfung der Daten auf (geometrische) Konsistenz.
- Die durch Geometriebehandlung implizit veränderten Objekte werden versioniert.
- Die Geometriebehandlung beschränkt sich auf Klassen-Themen; eine Geometriebehandlung bei Instanzen-Themen wird auch zukünftig nicht vorgesehen. Insofern wirkt sich eine Geometriebehandlung bei Klassen-Themen auch nicht auf Instanzen-Themen aus (keine „kaskadierende“ Geometriebehandlung).
- Bei Instanzen-Themen hat die AAA-Verarbeitungskomponente dafür zu sorgen, dass a) bei gewollter Identität (redundanzfreie Geometrie) ggf. ein Aufsplitten der Linien erfolgt und b) alle betroffenen Objekte im Fortführungsauftrag mitgeliefert werden.
- AX_Fortführungsauftrag wird um einen Steuerparameter (Geometriebehandlung ja/nein) ergänzt. Die AAA-Führungskomponente muss diesen Schalter auswerten und entsprechend reagieren, d.h. entweder die Geometriebehandlung ein- bzw. ausschalten oder den Fortführungsauftrag ablehnen. Die AAA-Verarbeitungskomponente hat dafür zu sorgen, dass die Schalterstellung dem Inhalt des Fortführungsauftrags entspricht.
- Eine Geometriebehandlung bei aufnehmenden Systemen im NBA-Verfahren wird nicht vorgesehen/erwartet. Es werden alle veränderten Objekte übermittelt, auch die, die lediglich durch Geometriebehandlung in der AAA-Führungskomponente geändert wurden.

Es gelten folgende geometrische Kriterien:

- Das Such- bzw. Trennkriterium für die Geometriebehandlung beträgt Wurzel 2 [mm]
- An der Geometriebehandlung nehmen Punkte/Stützpunkte und Linien teil.
- Bei Linien nehmen nur Geraden und Kreisbögen/Vollkreise an der Geometriebehandlung teil. Splines nehmen nicht an der Geometriebehandlung teil; hier muss die Verarbeitungskomponente dafür sorgen, dass alle betroffenen Objekte fortgeführt werden.

- Auch wenn eine Neulinie eingetragen wird, muss ein "darunter liegender" Altpunkt die Neulinie splitten. Diese muss über den Altpunkt geführt werden.

10.4.2 Implizite Funktionalität eines Systems für den Sekundärnachweis

Bei der Führung von Sekundärnachweisen (s. Abschnitt 10.5) über die Schnittstelle NAS baut das aufnehmende System (soweit vom Nutzer gewünscht) die Gegenreferenzen zu den ausgetauschten Referenzen auf und pflegt sie.

Replace-Befehle, bei denen das fortzuführende Objekt noch nicht im Datenbestand des Nutzers ist, sind bei der Übernahme wie *Insert*-Befehle zu behandeln. (Beispiel: Ein Nutzer erhält im Interessengebiet alle Flurstücke und die zugehörigen Eigentümer. Ein Flurstück wechselt seinen Eigentümer. Der Eigentümer ist aus Sicht des Nutzers neu (*Insert*) aus Sicht des ALKIS-Führungssystems aber alt (*Replace*), weil er bereits an Flurstücken außerhalb des Interessengebiets Eigentum hatte und deshalb seit langem im abgebenden System geführt wird, jedoch noch nie im System des Nutzers geführt wurde.)

10.5 Nutzerbezogene Bestandsdatenaktualisierung (NBA)

10.5.1 Fachliche Anforderungen

Die fachlichen Anforderungen zur Nutzerbezogene Bestandsdatenaktualisierung (NBA) gründen sich auf die vorhandenen Verfahren, wie sie in ALK/ATKIS und im ALB realisiert vorliegen. Diese Verfahren sind nicht identisch. Weitere fachliche Anforderungen lassen sich folgendermaßen zusammenfassen:

Änderungsdaten sind auf der Grundlage der Fortführungsdaten abzuleiten, die ihrerseits die Struktur der Bestandsdaten aufweisen. Änderungsdaten zur Nutzerbezogenen Bestandsdatenaktualisierung sollen

1. kontinuierlich und fortführungsfallbezogen (Änderungsdaten) und/oder
2. stichtagsbezogen (Differenzdaten) abgegeben werden können.

Fortführungsfallbezogen bedeutet, dass alle Veränderungen, die in einem zurückliegenden Zeitraum stattgefunden haben, der zeitlichen Reihenfolge nach aufgeführt werden. Damit wird es möglich, alle Prozesse schrittweise im aufnehmenden System nachzuvollziehen. Voraussetzung ist allerdings, dass auch alle Informationen in den Änderungsdaten enthalten sind, die das Erzeugen, Ändern und Löschen von Objekten in dem zurückliegenden Zeitraum betreffen.

Im Gegensatz dazu liefert das stichtagsbezogene Verfahren nur die Differenzdaten, die nötig sind, um den Ausgangszustand beim Nutzer auf den vom Nutzer gewünschten Endzustand zu bringen. Was auf dem Weg zum Endzustand mit den Objekten geschehen ist, kann in diesem Fall nicht nachvollzogen werden. Die stichtagsbezogenen Differenzdaten stellen eine Untermenge der Änderungsdaten dar und können durch Auswertung aus ihnen abgeleitet werden; sie umfassen alle neu entstandenen Objekte, die jeweils aktuellsten Versionen von fortgeführten Objekten sowie Angaben zu historisch gewordenen Objekten.

Für jeden Nutzer wird ein Profil angelegt, das beschreibt, nach welchen Kriterien der Nutzer mit Änderungsdaten aus dem einmal für das NBA - Verfahren vorgehaltenen Bestand versorgt werden soll. Dieses Profil muss nicht bei der ersten Datenabgabe erstellt werden, sondern kann auch nachträglich angelegt werden; die definierten Selektionsketten müssen aber in der Regel mit den bei der Erstdatenabgabe wirkenden Selektionsketten identisch sein.

Nutzerbezogene Selektionskriterien sind:

1. Fachlich durch Angabe von Objektarten, Attributarten und -werten sowie Relationen
2. Räumlich durch Angabe einer Fläche und
3. Zeitlich durch Angabe eines Zeitintervalls.

Objektarten, Attribute und Relationen bestimmen auch den inhaltlichen Umfang der abzugebenden Daten für den einzelnen Nutzer; diese Angaben sind ebenfalls im Nutzerprofil, das z.B. in ALKIS durch die Objektart AX_Benutzergruppe realisiert ist, zu hinterlegen.

10.5.2 Modellierung

Das NBA-Verfahren ist für alle Objektarten anzubieten, die eine datenführende Stelle im Bestand führt. Für den Nutzer kann die Selektion auf dem gesamten Vorrat der Objekteigenschaften aufsetzen; die Anforderungen des Datenschutzes sind dabei jedoch zu berücksichtigen. Als Ergebnis liefert das NBA - Verfahren als kleinste Einheiten der Änderungsdaten immer Fachobjekte. Diese Daten sind vollständig in Bezug auf das aktuelle Nutzerprofil; aus Sicht des gesamten Datenbestandes können diese Objekte unvollständig sein. Werden Fortführungsdaten für dasselbe Zeitintervall in mehreren Portionen an Nutzer abgegeben, stellt das abgebende System sicher, dass dieselbe Version eines Fachobjektes nur einmal an den Nutzer abgegeben wird.

Die räumliche Ausdehnung des Interessengebiets eines Nutzers wird durch Angabe beliebiger Flächen im Nutzerprofil beschrieben. Raumbezogene Elementarobjekte (REO) qualifizieren sich, sobald ein Teil von ihnen im angeforderten Gebiet liegt. In welchem Umfang Objekte durch Nachverfolgung von Relationen nachzuziehen sind, muss ebenfalls in den Selektionskriterien des Nutzerprofils beschrieben sein.

Der Zeitraum, für den die Bereitstellung von Änderungsdaten nach dem Verfahren NBA für verschiedene Nutzer sichergestellt werden muss, kann zeitlich begrenzt werden (zeitlicher Rahmen). Damit wird es möglich,

1. für jeden Nutzer Änderungsdaten rückwirkend innerhalb dieses Zeitraums anzufragen und
2. Änderungsdaten nutzerbezogen abzugeben, sie aber nicht nutzerbezogen vorhalten zu müssen.

Das Verfahren zur Nutzerbezogenen Bestandsdatenaktualisierung erfordert, dass für diesen Zeitraum Informationen über die Veränderung des Datenbestandes vorgehalten werden. Der Zeitraum wird durch die datenführende Stelle in Abstimmung mit den Nutzern bestimmt.

Die beim Verfahren NBA erforderliche Verwaltung der verschiedenen Ausprägungen eines Objektes über die Zeit wird durch das Versionskonzept abgedeckt. Deshalb wird

- die Datenhaltung der Änderungsdaten auf der Ebene der Bestandsdaten vorgenommen,
- die Führung der Informationen für das Verfahren der Nutzerbezogenen Bestandsdatenaktualisierung auf das Versionskonzept aufgesetzt und
- keine neue, zusätzliche und damit redundante Datenstruktur entwickelt.

Damit ist es möglich,

- aus einer Sammlung von Veränderungen,
- die jeweils die vollständigen Informationen zu den Objekten des Bestandes enthalten müssen,
- über den Zeitraum mehrerer Jahre hinweg (in Abhängigkeit vom zeitlichen Rahmen)
- Auswertungen nach
 - inhaltlichem Umfang durch Objektarten, Attribute und Relationen,
 - räumlicher Ausdehnung durch Flächen und
 - zeitlicher Ausdehnung durch Zeitintervalle

- nutzerbezogen

durchzuführen.

Um eindeutig die zu überschreibende Version zu kennzeichnen und Übermittlungsfehler im NBA - Verfahren aufzudecken, ist es erforderlich, den Objektidentifikator beim Datenaustausch um das Entstehungsdatum/-zeit zu ergänzen. Dies erfordert folgende Regeln:

- Das Entstehungsdatum im Objektidentifikator kann bei der Implementierung (z.B. im aufnehmenden System) weggelassen werden (Ersatz durch Zeitstempel der Versionen)
- Beim Datenaustausch mittels NBA - Verfahren mit fortführungsfallbezogener (kontinuierlicher) Datenabgabe wird beim Austausch von Objektversionen die Relation mit einem zum Entstehungsdatum der Objektversion passenden Entstehungsdatum der referenzierten Information ausgegeben.
- Beim Datenaustausch mittels NBA - Verfahren mit stichtagsbezogener Datenabgabe (Differenzdaten) wird beim Austausch von Objektversionen die Relation mit einem zum Stichtagsdatum passenden Entstehungsdatum der referenzierten Information ausgegeben.
- Bei der Erzeugung der Austauschdatei für die Nutzerbezogene Bestandsdatenaktualisierung muss das abgebende System folgende Funktionen erfüllen:
- Selektion der abzugebenden Daten aus dem (ggf. temporären) Historiennachweis entsprechend den im Nutzerprofil hinterlegten Selektionsketten und Filterangaben,
- Erzeugung der Fortführungsoperationen für das aufnehmende System aus dem Historiennachweis,
- Umwandlung der Daten in die Normbasierte Austauschschnittstelle.

Für die Ableitung der zu erzeugenden Fortführungsoperationen ist auszuwerten, ob das sich für die Datenausgabe qualifizierende Objekt aus Sicht der Datenhaltung eine erste, weitere oder letzte Version ist.

10.5.2.1 Abgabe von Änderungsdaten

Bei der kontinuierlichen, fortführungsfallbezogenen Datenabgabe (Änderungsdaten) werden alle sich für die Datenabgabe qualifizierenden Versionen eines Objektes verarbeitet. Das betrachtete Zeitintervall erstreckt sich von der letzten Datenabgabe bis maximal zur Gegenwart. Dabei ist aus Sicht der Datenhaltung auszuwerten, ob es sich um eine erste, weitere oder letzte Version eines Objektes handelt.

sich qualifizierende Version aus Sicht der Bestandsdaten-DB	auszugebende Operation
<u>erste</u> Version eines neuen Objekts	<Insert>
<u>weitere</u> Version eines Objektes	<Replace> der letzten übermittelten Version (Entstehungsdatum/zeit angeben)
<u>letzte</u> Version eines Objektes	<Delete> der letzten übermittelten Version (Entstehungsdatum/zeit angeben)

10.5.2.2 Abgabe von Differenzdaten

Bei der stichtagsbezogenen Datenabgabe (Differenzdaten) wird unter den Versionen eines Objektes jeweils nur die jüngste oder letzte Version verarbeitet, deren Entstehungs - bzw. Untergangszeit im betrachteten Zeitintervall liegt.

jüngste oder letzte sich qualifizierende Version aus Sicht der Bestandsdaten-DB	auszugebende Operation
<u>erste</u> Version eines neuen Objektes	<Insert> der <u>aktuellen</u> Version dieses Objektes
<u>Weitere</u> Version eines Objektes	<Replace> der letzten übermittelten Version (Entstehungsdatum/zeit angeben) mit der <u>aktuellen</u> Version dieses Objektes
<u>Letzte</u> Version eines Objektes	<Delete> der letzten übermittelten Version (Entstehungsdatum/zeit angeben)

10.5.2.3 Portionierung von NBA-Daten

Die Portionierung von NBA-Daten ist optional. Es besteht kein Zwang, diese zu verwenden, sie erlaubt jedoch den NBA-Beziehern die Übernahme der Daten in Etappen, was sich in der Vergangenheit bezüglich der Altverfahren insbesondere bei umfangreichen Datenbeständen bewährt hat.

Es werden insbesondere die folgenden Anforderungen adressiert: Bereitstellung der Daten von ALKIS und von ATKIS in geometrischer Portionierung. Die Portionsgröße soll variabel, aber einheitlich für einen Bezieher (Parametrisierung der Portionsgröße im Benutzerprofil) sein. NREO und ZUSO werden portionsbezogen über Relationen gemäß

Selektionskriterien im Nutzerprofil nachgezogen. Eine Portionierung allein nach Datenmenge ist nicht ausreichend.

