

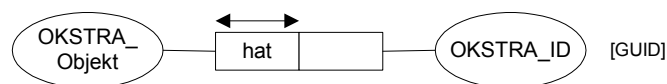


Zeitraum			D020.doc
Datum	Dok.	Oks.	Beschreibung der Änderungen
27.11.2008	1.013	1.013	Einführung der Objektarten "Verkehrslage" und "Verkehrsstau" gemäß N0094 Überarbeitung der Zeitangaben gemäß N0108 Überarbeitung des Schemas "Strassenzustandsdaten" gemäß N0107 Erweiterung der Informationen zu Straßenachsen gemäß N0109 Erweiterung der Modellierung von Straßenbaumaßnahmen gemäß N0104 Entfernung der Objektart "allgemeine_Eigenschaften" gemäß N0106
09.10.2007	1.012	1.012	Einführung eines Metamodells gemäß N0092 Entfernung von Objektarten zum Dynamischen Querprofil (geometrisches Modell) gemäß N0088 Entfernung der Schemata „Ingenieurbauwerke“, „Projektressourcen“ und „Ausstattung“ gemäß N0093 Überarbeitung des Objektes Böschung gemäß N0085 Entfernung von Längenangaben bei Strings gemäß N0090 Einführung einer Richtungsangabe für Streckenobjekte gemäß N0091 Erweiterung des OKSTRA um streckenbezogene Unfallkenngrößen gemäß N0082 Anpassung an die neue ASB, Teilsystem Bestandsdaten gemäß N0074 Entfernung der örtlichen und der verwaltungstechnischen Zuordnung gemäß N0100 Modellierung von Straßenbaumaßnahmen gemäß N0099 Auflösung des direct_positioning_Schemas Einführung des Präfix „S_“ für Schemanamen
02.08.2006	1.011	1.011	Ergänzung von RAS-Verm-Objekten gemäß N0071 Einführung OKSTRA-ID gemäß N0073 Einführung des Schemas Liegenschaftsverwaltung gemäß N0080
09.09.2005	1.010	1.010	Definition einer abstrakten Mixin-Klasse „Objekt_mit_ID“ für die Anbindung der (externen) Objekt-ID per Vererbung
28.05.2004	1.006	1.009	Anbindung Schema Kreuzungen gemäß ASB-Netzdaten, Stand September 2002

			Aufnahme des Dokuments aus Schema Administration
03.09.2003	1.005	1.008	Ergänzung der Währungsangabe (DM/EURO) Ergänzung Objekt_Id, geom. Ausprägung, Foto und Sensor
31.05.2002	1.004	1.007	Erweiterung der Wertebereiche von Meter, Kilometer und Jahr
12.09.2001	1.003	1.004	Ergänzung der Typen Währung, Kubikzentimeter, Promille und Anzahl_einstellig
11.12.2000	1.002	1.002	passive Versionierung durch Verschiebung der Objekte oertliche_Zuordnung, oertliche_Zuordnung_SP, nicht_nach_ASB_klass_Straße in Schema Straßennetz; verwaltungstechn_Zuordnung in Schema Administration
29.10.2000	1.001	1.001	passive Versionierung durch Ergänzung der Schemata "allgemeine Geometrieobjekte" und "Grunderwerbsverzeichnis"
15.10.1999	1.000	1.000	Erste Version des OKSTRA verabschiedet.

## OKSTRA\_Objekt

Alle OKSTRA-Objektarten (mit Ausnahme der Schlüsseltabellen und der konzeptionellen Objektarten) erben direkt oder mittelbar vom abstrakten Supertype *OKSTRA\_Objekt*. Das *OKSTRA\_Objekt* besitzt das optionale Attribut „OKSTRA\_ID“; dabei handelt es sich um einen Globally Unique Identifier (GUID) mit einer Länge von 128 Bit (entsprechend 32 Zeichen).



