

## **BSVRZ – Tutorial Konfiguration**

**Teil A – Schnelleinstieg Konfiguration**

**Teil B – Grundlagen der Konfiguration**

Teil C – Workflow bei der Konfiguration

Teil D – Richtlinien, Fehlersituationen und Lösungen

### **Auftraggeber**

Landesbetrieb für Straßenbau Saarland  
Geschäftsbereich 4 – Betrieb und Verkehr  
Lindenallee 2a  
66538 Neunkirchen

### **Ersteller**

inovat  
innovative systeme – verkehr – tunnel – technik  
Dipl.-Ing. H. C. Kniß  
Kölner Str. 30  
D-50859 Köln

### **Projekt**

sl.slf.07.02-Tutorial Konfiguration

## 0 Allgemeines

### 0.1 Verteiler

Organisationseinheit	Name	Anzahl Kopien	Vermerk
LFS, GB 4	Herr Dr. Norkauer	1	
LFS, GB 4	Frau Keinath	1	
Inovat	Herr Kniß	1	

Tabelle 0-1: Verteiler

### 0.2 Änderungsübersicht

Version	Datum	Kapitel	Bemerkungen	Bearbeiter
1.0	26.04.2008	alle	Erstellung	Dipl.-Ing. H. C. Kniß (inovat)
1.1	01.05.2008	3.1.5.1 3.1.5.2 3.1.5.6 3.4.1.4.6	Anpassung an neuen kv.kappich vom 27.04.2008	Dipl.-Ing. H. C. Kniß (inovat)
1.2	02.10.2008	3.1.5.1 3.1.5.2 3.1.5.6 3.4.1.4.6	Anpassung an neues Kernsoftwarerelease 3.4.7 (Ende September 2008)	Dipl.-Ing. H. C. Kniß (inovat)
2.0	07.01.2009	3.1.6 ff 3.4.1 3.4.2, 3.4.3, 3.4.4, 3.4.5 Tabelle 3-11, 3-12, 3-13 3.4.2, 3.4.3, 3.4.4 3.4.3 3.4.4 3.4.5 3.4.6 3.4.6.1.7 3.4.6.1.9 Anhang	Anpassungen gemäß Prüfanmerkungen vom 17.12.2008	Dipl.-Ing. H. C. Kniß (inovat)

Tabelle 0-2: Änderungsübersicht

## 0.3 Inhaltsverzeichnis

<b>0</b>	<b>Allgemeines</b>	<b>2</b>
0.1	Verteiler	2
0.2	Änderungsübersicht	2
0.3	Inhaltsverzeichnis	3
0.4	Abkürzungsverzeichnis	7
0.5	Definitionen	7
0.6	Referenzierte Dokumente	7
0.7	Abbildungsverzeichnis	7
0.8	Tabellenverzeichnis	8
<b>1</b>	<b>Zusammenfassung</b>	<b>9</b>
<b>2</b>	<b>Teil A – Schnelleinstieg Konfiguration</b>	<b>10</b>
2.1	Überblick Konfiguration	10
2.1.1	Konfiguration / Konfigurationsapplikation .....	10
2.1.2	Grundbegriffe der Konfiguration .....	12
2.1.2.1	Modelle .....	12
2.1.2.1.1	Datenmodell.....	12
2.1.2.1.2	Metamodell .....	12
2.1.2.1.3	Systemmodell / Fachmodell .....	13
2.1.2.2	Versorgung (Objektdefinition).....	13
2.1.2.3	Konfigurationsbereiche.....	13
2.1.2.4	Konfigurationsverantwortliche .....	14
2.1.2.5	Autarke Organisationseinheit (AOE) .....	15
2.1.2.6	Verwaltungsdatei.....	15
2.1.3	Konfigurationsapplikation.....	15
2.1.3.1	Anwendungsbetrieb.....	15
2.1.3.2	Konfigurationsbetrieb.....	16
2.1.3.2.1	Import.....	17
2.1.3.2.2	Konsistenzprüfung.....	17
2.1.3.2.3	Aktivierung .....	17
2.1.3.2.4	Export.....	17
2.1.3.2.5	Freigabe.....	17
<b>3</b>	<b>Teil B – Grundlagen der Konfiguration</b>	<b>18</b>
3.1	Konfigurationsbereiche im Detail	18
3.1.1	Historie.....	18
3.1.2	Konfigurationsverantwortlicher - Konfigurationsbereich .....	18

3.1.3	Konfigurationsdateien .....	18
3.1.4	Versorgungsdateien.....	19
3.1.4.1	Namensgebung .....	19
3.1.4.2	Inhalt .....	20
3.1.4.3	DTD einer Versorgungsdatei .....	20
3.1.4.4	Aufbau .....	22
3.1.4.4.1	Infobereich (→ info) .....	23
3.1.4.4.2	Änderungsübersicht (→ konfigurationsAenderung).....	23
3.1.4.4.3	Modellbereich (→ modell) .....	23
3.1.4.4.4	Objektbereich (→ objekte) .....	24
3.1.4.5	Verwaltungsdatei .....	24
3.1.5	Kriterien zur Aufteilung der Konfiguration in Konfigurationsbereiche .....	26
3.1.6	Beschreibung der aktuellen Standard-Konfigurationsbereiche .....	27
3.1.6.1	kv.kappich.....	28
3.1.6.2	kv.inovat.....	29
3.1.6.3	kv.bea .....	29
3.1.6.4	kv.bitctrl.....	29
3.1.6.5	kv.dambach .....	31
3.1.6.6	kv.logos.....	31
3.1.6.7	kv.ptv .....	32
3.1.7	Abhängigkeiten zwischen Konfigurationsbereichen .....	32
3.1.7.1	Was sind Abhängigkeiten?.....	32
3.1.7.2	Dokumentation der Abhängigkeiten .....	32
3.1.8	Kodierungsschemata und Namenskonventionen für KV und AOE .....	33
3.1.8.1	Kodierung für Konfigurationsverantwortlichen / AOE.....	34
3.1.8.2	PID von Konfigurationsverantwortlichen / AOE.....	34
3.1.8.3	Länderkürzel .....	35
3.1.8.4	PID von Konfigurationsbereichen.....	36
3.2	Konfigurationsverantwortliche und AOE .....	37
3.2.1	Konfigurationsverantwortliche.....	37
3.2.2	Autarke Organisationseinheit (AOE).....	37
3.3	Konfigurationsapplikation im Detail .....	38
3.3.1	Anwendungsbetrieb .....	38
3.3.2	Konfigurationsbetrieb .....	38
3.3.2.1	Übersicht der Modi und zusätzlichen Aufrufparameter .....	38
3.3.2.2	Konfigurationsumgebung .....	40
3.3.2.3	Import.....	40