10.5.2.3.1 Formale Form der Portionierung

Die Abwicklung erfolgt über im automatisierten NBA-Ablauf systemseitig erzeugte AX_NutzerbezogeneBestandsdatenaktualisierung_NBA-Dokumente. Diese werden zu Zeitpunkten gemäss den Vorgaben in AX_BenutzergruppeNBA erzeugt. Die zur jeweiligen Auftragsabwicklung verwendete Anzahl von NBA-Dokumenten bleibt dabei der Implementierung überlassen, allerdings soll es ein zusammenfassendes Auftragsprotokoll zu den ggf. n Ergebnisdateien geben. Aufgrund von [1..n] Benutzungsaufträgen mit Anlass "Nutzerbezogene Bestandsdatenaktualisierung NBA" (0040) werden alle erforderlichen Portionen als eigene unabhängige "Fortführungsaufträge"⁵ abgegeben.

Zur redundanzfreien Abgabe von Objekten folgen Erläuterungen bei der Parametrisierung der Portionierung (siehe unten).

Alle Portionen eines Benutzungsauftrages erhalten dieselbe Antragsnummer und dieselbe Auftragsnummer. Bei Folgelieferungen erhöht sich die Auftragsnummer.

Zu jeder Portion sind Metadaten zu erzeugen, aus denen mindestens das geometrische und logische Gebiet der Portion hervorgeht (wird nach Abschluss der Arbeiten der AdV zum Thema Metadaten nachgereicht).

10.5.2.3.2 Anforderungen an das abgebende System

10.5.2.3.2.1 Parametrisierung der Portionierung

Auf Ebene des Nutzerprofils für NBA kann die Portionierung durch Angabe des Portionierungsparameters aktiviert bzw. durch weglassen deaktiviert werden.

Die Selektion und Portionierung erfolgt als zweistufiger Prozess:

1. Die Selektionskriterien in AX_BenutzergruppeNBA dienen zur Auswahl der insgesamt in der Lieferung abzugebenden Objekte (unabhängig davon, ob eine Portionierung erfolgt oder nicht).

⁵ Hier ist der Begriff "Fortführungsauftrag" nicht wortwörtlich im Sinne des AX_Fortfuehrungsauftrag zu verstehen sondern in Form der Response AX_NutzerbezogeneBestandsdatenaktualisierung_NBA, welche die WFS-Operation "Transaction" wie der Fortführungsauftrag enthält.

2. Die Aufteilung der selektierten Objekte in Portionen erfolgt anhand des Portionierungsparameters wie nachfolgend beschrieben. Bei Einrichtung des NBA-Profiles muss auf sinnvolle Parametrisierung der Portionierung geachtet werden.

Fallen in einer Portion keine Fortführungsdaten für die betreffende Lieferung an, so muss diese Portion nicht erzeugt werden. Hierdurch wird die Anzahl der Portionen pro Lieferung variabel, jedoch wird durch die Festlegungen bzgl. Dateinamen klar, welchen Umfang die Gesamtlieferung hat.

Der folgende Portionierungsparameter steht zur Verfügung: **Angabe der Seitenlänge** in Metern. Sofern gesetzt, handelt es sich um einen positiven, von Null verschiedenen Integer-Wert. Das abgebende System unterteilt automatisch das in den Selektionskriterien insgesamt angegebene Gebiet in entsprechende Quadrate. Dabei gelten die folgenden Regeln:

- Das Gebiet wird erst von West nach Ost, dann von Süd nach Nord abgearbeitet. Die erste linke untere Ecke ergibt sich dadurch, dass vom südwestlichsten Punkt des Abgabegebietes auf das nächste Koordinatenpaar mit vollen Meterwerten gegangen wird, das südwestlich davon liegt. Ist der südwestlichste Punkt des Abgabegebietes bereits ein Koordinatenpaar auf volle Meterwerte, so wird er direkt verwendet.
- Alle REOs, die innerhalb eines Portionsquadrates liegen, sowie alle über die Selektionskriterien zusätzlich angeforderten, mit den jeweiligen REOs assoziierten NREO und ZUSO gemeinsam in eine Portion.
- Würde ein Objekt aufgrund seiner Ausdehnung in mehreren Portionen auftauchen, so wird es lediglich mit dem in der Reihenfolge am Anfang stehenden Quadrat abgegeben. Ein Beispiel: Ein Flächenobjekt erstreckt sich über Portion (=Quadrat) 12, 13, 21 und 22. Es wird nur in Portion 12 abgegeben.

Anhängende NREO und ZUSO werden nur in der jeweils ersten Portion ihres Auftretens abgegeben.

10.5.2.3.2.2 Klammerung der Lieferungsportionen

Die Portionen einer Lieferung werden über geeignete Portionskennungen als zusammengehörig kenntlich gemacht. Diese Kennung ist abzulegen

- im Attribut AX_NutzerbezogeneBestandsdatenaktualisierung_NBA.portionskennung,
- im Dateinamen der Portion.

Die Portionskennungen setzen sich wie folgt zusammen:

```

<NBA-Profilkennung6>
<_>
<Datum der NBA-Erzeugung im Format DateTime7>
<_>
<Laufende Nummer der Portion, ohne führende Nullen>
<von>
<Gesamtzahl der Portionen der Lieferung, ohne führende Nullen8>
<_>
<Koordinatenpaar der linken unteren Ecke der jeweiligen Portion, durch Unterstrich
getrennt9>

```

Sinn und Zweck dieser Namenskonvention ist a) die Klammerung der Portionen einer Lieferung und b) es Nutzern zu ermöglichen, Portionen räumlich zuzuordnen zu können ohne dazu in das NBA-Dokument schauen zu müssen.

Insgesamt ergeben sich Dateinamen z.B. wie folgt:

- *Firmaxy_041201_124von211_3401559_5572720.xml*
- *RMR_040715_7von31_3401449_5573000.xml*

Erläuterndes Beispiel:

1. In den Selektionskriterien bei der AX_BenutzergruppeNBA steht:

```

<wfs:Query typeName="AX_PunktortTA">
  <adv:XlinkPropertyPath>istTeilVon/AX_Grenzpunkt</adv:XlinkPropertyPath>
  <ogc:Filter>
    <ogc:BBOX>
      <ogc:PropertyName>position</ogc:PropertyName>
      <gml:Envelope srsName="urn:adv:crs:DE_DHDN_3GK3">
        <gml:pos>3353100.000 5532300.000</gml:pos>
        <gml:pos>3353300.000 5532500.000</gml:pos>
      </gml:Envelope>
    </ogc:BBOX>
  </ogc:Filter>
</wfs:Query>

```

2. In den zusätzlich wirkenden Portionierungsparametern steht:

⁶ Die Profilkennung sollte eine prägnante, nicht zu lange Bezeichnung sein. Bei Dateinamen problematische Zeichen (z.B. Leerzeichen) werden vermieden, indem sie bei der Ableitung dieses Namensbestandteils aus der Profilkennung nicht übernommen werden.

⁷ Bezug: Tagesaktualität der Datenbestände. Will man öfter fortführen, so ist eine Ergänzung um entsprechend genaue Uhrzeitangaben nötig. Der Zeitpunkt sollte dem Wert entsprechen, der bei AX_Benutzer im Attribut „letzteAbgabeZugriff“ gestzt wird (exakter Systemzeitpunkt der Abgabe).

⁸ Die Gesamtzahl wird getrennt für geometrische und nicht-geometrische Portionen ermittelt.

⁹ Genauigkeit auf volle Meter, d.h. ohne Nachkommastellen. Beispiel: "3401559_5572720"

Seitenlänge = 100m

3. Ergebnis:

Es entstehen maximal vier Portionen, gefüllt mit AX_PunktortTA und zugehörigen AX_Grenzpunkt

Aus Profilkennung und Datum können zur Erhöhung der Übersichtlichkeit geeignete Verzeichnisstrukturen generiert werden.

10.5.2.3 Anforderungen an das aufnehmende System - Verarbeitung der Lieferung

Anhand der Portionskennung ("3von8") wird ausgewertet, ob alle Portionen einer Lieferung übernommen worden sind. Dem aufnehmenden System muss hierzu bekannt sein, aus welchen Portionen die Gesamtlieferung besteht. Erst nach vollständiger Übernahme einer NBA-Lieferung kann diese auf Anforderung quittiert werden und es darf mit der Übernahme der Folgelieferung begonnen werden.

10.5.2.4 Quittierung von NBA-Lieferungen

Bei Übernahme einer NBA-Lieferung kann eine Quittierung an die liefernde Stelle in Form des NBA-Quittierungs-Auftrags AX_NBAQuittierung erfolgen, soweit dies länderspezifisch gewünscht wird und in der NBA-Lieferung angefordert wurde. Die Art der Verarbeitung der NBA-Quittierung obliegt länderspezifischen Regelungen.

11 Koordinatenreferenzsysteme und Maßeinheiten

Siehe separates Dokument „Koordinatenreferenzsysteme und Maßeinheiten für AFIS-ALKIS-ATKIS“.

12 Qualitätssicherung

12.1 AdV-Qualitätssicherungssystem

Die AdV hat folgende Eckpunkte des Qualitätssicherungssystem für die Geodaten des amtlichen Vermessungswesens beschlossen:

„Durch bundeseinheitliche Festlegung, Benennung und beschreibende und quantitative Qualitätsmerkmale kennzeichnet und sichert die AdV die Qualität der geotopographischen und liegenschaftsbeschreibenden Produkte des amtlichen Vermessungswesens. Dabei sind die bundesweite Aktualität, Einheitlichkeit, Vollständigkeit und Verfügbarkeit der Produkte wesentliche Qualitätsmerkmale. Die Vermessungsverwaltungen gewährleisten die Einhaltung der AdV-Produktqualität durch standardisierte Prüfverfahren und erklären die Konformität mit den AdV-Standards.“

Ziel ist eine umfassende Qualitätssicherung für die Geodaten des amtlichen Vermessungswesens als Ergebnis des Konzeptions- und Produktionsprozesses. Die Konzeption (AAA-Basisschema, AAA-Fachschemata) liegt in den Händen der Ländergemeinschaft, vertreten durch die AdV, während die Produktion der Datenbestände im Einklang mit dem AAA-Anwendungsschema Aufgabe der Vermessungsverwaltung eines jeden einzelnen Landes ist.

12.2 Qualitätssicherungsmodell

Das Beziehungsgefüge der Qualitätsprüfaspekte ist in nachfolgendem Qualitätssicherungsmodell für das AAA-Anwendungsschema dargestellt:

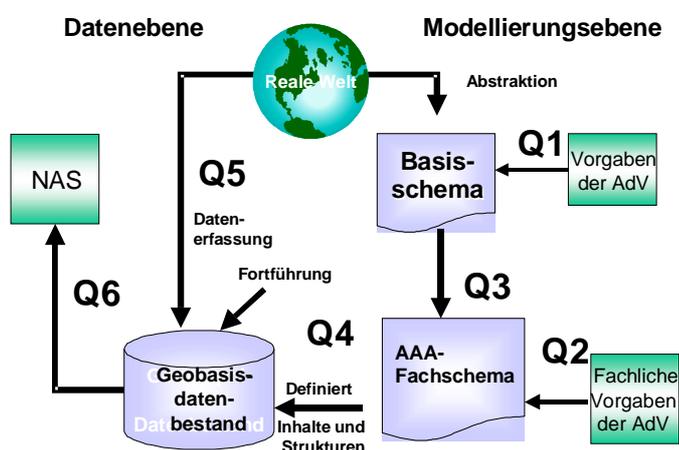


Abbildung 12-1: Das Qualitätssicherungsmodell des AFIS-ALKIS-ATKIS-Projektes

Q1 misst das AAA-Basisschema an den strategisch-fachlichen Vorgaben der AdV, Q2 misst das AAA-Fachschemata an den fachlichen Vorgaben der AdV. Mit Q3 wird festgestellt, ob das AAA-Fachschemata den Regeln des AAA-Basisschemas entspricht. Q1, Q2 und Q3 prüfen die konzeptionelle, interne Qualität.

Q4 prüft den Geobasisdatenbestand intern als Produkt auf logische Übereinstimmung mit dem AAA-Anwendungsschema und auf die Einhaltung der dort niedergelegten Qualitätsangaben, während Q5 den Geodatenbestand extern mit der realen Welt vergleicht. Q6 betrifft die Qualität der NAS zum Nutzer.

Im einzelnen ergibt sich folgendes Qualitätsprüfungsschema:

		AdV	Länder
1.	AdV-Regelwerke und Standards zur Entwicklung von Verfahren und Programmsystemen		
	Qualitätssicherung des AAA-Basisschemas gegenüber den Vorgaben der AdV (Q1)	X	
	Qualitätssicherung des gemeinsamen AAA-Fachschemas gegenüber den fachlichen Vorgaben der AdV (Q2)	X	
	Qualitätssicherung des gemeinsamen AAA-Fachschemas gegenüber dem AAA-Basisschema (Q3)	X	
	Qualitätssicherung der Datenbestände (ALKIS/ATKIS/AFIS) gegenüber dem gemeinsamen AAA-Anwendungsschema (Q4)		X
	Qualitätssicherung der Austauschdaten gegenüber der NAS (Q6)	Grund-sätze	X
2.	Vorgaben für die AdV-Produktqualität		
	Festlegung von beschreibenden und bewertenden Qualitätsmerkmalen für einheitliche Produkte einschl. Aktualität, Einheitlichkeit, Vollständigkeit und Verfügbarkeit.	X	
3.	Vorgaben für Qualitätssicherung der Bestandsdaten		
	Qualitätssicherung der Bestandsdaten gegenüber der fachlichen Realität (Q5)		X
4.	Qualitätssicherung (als Teil des Qualitätsmanagements)		
	Konformitätserklärung durch die Vermessungsverwaltungen		X

Die Qualitätssicherungsgrundsätze zu Q6 gehen davon aus, dass bei Datenabgaben aus AFIS/ALKIS/ATKIS keine Überprüfung der entstehenden NAS-Dateien gegenüber dem Modell vorgenommen werden muss. Die modellkonforme Implementierung hat dies anhand der jeweils gültigen XML-Schemadateien (XSD) sicher zu stellen; die Interoperabilität ist zu gewährleisten. Die Datenübernahme ist Bestandteil des Qualifizierungsprozesses.

ses. In diesem Rahmen müssen entsprechende Prüfwerkzeuge zur Verfügung stehen, die anhand der jeweils gültigen XML-Schemadateien (XSD) die Qualität der Übernahmedaten sicherstellen. Die Prüfung der Austauschdaten gegenüber den NAS-Schema unterscheidet die Prüfung der Wohlgeformtheit der XML-Datei (Prüfwerkzeug z.B. xmlint.exe) und die Prüfung der Gültigkeit der XML-Datei (Prüfwerkzeug z.B. XSV.exe).