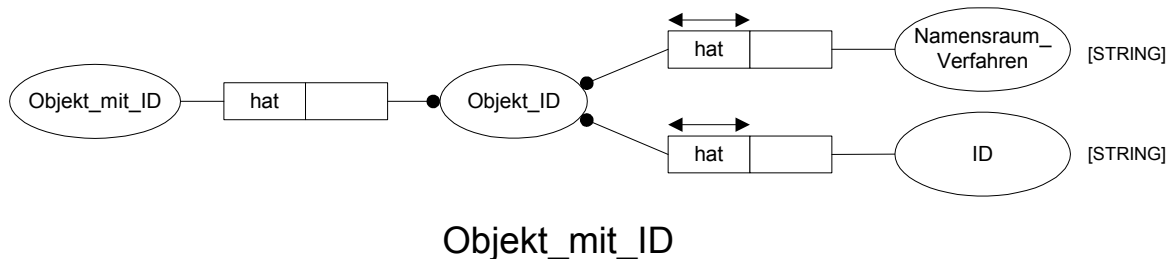
### OKSTRA\_Objekt

Innerhalb einer GUID sind folgende Zeichen zulässig: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. Der Vorteil von GUIDs besteht darin, dass sie mittels geeigneter Algorithmen dezentral, aber trotzdem global eindeutig erzeugt werden können (vgl. dazu Paul J. Leach und Rich Salz: UUIDs and GUIDs, 1998, <http://www.opengroup.org/dce/info/draft-leach-uuids-guids-01.txt>.)

Bei historisierbaren Objektarten bekommen alle Versionen eines Objektes, d.h. alle Instanzen, die den Zustand desselben Objektes in verschiedenen Zeiträumen beschreiben, dieselbe GUID (wenn sie vergeben wird). In einem System, das die OKSTRA®-Historisierung unterstützt, ist somit nicht die „OKSTRA\_ID“ allein, sondern die Kombination aus „OKSTRA\_ID“ und Zeitpunkt eindeutig. Diese Konvention ist nötig, um die „OKSTRA\_ID“ bei der Kommunikation zwischen historiefähigen und nicht historiefähigen Systemen verwenden zu können.

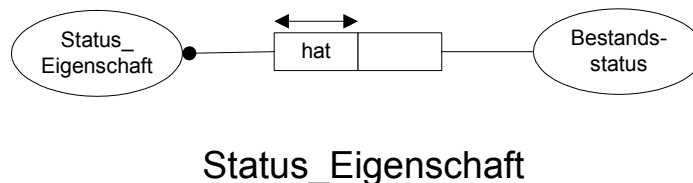
## Objekt\_ID

OKSTRA®-Objektarten, die von dem abstrakten Mixin *Objekt\_mit\_ID* erben, sind in der Lage, *Objekt\_IDs* anderer Fachsysteme zu verwalten. Eine *Objekt\_ID* besitzt das STRING-Attribut „ID“ zur Aufnahme des eigentlichen Identifikators und das STRING-Attribut „Namensraum\_Verfahren“, in dem das erzeugende Fachsystem/Verfahren eingetragen werden kann.



## Status\_Eigenschaft

Durch Erben vom abstrakten Supertypen *Status\_Eigenschaft* wird eine Objektart in die Lage versetzt, eine Information zum Bestandsstatus zu tragen (Bestand\_erfasst, Bestand\_amtlich, geplant/neu, geplant/Abriss, zerstört, unbekannt).




## Datum, Zeit, Dauer

Der OKSTRA® besitzt die vier nachstehend aufgeführten Kategorien von Datums- und Zeitangaben. Da derartige Angaben generell nur attributive Funktion haben, werden zu ihrer Darstellung keine vollwertigen Objektarten verwendet, sondern attributive Konstrukte (Datentypen, Schlüssel Tabellen und konzeptionelle Objektarten).

### Datentypen und Schlüssel Tabellen für Datums- und Zeitangaben

In diese Kategorie fallen

- die Datentypen *Datum* und *Uhrzeit*, die sich auf den elementaren Datentypen STRING zurückführen lassen,
- die Schlüssel Tabellen *Monat* und *Wochentag* sowie
- die Datentypen *Jahr*, *Woche*, *Tag*, *Stunde*, *Minute*, *Sekunde* und *Millisekunde*, die auf dem elementaren Datentyp INTEGER basieren.

	<b>Objektkatalog für das Straßen- und Verkehrswesen</b> <b>Schema Allgemeine Objekte</b>	<b>Seite: 4 von 10</b> <b>Name: D020</b> <b>Stand: 27.11.2008</b>
--	---	---

Der Wertebereich der STRING- und INTEGER-Typen wird dabei durch WHERE-Klauseln sinnvoll beschränkt, soweit dies möglich ist. Dabei werden folgende Definitionen vorausgesetzt:

*Jahr*: Jahreszahl eines Jahres,

*Woche*: Nummer einer Woche in einem Jahr,

*Stunde*: Eine Stunde innerhalb eines Tages,

*Minute*: Eine Minute innerhalb einer Stunde,

*Sekunde*: Eine Sekunde innerhalb einer Minute,

*Millisekunde*: Eine Millisekunde innerhalb einer Sekunde.