3.3.2.3.1	Funktion .....	40
3.3.2.3.2	Aufrufparameter .....	41
3.3.2.3.3	Startscript für den Import .....	41
3.3.2.4	Konsistenzprüfung .....	42
3.3.2.4.1	Funktion .....	42
3.3.2.4.2	Aufrufparameter .....	42
3.3.2.4.3	Startscript für die Konsistenzprüfung .....	43
3.3.2.5	Aktivierung .....	43
3.3.2.5.1	Funktion .....	43
3.3.2.5.2	Aufrufparameter .....	43
3.3.2.6	Export .....	44
3.3.2.6.1	Funktion .....	44
3.3.2.6.2	Aufrufparameter .....	44
3.3.2.6.3	Startscript für den Export .....	45
3.3.2.7	Freigabe .....	45
3.3.2.8	Freigabe zur Übernahme .....	45
3.3.2.8.1	Funktion .....	45
3.3.2.8.2	Aufrufparameter .....	45
3.3.2.8.3	Startscript für die „Freigabe zur Übernahme“ .....	46
3.3.2.9	Freigabe zur Aktivierung .....	46
3.3.2.9.1	Funktion .....	46
3.3.2.9.2	Aufrufparameter .....	46
3.3.2.9.3	Startscript für die „Freigabe zur Aktivierung“ .....	47
3.3.2.10	Freigabe zur Aktivierung ohne lokale Aktivierung durch den KV .....	47
3.3.2.10.1	Funktion .....	47
3.3.2.10.2	Aufrufparameter .....	47
3.4	Standardworkflows .....	48
3.4.1	Grundsätzliches .....	48
3.4.2	Modelle .....	49
3.4.3	Versorgung (Objekte) .....	50
3.4.4	Betrieb .....	52
3.4.5	Projekt .....	53
3.4.6	Beschreibung einzelner Teilschritte für die Workflow .....	54
3.4.6.1	Konfigurationsverantwortlichen oder AOE erstellen .....	55
3.4.6.1.1	Schritt 1: Beantragung einer Kodierungsbasis bei der ZS für die Organisation .....	55
3.4.6.1.2	Schritt 2: Festlegung der Kodierung für diesen KV / diese AOE .....	55

3.4.6.1.3	Schritt 3: Festlegung des Typs des Konfigurationsverantwortlichen („einfacher“ KV oder „Autarke Organisations-Einheit“ AOE) .....	56
3.4.6.1.4	Schritt 4: Festlegung der PID für den KV / die AOE .....	56
3.4.6.1.5	Schritt 5: Erstellung des Konfigurationsbereichs für den KV oder die AOE (Anpassung der Vorlage) .....	57
3.4.6.1.6	Schritt 6: Verwaltungsdatei erstellen (Anpassung Vorlage) .....	60
3.4.6.1.7	Schritt 7: Konfigurationsdatei für KV oder AOE erstellen (Import – Aktivierung).....	61
3.4.6.1.8	Schritt 8 (nur bei Erstellung AOE): Erstellung des Defaultbereichs (Anpassung Vorlage).....	62
3.4.6.1.9	Schritt 9 (nur bei Erstellung AOE): Konfigurationsdatei für Defaultbereich erstellen (Import – Aktivierung) .....	62
3.4.6.1.10	Schritt 10: Konfigurationsbereiche zur Aktivierung freigeben (Freigabe Aktivierung) .....	63
3.4.6.2	Vorlage für eine Versorgungsdatei für Modell- und/oder Objektdefinitionen .....	63
3.4.6.3	Modellierungsmöglichkeiten (Datenmodell, Metamodell) .....	63
3.4.7	Übernahme der Änderungen .....	64
3.5	Weitergeben von Konfigurationen .....	65
3.5.1	Weitergabe mit „Freigabe zur Übernahme“ .....	66
3.5.2	Weitergabe mit „Freigabe zur Aktivierung“ .....	66
3.5.3	Weitergabe mit „Freigabe zur Aktivierung ohne lokale Aktivierung durch den KV“ .....	66
3.5.4	Zusammenstellung eines KV zur Weitergabe an die ZS .....	66
3.6	Übernehmen von Konfigurationen .....	67
3.6.1	Anpassung der Verwaltungsdatei .....	68
3.6.2	Aktivierung einer übernommenen Konfiguration .....	69
<b>4</b>	<b>Teil C – Workflow bei der Konfiguration</b> .....	<b>70</b>
4.1	Organisation einer Konfigurationsumgebung .....	70
4.2	Komplexe Workflow .....	70
4.3	Sonderfälle beim Workflow .....	70
<b>5</b>	<b>Teil D – Richtlinien, Fehlersituationen und Lösungen</b> .....	<b>71</b>
5.1	Richtlinien .....	71
5.2	Fehlersituationen .....	71
<b>6</b>	<b>Anhang</b> .....	<b>72</b>
6.1	Übersichtsdarstellung „BSVRZ Tutorial Konfiguration - Standardworkflow“ .....	72

## 0.4 Abkürzungsverzeichnis

siehe [AbkBLAK]