12.3 Systematik und Dokumentation der Qualitätssicherung

Auf der Basis der ISO-Norm 19105 „Geographic Information - Conformance and testing“ sind Abstract test suites (ATS) zu formulieren, welche zur Konformitätsuntersuchung herangezogen werden sollen. Danach wird jedes AAA-Qualitätskriterium nach folgendem Schema analysiert und dokumentiert:

- These (conformance requirements).
- Untersuchungsansätze, formuliert in Fragestellungen.
Jede der Fragestellungen kann zu separaten Testmodulen und dort zu Testfällen führen, die wie folgt strukturiert sind:
 - a) Testzweck
 - b) Testmethode
 - c) Referenz
 - d) Testtyp.
- Test zur Bestätigung oder Widerlegung der These (executable test suite - ETS mit executable test cases).

Die Ergebnisse der Qualitätssicherung für das AAA-Anwendungsschema sind in folgenden Dokumenten unter <http://www.adv-online.de/veroeffentlichungen> veröffentlicht:

- „Aufstellung von Qualitätskriterien und Aussagen zu deren Umsetzung für die GeoInfo-Dok“, Stand 26.10.2000.
- „Grundsätze für Qualitätskriterien und standardisierte Prüfverfahren für die Anwendung des AFIS-ALKIS-ATKIS-Basisschemas bei der Entwicklung der Fachschemata“ mit Anlage „Qualitätskriterien (Q3) zur Prüfung des AFIS-ALKIS-ATKIS-Fachschemas gegenüber dem AFIS-ALKIS-ATKIS-Basisschemas“; Stand:10.07.2002.
- „Dokumentation zur Modellierung der Geoinformationen des amtlichen Vermessungswesens (GeoInfoDok) - Qualitätssicherung ALKIS -; Stand. 22.11.2002.

Die Dokumentationen zur Qualitätssicherung von AFIS und ATKIS liegen noch nicht vor.

13 Glossar, Abkürzungen und Abbildungsverzeichnis

13.1 Fachbegriffe und ihre englische Übersetzung

Fachbegriff (deutsch)	Erläuterung	Fachbegriff (englisch)
AdV-Standard	Die AdV schafft Regelwerke zur Entwicklung von <i>Verfahren</i> und <i>Programmsystemen</i> und zur Herstellung von <i>Produkten</i> . AdV-Regelwerke, die der Festlegung von bundeseinheitlichen Grunddatenbeständen, Datenaustauschnittstellen und Standardprodukten dienen, werden durch Verpflichtung der Mitgliedsverwaltungen zu ihrer Einhaltung zu AdV-Standards erhoben.	AdV-standard
AFIS-ALKIS-ATKIS-Anwendungsschema	Das Basisschema und das anwendungsspezifische Subschema von AFIS, ALKIS und ATKIS (AAA-Fachschemata) bilden zusammen das gemeinsame AFIS-ALKIS-ATKIS-Anwendungsschema.	AFIS-ALKIS-ATKIS application schema
AFIS-ALKIS-ATKIS-Basisschema	↔ siehe Basisschema	AFIS-ALKIS-ATKIS basic schema
AFIS-ALKIS-ATKIS-Referenzmodell	Das AFIS-ALKIS-ATKIS-Referenzmodell ist ein gemeinsames Rahmenmodell, in dem die Strukturen und Inhalte der Produkte von AFIS, ALKIS und ATKIS, die Datenerfassungsquellen, Bestandsdaten sowie deren digitale und analoge Auszüge aus AFIS, ALKIS und ATKIS sowie die Abgabe der Daten an den Nutzer als Komponenten mit ihren gegenseitigen Beziehungen definiert sind.	AFIS-ALKIS-ATKIS reference model
Anlass	Der Anlass gibt den Grund einer Veränderung eines Objektes wieder. Er wird als Attribut bei AA_Objekt neben dem Objektidentifikator und dem Lebenszeitintervall geführt.	cause (for a change)
Anwendungsschema	Ein Anwendungsschema ist ein konzeptuelles Schema für Daten, die von einer oder mehreren Anwendungen benötigt werden. conceptual schema for data required by one or more applications	application schema

Attribut	Attribute sind selbstbezogene Eigenschaften eines Objekts. Deren individueller Aufbau wird bei jeder Objektart als Attributart in den Objektartenkatalogen beschrieben.	attribute
Ausgabekatalog	Im Ausgabekatalog ist die Art und Weise der Aufbereitung und Ausgabe der Daten und Auszüge aus AFIS, ALKIS und ATKIS an den Nutzer spezifiziert.	output catalogue
Auszüge	Auszüge sind nach Inhalt, Gebiet und/oder Zeitraum (wie z. B. Fortführungsdatenbestände) selektierte Datenbestände, die an den Nutzer als objekt- oder bildstrukturierte Daten, aufbereitete Informationen oder analoge Auszüge abgegeben werden.	extracts
Basisschema	Das Basisschema ist ein Schema, das die grundlegenden Eigenschaften für eine oder mehrere Anwendungen beschreibt. Es enthält den einheitlichen und objektorientierten Modellansatz, auf dem die Subschemata von AFIS, ALKIS und ATKIS aufbauen.	basic schema
Bestandsdaten	Bei Bestandsdaten handelt es sich um Geoinformationen des amtlichen Vermessungswesens in AFIS, ALKIS und ATKIS. Sie enthalten die vollständige Beschreibung von Fachobjekten einschließlich der Daten zu ihrer kartographischen oder textlichen Darstellung in einem oder mehreren Zielmaßstäben.	(geographic) data in primary database
Bestandsdaten-aktualisierung	Die Bestandsdatenaktualisierung ist ein Verfahren zur Fortführung von Sekundärdatenbeständen bei Nutzern mit Hilfe der Normbasierten Austausch-schnittstelle (NAS). Das Verfahren wird mit "NBA - Verfahren" abgekürzt.	update of primary database
Bestandsobjekte	Bestandsobjekte sind Fachobjekte des Liegenschaftskatasters, die nach dem AFIS-ALKIS-ATKIS-Datenmodell modelliert wurden.	features in primary database
Datenmodell	Ein Datenmodell beschreibt die grundlegenden Eigenschaften, die für alle Erscheinungen einer bestimmten (fachbezogenen) Sicht auf die Wirklichkeit eine einheitliche Abbildung erleichtern. Es bestimmt die grundsätzlichen Strukturen, die prinzipiell	data model

	möglichen Beziehungen und die Eigenschaften, die zugeordnet werden können. à siehe auch Modell	
Datenmodellierungssprache	à siehe Modellierungssprache	data modeling language
Differenzdaten	Differenzdaten sind stichtagsbezogene Änderungsdaten, die nötig sind, um den Ausgangszustand der Bestandsdaten beim Nutzer auf den gewünschten Endzustand (Stichtag) zu bringen. Sie umfassen alle neu entstandenen Objekte, die jeweils aktuellsten Versionen fortgeführter Objekte sowie Angaben zu historisch gewordenen Objekten. Die Differenzdaten stellen eine Untermenge der Änderungsdaten dar.	change-only data, differential update
Digitales Bildmodell	Ein Digitales Bildmodell ist ein Modell zur Speicherung von Bilddaten, z.B. digitalen Orthophotos.	digital image model
Digitales Geländemodell	Ein Digitales Geländemodell ist ein Digitales Höhenmodell mit zusätzlichen topographischen Informationen wie Bruchkanten etc.	digital terrain model
Digitales Höhenmodell	Ein Digitales Höhenmodell speichert Informationen über die Höhe von diskreten Punkten, die i.d.R. in einem regelmäßigen Gitter angeordnet sind. Diese Höheninformationen werden genutzt, um Höhen für alle anderen Positionen zu berechnen bzw. zu interpolieren.	digital elevation model
Elementarobjekte	Elementarobjekte stellen die kleinsten, fachlich eigenständigen Einheiten dar. Sie setzen sich nicht aus anderen eigenständigen Einheiten zusammen. Es gibt in der Modellierung für AFIS, ALKIS und ATKIS folgende Arten von Elementarobjekten: Raumbezogener Elementarobjekte (REO) Raumbezogene Elementarobjekte sind zu bilden, wenn zusätzlich zu fachlichen Eigenschaften auch geometrische oder topologische Eigenschaften nachgewiesen werden sollen. Nicht raumbezogener Elementarobjekte (NREO) Nicht raumbezogene Elementarobjekte	elementary objects

	sind zu bilden, wenn neben den fachlichen Eigenschaften keine geometrischen oder topologischen Eigenschaften nachgewiesen werden können. à siehe auch "Zusammengesetzte Objekte (ZUSO)"	
Erhebungsdaten	Die Erhebungsdaten stellen die Grundlage zur Fortführung der amtlichen Geoinformationen dar. Sie werden durch Erhebungsprozesse aus Quelldaten, die mit den bekannten geodätischen Mess- und Erkundungsmethoden in der realen Welt erhoben oder aus kartographischen Darstellungen und anderen Unterlagen erfasst werden, gebildet.	collected data
Erhebungsprozess	Der Erhebungsprozess erzeugt zur Qualifizierung und Fortführung der amtlichen Geoinformationen aus Quelldaten Erhebungsdaten. Der Erhebungsprozess ist nicht Bestandteil des Anwendungsschemas ALKIS und wird länderspezifisch modelliert.	data collection process
Fachdaten	Fachdaten sind anwendungsspezifische Daten eines Fachanwenders, z.B. Leitungsdaten oder Kundendaten eines Versorgungsunternehmens. Diese können mit einem Raumbezug versehen werden.	technical data
Fachdatenobjekt	Fachdatenobjekte sind Objekte in Fachinformationssystemen anderer Fachbereiche.	technical data object
Fachdaten- verbindung	Die Fachdatenverbindung beinhaltet die Integrations- und Verknüpfungsmöglichkeiten zwischen den Daten der Vermessungsverwaltung (Basisdaten) und den Fachdaten in Form von Referenzen. Diese Verknüpfung kann entweder in den raumbezogenen Basisinformationssystemen der Vermessungsverwaltung, im Fachinformationssystem (einseitige Verknüpfung) oder gegenseitig in beiden Informationssystemen (gegenseitige Verknüpfung) erfolgen.	association to technical data
Fachinformationssystem	System, das Informationen fachlicher Art enthält und Geobasisinformationen der Vermessungs- und Katasterverwaltung als Grundlage nutzt.	technical information system

Fachobjekt	Ein Fachobjekt entsteht durch Abstraktion eines Gegenstandes oder Sachverhaltes der realen Welt. Im Anwendungsbereich von AFIS, ALKIS und ATKIS ist dies eingeschränkt auf die Gegenstände und Sachverhalte, die den fachlichen Gehalt von AFIS, ALKIS und ATKIS ausmachen. à Objekt abstraction of real world phenomena NOTE 1 A feature may occur as a type or an instance. Feature type or feature instance should be used when only one is meant. NOTE 2 UML uses feature for another concept than the use of feature within this standard. In UML, a property, such as operation or attribute, is encapsulated as part of a list within a classifier, such as an interface, a class or a data type.	feature
Festpunkt		geodetic control station
Fortführung	Fortführung ist die Aktualisierung von Bestandsdaten. Die Fortführungsdaten (Daten und Metadaten) werden dabei durch Anwendung geeigneter Methoden in den Bestand überführt.	update, revision
Fortführungsauftrag	Der Fortführungsauftrag ist eine Objektart, die ein oder mehrere Fortführungsfälle zu einer Einheit zusammenfasst. Sie steuert das Verfahren der Datenaktualisierung für sämtliche Bestandsobjekte.	revision case or instance
Führungsprozess	Beim Führungsprozess handelt es sich um die Ersteinrichtung bzw. Fortführung der Bestandsdaten (Geobasisdaten und Metadaten).	process of updating
Geobasisdaten	Geobasisdaten sind grundlegende amtliche Geodaten, welche die Landschaft (Topographie), die Flurstücke und die Gebäude im einheitlichen geodätischen Raumbezug anwendungsneutral beschreiben. Geobasisdaten werden durch die Vermessungsverwaltungen der Länder erhoben, geführt und bereitgestellt. Sie erfüllen die Funktion der Basisdaten für Geofachdaten.	(geographic) reference data
Geodaten	Geodaten sind Daten, die sich auf räumliche Objekte in Relation zum Erdkörper beziehen.	geographic data
Geodatenbestand	Geodatenbestand umfasst die Gesamtheit der geographischen Daten, die in einer	Geographic database