## Datentypen für elementare Dauerangaben

In diese Kategorie fallen die Datentypen *Jahre*, *Monate*, *Wochen*, *Tage*, *Stunden*, *Minuten*, *Sekunden* und *Millisekunden*. Sie dienen zur Angabe von Zeitdauern in verschiedenen Einheiten und werden sämtlich auf den Datentyp *Anzahl* zurückgeführt.

## Zeitpunkt und Zeitabschnitt

In dieser Kategorie existieren die beiden konzeptionellen Objektarten *Zeitpunkt* und *Zeitabschnitt*. Ein *Zeitpunkt* wird durch die Angabe eines *Datums* und ggf. einer *Uhrzeit* definiert. Ein *Zeitabschnitt* bezeichnet ein zusammenhängendes Zeitintervall und besitzt einen Start- und i.d.R. auch einen End-*Zeitpunkt*. Fehlt dieser, ist der *Zeitabschnitt* noch nicht beendet.

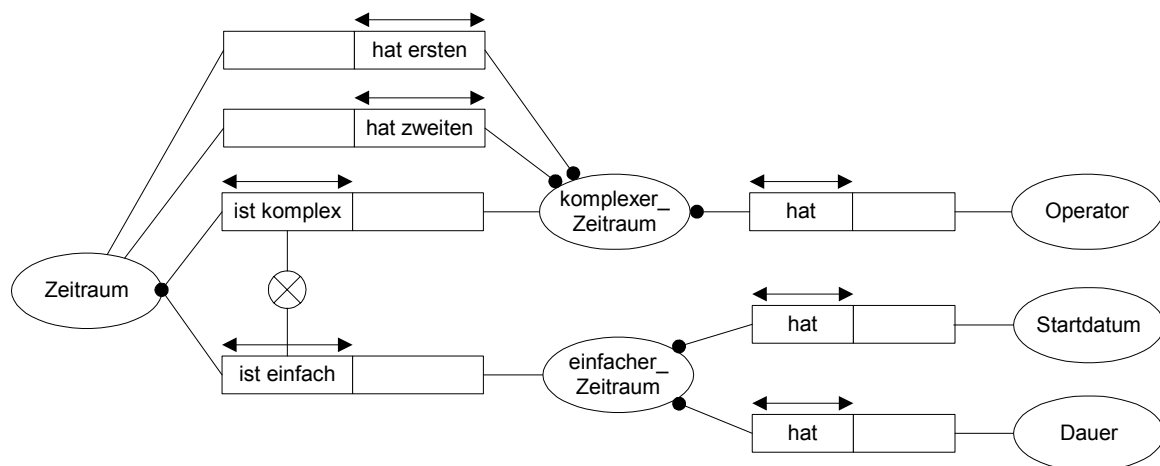
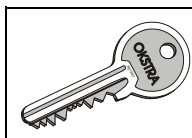
## GDF-Zeitraummodell

Dieses Modell wurde dem Zeitraummodell aus GDF nachgebildet („CEN Road Traffic and Transport Telematics, Geographic Road Database, GDF for Road Traffic and Transport Telematics“, Time Domain Kapitel 10.1.1 einschließlich Anhang A1.15). Es besitzt gegenüber dem *Zeitabschnitt* folgende Erweiterungen:

- Ein *Zeitraum* kann aus beliebig vielen Zeitintervallen zusammengesetzt sein.
- Die einzelnen Zeitintervalle eines *Zeitraums* müssen auf der Zeitachse nicht zusammenhängend sein.
- Die *Dauer* eines *Zeitraums* kann in verschiedenen Einheiten angegeben werden.
- Das *Startdatum* eines *Zeitraums* kann „unscharf“ bzw. regelmäßig wiederkehrend sein (z.B. „jeder zweite Sonntag im Juli“).

Kern der Modellierung ist die konzeptionelle Objektart *Zeitraum*, die entweder einen *einfachen Zeitraum* oder einen *komplexen Zeitraum* enthält. Ein *einfacher Zeitraum* beschreibt ein einfaches Zeitintervall und ist durch die Angabe eines *Startdatums* und der *Dauer* des Intervalls charakterisiert. Ein *komplexer Zeitraum* ist aus genau zwei *Zeiträumen* zusammengesetzt. Da diese *Zeiträume* ihrerseits ebenfalls *einfache Zeiträume* oder *komplexe Zeiträume* enthalten, lassen sich durch rekursive Schachtelung aus beliebig vielen Zeitintervallen zusammengesetzte *Zeiträume* beschreiben.