ZS	<b>Z</b> entrale <b>S</b> telle der Länder zur Verwaltung des BSVRZ (vormals ZID, Zentraler Informationsdienst des Bundes).
BSVRZ	<b>B</b> asis- <b>S</b> ystem <b>V</b> erkehrsrechnerzentralen, Softwaresystem auf Basis des Datenverteilersystems entsprechend den Vorgaben des AK-VRZ
AK-VRZ	<b>A</b> rbeitskreis <b>V</b> erkehrsrechnerzentralen

## 0.5 Definitionen

Keine

## 0.6 Referenzierte Dokumente

AbkBLAK	Abkürzungsverzeichnis zum VRZ-Basissystem gemäß BLAK-VRZ, aktueller Stand SE-02.00.00.00.00-Abk
VModell	V-Modell – Entwicklungsstandard für IT-Systeme (EstdIT), Version 1997.
AFoBLAK	Anwenderforderungen zum VRZ-Basissystem gemäß BLAK-VRZ, aktueller Stand SE-02.00.00.00.00-AFo.
SysArcBLAK	Systemarchitektur zum VRZ-Basissystem gemäß BLAK-VRZ, aktueller Stand SE-02.00.00.00.00-SysArc.
TutorialBSVRZ	Tutorial zur Kernsoftware des BSVRZ, aktueller Stand V 1.0 vom 08.09.2006.
K2SDTD	Document Type Definition (DTD) zum Datenkatalog des VRZ-Basissystem gemäß BLAK-VRZ, aktueller Stand - Datei K2S.dtd, Bestandteil des Kernsoftware Pakets de.bsvrz.puk.config.xmlFile.resolver.K2S.dtd

## 0.7 Abbildungsverzeichnis

Abbildung 2-1: Konfigurationsapplikation und Konfiguration im Datenverteilersystem .....	11
Abbildung 2-2: Weitere Funktionen der Konfigurationsapplikation .....	16
Abbildung 3-1: Zu verwendende Elemente der [K2SDTD] .....	21
Abbildung 3-2: Kopfbereich einer Versorgungsdatei .....	22
Abbildung 3-3: Aufbau eines Konfigurationsbereichs bzw. einer Versorgungsdatei .....	22
Abbildung 3-4: Aufbau der Versorgungsdatei am Beispiel des Teilmodells Verkehr Global .....	23
Abbildung 3-5: Verwaltungsdatei.....	25
Abbildung 3-6: Abhängigkeiten des kb.tmVerkehrGlobal in der Version 25 Eintrag in der Versorgungsdatei nach dem Export.....	33
Abbildung 3-7: Einrichten von Konfigurationsverantwortlichen(KV oder AOE).....	57
Abbildung 3-8: Vorlage einer Versorgungsdatei für einen KV (kb.kv.ORG.ZUSATZ.xml) .....	58

Abbildung 3-9: Vorlage einer Versorgungsdatei für eine AOE (kb.kv.aoe.ORG.ZUSATZ.xml) ....	59
Abbildung 3-10: Vorlage der Verwaltungsdatei für KV und AOE (verwaltungsdaten.xml).....	61
Abbildung 3-11: Versorgungsdatei Default-Bereich für eine AOE (kb.default.kv.aoe.ORG.ZUSATZ.xml) .....	62
Abbildung 3-12: Versorgungsdatei Vorlage für Modell- und/oder Objektdefinitionen(kb.tmModellXyz.xml bzw. kb.objekteXyz.xml) .....	63

## 0.8 Tabellenverzeichnis

Tabelle 0-1: Verteiler .....	2
Tabelle 0-2: Änderungsübersicht .....	2
Tabelle 3-1: Konfigurationsbereiche des kv.kappich .....	28
Tabelle 3-2: Konfigurationsbereiche des kv.inovat .....	29
Tabelle 3-3: Konfigurationsbereiche des kv.bea .....	29
Tabelle 3-4: Konfigurationsbereiche des kv.bitctrl .....	30
Tabelle 3-5: Konfigurationsbereiche des kv.dambach .....	31
Tabelle 3-6: Konfigurationsbereiche des kv.logos .....	31
Tabelle 3-7: Konfigurationsbereiche des kv.ptv .....	32
Tabelle 3-8: Beispiele für Konfigurationsverantwortliche /AOE .....	35
Tabelle 3-9: Länderkürzel für Ortsbezüge.....	36
Tabelle 3-10: Übersicht der relevanten Aufrufparameter der Konfigurationsapplikation .....	40
Tabelle 3-11: Arbeitsablauf für die Modellpflege.....	49
Tabelle 3-12: Arbeitsablauf für die Objektpflege .....	51
Tabelle 3-13: Arbeitsablauf für den Betrieb (AOE, Defaultbereich) .....	52
Tabelle 3-14: Arbeitsablauf für die Projektpflege .....	54

# 1 Zusammenfassung

Von Seiten der Anwender des Basis-Systems VRZ (BSVRZ) besteht – alleine schon auf Grund der hohen Komplexität des Systems – Bedarf an einführenden Dokumentationen (Tutorials) zu den wesentlichen Komponenten des BSVRZ. Für die eigentliche Kernsoftware wurde daher bereits ein „Tutorial Kernsoftware“ erstellt und über die ZS den Anwendern zur Verfügung gestellt.

Mit der Einführung des neuen Release des Basis-Systems VRZ, welches erstmals die endgültige Version der neuen Konfiguration enthält, müssen alle Anwender des Systems bestehende Systeme auf diese neue Konfiguration umstellen und neu zu erstellende Systeme mit dieser Konfiguration betreiben. Die zur Konfiguration existierende Dokumentation auf Basis der V-Modell-Dokumente ist als Einstieg und Leitfaden für Anwender nicht geeignet.

Ziel dieses „BSVRZ Tutorial Konfiguration“ ist es, den Einstieg in den Umgang mit der Konfiguration zu vereinfachen.