	Datenbank vorgehalten werden.	
Geoinformationen	Geoinformationen sind Geodaten, die für eine bestimmte Anwendung ausgewählt, bearbeitet und aggregiert wurden.	geoinformation
Geoinformationssystem	Ein Geoinformationssystem ist ein System zur Erfassung, Speicherung, Prüfung, Veränderung, Integration, Analyse und Darstellung von Geoinformationen.	geographic information system
Geokodierung	Unter Geokodierung versteht man die Zuordnung von Objekten (Daten, Informationen) zur Erdoberfläche mit Hilfe eines (räumlichen) Referenzsystems.	geocoding
Grunddatenbestand	Als Grunddatenbestand wird der von allen Vermessungsverwaltungen der Länder der Bundesrepublik Deutschland bundeseinheitlich zu führende und dem Nutzer länderübergreifend zur Verfügung stehende Datenbestand (in AFIS, ALKIS und ATKIS) bezeichnet.	(geographic) core data inventory
Historisierung	Als Historisierung bezeichnet man das Entstehen der letzten Version (Untergang) eines Fachobjektes.	historization
Identifikator	Der Identifikator kennzeichnet ein Objekt eindeutig (unique). Er ist eine besondere selbstbezogene Eigenschaft des Objekts und steht stellvertretend für das Objekt, das er repräsentiert. Er bleibt so lange unverändert, wie das entsprechende Objekt existiert. Die für den AFIS-ALKIS-ATKIS-Datenaustausch definierte Austauschschnittstelle beruht auf der Anwendung der Norm ISO 19118 Encoding. Die daher Normbasierte Austauschschnittstelle wird mit "NAS" abgekürzt.	identifier
Kardinalität	Die Kardinalität ist die Mächtigkeit einer Menge bzw. die Anzahl der Elemente einer endlichen Menge. In der Modellierung wird dies durch den Bereich möglicher Kardinalitäten ausgedrückt. Gebräuchliche Bereichsangaben in den Objektartenkatalogen sind z.B.: 1:1 Kommt genau einmal vor. 1:? Kommt einmal oder beliebig oft vor. 0:1 Kommt keinmal oder einmal vor. 0:? Kommt keinmal oder beliebig oft vor.	cardinality

Kartengeometrieobjekt	Kartengeometrieobjekte sind Fachobjekte, die bei der Ableitung für einen bestimmten Kartenmaßstab aus Gründen der kartographischen Generalisierung ihre geometrische Form und/oder Lage verändert haben.	map geometry object
Klasse	<p>Eine Klasse ist ein Begriff aus der objektorientierten Modellierung und beschreibt eine Menge von Objekten, die sich durch die gleichen Attribute, Methoden, Relationen und das gleiche (dynamische) Verhalten auszeichnen.</p> <p>descriptor of a set of objects that share the same attributes, operations, methods, relationships, and behaviour</p> <p>NOTE A class represent a concept within the system being modelled. Depending on the kind of model, the concept may be real-world (for an analysis model), or it may also contain algorithmic and computer implementation concepts (for a design model). A classifier is a generalization of class that includes other class-like elements, such as data type, actor and component.</p> <p>NOTE A class may use a set of interfaces to specify collections of operations it provides to its environment.</p>	class
Kodierung	Die Kodierung ist die Abbildung von Informationen (Daten, Objekte) in ein (maschinenlesbares) Schlüsselssystem (Verschlüsseln); die inverse Abbildung ist die Dekodierung.	encoding
konzeptuelles Modell	<p>Ein konzeptuelles Modell ist als Abbild der realen Welt bezüglich konkreter Fachthemen zu verstehen.</p> <p>model that defines the concepts of a universe of discourse</p>	conceptual model
konzeptuelles Schema	<p>Das konzeptuelle Schema beschreibt das konzeptuelle Modell mit Hilfe einer formellen Sprache.</p> <p>schema of a conceptual model</p> <p>A conceptual schema classifies objects into types and classes, identifying types of objects according to their properties and associations between types of objects.</p>	conceptual schema
Metadaten	Metadaten sind Daten über Daten. Sie dienen der Beschreibung der Geodaten hinsichtlich nutzerrelevanter Aspekte zur Bewertung der Eignung der Daten und	metadata

	<p>des Zugriffs auf dieselben. ISO unterscheidet etwa 400 optionale, obligatorische und bedingt obligatorische Metadatenelemente.</p> <p>data describing and documenting data</p>	
Metadatenkatalog	<p>Ein Metadatenkatalog ist ein Katalog mit beschreibenden Daten (Metadaten). Er enthält für jeden Datenbestand insbesondere Angaben über den Inhalt, die Darstellung, die Ausdehnung (sowohl geometrisch als auch zeitlich), den Raumbezug, die Qualität und die verantwortliche Institution, aufgrund derer ein Nutzer die Verfügbarkeit und Eignung der Geodaten für seine Zwecke bewerten kann.</p>	metadata catalogue
Metaobjektklasse	<p>Metaobjektklassen bzw. Metaklassen werden definiert, um auf deren Basis Fachobjekte zu instanzieren. Bei der Modellierung der Basisklassen wurde eine raumbezogene Metaobjektklasse (GF_FeatureType aus ISO 19109) verwendet.</p>	metaclass
Methode	<p>Eine Methode ist eine an ein Objekt gebundene Funktion. Sie hat nur Auswirkungen auf dieses Objekt selbst bzw. auf dessen Eigenschaften (Attribute, Geometrie und Relationen).</p>	method
Modell	<p>Ein Modell ist eine vereinfachende bildliche oder mathematische Darstellung von Strukturen und des Verhaltens komplexer Sachverhalte der realen Welt. Es dient der Lösung bestimmter Aufgaben, deren Bewältigung am Original unmöglich oder unzumutbar ist.</p> <p>model</p> <p>abstraction of some aspects of reality</p>	model
Modellierungssprache	<p>Eine Modellierungssprache bietet darstellende und/oder lexikalische (textliche) Elemente zur Beschreibung eines Modells. Für die Modellierung im Fachbereich AFIS-ALKIS-ATKIS wird gemäß ISO19103 die Unified Modeling Language (UML) verwendet.</p> <p>formal language based on a conceptual</p>	conceptual schema language

	<p>formalism for the purpose of representing conceptual schemas</p> <p>EXAMPLE UML, EXPRESS, IDEFIX</p> <p>NOTE A conceptual schema language may be lexical or graphical.</p>	
Normen	<p>Normen dienen der Standardisierung verschiedenster Bereiche menschlichen Wirkens. Eine Art von Normen sind ISO-Normen: Dokumente, die von Mitgliedern der International Organization for Standardization (ISO) in sogenannten Technical Committees (TC) im Rahmen eines mehrstufigen Entwicklungsprozesses erstellt werden. Für Geoinformation ist das TC211 "Geographic information/Geomatics" zuständig (siehe http://www.isotc211.org/). Dabei durchlaufen diese Dokumente mehrere Reifestadien. Endstadium ist das des "International Standard". Für nähere Informationen: Siehe http://www.iso.ch/.</p>	de-jure standards
Nutzer-spezifische Bestandsdaten-aktualisierung	<p>Operation zur Fortführung von sekundären Datenbeständen mit Hilfe von Differenzdaten.</p>	user-specific updating of secondary databases
Objekt	<p>Ein Objekt (Instanz einer Klasse) ist ein materieller oder immaterieller Gegenstand der fachlichen Realität, der eindeutig identifizierbar und durch Abstraktion auf seine relevanten Eigenschaften beschränkt ist. Dies schließt seinen Zustand und sein Verhalten ein.</p> <p>à Fachobjekt</p> <p>a discrete entity with a well-defined boundary and identity that encapsulates state and behaviour; an instance of a class</p>	object
Objektart	<p>Objekte werden nach verschiedenen Objektarten klassifiziert. Für jede Objektart werden im Objektartenkatalog alle erlaubten Eigenschaften festgelegt (Typenebene). Diese Festlegungen gelten dann für alle Ausprägungen (Instanzenebene), das sind die einzelnen Objekte dieser Art, uneingeschränkt. Jedes Objekt gehört zu genau einer Objektart.</p> <p>class of real world phenomena with common</p>	feature type

	<p>properties</p> <p>EXAMPLE The phenomenon 'Eiffel Tower' may be classified with other similar phenomena into a feature type 'tower'.</p> <p>NOTE In a feature catalogue, the basic level of classification is the feature type.</p>	
Objektartenkatalog	<p>Der Objektartenkatalog führt für alle Objektarten abschließend die auf der Grundlage des AFIS-ALKIS-ATKIS-Anwendungsschemas modellierten Datenelemente mit ihren Festlegungen auf.</p> <p>catalogue containing definitions and descriptions of the feature types, feature attributes, and feature relationships occurring in one or more sets of geographic data, together with any feature operations that may be applied</p>	feature catalogue
Objektbehälter	<p>Der Objektbehälter bildet eine datentechnische Klammer um die verschiedenen Versionen eines Objekts, die dieses im Verlaufes seines Lebens durchläuft. Durch "Klammerung" der Versionen innerhalb eines Objektbehälters bleibt die fachliche Objektsicht stets erhalten.</p>	container for feature versions
Objektidentifikator	à Identifikator	object identifier
Objektorientierung	<p>Grundlage der Objektorientierung, die sowohl bei der objektorientierten Modellierung von Systemen und Prozessen, bei der objektorientierten Programmierung als auch bei objektorientierten Datenbankmanagementsystemen eingesetzt wird, ist die Abstraktion der Realität in Objekte, Klassen und Beziehungen. Die Objektorientierung ist damit eine Methode (Konzept, Sprache) zur Modellierung von Sachverhalten, bei der sämtliche erforderlichen Informationen (Daten und Methoden) als gekapselte Objekte, die miteinander kommunizieren können, aufgefasst werden.</p>	object orientation
Objektstrukturierung	<p>Objektstrukturierung besagt, dass die in einem Anwendungsschema modellierten Sachverhalte in der Struktur von Objekten vorliegen und nach Objekten geordnet sind. Im Gegensatz zur Objektorientierung wird bei der objektstrukturierten Modellierung das</p>	object structuring

	Verhalten eines Objekts, das durch seine Methoden repräsentiert wird, nicht beschrieben.	
Präsentationsobjekt	Präsentationsobjekte sind raumbezogene Elementarobjekte, welche die Fachobjekte um Angaben zur Darstellung von Schrift und Signaturen ergänzen. Dabei werden all jene Texte und Signaturen definiert, die nicht vollautomatisch für einen bestimmten Zielmaßstab einer Karte erzeugt und platziert werden können. Präsentationsobjekte sind in dem Objektartenkatalog zu definieren, auf dem sie aufbauen (z.B. ATKIS-Basis-OK).	presentation object
Primärnachweis	Der Primärnachweis ist der originäre, von der entsprechend fachlich zuständigen Stelle (Datenherr) geführte Datenbestand.	primary database
Protokollobjekt	Ein Protokollobjekt dient der Übermittlung von Protokollinformationen.	protocol object
Prozess	Ein Prozess überführt einen Quelldatenbestand in einen Zieldatenbestand. Zur Beschreibung von Prozessen (Vorgänge, Methoden) werden die Sprachmittel textliche, formularmäßige Beschreibung und Pseudocode verwendet. Die "Prozesse in ALKIS" enthalten die Definitionen und Beschreibungen der Methoden und Vorgänge sowie die Prozessobjektarten zur Steuerung der Prozesse.	process
Pseudocode	Der Pseudocode ist ein Sprachmittel zur Beschreibung eines Prozesses. In ihm erfolgt die Beschreibung der Bearbeitungsschritte eines Vorgangs mit der folgenden Notation: "objektart.methode (parameter)".	pseudocode
PunktLinienThema	Ein PunktLinienThema im Sinne der Modellierung beinhaltet die Möglichkeit, Fachobjekte so zu gruppieren, dass sie Geometrien gemeinsam nutzen. Dies führt dazu, dass exakt übereinanderliegende Linien und Punkte sich gegenseitig zerschlagen und zu redundanzfreien Geometrien vereinigen. Sich kreuzende Linien führen nicht zur	point and line theme

	gegenseitigen Zerschlagung. Überlap-pende Flächen zerschlagen sich nicht zu den jeweils kleinstmöglichen Teilflächen.	
Qualifizierungsprozess	Der Qualifizierungsprozess überführt die Erhebungsdaten (Ausgangsdaten) in die Fortführungsdaten (Zieldaten). Er dient der Qualitätssicherung und stellt sicher, dass die Fortführungsdaten den Qualitätsanforderungen entsprechen.	qualifying process
Raumbezug	Der Raumbezug ist die geometrische (Lage und Form des Objekts) und/oder die topologische (Lagebeziehungen zwischen Objekten) Beschreibung eines Objekts und stellt somit den Bezug des Objekts zu einem räumlichen Ausschnitt der Erde her.	spatial reference
Raumbezugsgrundform	Raumbezugsgrundformen sind von der ISO-Norm 19107 <i>Spatial schema</i> für die Verwendung in Anwendungsschemata zur Verfügung gestellte, vordefinierte "Geometrische Objekte" (GM_Objekt) und "Topologische Objekte" (TP_Objekt), die als UML-Klassen beschrieben sind. Die Raumbezugsgrundformen werden in der Regel als Attributwerte der Objekte geführt.	geometrical and topological primitives
Relation	Unter dem Begriff " <i>Relation</i> " wird ganz allgemein eine semantische Verbindung zwischen Modellelementen verstanden. <i>Relation</i> ist der Oberbegriff, unter dem die Begriffe <i>Assoziation</i> , <i>Generalisierung/Spezialisierung</i> , <i>Abhängigkeit</i> und <i>Realisierung/Verfeinerung</i> subsummiert werden.	relation
Schema	Ein Schema ist eine anschauliche (bildliche) Darstellung des Wesentlichen eines Sachverhalts. Es ist das Ergebnis der darstellenden und/oder lexikalischen (textlichen) Beschreibung eines Modells mit Hilfe einer (normierten) Modellierungssprache. formal description of a model	schema
Sekundärnachweis	Der Sekundärnachweis beinhaltet eine Kopie des gesamten Primärnachweises oder von Teilen desselben, die laufend aktualisiert wird. Die Fortführung des Sekundärnachweises erfolgt über die Nutzerbezogene Bestandsdatenaktuali-	secondary database

	sierung (NBA).	
Signaturenkatalog	Ein Signaturenkatalog enthält Regeln, nach denen die im Ausgabekatalog definierten Ausgaben von Geodaten in Abhängigkeit von ihrem Objekttyp, von bestimmten Attributen/Attributwerten, von bestimmten Referenzbedingungen und/oder von zu berechnenden Werten signaturiert werden, und die Beschreibung aller vorkommenden Signaturen. Er ist an den jeweiligen Zielmaßstab angepasst.	portrayal catalogue
Standard	Ein Standard ist ein breit akzeptiertes und angewandtes Regelwerk. Er wird meist nur von <i>einer</i> Institution erzeugt, d.h. es existiert dafür kein internationales Gremium. Die Verbindlichkeit eines Standards geht oft nicht über eine einzelne Organisation hinaus. Ein Standard wird nicht offiziell international herausgegeben, wie dies bei Normen der Fall ist. Einen regulären Ablauf der Entstehung (wie bei Normen z.B. von DIN, ISO oder CEN) gibt es nicht. à AdV-Standard	de-facto standard
Standardausgaben	Mit Standardausgaben werden Regelfälle der Benutzung (auch im Sinne einheitlicher Produkte der AdV) abgedeckt. Es sind Ausgabeprodukte der AFIS-ALKIS-ATKIS-Daten, die normalen bzw. „normierten“ Ansprüchen an die entsprechenden Datenbestände Genüge tun. Sie werden über die Definition einheitlicher Selektions- und Filterkriterien festgelegt. Beispiele von Standardausgaben für ALKIS sind die Liegenschaftskarte, der Flurstücks- und Eigentüternachweis und die Liegenschaftskarte mit Flurstücks- und Eigentümerangaben.	standard output
Subschema	à Basisschema	subschema
Transferprozess	<i>Siehe GeoInfoDok, 3.3.9.6</i>	transfer process
URI	Uniform Resource Identifier Zeichenkette, die eindeutig auf eine Ressource (Name, Datei etc.) verweist. Der Ort der Ressource ist nicht eingeschränkt (www, LAN, ...). URLs (Uniform Resource Locator) und URNs (Uniform Resource Name) sind Teilmengen von URIs.	URI (Uniform Resource Identifier) generic set of all names/addresses that are short strings that refer to resources