Die Art der Verknüpfung von zwei *Zeiträumen* zu einem *komplexen Zeitraum* wird durch die Schlüsseltabelle *Operator* beschrieben. Mögliche Operationen sind „Vereinigung“ (Vereinigungsmenge beider *Zeiträume*), „Durchschnitt“ (diejenigen Zeitintervalle, die in beiden *Zeiträumen* enthalten sind) und „Differenz“ (der erste *Zeitraum* abzüglich der Zeitintervalle, die in beiden *Zeiträumen* enthalten sind).

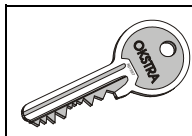


## Zeitraum

Die konzeptionelle Objektart *Startdatum* ermöglicht die Angabe einer Vielzahl von Varianten für den Beginn eines *Zeitraums*. Über die Attribute „Jahr“, „Monat\_im\_Jahr“ und „Tag\_im\_Monat“ kann beispielsweise ein normales Datum codiert werden. Falls übergeordnete Angaben fehlen, wird das *Startdatum* als regelmäßig wiederkehrend interpretiert. Wenn z.B. kein „Jahr“ angegeben wird, sondern nur die Attribute „Monat\_im\_Jahr“ und „Tag\_im\_Monat“, dann bezeichnet das *Startdatum* den entsprechenden Tag in *jedem* Jahr (z.B. „jedes Jahr am 13. Mai“). Anstelle eines Monats kann auch die Nummer der Woche im Jahr angegeben werden (Attribut „Woche\_im\_Jahr“, z.B. „34“). Die Schlüsseltabelle *Nummer\_des\_Wochentages* dient für Zeitangaben der Form „der zweite Sonntag im Juni“ (für dieses Beispiel müsste man die Attribute „Monat\_im\_Jahr“, „Tag\_in\_der\_Woche“ und „Nummer\_des\_Wochentages“ belegen; sofern kein „Jahr“ angegeben wird, ist der zweite Sonntag im Juni in *jedem* Jahr gemeint).

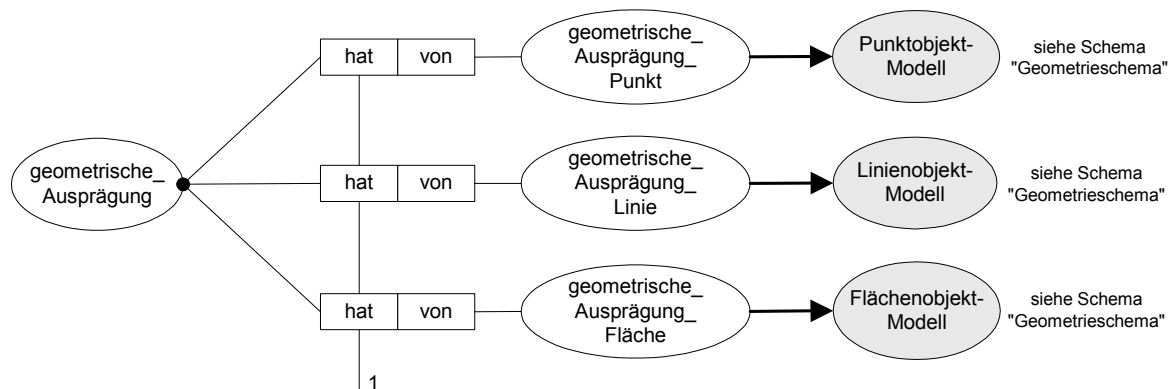
Die konzeptionelle Objektart *Dauer* ermöglicht die Angabe einer Zeitdauer in verschiedenen Einheiten (von *Jahre* bis *Sekunden*); sofern mehrere Attribute belegt werden, müssen die einzelnen Inhalte addiert werden (z.B. „drei Jahre und fünf Monate“).