Dazu ist das Gesamttutorial in vier aufeinander aufbauende Teilen gegliedert:

- Teil A: Schnelleinstieg Konfiguration
- Teil B: Grundlagen der Konfiguration
- Teil C: Workflow bei der Konfiguration
- Teil D: Anhang mit Richtlinien und Fehlersituationsbeschreibung

Ziel des ersten Teils des Tutorials ist es, dem Leser einen schnellen Überblick über die Aufgaben und die Grundbegriffe der Konfiguration sowie die Bedeutung der Konfiguration innerhalb des BSVRZ zu geben.

Ziel des zweiten Teils des Tutorials ist die detaillierte Beschreibung der einzelnen Bestandteile einer Konfiguration, dem Zusammenhang zwischen diesen Komponenten und den notwendigen Arbeitsschritten im Umgang mit der Konfiguration.

Der zweite Teil enthält auch die detaillierte Beschreibung für den Standardworkflow im Umgang mit der Konfiguration für die Bereiche Modellpflege, Objektpflege, Betrieb und Projekte.

Ziel des dritten Teils des Tutorials ist die Darstellung der genauen Arbeitsabläufe, die beim Umgang mit der Konfiguration notwendig sind. Dabei wird zuerst die Organisation einer in der Praxis erprobten Konfigurationsumgebung dargestellt. Darauf aufbauend werden komplexere Abläufe und ausgewählte Sonderfälle, die in der Praxis bei projekt- und firmenübergreifenden Konfigurationsänderungen durchgeführt werden müssen, beschrieben.

Im letzten Teil des Tutorials werden Richtlinien, die bei der Erstellung von Konfigurationen eingehalten werden müssen, beschrieben. Des Weiteren werden hier häufig auftretenden Fehlersituationen und deren Lösungsmöglichkeiten beschrieben.

## Hinweis

Das vorliegende Tutorial umfasst die Teile A + B.

## 2 Teil A – Schnelleinstieg Konfiguration

### Teil A – Schnelleinstieg Konfiguration →

- Konfiguration / Konfigurationsapplikation
- Überblick über die Grundlagen der Konfiguration.
- Überblick über die Grundfunktionen der Konfigurationsapplikation

### 2.1 Überblick Konfiguration

#### 2.1.1 Konfiguration / Konfigurationsapplikation

Mit dem Begriff Konfiguration werden synonym zwei unterschiedliche Dinge bezeichnet:

- Konfigurationsapplikation
- Konfigurationsdaten

Zum einen ist mit der Konfiguration die **Applikation Konfiguration** im Kernsystem gemeint. Diese sorgt unter anderem dafür, dass der Datenverteiler und die angebotenen Applikationen Zugriff auf die notwendigen Konfigurationsinformationen haben.

Zum anderen bezeichnet Konfiguration die **Konfigurationsdaten** (Datenmodell(e), Versorgungsdaten) selbst, die von der Konfigurationsapplikation verwaltet werden und dem Datenverteiler und angebotenen Applikationen auf Anfrage zur Verfügung gestellt werden.

Im Folgenden wird immer explizit von der **Konfigurationsapplikation** gesprochen, wenn die Applikation Konfiguration gemeint ist, ansonsten wird der Begriff Konfiguration als Synonym für die Konfigurationsdaten verwendet.

Abbildung 2-1 zeigt den Zusammenhang zwischen der Konfigurationsapplikation und der Konfiguration im Datenverteilersystem.

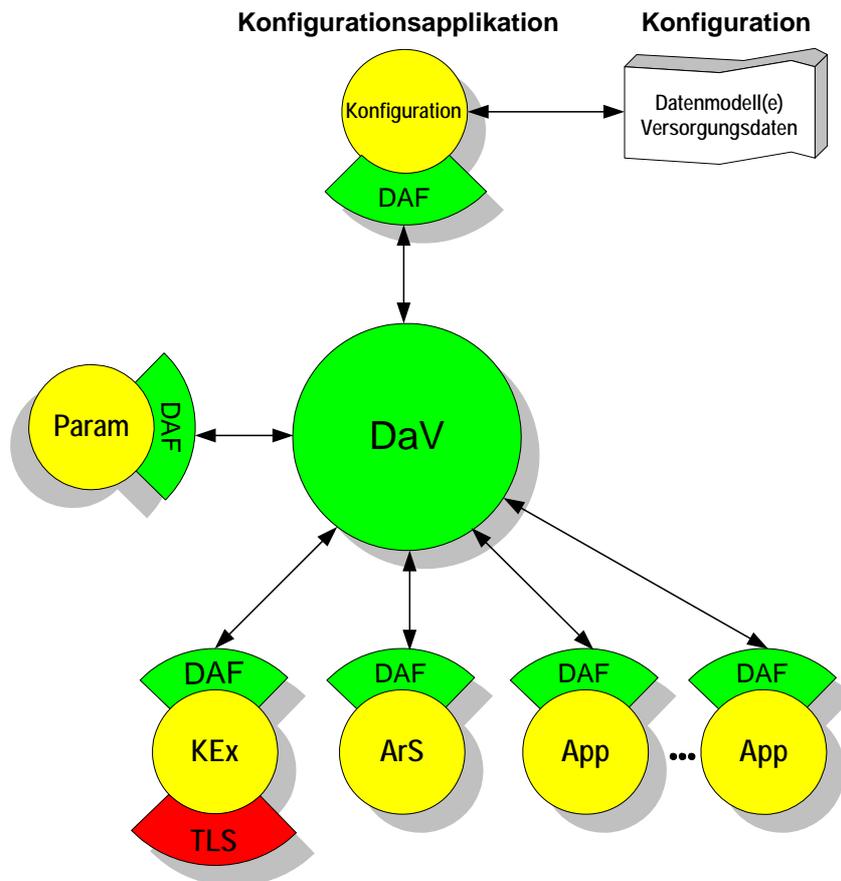


Abbildung 2-1: Konfigurationsapplikation und Konfiguration im Datenverteilersystem

Das BSVRZ besteht neben den Applikationen aus dem Datenverteiler, der **Konfigurationsapplikation** und der Parametrierung. Die Konfigurationsapplikation und die Parametrierung verwalten die Konfiguration(sdaten) und die Parameterdaten, die von den einzelnen Applikationen des Systems benötigt werden.