Versionierung	Versionierung ist die zeitlich geordnete Veränderung von Fachobjekten durch die Fortführung. Kernpunkt des Versionskonzeptes ist die Überlegung, dass jedes Fachobjekt neben anderen Informationen ein Lebenszeitintervall (bestehend aus Entstehungs- und Untergangsdatum) führt. à Versionierungsschema	versioning
Versionierungsschema	Das Versionierungsschema ist Teil des konzeptuellen Basisschemas und beschreibt Aspekte der zeitlichen Veränderung der Fachobjekte durch Fortführungen (siehe Abb. 3.4-1). à Objektbehälter à Versionierung	versioning schema
Vorgang	Siehe 3.7.2.1	operation
XML-Schema	Das XML-Schema ist die lexikalische Beschreibung eines Anwendungsschemas auf der Basis von XML (Extensible Markup Language). Auf der Grundlage der im XML-Schema festgelegten Strukturen können XML-Dokumente zum Austausch von Daten geschaffen werden. Vgl. Punkt 10.1.1 und http://www.w3.org/TR/xmlschema-0/ .	XML schema
Zeitstempel	Der Zeitstempel besteht aus Entstehungsdatum und –zeit, welche aus dem Attribut “Lebenszeitintervall” übernommen werden. Er ist als Ergänzung zum Objektidentifikator gedacht und soll bei der Fortführung das gezielte Identifizieren von Objektversionen ermöglichen. Siehe auch 10.1.3.2.	time stamp
zusammengesetzte Objekte (ZUSO)	Zusammengesetzte Objekte werden gebildet, um den Zusammenhang zwischen einer beliebigen Zahl und Mischung semantisch zusammengehörender raumbezogener Elementarobjekte, nicht raumbezogener Elementarobjekte oder zusammengesetzter Objekte herzustellen. Ein zusammengesetztes Objekt muss aber mindestens ein Elementarobjekt als Bestandteil besitzen. à siehe auch “Elementarobjekt”	composed object or complex object

13.2 Abkürzungsverzeichnis

Abkürzung	Langtext
AdV	Arbeitsgemeinschaft der Vermessungsverwaltungen der Länder der Bundesrepublik Deutschland
AFIS	Amtliches Festpunktinformationssystem
ALB	Automatisiertes Liegenschaftsbuch
ALK	Automatisierte Liegenschaftskarte
ALKIS	Amtliches Liegenschaftskataster Informationssystem
ATKIS	Amtliches Topographisch-Kartographisches Informationssystem
ATS	Abstract Test Suite
BKG	Bundesamt für Kartographie und Geodäsie
CD	Committee Draft
CRS	Coordinate Reference System
CSL	Conceptual Schema Language
DB	Datenbank
DBM	Digitales Bildmodell
DGM	Digitales Geländemodell
DLM	Digitales Landschaftsmodell
DOP	Digitales Orthophoto
DTD	Document Type Definition
DTK	Digitale Topographische Karte
DXF	Data Exchange Format
FIS	Fachinformationssystem
GeoInfoDok	Dokumentation zur Modellierung der Geoinformationen des amtlichen Vermessungswesens
GIS	Geo-Informationssystem
GML	Geography Markup Language
ID	Identifikator / Identifier
ISO	International Organization for Standardization
NAS	Normbasierte Austauschschnittstelle
NBA	Nutzerbezogene Bestandsdatenaktualisierung
NREO	Nicht raumbezogenes Elementarobjekt
OGC	Open Geospatial Consortium
OK	Objektartenkatalog
REO	Raumbezogenes Elementarobjekt
SK	Signaturenkatalog
TC	Technical Committee
TK	Topographische Karte
UML	Unified Modeling Language
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
URN	Uniform Resource Name
UUID	Universally Unique Identifier
XML	Extensible Markup Language
ZUSO	Zusammengesetztes Objekt

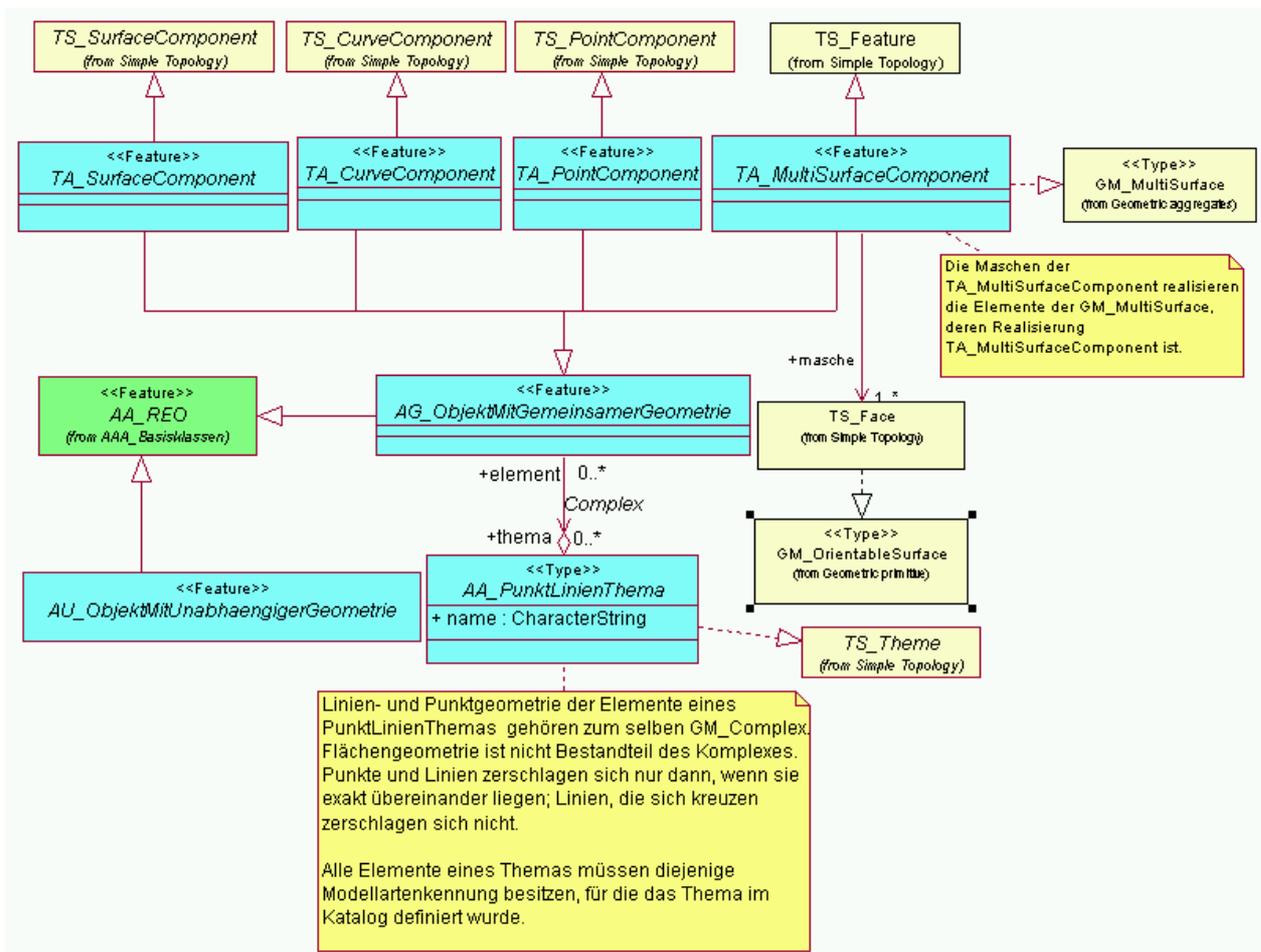
13.3 Abbildungsverzeichnis

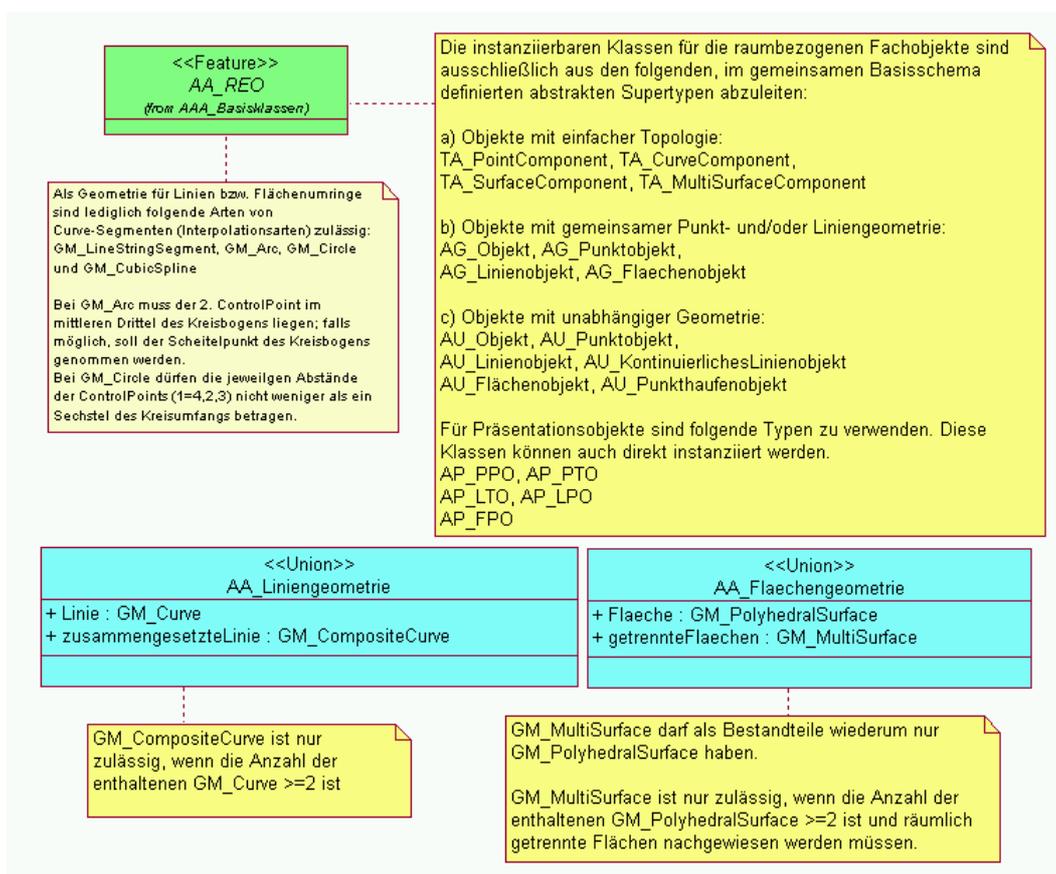
Abbildung 2-1: Gemeinsames AFIS-ALKIS-ATKIS-Referenzmodell 10

Abbildung 3-1: Die Rolle des Anwendungsschemas	13
Abbildung 3-2: Abhängigkeit des AFIS-ALKIS-ATKIS-Anwendungsschemas von den genormten Strukturen aus ISO 19100.....	14
Abbildung 3-3: Verwendete Teile aus der Normfamilie ISO 19100	14
Abbildung 3-4: Die Bestandteile des AFIS-ALKIS-ATKIS-Anwendungsschemas.....	15
Abbildung 3-5: Das Basisschema als Grundlage der Modellierung von anwendungsspezifischen Fachschemata (z.B. AFIS, ALKIS und ATKIS)16	
Abbildung 3-6: Bestandteile des Basisschemas	17
Abbildung 3-7: Modellierung der Basisklassen	21
Abbildung 3-8: Zusammenfassende Darstellung der für AFIS-ALKIS-ATKIS erforderlichen Ergänzungen am genormten Spatial Schema.....	24
Abbildung 3-9: Restriktionen bezüglich der Geometrie und instanziierebare Klassen	25
Abbildung 3-10: Objekte mit gemeinsamer Geometrie.....	27
Abbildung 3-11: Objekte mit unabhängiger Geometrie	28
Abbildung 3-12: Präsentationsobjekte.....	30
Abbildung 3-13: Modellierung der Punktmengenobjekte	35
Abbildung 3-14: Versionierungsschema	40
Abbildung 3-15: Beispiel zur Versionierung nach Änderung von Attributen	43
Abbildung 3-16: Beispiel zur Versionierung nach Änderung von Relationen	44
Abbildung 3-17: Erweiterungen der genormten Struktur der Objektartenkataloge	46
Abbildung 3-18: Prozesse und Daten der Geoinformationen des amtlichen Vermessungswesens.....	47
Abbildung 3-19: Vorgänge im AAA-Anwendungsschema.....	48
Abbildung 3-20: Klassendiagramm "AA_Antrag"	53
Abbildung 3-21: Klassendiagramm "AA_Projektsteuerungskatalog"	54
Abbildung 3-22: Klassendiagramm "AA_Meilenstein"	55
Abbildung 10-1: Einbettung der NAS in Normen und Standards.....	75
Abbildung 10-2: XML - basierende Kodierungsregeln gemäß ISO 19118.....	76
Abbildung 10-3: Automatische Ableitung der NAS-Schemata	77
Abbildung 10-4: Grundstruktur der zentralen XML Schemata der NAS.....	82
Abbildung 10-5: Erläuterung zur Linienteilung.....	92
Abbildung 10-6: Das UML-Paket "NAS-Operationen" im Kontext der Bestandteile des Anwendungsschemas	106
Abbildung 12-1: Das Qualitätssicherungsmodell des AFIS-ALKIS-ATKIS-Projektes ...	131