Objekt	Definition, Erläuterung
<i>Dauer</i>	Dauer (zeitliche Länge eines <i>einfachen Zeitraums</i> )
<i>einfacher Zeitraum</i>	beschreibt ein Zeitintervall
<i>komplexer Zeitraum</i>	verknüpft jeweils zwei <i>Zeiträume</i> mittels einer binären Operation (siehe <i>Operator</i> )
<i>Operator</i>	Schlüsseltabelle, deren Inhalt den binären Operator beschreibt, mit dem zwei <i>Zeiträume</i> zu einem <i>komplexen Zeitraum</i> verknüpft werden. Mögliche Operationen sind „Vereinigung“ (Vereinigungsmenge beider <i>Zeiträume</i> ), „Durchschnitt“ (diejenigen Zeitintervalle, die in beiden <i>Zeiträumen</i> enthalten sind) und „Differenz“ (der erste <i>Zeitraum</i> abzüglich der Zeitintervalle, die in beiden <i>Zeiträumen</i> enthalten sind).
<i>Startdatum</i>	Beschreibt den Beginn eines <i>einfachen Zeitraums</i> mit Datum und Uhrzeit; ergibt in Verbindung mit der <i>Dauer</i> das Endedatum
<i>Zeitraum</i>	allgemeiner Zeitraum; abstrakter Supertyp für den <i>einfachen Zeitraum</i> und den <i>komplexen Zeitraum</i>



## Geometrische Ausprägung

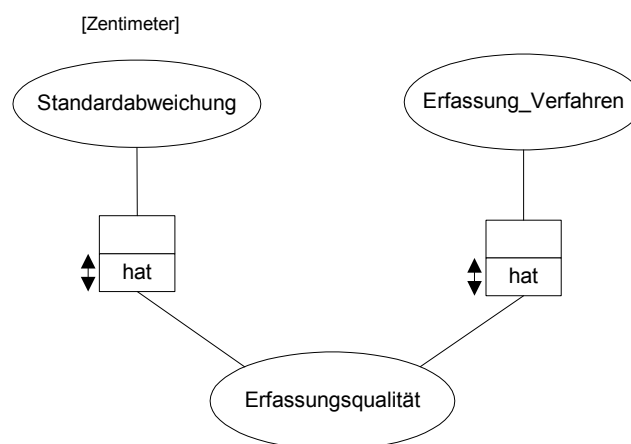
Mit dem Objekt *Geometrische Ausprägung* wird eine wahlweise Darstellung eines OKSTRA®-Objekts als Punkt, Linie oder Fläche ermöglicht. Genau eine dieser Möglichkeiten muss gewählt werden.



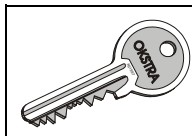
Geometrische Ausprägung

## Erfassungsqualität

Mit der konzeptionellen Objektart *Erfassungsqualität* können für eine Objektart, die über Geometrie verfügt, Informationen zur geometrischen Genauigkeit und zum Erfassungsverfahren angegeben werden.