Ein wesentlicher Aspekt der Architektur des Datenverteilersystems ist die Möglichkeit, Datenverteiler miteinander zu koppeln. Jede Verkehrsrechnerzentrale und Unterzentrale stellt ein System (Autarke Organisationseinheit **AOE**, welches aus mehreren Rechnern bestehen kann) dar, welches jeweils eine eigene Konfiguration (Konfigurationsapplikation und Konfigurationsdaten) hat.

Damit zwischen diesen System Daten ausgetauscht werden können, müssen die Konfigurationsapplikationen für die entsprechenden konkreten Objekte (z.B. Messquerschnitte) und die entsprechenden Beschreibungen der Daten (z.B. Attributgruppe Verkehrsdaten) die gleichen **Konfigurationen** (Konfigurationsdaten) zur Verfügung haben.

Die Sicherstellung der Identität der Konfigurationen, die in unterschiedlichen Systemen eingesetzt werden, wird durch die Aufteilung der Konfiguration in **Konfigurationsbereiche** und die Zuordnung zu **Konfigurationsverantwortlichen** sichergestellt.

## 2.1.2 Grundbegriffe der Konfiguration

### 2.1.2.1 Modelle

#### 2.1.2.1.1 Datenmodell

Aufgabe der Modellbildung ist die Abbildung der realen Welt auf ein Modell, mit dem alle wichtigen Belange dargestellt werden.

Das **Datenmodell** legt die Sicht des Systems auf die reale Welt fest. Ein Gegenstand oder Konzept der realen Welt und dessen Beziehungen müssen im Datenmodell definiert werden, damit die Konfigurationsapplikation und die darauf basierenden Applikationen damit arbeiten können. Beispielsweise muss jeder einzelne Messquerschnitt, der Daten über den Datenverteiler senden soll, im Datenmodell eingetragen werden.

Das vom BSVRZ verwendete Datenmodell unterscheidet zwei grundsätzliche Modelltypen:

- Metamodel
- Systemmodell / Fachmodell

#### 2.1.2.1.2 Metamodel

Das Metadatenmodell definiert die zur Modellierung verfügbaren Elemente wie Objekttypen, Attributgruppen, Attributtypen etc. Diese bilden die Basis für die Definition der System- /Fachmodelle. Die folgende Abbildung zeigt die wichtigsten Bausteine des Datenmodells:

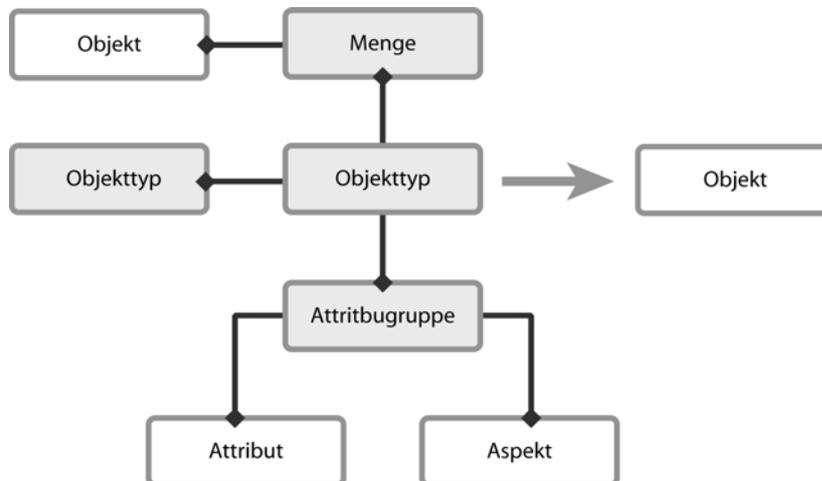


Abbildung 2-1: Bausteine des Datenmodells

- **Objekt:** Ein Objekt repräsentiert einen bestimmten Gegenstand oder ein Konzept aus der realen Welt, z.B. einen bestimmten Messquerschnitt oder Stau. Die Eigenschaften und Struktur eines Objekts werden durch den ihm zugeordneten Objekttyp festgelegt. Ein Objekt ist also eine Instanz, d.h. eine konkrete Ausprägung eines Objekttyps. Jedes Objekt basiert auf genau einem Objekttyp.
- **Objekttyp:** Der Objekttyp eines Objekts legt dessen Eigenschaften und Beziehungen zu anderen Elementen des Datenmodells fest.
- **Attributgruppe:** Die Attributgruppe fasst Attribute und Attributlisten in Gruppen zusammen und legt so die Struktur von Datensätzen fest.

- **Attributliste:** Die Attributliste fasst Attribute in Gruppen und ermöglicht so, diese Zusammenstellung in unterschiedlichen Attributgruppen wieder zu verwenden.
- **Attribut:** Ein Attribut repräsentiert ein einzelnes Datum innerhalb einer Attributliste bzw. einer Attributgruppe.
- **Aspekt:** Der Aspekt bezeichnet eine Sicht auf Daten, etwa, ob es sich um aggregierte Daten handelt.
- **Menge:** Eine Menge enthält (Referenzen auf) Objekte, so dass Beziehungen zwischen Objekten modelliert werden können.

### 2.1.2.1.3 Systemmodell / Fachmodell

Die Fachmodelle<sup>1</sup> setzen fachliche Anforderungen in Objekttypen, Beziehungen und Datenstrukturen um. So sind z.B. im Teilmodell `TeilmodellVerkehrGlobal` die Zusammenhänge wichtiger verkehrlicher Infrastrukturelemente beschrieben. Hier erfolgt unter anderem die Definition von Messquerschnitten und die Definition der Datenstrukturen (Attributgruppen), die für Messquerschnitte relevant sind (z. B. Verkehrsdaten (Kurzzeit)).