Anlage: Geometrische Umsetzung der AAA-REO mit GML

Diese Anlage zeigt eine vollständige Auflistung der im AAA-Basischema erlaubten Geometrie- und Topologieelemente mit ihrer jeweiligen Repräsentierung im GML-Profil und damit der NAS.





```

+ TA_SurfaceComponent
|----+ AX_TatsaechlicheNutzung
|   |---- AX_Hafenbecken
|   |---- AX_StehendesGewaesser
|   |---- AX_Meer
|   |---- AX_Fliessgewaesser
|   |---- AX_Wohnbauflaechen
|   |---- AX_IndustrieUndGewerbeflaechen
|   |---- AX_Halde
|   |---- AX_Bergbaubetrieb
|   |---- AX_TagebauGrubeSteinbruch
|   |---- AX_FlaechenBesondererFunktionalerPraegung
|   |---- AX_SportFreizeitUndErholungsflaechen
|   |---- AX_Friedhof
|   |---- AX_FlaechenGemischterNutzung
|   |---- AX_Siedlungsflaechen
|   |---- AX_Landwirtschaft
|   |---- AX_Wald
|   |---- AX_Gehoelz
|   |---- AX_Heide
|   |---- AX_Moor
|   |---- AX_Sumpf
|   |---- AX_UnlandVegetationsloseFlaechen
|   |---- AX_FlaechenZurZeitUnbestimmbar

```

```
| |---- AX_Strassenverkehr
| |---- AX_Platz
| |---- AX_Weg
| |---- AX_Bahnverkehr
| |---- AX_Flugverkehr
| |---- AX_Schiffsverkehr
```

TA_SurfaceComponent wird in GML zu:

```
<xs:element name="position" type="gml:SurfacePropertyType"/>
```

Dies bedeutet: gml:_Surface

Diese SubstitutionGroup hat als Mitglieder (profilbezogen vollständige Aufzählung):

[gml:Surface](#)

```
+ TA_CurveComponent
|---- AX_BesondereFlurstuecksgrenze
|---- AX_Gewaesserachse
|---- AX_Bahnstrecke
|---- AX_Strassenachse
|---- AX_Fahrbahnachse
|---- AX_Fahrwegachse
|---- AX_Gewaesserstationierungsachse
|---- AX_Gebietsgrenze
```

TA_CurveComponent wird in GML zu:

```
<xs:element name="position"><xs:complexType><xs:choice><xs:element
ref="gml:Curve"/><xs:element
ref="gml:CompositeCurve"/></xs:choice></xs:complexType></xs:element>
```

Sprung zu [gml:Curve](#), [gml:CompositeCurve](#)

```
+ TA_PointComponent
|---- AX_PunktortTA
```

'PunktortTA' ist ein Punktort, der in der Flurstücksgrenze liegt und einen Grenzpunkt verortet.

TA_PointComponent wird in GML zu:

```
<xs:element name="position" type="gml:PointPropertyType"/>
```

Dies bedeutet: gml:Point mit Attribut gml:pos.

```
+ TA_MultiSurfaceComponent
|---- AX_Flurstueck
|---- AX_Bodenschaetzung
|----+ AX_Gebiet
| |---- AX_KommunalesGebiet
| |---- AX_Gebiet_Kreis
| |---- AX_Gebiet_Regierungsbezirk
| |---- AX_Gebiet_Bundesland
| |---- AX_Gebiet_Nationalstaat
```

TA_MultiSurfaceComponent wird in GML zu:

```
<xs:element name="position"><xs:complexType><xs:choice><xs:element
ref="gml:Surface"/><xs:element
ref="gml:MultiSurface"/></xs:choice></xs:complexType></xs:element>
```

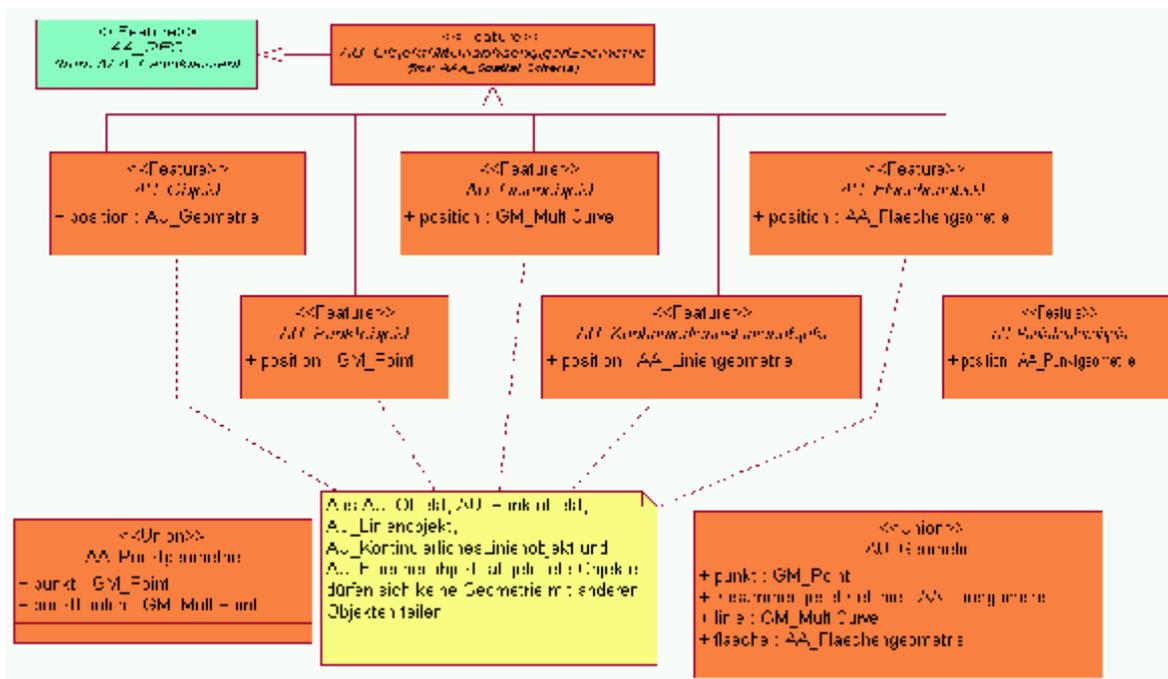
Das führt zu gml:Surface oder zu gml:MultiSurface

Sprung zu [gml:Surface](#)

gml:Multisurface führt über [1..n] Attribute gml:surfaceMember zu Mitgliedern der SubstitutionGroup gml:_Surface

Die SubstitutionGroup gml:_Surface hat als Mitglieder (profilbezogen vollständige Aufzählung): gml:Surface

Hinweis: gml:MultiSurface ist nur bei mehreren getrennten Flächen erlaubt! Bei einer einzigen Fläche ist stets gml:Surface zu verwenden.



- + AU_Objekt
 - |---- AX_Transportanlage
 - |---- AX_Strassenverkehrsanlage
 - |---- AX_WegPfadSteig
 - |---- AX_Bahnverkehrsanlage
 - |---- AX_Gleis
 - |---- AX_Flugverkehrsanlage
 - |---- AX_EinrichtungenFuerDenSchiffsverkehr
 - |---- AX_Ortslage
 - |---- AX_Hafen
 - |---- AX_Schleuse
 - |---- AX_Grenzuebergang
 - |---- AX_Testgelaende
 - |---- AX_UntergeordnetesGewaesser
 - |---- AX_Gewaessermerkmal
 - |---- AX_Vegetationsmerkmal
 - |---- AX_FelsenFelsblockFelsnadel
 - |---- AX_MusterLandesmusterUndVergleichsstueck
 - |---- AX_Landschaft
 - |---- AX_KleinraeumigerLandschaftsteil
 - |---- AX_Gewann
 - |---- AX_Insel

AU_Objekt wird in GML zu:

```
<xs:element name="position"><xs:complexType><xs:choice><xs:element
ref="gml:Point"/><xs:element ref="gml:Curve"/><xs:element
ref="gml:CompositeCurve"/><xs:element ref="gml:MultiCurve"/><xs:element
ref="gml:Surface"/><xs:element
ref="gml:MultiSurface"/></xs:choice></xs:complexType></xs:element>
```

Siehe Abbildung zur unabhängigen Geometrie zur Weiterverfolgung dieser Verzweigungen.

```

+ AU_Punktobjekt
|---- AP_PTO
|---- AX_HeilquelleGasquelle
|---- AX_Wasserspiegelhoehe
|---- AX_Nullpunkt
|---- AX_Hoehleneingang
|---- AX_MarkanterGelaendepunkt
|---- AX_BesondererHoechenpunkt
|---- AX_GrablochDerBodenschaetzung
|---- AX_Wohnplatz
|---- AX_PunktortAU
|---- AX_GeoreferenzierteGebaueadresse

```

'PunktortAU' ist ein Punktort mit unabhängiger Geometrie ohne Zugehörigkeit zu einem Geometriethema. Er kann zu ZUSOs der folgenden Objektarten gehören: Grenzpunkt, Besonderer Gebäudepunkt, Besonderer Bauwerkspunkt, Aufnahmepunkt, Sicherungspunkt, Sonstiger Vermessungspunkt, Besonderer topographischer Punkt, Lagefestpunkt, Höhenfestpunkt, Schwerefestpunkt, Referenzstationspunkt.

AU_Punktobjekt wird in GML zu:

```
<xs:element name="position" type="gml:PointPropertyType"/>
```

gml:PointPropertyType bedeutet: gml:Point mit Attribut gml:pos.

```

+ AU_Linienobjekt
|---- AP_LPO
|---- AX_SeilbahnSchwebbahn
|---- AX_Hoehenlinie
|---- AX_Einschnitt
|---- AX_StrukturiertErfassteGelaendepunkte
|---- AX_Gewaesserbegrenzung
|---- AX_Geripplinie
|---- AX_AbgeleiteteHoechenlinie
|---- AX_Gebaueausgestaltung
|---- AX_TopographischeLinie

```

AU_Linienobjekt wird in GML zu:

```
<xs:element name="position" type="gml:MultiCurvePropertyType"/>
```

Dies führt zu gml:MultiCurve. gml:MultiCurve hat [1..n] Attribute curveMember, welche zur SubstitutionGroup gml:_Curve führen.

Diese SubstitutionGroup hat als Mitglieder (profilbezogen vollständige Aufzählung):

[gml:Curve](#), [gml:CompositeCurve](#)

Hinweis: Hier ist gml:LineString durch GID Hauptdokument, 10.1.6 außen vor.

+ AU_KontinuierlichesLinienobjekt
 |---- AP_LTO
 |---- AX_Leitung
 |---- AX_Abschnitt
 |---- AX_Ast

AU_KontinuierlichesLinienobjekt wird in GML zu:

```
<xs:element name="position"><xs:complexType><xs:choice><xs:element
ref="gml:Curve"/><xs:element
ref="gml:CompositeCurve"/></xs:choice></xs:complexType></xs:element>
```

Sprung zu [gml:Curve](#), [gml:CompositeCurve](#)

+ AU_Flaechenobjekt
 |---- AP_FPO
 |---- AA_Antragsgebiet
 |---- AX_Polder
 |---- AX_Duene
 |---- AX_Soll
 |---- AX_Aussparungsflaeche
 |---- AX_Baublock
 |---- AX_Kondominium
 |---- AX_HistorischesFlurstueck

AU_Flaechenobjekt wird in GML zu:

```
<xs:element name="position"><xs:complexType><xs:choice><xs:element
ref="gml:_Surface"/><xs:element
ref="gml:MultiSurface"/></xs:choice></xs:complexType></xs:element>
```

Das führt zu gml:Surface oder zu gml:MultiSurface

Sprung zu [gml:Surface](#)

gml:MultiSurface führt über [1..n] Attribute gml:surfaceMember zu Mitgliedern der SubstitutionGroup gml:_Surface

Die SubstitutionGroup gml:_Surface hat als Mitglieder (profilbezogen vollständige Aufzählung): [gml:Surface](#)