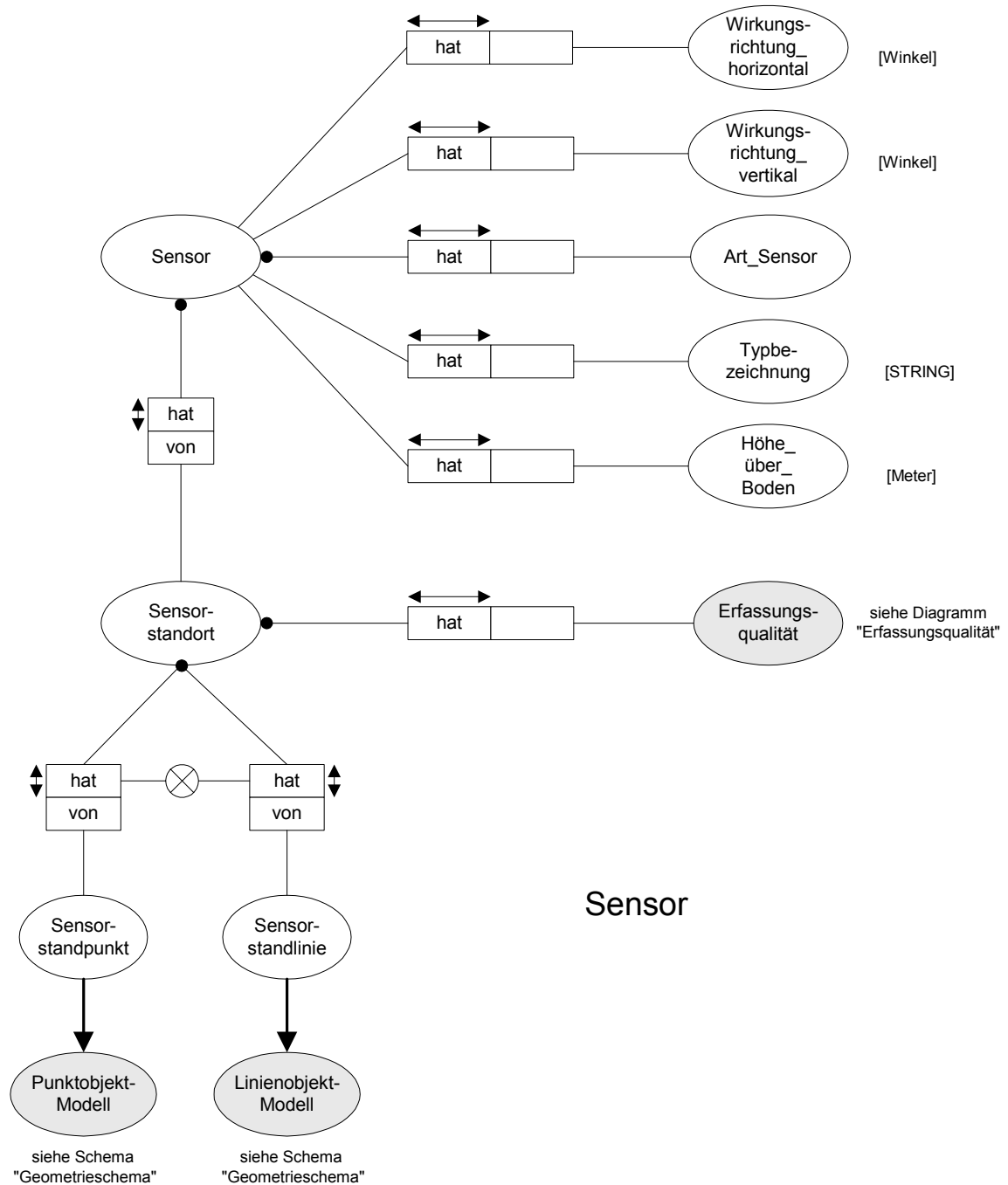


Erfassungsqualität



## Sensor

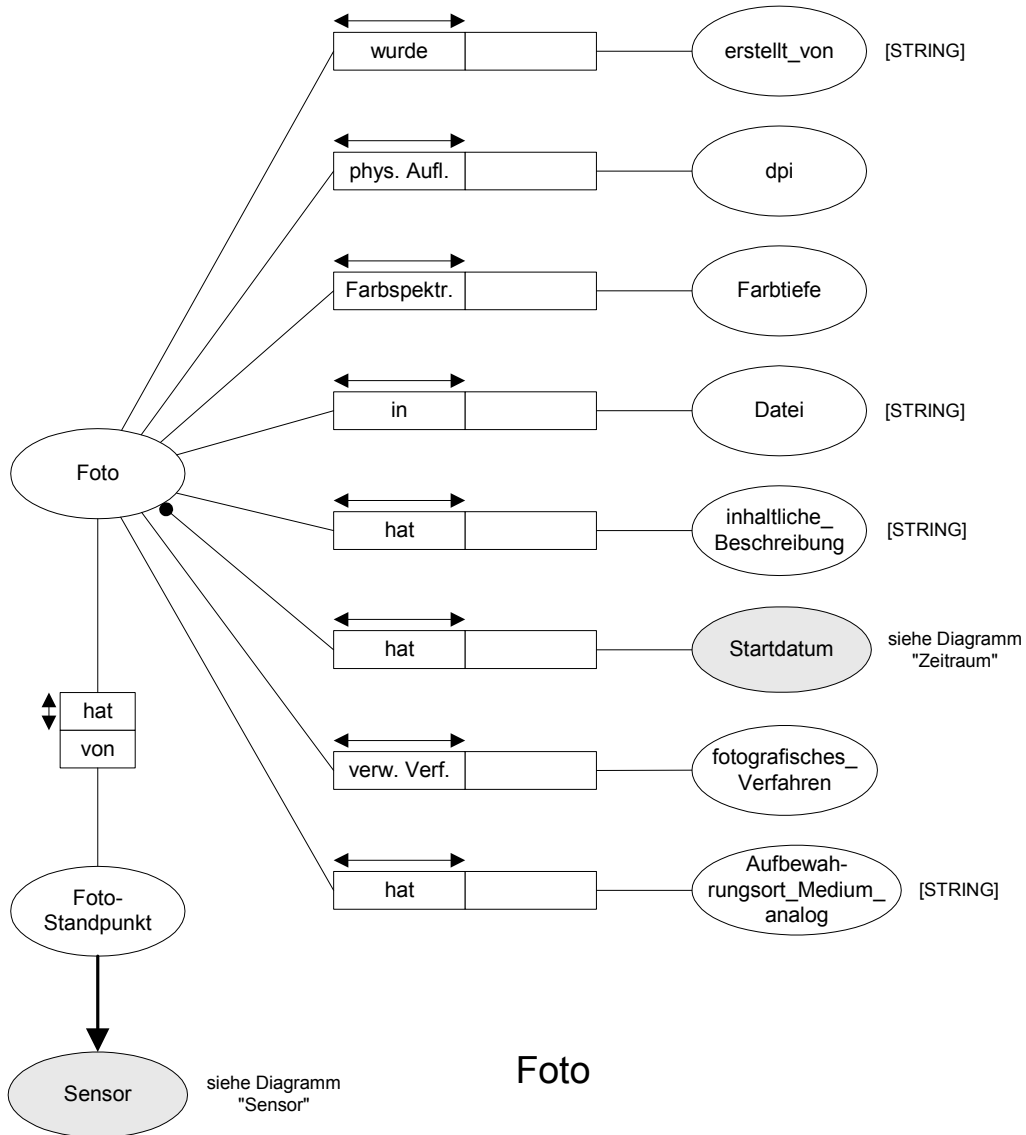
Die Objektart *Sensor* beschreibt ein Gerät zur audiovisuellen Aufzeichnung. Hierzu zählen z.B. Fotokamera, Videokamera, Mikrofon und Wärmebildkamera.