### 2.1.2.2 Versorgung (Objektdefinition)

Bei der Versorgung findet die konkrete Abbildung der realen Welt auf Objekte statt.

### 2.1.2.3 Konfigurationsbereiche

In der Konfiguration werden verschiedene Konzepte (Modelle und Objektdefinitionen) zusammengefasst. Die Definition der Modelle und Objekte erfolgt dabei in **Konfigurationsbereichen**.

Über Konfigurationsbereiche erfolgt die logische und physische Aufteilung einer Gesamtkonfiguration aus Modellen und Objekten. Ein Konfigurationsbereich besteht dabei physisch aus einer Datei<sup>2</sup>.

Die Aufteilung der gesamten Konfiguration in Konfigurationsbereiche erfolgt nach fachlichen, versorgungstechnischen und verwaltungstechnischen Kriterien.

Innerhalb eines Konfigurationsbereichs wird zwischen dem Modellbereich und dem Objektbereich unterschieden.

In der Regel werden für die Datenmodellierung (Modellbereich in einem Konfigurationsbereich) und die Versorgung (Objektbereich in einem Konfigurationsbereich) unterschiedliche Konfigurationsbereiche angelegt.

Die Aufteilung der Gesamtkonfiguration in einzelne Konfigurationsbereiche erfolgt dabei im Wesentlichen nach folgenden Kriterien:

- **Konfigurationsverantwortlicher**  
Jeder Konfigurationsbereich ist einem Konfigurationsverantwortlichen zugeordnet. Nur dieser kann den Konfigurationsbereich ändern und ist für die Pflege zuständig. Die Konfigurationsbereiche müssen also in jedem Fall sinnvoll nach den Konfigurationsverantwortlichen aufgeteilt werden.
- **Trennung Modelldefinitionen und Objektdefinitionen**  
In der Regel sollten die Datenmodelle und die konkreten Objektdefinitionen in unterschiedlichen

---

<sup>1</sup> Das Systemmodell ist faktisch auch ein Fachmodell. Allerdings wurden hier die systemübergreifenden Modellierungen zusammengefasst (wie. z. B. die Benutzerverwaltung etc.).

<sup>2</sup> Genau genommen sind es zwei Dateien: Zum einen die **Versorgungsdatei** (XML-Format, \*.xml-Datei), die durch den Anwender bearbeitet werden kann, zum anderen eine daraus generierte **Konfigurationsdatei** (BINÄR-Format, \*.config-Datei), die ausschließlich von der Konfigurationsapplikation verwendet wird.

Konfigurationsbereichen erfolgen. Dies ergibt sich i. d. R: bereits aus den unterschiedlichen Verantwortlichkeiten für die Modellpflege (projektübergreifend) und die konkrete Anlagenversorgung (projekt- / hersteller- / oranisationspezifisch).

- **Austausch / Weitergabe von Konfigurationsbreichen**

Die Konfigurationsdaten werden i. d. R. in verschiedenen AOE (Autarken Organisationseinheiten wie VRZ, UZ) benötigt. Konfigurationsbereiche sollten deshalb auch nach potenziell auszutauschenden Konfigurationsinformationen aufgeteilt werden.

- **Quelle von Versorgungsdaten**

Insbesondere bei der Erstellung der Objektdefinitionen für ein Projekt müssen häufig Massendaten (TLS-Versorgungen, Karteninformationen eines Netzes, Messquerschnitte etc.) in einen Konfigurationsbereich eingetragen werden. Um eine automatische Generierung zu ermöglichen, sollten Konfigurationsbereiche auch abhängig von der Informationsquelle aufgeteilt werden<sup>3</sup>.

#### Anmerkung

- Die unterschiedlichen Aufteilungen der Gesamtkonfiguration auf Konfigurationsbereiche dienen lediglich der logischen Strukturierung.
- Die logische Strukturierung der Konfigurationsbereiche kann sich überschneiden. So kann ein Konfigurationsbereich sowohl Modell- als auch Objektbereiche enthalten. Ebenso können sich Objektdefinitionen oder modellspezifische Bereiche über mehrere Konfigurationsbereiche erstrecken.

#### Tipp

- In einem Konfigurationsbereich sollte i. d. R. entweder ein Modell oder konkrete Objekte für eine Versorgung definiert werden.
- Modelldaten und Projekt- /anlagenspezifische Daten sollten jeweils unter eigenen Konfigurationsverantwortlichen erstellt werden.

### 2.1.2.4 Konfigurationsverantwortliche

Jedem Konfigurationsbereich ist ein Konfigurationsverantwortlicher zugeordnet. Nur die Konfigurationsapplikation, die unter dem Konfigurationsverantwortlichen des Konfigurationsbereichs gestartet wurde, hat die Möglichkeit, die enthaltenen Konfigurationsdaten zu ändern oder zu erweitern.

Der Konfigurationsverantwortliche ist seinerseits auch ein Konfigurationsobjekt, das in einem Konfigurationsbereich versorgt werden muss. Da die Konfigurationsapplikation potentiell unter jedem Konfigurationsverantwortlichen gestartet werden kann, muss der Konfigurationsverantwortliche in einem eigenen Konfigurationsbereich definiert werden.

#### Anmerkung

<sup>3</sup> Dies ist auch eine Aufteilung nach der Verantwortlichkeit, hier aber der Verantwortlichkeit für die Erstellung und nicht die der Pflege (→ Konfigurationsverantwortlicher).

- **Konfigurationsverantwortliche** werden für die **Definition** von Modellen und Objekten verwendet.