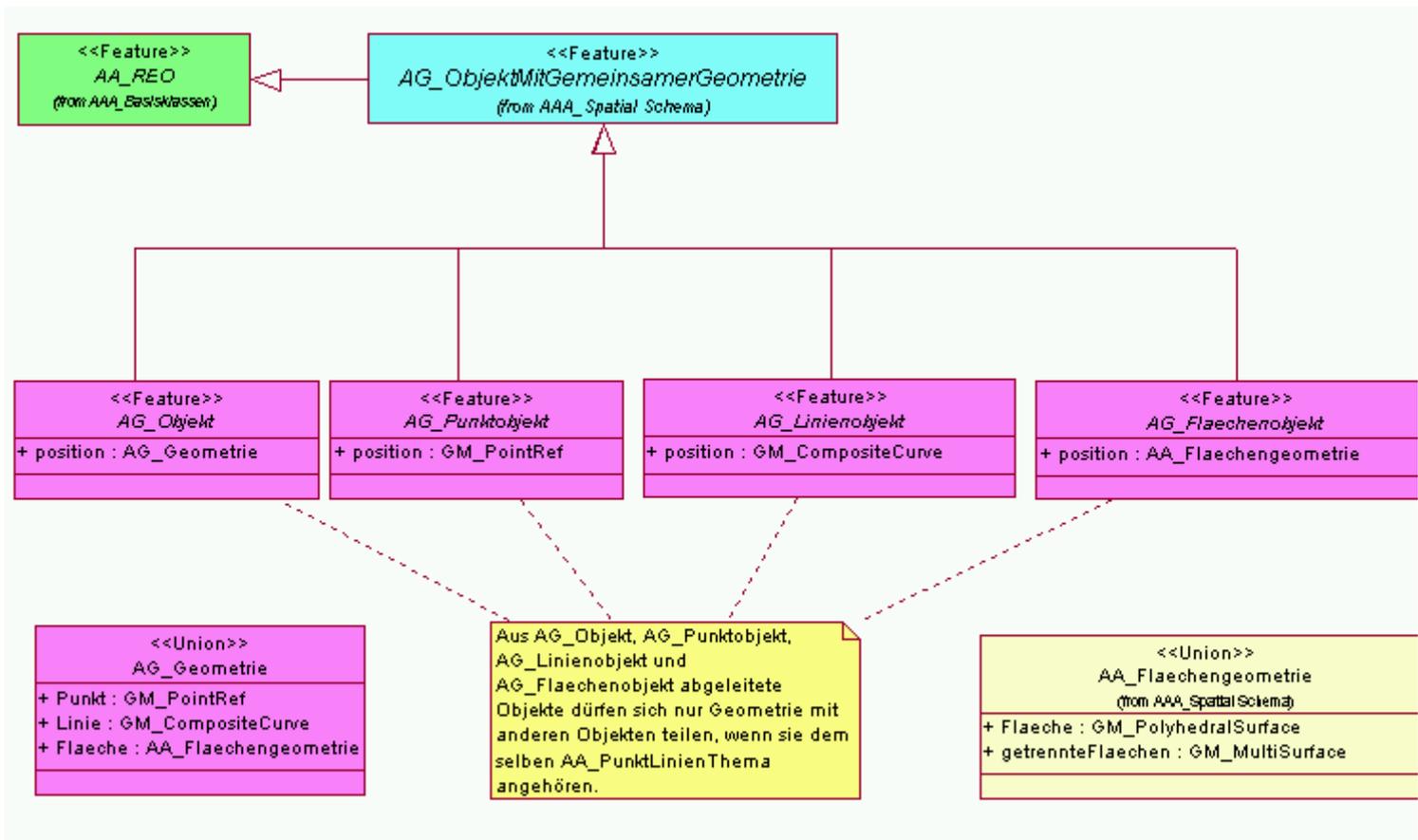
+ AU_Punkthaufenobjekt
 |---- AP_PPO

AU_Punkthaufenobjekt wird in GML zu:

```
<xs:element name="position"><xs:complexType><xs:choice><xs:element
ref="gml:Point"/><xs:element
ref="gml:MultiPoint"/></xs:choice></xs:complexType></xs:element>
```

gml:Point bedeutet: gml:Point mit Attribut gml:pos.

gml:MultiPoint führt über [1..n] Attribute gml:pointMember zu gml:Point



```

+ AG_Objekt
|---- AX_Gebaeude
|---- AX_Turm
|---- AX_BauwerkOderAnlageFuerIndustrieUndGewerbe
|---- AX_VorratsbehaelterSpeicherbauwerk
|---- AX_BauwerkOderAnlageFuerSportFreizeitUndErholung
|---- AX_HistorischesBauwerkOderHistorischeEinrichtung
|---- AX_SonstigesBauwerkOderSonstigeEinrichtung
|---- AX_EinrichtungInOeffentlichenBereichen
|---- AX_BauwerkImVerkehrsbereich
|---- AX_BauwerkImGewaesserbereich
|---- AX_SchifffahrtslinieFaehrverkehr
|---- AX_DammWallDeich
|---- AX_KlassifizierungNachStrassenrecht
|---- AX_AndereFestlegungNachStrassenrecht
|---- AX_NaturUmweltOderBodenschutzrecht
|---- AX_Denkmalschutzrecht

```

AG_Objekt wird in GML zu:

```

<xs:element name="position"><xs:complexType><xs:choice><xs:element
ref="gml:Point"/><xs:element ref="gml:CompositeCurve"/><xs:element
ref="gml:Surface"/><xs:element
ref="gml:MultiSurface"/></xs:choice></xs:complexType></xs:element>

```

Dies führt zu gml:Point, gml:CompositeCurve, gml:Surface oder zu gml:MultiSurface.

gml:Point bedeutet: gml:Point mit Attribut gml:pos.

Sprung zu [gml:CompositeCurve](#)

Sprung zu [gml:Surface](#)

gml:Multisurface führt über [1..n] Attribute gml:surfaceMember zu Mitgliedern der SubstitutionGroup gml:_Surface

Die SubstitutionGroup gml:_Surface hat als Mitglieder (profilbezogen vollständige Aufzählung): [gml:Surface](#)

```

+ AG_Punktobjekt

```

```

|---- AX_PunktortAG

```

'PunktortAG' ist ein Punktort mit redundanzfreier Geometrie (Besonderer Gebäudepunkt, Besonderer Bauwerkspunkt) innerhalb eines Geometriethemas.

AG_Punktobjekt wird in GML zu:

```

<xs:element name="position" type="gml:PointPropertyType"/>

```

Dies bedeutet: gml:Point mit Attribut gml:pos.

+ AG_Linienobjekt
|---- AX_BesondereGebaueudelinie
|---- AX_Firstlinie
|---- AX_Gelaendekante
|---- AX_Sickerstrecke

AG_Linienobjekt wird in GML zu:

```
<xs:element name="position"><xs:complexType><xs:sequence><xs:element  
ref="gml:CompositeCurve"/></xs:sequence></xs:complexType></xs:element>  
Sprung zu gml:Curve, gml:CompositeCurve
```

+ AG_Flaechenobjekt
|---- AX_Bauteil
|---- AX_Boeschungsflaeche
|---- AX_Bewertung
|---- AX_Tagesabschnitt
|---- AX_KlassifizierungNachWasserrecht
|---- AX_AndereFestlegungNachWasserrecht
|---- AX_BauRaumOderBodenordnungsrecht
|---- AX_Forstrecht
|---- AX_SonstigesRecht
|---- AX_Schutzzone

AG_Flaechenobjekt wird in GML zu:

```
<xs:element name="position"><xs:complexType><xs:choice><xs:element  
ref="gml:Surface"/><xs:element  
ref="gml:MultiSurface"/></xs:choice></xs:complexType></xs:element>
```

Das führt zu gml:Surface oder zu gml:MultiSurface

Sprung zu [gml:Surface](#)

gml:Multisurface führt über [1..n] Attribute gml:surfaceMember zu Mitgliedern der SubstitutionGroup gml:_Surface

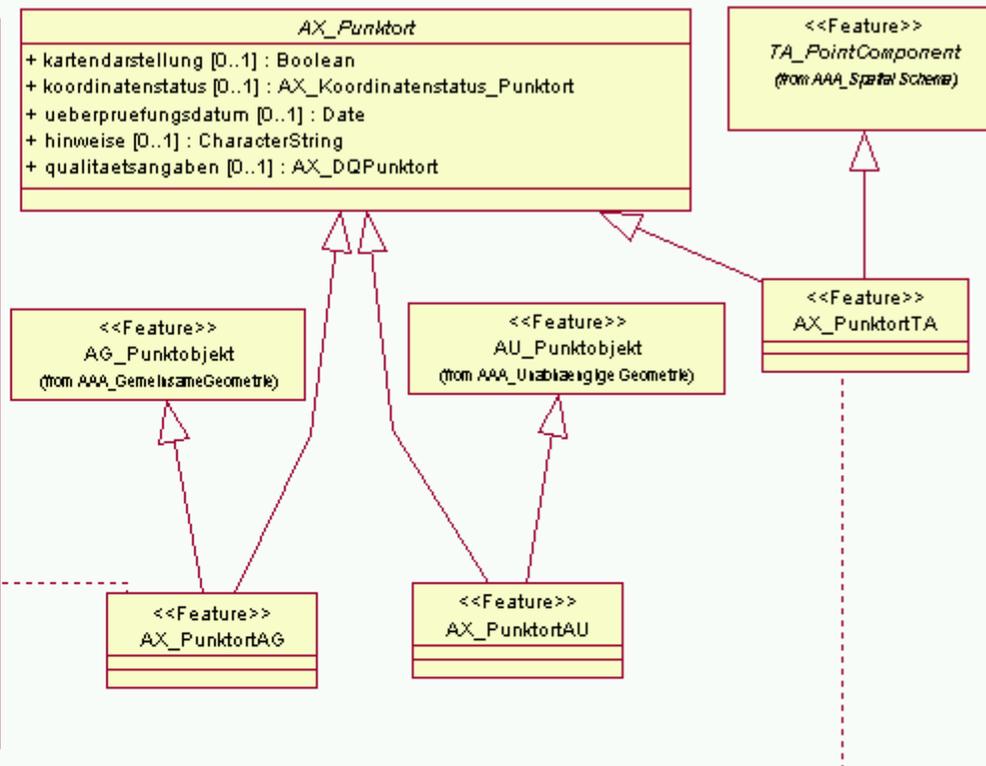
Die SubstitutionGroup gml:_Surface hat als Mitglieder (profilbezogen vollständige Aufzählung): [gml:Surface](#)

```

context AX_PunktortAG inv:
self.kartendarstellung = true

context po: AX_PunktortAG inv:
AX_Gebaeude.allInstances->exists(
gposition.generator->exists( os : GM_OrientableSurface |
os.boundary().exterior->exists( r : GM_Ring |
r->exists( oc : GM_OrientableCurve |
oc.primitive.segment->exists( cs : GM_CurveSegment |
cs.controlPoint->exists( p : GM_Point | p.composite.contains(po.gposition) )))
or
os.boundary().interior->exists( r : GM_Ring |
r->exists( oc : GM_OrientableCurve |
oc.primitive.segment->exists( cs : GM_CurveSegment |
cs.controlPoint->exists( p : GM_Point | p.composite.contains(po.gposition) ))))
or
AX_Bauteil.allInstances.exists(
gposition.generator->exists( os : GM_OrientableSurface |
os.boundary().exterior->exists( r : GM_Ring |
r->exists( oc : GM_OrientableCurve |
oc.primitive.segment->exists( cs : GM_CurveSegment |
cs.controlPoint->exists( p : GM_Point | p.composite.contains(po.gposition) ))))
or
os.boundary().interior->exists( r : GM_Ring |
r->exists( oc : GM_OrientableCurve |
oc.primitive.segment->exists( cs : GM_CurveSegment |
cs.controlPoint->exists( p : GM_Point | p.composite.contains(po.gposition) ))))

```



```

<<CodeList>>
AX_Koordinatenstatus_Punktort

+ Amtliche Koordinaten bzw. amtliche Höhe = 1000
+ Weitere gültige Koordinaten bzw. weitere gültige Höhe = 2000
+ Vorläufige Koordinaten bzw. vorläufige Höhe = 3000
+ Zu keiner Zeit gültig gewesene Koordinaten bzw. Höhe = 4000
+ Historische (nicht mehr gültige) Koordinaten bzw. Höhe = 5000
+ Koordinaten bzw. Höhe, die sich als fehlerhaft herausgestellt haben = 5100

```

```

context AX_PunktortTA inv:
self.kartendarstellung = true

context po: AX_PunktortTA inv:
AX_Flurstueck.allInstances->exists(
masche->exists( f : TS_Face |
f.boundary->exists( de : TS_DirectedEdge |
de.topo.boundary[1].topo.composite->exists( obj | obj = po ) or
de.topo.boundary[2].topo.composite->exists( obj | obj = po )))

```

gml:Surface hat als Attribut

[1] patches

patches hat den type gml:SurfacePatchArrayPropertyType

Dieser Typ führt auf eine Folge von Elementen der SubstitutionGroup gml:_SurfacePatch in der Häufigkeit [0..n].

gml:_SurfacePatch umfaßt (profilbezogen vollständige Aufzählung):

[gml:PolygonPatch](#)

gml:PolygonPatch A PolygonPatch is a surface patch that is defined by a set of boundary curves and an underlying surface to which these curves adhere. The curves are coplanar and the polygon uses planar interpolation in its interior. Implements GM_Polygon of ISO 19107.

[Zu Instanzen einer gml:Surface.](#)

gml:PolygonPatch hat als Elemente

[1] exterior

exterior führt auf die SubstitutionGroup gml:_Ring

Die SubstitutionGroup gml:_Ring umfaßt (profilbezogen vollständige Aufzählung):

gml:Ring

hat als Attribut [1..n] curveMember.

Diese führen auf die SubstitutionGroup gml:_Curve

Die SubstitutionGroup gml:_Curve umfaßt (profilbezogen vollständige Aufzählung):

[gml:Curve](#), [gml:CompositeCurve](#)

[0..n] interior

interior führt auf die SubstitutionGroup gml:_Ring

Die SubstitutionGroup gml:_Ring umfaßt (profilbezogen vollständige Aufzählung):

gml:Ring

hat als Attribut [1..n] curveMember.

Diese führen auf die SubstitutionGroup gml:_Curve

Die SubstitutionGroup gml:_Curve umfaßt (profilbezogen vollständige Aufzählung):

[gml:Curve](#), [gml:CompositeCurve](#)

sowie [1] Attribut interpolation mit dem festen Wert "planar".

Hinweis: **Geometrieteilung** kann nicht auf Segmentebene sondern nur auf Ebene von gml:_Curve stattfinden.

[Zu Instanzen eines gml:PolygonPatch.](#)

gml:Polygon hat als Attribute

[1] exterior

exterior führt auf die SubstitutionGroup gml:_Ring

Die SubstitutionGroup gml:_Ring umfaßt (profilbezogen vollständige Aufzählung):

gml:LinearRing

gml:LinearRing hat mindestens vier gml:pos Attribute.

gml:Ring

hat als Attribut [1..n] curveMember.

