



## Informationsarchitektur - Daten:

### RAIM-Referenzmodelle und -Werkzeuge für die Datenarchitektur



### 1. DATENARCHITEKTUR

In der Datenarchitektur werden die Daten mit ihren Beziehungen, die für die Durchführung der Geschäftsprozesse benötigt werden, identifiziert und beschrieben. Dies erfolgt in einem Modell und einer Darstellungsform, die stabil, vollständig, konsistent und für alle Beteiligten verständlich ist (vgl. Datenmodell).

### 2. HISTORISCHE ENTWICKLUNG

Im Laufe der vergangenen Jahre und Jahrzehnte wurden in den verschiedenen IVS-Domänen Schnittstellen zum Austausch von Daten entwickelt. Einige dieser Schnittstellenbeschreibungen enthalten explizite Datenbeschreibungen (z. T. in formalen Datenbeschreibungssprachen wie z. B. UML), andere Schnittstellenbeschreibungen enthalten implizite Datenbeschreibungen (auch hier z. T. in formalen Datenbeschreibungssprachen).

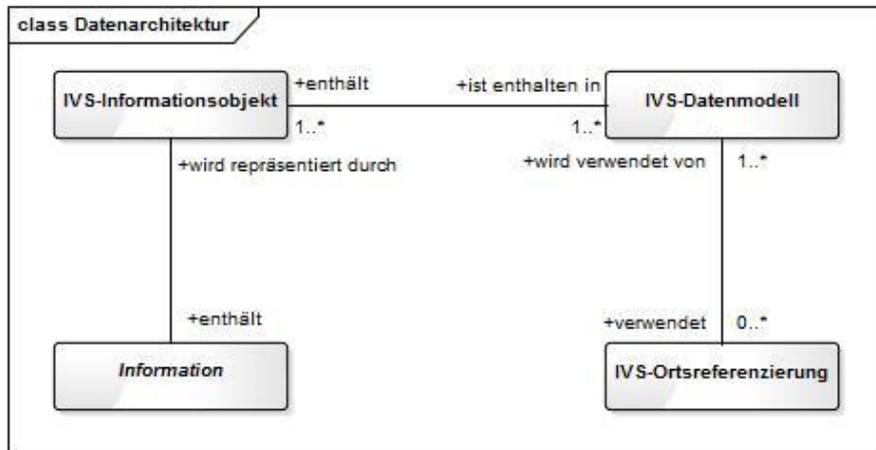
Diese Standards sind unabhängig voneinander entstanden und für verschiedene Anwendungsbereiche konzipiert worden. Mittlerweile haben sich die Standards weiterentwickelt und überlappen sich zum Teil inhaltlich. Aufgrund der unterschiedlichen Anwendungsbereiche und je nach Zeitpunkt der Entstehung werden die Daten auf unterschiedlichen Abstraktionsniveaus beschrieben.

Mittlerweile existieren Anwendungen, die mit zwei oder mehr domänenspezifischen Datenmodellen arbeiten. Da kein übergeordnetes Datenmodell existiert, müssen solche Anwendungen bisher die Abbildungen zwischen den Datenmodellen anwendungsspezifisch erarbeiten.

Ein wesentlicher Bestandteil verkehrsbezogener Daten ist der Ort, für den diese Daten gelten. Um die räumliche Gültigkeit von verkehrsbezogenen Daten beschreiben zu können, wurden verschiedene Verfahren (sogenannte Ortsreferenzierungssysteme bzw. englisch location referencing systems) entwickelt. Die domänenspezifischen Datenmodelle verwenden unterschiedliche, z. T. mehrere Ortsreferenzierungssysteme. Auch bei den Ortsreferenzierungssystemen besteht das Problem, dass die Konvertierung von Ortsreferenzen zwischen verschiedenen Systemen z. T. nicht möglich, oft jedoch aufwändig und fehleranfällig ist. Wie bei den domänenspezifischen Datenmodellen gibt es auch hier kein übergeordnetes System, in das alle Ortsreferenzierungssysteme ohne Weiteres eingebettet werden könnten.

### 3. BEGRIFFSDEFINITIONEN DER DATENARCHITEKTUR

Das folgende UML-Diagramm zeigt die Architekturbausteine der Datenarchitektur und deren Beziehungen untereinander:



*Datenarchitektur*

In der Datenarchitektur werden die folgenden Architekturbausteine beschrieben:

- Ein IVS-Informationsobjekt enthält eine Information mit einer festgelegten Bedeutung (Semantik).
- Ein IVS-Datenmodell enthält IVS-Informationsobjekte und legt deren Format (Syntax) fest.
- Eine IVS-Ortsreferenzierung beschreibt ein Verfahren zur Beschreibung von geografischen Orten und wird in IVS-Datenmodellen verwendet.

#### 4. IVS-INFORMATIONSOBJEKTE

Ein IVS-Informationsobjekt enthält die semantische Beschreibung inhaltlich zusammengehöriger Informationen, die als Input bzw. Output eines Geschäftsprozessschrittes verwendet werden. Jede RAIM muss einen Katalog der verwendeten IVS-Informationsobjekte erstellen. Um diesen Katalog sinnvoll erstellen zu können, wird aus dem Katalog der bestehenden IVS-Geschäftsprozesse ermittelt, welche Input- bzw. Output-Informationsobjekte zur Umsetzung dieser Geschäftsprozesse benötigt werden. Wenn der Katalog fertiggestellt ist, kann eine Matrix der Informationsobjekte/Geschäftsprozesse erstellt werden.