### 2.1.2.5 Autarke Organisationseinheit (AOE)

Ein wesentlicher Aspekt der Architektur des Datenverteilersystems ist die Möglichkeit, Datenverteiler miteinander zu koppeln. Jede Verkehrsrechnerzentrale und Unterzentrale stellt ein System dar, welches aus einem oder mehreren Rechnern bestehen kann. Jedes dieser Systeme ist unabhängig (autark) von anderen Systemen, mit denen es über den Datenverteiler verbundenen ist<sup>4</sup>. Ein solches unabhängiges System wird im BSVRZ als **Autarke Organisationseinheit (AOE)** bezeichnet. Jede AOE hat (mindestens) einen eigenen Datenverteiler, genau eine eigene Konfigurationsapplikation und eine eigene Konfiguration.

Analog zum Konfigurationsverantwortlichen ist auch die AOE ein eigenes Konfigurationsobjekt. Die AOE ist vom Konfigurationsverantwortlichen abgeleitet, d. h. ein

- **AOE ist auch ein Konfigurationsverantwortlicher**<sup>5</sup>.

Es muss unterschieden werden, ob der ein Konfigurationsverantwortlicher für die Datenmodellierung und Versorgung verwendet werden soll, oder ob der Konfigurationsverantwortliche einer Laufzeitumgebung eines Systems mit eigener Konfiguration für den Betrieb zugeordnet werden soll. Im letzteren Fall wird dann AOE verwendet.

#### Anmerkung

- **AOE** werden für den **Betrieb** einer Konfiguration verwendet, d. h. jeweils für ein konkretes **Projekt**.

### 2.1.2.6 Verwaltungsdatei

Damit die Konfigurationsapplikation weiß, welche Konfigurationsbereiche unter welchem Konfigurationsverantwortlichen verwaltet werden müssen, verwendet diese die Verwaltungsdatei (Datei: verwaltungsdaten.xml)

In der Verwaltungsdatei werden für die jeweilige Umgebung der Konfiguration dokumentiert, welche Versionen der einzelnen Konfigurationsbereich lokal für diese Konfiguration aktuell aktiviert sind und welche Versionen wann aktiv waren → Kapitel 3.1.4.5.

## 2.1.3 Konfigurationsapplikation

### 2.1.3.1 Anwendungsbetrieb

Im Anwendungsbetrieb arbeitet die Konfiguration als dienstleistender Hintergrundprozess für andere Applikationen des Systems. Sie verwaltet die Konfigurationsdaten und beantwortet Konfigurationsanfragen von Applikationen und ändert die Konfigurationsdaten im Auftrag von anderen Applikationen (z.B. bei dynamischen Objekten). Weiter prüft die Konfiguration die Authentifizierung von sich anmeldenden Applikationen (Benutzern) im Auftrag des Datenverteilers.

<sup>4</sup> Autark bedeutet in diesem Zusammenhang, dass jedes System für sich gestartet und betrieben werden kann. Das durch die fehlende Kopplung Einschränkungen der Funktionen gegeben sein können, ist natürlich möglich.

<sup>5</sup> Aber nicht umgekehrt! Ein AOE ist also ein Konfigurationsverantwortlicher mit zusätzlichen Eigenschaften. Die Details werden im Teil B beschrieben.

Die Beschreibung der Funktionen und die Nutzung der Konfigurationsapplikation im Anwendungsbetrieb sind nicht Bestandteil dieses Tutorial sondern werden im Tutorial [TutorialBSVRZ] näher erläutert.

### 2.1.3.2 Konfigurationsbetrieb

Abbildung 2-2 skizziert weitere Funktionen der Konfiguration und gibt gleichzeitig einen Überblick von Teilen des Workflow für die Pflege der Konfiguration.

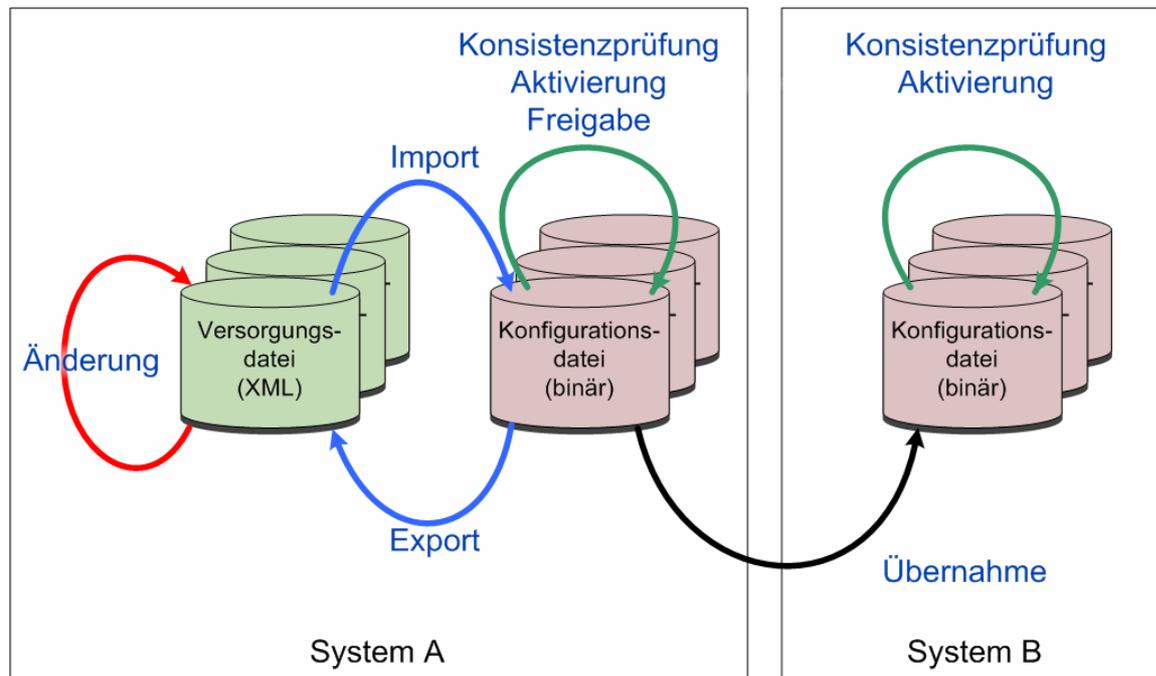


Abbildung 2-2: Weitere Funktionen der Konfigurationsapplikation

Die Konfiguration eines Datenverteilersystems benötigt grundsätzlich bestimmte Konfigurationsbereiche, um ihre Arbeit überhaupt verrichten zu können. Hierzu gehören u. a. die Konfigurationsbereiche mit dem Metadatenmodell, mit dem globalen Fachmodell und dem Systemdatenmodell.