Sensor

## Foto

Die Objektart *Foto* beschreibt eine auf optisch-digitalem bzw. optisch-chemischem Weg generierte Abbildung einer realen Situation.

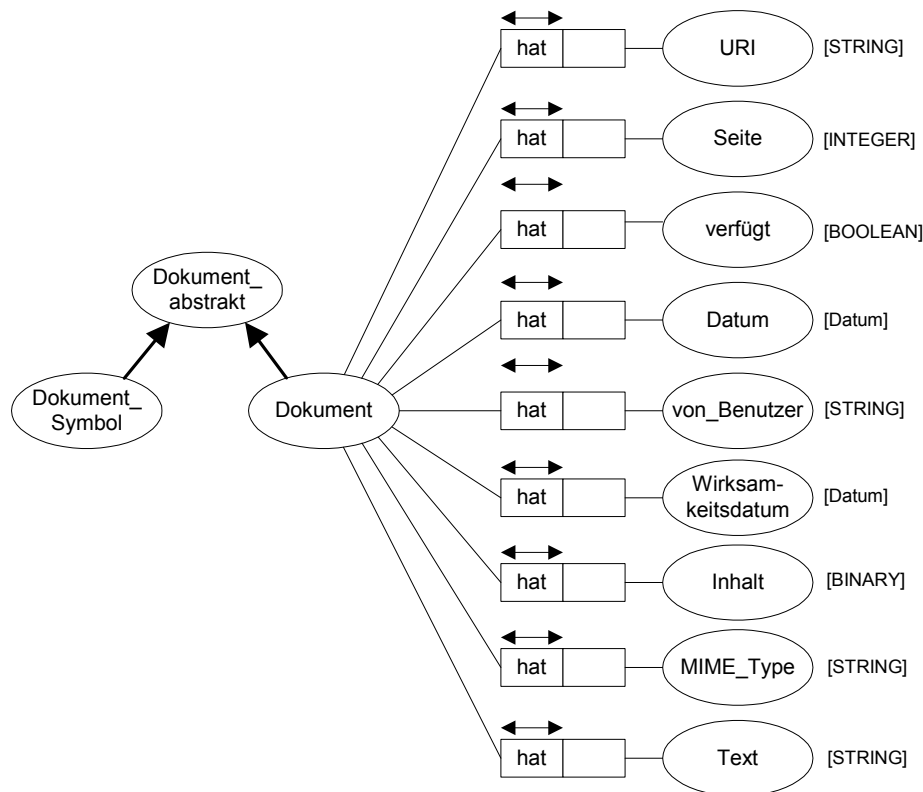






## Dokument

Mit der Objektart *Dokument* können Metadaten über ein bestehendes Dokument (z.B. das Aktenzeichen einer Verwaltungsakte) angegeben bzw. Verweise auf ein solches Dokument realisiert werden. Darüber hinaus besteht auch die Möglichkeit, ein Dokument im IT-technischen Sinn – d.h. den Inhalt einer Datei eines bestimmten Typs - zu transportieren. Damit lassen sich z.B. Grafiken über den OKSTRA® austauschen.