Diese führen auf die SubstitutionGroup gml:_Curve

Die SubstitutionGroup gml:_Curve umfaßt (profilbezogen vollständige Aufzählung):

[gml:Curve](#), [gml:CompositeCurve](#)

[0..n] interior

interior führt auf die SubstitutionGroup gml:_Ring

Die SubstitutionGroup gml:_Ring umfaßt (profilbezogen vollständige Aufzählung):

gml:LinearRing

gml:LinearRing hat mindestens vier gml:pos Attribute.

gml:Ring

hat als Attribut [1..n] curveMember.

Diese führen auf die SubstitutionGroup gml:_Curve

Die SubstitutionGroup gml:_Curve umfaßt (profilbezogen vollständige Aufzählung):

[gml:Curve](#), [gml:CompositeCurve](#)

Hinweis: gml:Polygon darf nicht in REO-Geometrie der NAS verwendet werden, allenfalls in Filterausdrücken.

[Zur Instanz eines gml:Polygon](#)**gml:LineString** hat als Attribute:

[2..n] pos

gml:LineString darf nicht in REO-Geometrie der NAS verwendet werden, allenfalls in Filterausdrücken.

[Zur Instanz eines gml:LineString](#)

gml:Curve hat als Attribut

[1] segments

segments kann als Werte eine beliebige Folge der folgenden Elemente haben (profilbezogen vollständige Aufzählung):

gml:LineStringSegment

gml:Arc (als Stellvertreter des abstrakten Elements gml:ArcString!)

gml:Circle (als Angehöriger der SubstitutionGroup gml:Arc)

gml:CubicSpline

GML-Profil der NAS besagt hierzu in 10.1.6 Nr. 2:

Bei der Darstellung von GM_Curve in AAA-Daten wird die Verwendung von gml:Curve mit genau einem gml:LineStringSegment vorgeschrieben, sofern die GM_Curve zwischen allen Stützpunkten linear interpoliert wird. gml:LineString darf in diesen Fällen nicht verwendet werden!

[Zur Instanz einer gml:Curve](#)

gml:LineStringSegment hat als Attribute

[2..n] pos (Interpolation ist fixiert auf "linear")

[Zur Instanz eines gml:LineStringSegment](#)

gml:ArcString ist ein Liniensegment und abstrakt und daher nicht direkt verwendbar. Jedoch wurde dem

gml:Arc die Möglichkeit eingeräumt, an die Stelle eines ArcString zu treten und gml:Circle wiederum an die Stelle eines gml:Arc.

gml:Arc hat als Attribute

[3] pos

Bei GM_Arc muss der 2. ControlPoint im mittleren Drittel des Kreisbogens liegen; falls möglich, soll der Scheitelpunkt des Kreisbogens genommen werden.

gml:Circle hat als Attribute

[3] pos

Die ControlPoints sollen über den Kreis verteilt liegen.

gml:CubicSpline hat als Attribute

[3..n] pos (Interpolation ist fixiert auf "cubic spline")

[1] vectorAtStart

[1] vectorAtEnd

gml:CompositeCurve hat als Attribute

[1..n] curveMember. Inhalt jeweils ein Element der SubstitutionGroup gml:_Curve
Diese SubstitutionGroup hat als Mitglieder (profilbezogen vollständige Aufzählung): [gml:Curve](#), [gml:CompositeCurve](#)

gml:CompositeCurve ist nur zulässig, wenn die Anzahl der enthaltenen _Curves-Repräsentanten größer gleich zwei ist!

[Zur Instanz einer gml:CompositeCurve](#)

Hinweis: Das GML-Attribut srsName fand bei den Beispielen bewusst keine Berücksichtigung, für jede NAS-Geometrie muss jedoch das zugehörige Koordinatenreferenzsystem mindestens auf der höchsten GML-Ebene und im Auftrag spezifiziert sein.

Instanzen einer gml:Surface

- a) Inselartige Fläche, eine einzige Liniengeometrie linearer Interpolation

```
<gml:Surface srsName="urn:adv:crs:DE_DHDN_3GK3">
  <gml:patches>
    <gml:PolygonPatch>
      <gml:exterior>
        <gml:Ring>
          <gml:curveMember>
            <gml:Curve>
              <gml:segments>
                <gml:LineStringSegment>
                  <gml:pos>350606.989 5533364.288</gml:pos>
                  <gml:pos>350577.082 5533356.950</gml:pos>
                  <gml:pos>350553.704 5533364.875</gml:pos>
                  <gml:pos>350606.989 5533364.288</gml:pos>
                </gml:LineStringSegment>
              </gml:segments>
            </gml:Curve>
          </gml:curveMember>
        </gml:Ring>
      </gml:exterior>
    </gml:PolygonPatch>
  </gml:patches>
</gml:Surface>
```

- b) Fläche, die sich Geometrie (gelb, grün, türkis, rotbraun und rosa eingefärbte Repräsentanten von _Curve) mit mehreren anderen Objekten teilt und aus zwei getrennt liegenden Teilflächen besteht.

```
<gml:MultiSurface>
  <gml:surfaceMember>
    <gml:Surface>
      <gml:patches>
        <gml:PolygonPatch srsName="urn:adv:crs:DE_DHDN_3GK3">
          <gml:exterior>
            <gml:Ring>
              <gml:curveMember>
                <gml:Curve>
                  <gml:segments>
                    <gml:LineStringSegment>
                      <gml:pos>350606.989 5533364.288</gml:pos>
                      <gml:pos>350577.082 5533356.950</gml:pos>
                    </gml:LineStringSegment>
                    <gml:LineStringSegment>
                      <gml:pos>350577.082 5533356.950</gml:pos>
```

```
<gml:pos>350553.704 5533364.875</gml:pos>
</gml:LineStringSegment>
<gml:Arc>
  <gml:pos>350553.704 5533364.875</gml:pos>
  <gml:pos>350545.129 5533368.066</gml:pos>
  <gml:pos>350548.320 5533376.641</gml:pos>
</gml:Arc>
</gml:segments>
</gml:Curve>
</gml:curveMember>
<gml:curveMember>
  <gml:Curve>
    <gml:segments>
      <gml:LineStringSegment>
        <gml:pos>350548.320 5533376.641</gml:pos>
        <gml:pos>350562.100 5533365.299</gml:pos>
        <gml:pos>350575.795 5533362.317</gml:pos>
      </gml:LineStringSegment>
    </gml:segments>
  </gml:Curve>
</gml:curveMember>
<gml:curveMember>
  <gml:CompositeCurve>
    <gml:curveMember>
      <gml:Curve>
        <gml:segments>
          <gml:LineStringSegment>
            <gml:pos>350575.795 5533362.317</gml:pos>
            <gml:pos>350602.304 5533368.809</gml:pos>
            <gml:pos>350605.944 5533369.694</gml:pos>
          </gml:LineStringSegment>
        </gml:segments>
      </gml:Curve>
    </gml:curveMember>
    <gml:curveMember>
      <gml:Curve>
        <gml:segments>
          <gml:LineStringSegment>
            <gml:pos>350605.944 5533369.694</gml:pos>
            <gml:pos>350637.258 5533374.706</gml:pos>
            <gml:pos>350646.578 5533376.195</gml:pos>
          </gml:LineStringSegment>
        </gml:segments>
      </gml:Curve>
    </gml:curveMember>
  </gml:CompositeCurve>
</gml:curveMember>
<gml:curveMember>
  <gml:Curve>
    <gml:segments>
      <gml:LineStringSegment>
```

```

    <gml:pos>350646.578 5533376.195</gml:pos>
  <gml:pos>350674.548 5533373.482</gml:pos>
  <gml:pos>350698.799 5533368.356</gml:pos>
  <gml:pos>350715.506 5533364.083</gml:pos>
  <gml:pos>350731.341 5533360.025</gml:pos>
  <gml:pos>350746.098 5533354.579</gml:pos>
  <gml:pos>350753.466 5533351.861</gml:pos>
  <gml:pos>350773.202 5533339.498</gml:pos>
  <gml:pos>350780.749 5533332.990</gml:pos>
  <gml:pos>350792.757 5533322.859</gml:pos>
  <gml:pos>350810.440 5533305.764</gml:pos>
  <gml:pos>350825.882 5533286.907</gml:pos>
  <gml:pos>350837.856 5533267.468</gml:pos>
  <gml:pos>350840.348 5533262.724</gml:pos>
  <gml:pos>350855.370 5533234.144</gml:pos>
  <gml:pos>350859.100 5533227.047</gml:pos>
  <gml:pos>350874.548 5533197.649</gml:pos>
  <gml:pos>350865.874 5533202.262</gml:pos>
  <gml:pos>350833.030 5533264.699</gml:pos>
  <gml:pos>350821.417 5533283.662</gml:pos>
  <gml:pos>350806.235 5533301.979</gml:pos>
  <gml:pos>350789.163 5533318.859</gml:pos>
  <gml:pos>350769.945 5533334.914</gml:pos>
  <gml:pos>350750.935 5533346.907</gml:pos>
  <gml:pos>350729.689 5533354.815</gml:pos>
  <gml:pos>350697.492 5533362.983</gml:pos>
  <gml:pos>350673.787 5533367.967</gml:pos>
  <gml:pos>350646.746 5533370.652</gml:pos>
  <gml:pos>350606.989 5533364.288</gml:pos>
</gml:LineStringSegment>
</gml:segments>
</gml:Curve>
</gml:curveMember>
</gml:Ring>
</gml:exterior>
</gml:PolygonPatch>
</gml:patches>
</gml:Surface>
</gml:surfaceMember>
<!-- Hier kommt nun die zweite Teilfläche -->
<gml:surfaceMember>
<gml:Surface srsName="urn:adv:crs:DE_DHDN_3GK3">
<gml:patches>
<gml:PolygonPatch>
<gml:exterior>
<gml:Ring>
  <gml:curveMember>
    <gml:CompositeCurve>
      <gml:curveMember>
        <gml:Curve>
          <gml:segments>

```

```

    <gml:LineStringSegment>
      <gml:pos>350975.795 5533362.317</gml:pos>
      <gml:pos>351002.304 5533368.809</gml:pos>
      <gml:pos>351005.944 5533369.694</gml:pos>
    </gml:LineStringSegment>
  </gml:segments>
</gml:Curve>
</gml:curveMember>
<gml:curveMember>
  <gml:Curve>
    <gml:segments>
      <gml:LineStringSegment>
        <gml:pos>351005.944 5533369.694</gml:pos>
        <gml:pos>351037.258 5533374.706</gml:pos>
        <gml:pos>351046.578 5533376.195</gml:pos>
        <gml:pos>351100.508 5533380.185</gml:pos>
        <gml:pos>350975.795 5533362.317</gml:pos>
      </gml:LineStringSegment>
    </gml:segments>
  </gml:Curve>
</gml:curveMember>
</gml:CompositeCurve>
</gml:curveMember>
</gml:Ring>
</gml:exterior>
</gml:PolygonPatch>
</gml:patches>
</gml:Surface>
</gml:surfaceMember>
</gml:MultiSurface>

```

Instanz eines gml:Polygon

- Fläche einer einzigen Liniengeometrie linearer Interpolation (nur in Filtern verwendbar!!!)

```

<gml:Polygon srsName="urn:adv:crs:DE_DHDN_3GK3">
  <gml:exterior>
    <gml:LinearRing>
      <gml:pos>350606.989 5533364.288</gml:pos>
      <gml:pos>350577.082 5533356.950</gml:pos>
      <gml:pos>350553.704 5533364.875</gml:pos>
      <gml:pos>350606.989 5533364.288</gml:pos>
    </gml:LinearRing>
  </gml:exterior>
</gml:Polygon>

```

Instanz eines gml:LineString (nur in Filtern verwendbar!!!)

```

<gml:LineString srsName="urn:adv:crs:DE_DHDN_3GK3">
  <gml:pos>350605.944 5533369.694</gml:pos>

```

```
<gml:pos>350637.258 5533374.706</gml:pos>
  <gml:pos>350646.578 5533376.195</gml:pos>
</gml:LineString>
```

Instanz einer gml:Curve

```
<gml:Curve srsName="urn:adv:crs:DE_DHDN_3GK3">
  <gml:segments>
    <gml:LineStringSegment>
      <gml:pos>350606.989 5533364.288</gml:pos>
      <gml:pos>350577.082 5533356.950</gml:pos>
      <gml:pos>350553.704 5533364.875</gml:pos>
    </gml:LineStringSegment>
    <gml:Arc>
      <gml:pos>350553.704 5533364.875</gml:pos>
      <gml:pos>350545.129 5533368.066</gml:pos>
      <gml:pos>350548.320 5533376.641</gml:pos>
    </gml:Arc>
  </gml:segments>
</gml:Curve>
```

Instanz eines gml:LineStringSegment: Siehe [innerhalb der gml:Curve](#)

```
<gml:Curve srsName="urn:adv:crs:DE_DHDN_3GK3">
  <gml:segments>
    <gml:LineStringSegment>
      <gml:pos>350606.989 5533364.288</gml:pos>
      <gml:pos>350577.082 5533356.950</gml:pos>
      <gml:pos>350553.704 5533364.875</gml:pos>
    </gml:LineStringSegment>
  </gml:segments>
</gml:Curve>
```

Instanz einer gml:CompositeCurve

```
<gml:CompositeCurve srsName="urn:adv:crs:DE_DHDN_3GK3">
  <gml:curveMember>
    <gml:Curve>
      <gml:segments>
        <gml:LineStringSegment>
          <gml:pos>350975.795 5533362.317</gml:pos>
          <gml:pos>351002.304 5533368.809</gml:pos>
          <gml:pos>351005.944 5533369.694</gml:pos>
        </gml:LineStringSegment>
      </gml:segments>
    </gml:Curve>
  </gml:curveMember>
  <gml:curveMember>
    <gml:Curve>
      <gml:segments>
        <gml:LineStringSegment>
          <gml:pos>351005.944 5533369.694</gml:pos>
          <gml:pos>351037.258 5533374.706</gml:pos>
          <gml:pos>351046.578 5533376.195</gml:pos>
          <gml:pos>351100.508 5533380.185</gml:pos>
          <gml:pos>350975.795 5533362.317</gml:pos>
        </gml:LineStringSegment>
      </gml:segments>
    </gml:Curve>
  </gml:curveMember>
</gml:CompositeCurve>
```