Der bereits vorausgefüllte Katalog der IVS-Informationsobjekte kann ebenfalls dazu verwendet werden. Dazu kann eine Kopie dieses Kataloges angelegt werden. In dieser Kopie sind dann die für die RAIM relevanten Informationsobjekte (durch Hinzufügen bzw. Löschen von Informationsobjekten) zu beschreiben.

#### 5. IVS-DATENMODELLE

Ein IVS-Datenmodell besteht aus einer Sammlung von Informationsobjekten und einer Vorgabe, wie die semantischen Informationen der Informationsobjekte syntaktisch korrekt kodiert werden müssen. Jede RAIM muss einen Katalog der vorhandenen Datenmodelle erstellen. Der bereits vorausgefüllte Katalog der IVS-Datenmodelle kann dazu verwendet werden. Dazu kann eine Kopie dieses Kataloges angelegt werden. In dieser Kopie sind dann die für die RAIM relevanten Datenmodelle (durch Hinzufügen bzw. Löschen von Datenmodellen) zu beschreiben.

Jede RAIM muss eine Matrix erstellen, in der die Zuordnung zwischen Informationsobjekten und Datenmodellen beschrieben werden. Die bereits vorausgefüllte Matrix der IVS-Informationsobjekte-Datenmodelle kann dazu verwendet werden. Dazu kann eine Kopie dieser Matrix angelegt werden. In dieser Kopie sind dann die für die RAIM relevanten Informationsobjekte und Datenmodelle aufzunehmen und die Zuordnung zwischen Informationsobjekten und Datenmodellen zu



beschreiben.

## 6. IVS-ORTSREFERENZIERUNGSSYSTEME

Ein IVS-Ortsreferenzierungssystem legt die Syntax und Semantik für ein Verfahren fest, mit dem geografische Orte auf der Erde oder speziell Positionen in bzw. Teilen von Verkehrsnetzen beschrieben werden können. Jede RAIM muss einen Katalog der vorhandenen Ortsreferenzierungssysteme erstellen. Der bereits vorausgefüllte Katalog der IVS-Ortsreferenzierungen kann dazu verwendet werden. Dazu kann eine Kopie dieses Kataloges angelegt werden. In dieser Kopie sind dann die für die RAIM relevanten Ortsreferenzierungen (durch Hinzufügen bzw. Löschen von Ortsreferenzierungen) zu beschreiben.

Jede RAIM muss eine Matrix erstellen, in der die Zuordnung zwischen Datenmodellen und Ortsreferenzen beschrieben wird. Die bereits vorausgefüllte Matrix der IVS-Datenmodell-Ortsreferenzen kann dazu verwendet werden. Dazu kann eine Kopie dieser Matrix angelegt werden. In dieser Kopie sind dann die für die RAIM relevanten Datenmodelle und Ortsreferenzierungen aufzunehmen und die Zuordnung zwischen Datenmodellen und Ortsreferenzierungen zu beschreiben.

## 7. VERWENDUNG UNTERSCHIEDLICHER WERKZEUGE

Bei der Festlegung auf IVS-Referenzmodelle und Werkzeuge muss berücksichtigt werden, dass zurzeit viele unterschiedliche, domänenspezifische IVS-Datenmodelle existieren und auch weiterhin existieren werden (siehe Anleitung zur IVS-Datenarchitektur). Diese Datenmodelle sind mit unterschiedlichen Werkzeugen und unter Anwendung unterschiedlicher IVS-Referenzmodelle erzeugt worden. Da diese Datenmodelle auch von unterschiedlichen Communities gepflegt werden, ist eine Vereinheitlichung der Modellierungsprinzipien nur schwer zu erreichen.

Falls eines oder mehrere der bereits vorhandenen IVS-Datenmodelle verwendet werden (müssen), sind die Modellierungsprinzipien und -werkzeuge bereits vorgegeben. Lediglich für neu zu entwickelnde IVS-Datenmodelle werden die im Folgenden angegebenen Prinzipien und Werkzeuge festgelegt.

## 8. MODELLIERUNGSPRINZIPIEN

### 8.1 Verwendung von Standards

Falls möglich, sollen für den Austausch von Daten Standards verwendet werden.

### 8.2 Definition von technologie- und plattformunabhängigen Datenmodellen

Grundsätzlich sollen Datenmodelle in einem technologie- und plattformunabhängigen Format, einem sogenannten Platform-independent model (PIM) beschrieben werden. Aus dem PIM werden dann sogenannte Platform-specific models (PSM) möglichst automatisch generiert. Diese Vorgehensweise hat sich bereits bei der Modellierung des DATEX II Datenmodells bewährt. Sie hat den Vorteil, dass die spezifische Implementation des Datenmodells ausgetauscht werden kann, ohne das Datenmodell zu verändern.

Dadurch werden u.a. die folgenden Einsatzszenarien ermöglicht:

- Spezifikation verschiedener PSMs (Austauschformate), die gleichzeitig verwendet werden (z. B. binäres Datenformat für bandbreitenbeschränkten Einsatz und XML-Format für sonstigen Einsatz).
- Upgrade von einem veralteten PSM auf ein neueres PSM.

Diese Veränderungen können durchgeführt werden, ohne das PIM-Datenmodell zu verändern.



## 9. DATENMODELLIERUNGSSPRACHE

Für neu zu entwickelnde Datenarchitekturen bzw. Datenmodelle wird die Unified Modeling Language (UML) als Modellierungssprache vorgeschlagen. UML hat sich mittlerweile als De-facto-Standard bei der technologie-agnostischen Modellierung von Daten etabliert. Des Weiteren hat UML den Vorteil, dass mit dem XML Metadata Interchange (XMI) ein Datenaustauschformat vorliegt, das unabhängig von den eingesetzten Tools ist und zunehmend auch von Software-Entwicklungswerkzeugen verwendet wird.