### Dokument

Für ein *Dokument* lässt sich ein URI (Uniform Ressource Identifier) angeben, mit dem das *Dokument* als allgemeine, ggf. im Internet verfügbare Ressource spezifiziert werden kann. Falls ein Aktenzeichen für das *Dokument* angegeben werden soll, geschieht dies durch die Angabe des URNs „urn:okstra:dok:[Aktenzeichen]“ im URI-Attribut. Der URI dient auch als Kennung für die abstrakten Verweise auf *Dokumente*.

Falls ein Dateiinhalt in einer Instanz der Objektart *Dokument* transportiert werden soll, sind die Attribute „Inhalt“ und „MIME\_Type“ von Bedeutung. Der Dateiinhalt wird im Attribut „Inhalt“ in Form von Binärdaten eingebettet, wobei die formatspezifischen Codierungen für Binärdaten zu beachten sind (für das CTE-Format siehe z.B. die Festlegungen zur Darstellung des EXPRESS-Typen BINARY gemäß ISO 10303-21). Der Typ des Dokuments wird im Attribut „MIME\_Type“ abgelegt (z.B. „application/msword“ für Word-Dateien oder „image/jpeg“ für JPEG-Bilddaten; Listen mit gängigen MIME\_Types finden sich im Internet). Diese Angabe ist nötig, damit ein Empfänger der Daten weiß, wie er die im Attribut „Inhalt“ enthaltenen Daten behandeln muss bzw. mit welcher Applikation er sie öffnen kann.

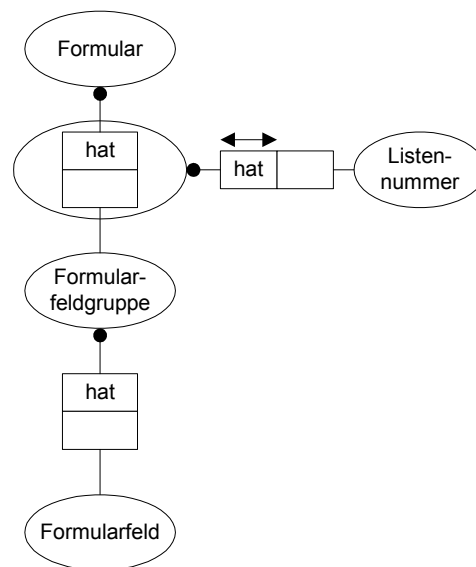
## Formular

Die Objektart *Formular* dient gleichzeitig zur Angabe des formalen Aufbaus eines Formulars wie auch der Werte, die dort ggf. eingetragen werden. Ein *Formular* besteht aus einer geordneten Menge von *Formularfeldgruppen*, die wiederum *Formularfelder* besitzen. *Formulare*, *Formularfeldgruppen* und *Formularfelder* besitzen Namen (Pflichtattribute), die Aufschluss über die Art des *Formulares*, die jeweilige *Formularfeldgruppe* und das jeweils bezeichnete *Formularfeld* geben.

Ein *Formular* kann mehrere *Formularfeldgruppen* gleichen Namens enthalten; damit können Wiederholungen von Teilen des *Formulars* beschrieben werden. Die Namen der *Formularfelder* innerhalb einer *Formularfeldgruppe* müssen hingegen eindeutig sein, um eine zweifelsfreie Zuordnung zu ermöglichen.

Ein *Formularfeld* besitzt das optionale Stringattribut „Wert“, in dem ein in das Feld eingetragener Wert abgelegt werden kann. Bei einem Ankreuzfeld sind die Werte „1“ für „angekreuzt“ und „0“ für „nicht angekreuzt“ zu verwenden.

Die Objektarten *Formularfeldgruppe* und *Formularfeld* sind konzeptionell, d. h. sie können nicht eigenständig existieren, sondern nur im Kontext eines *Formulars*. Die *Formularfeldgruppen* eines *Formulars* sind in der Reihenfolge ihres Auftretens im Papierformular zu ordnen.



Formular