Um das Datenverteilersystem an die entsprechende Umgebung anzupassen, muss die Konfiguration um weitere Konfigurationsbereiche erweitert werden. Z. B. kann in System A eine Versorgungsdatei für den neuen Konfigurationsbereich X erstellt werden, in der die Objekte aus dem TLS-Umfeld (SM, EAK, DE) versorgt sind. Durch die Import-Funktion wird eine entsprechende Konfigurationsdatei erzeugt. Wenn dies fehlerfrei funktioniert hat, kann die Änderung der Konfiguration durch die Aktivieren-Funktion aktiviert werden und beim nächsten Systemstart können alle Applikationen auf die neue Version zugreifen.

Jede Konfigurationsdatei kann durch die Export-Funktion in eine entsprechende Versorgungsdatei, die den aktuellen Stand des Konfigurationsbereichs enthält, umgewandelt werden. Wenn bei dem Konfigurationsbereich der Konfigurationsverantwortliche mit dem übereinstimmt, unter dem die Konfiguration gestartet wird, können Änderungen gemacht werden. Nach der Überarbeitung kann die neue Version wieder importiert und aktiviert werden.

Damit ein System B den Konfigurationsbereich X von dem System A übernehmen kann, muss der Konfigurationsbereich in System A zur Übernahme/Aktivierung freigegeben werden. Die Freigabe erfolgt über eine weitere Funktion der Konfiguration.

Nach der Freigabe zur Übernahme/Aktivierung kann der Konfigurationsbereich X von System B übernommen werden. Dazu wird die entsprechende Konfigurationsdatei übernommen (kopiert), in den Verwaltungsdaten eingetragen und die Konfiguration in System B wird aktiviert.

Nachfolgend sind die Funktionen beschrieben, welche für die Konfigurationsapplikation im Konfigurationsbetrieb zur Verfügung stehen.

#### 2.1.3.2.1 Import

Beim Import werden einzelne oder mehrere, neu angelegte oder geänderte, Versorgungsdateien (\*.xml-Dateien) in die Konfiguration übernommen. Dabei werden Änderungen in die bestehenden korrespondierenden Konfigurationsdateien (\*.config-Dateien) übernommen bzw. neu erstellte Versorgungsdateien werden in neue Konfigurationsdateien konvertiert.

#### 2.1.3.2.2 Konsistenzprüfung

Die Konsistenzprüfung prüft die neuste Version der eigenen<sup>6</sup> Konfigurationsbereiche und freigegebene Versionen von fremden<sup>7</sup> Konfigurationsbereichen. Dabei werden die Konfigurationsdaten u. a. auf Fehler / Warnungen geprüft:

#### 2.1.3.2.3 Aktivierung

Damit die importierten Änderungen für die Konfigurationsapplikation im Anwendungsbetrieb verfügbar sind, müssen diese aktiviert werden.

Durch die Aktivierung stehen beim nächsten Start der Konfigurationsapplikation die neusten Versionen der eigenen Bereiche und die neusten zur Aktivierung freigegebenen Versionen von übernommenen Konfigurationsbereichen bereit.

Bei der Funktion Aktivierung wird die Funktion Konsistenzprüfung, die auch alleine aufgerufen werden kann, implizit durchgeführt.

Bei der Aktivierung werden die Versionsnummern in der Verwaltungsdatei und in Konfigurationsdateien von eigenen Bereichen aktualisiert. Die Aktivierung wird mit dem nächsten Systemstart abgeschlossen.

#### 2.1.3.2.4 Export

Die Export-Funktion erzeugt aus einzelnen oder mehreren Konfigurationsdateien entsprechende Versorgungsdateien (Konvertierung der \*.config-Dateien in \*.xml-Dateien). Diese Versorgungsdateien enthalten die Definition von Typen, Attributgruppen etc. sowie die Objekte des Konfigurationsbereichs jeweils nach PID sortiert. Dabei sind nur Informationen enthalten, die von außen vorgegeben werden können (also z. B. keine Objekt-ID's). Die Versorgungsdateien enthalten zudem keine Historie. Exportiert wird also genau die neuste Version des entsprechenden Konfigurationsbereichs.

#### 2.1.3.2.5 Freigabe

Bei der Funktion Freigabe werden die Varianten

- Freigabe zur Übernahme
- Freigabe zur Aktivierung
- Freigabe zur Aktivierung ohne lokale Aktivierung durch den KV

unterschieden.

---

<sup>6</sup> Konfigurationsbereiche, die unter der Verwaltung des aktuellen Konfigurationsverantwortlichen liegen → Verwaltungsdatei.

<sup>7</sup> Konfigurationsbereiche, die von anderen Konfigurationsverantwortlichen übernommen wurden.

Abhängig von der gewählten Variante werden die freigegebenen Bereich intern so gekennzeichnet, das eine Verwendung des Bereichs durch andere Konfigurationsverantwortliche in deren Konfiguration ermöglicht wird. Durch die Variante kann die Nutzung durch den freigebenden Konfigurationsverantwortlichen bestimmt werden.

## 3 Teil B – Grundlagen der Konfiguration

### Teil B – Grundlagen Konfiguration →

- Konfigurationsbereiche im Detail
- Konfigurationsverantwortliche und AOE.
- Konfigurationsapplikation im Detail
- Standardworkflow
  - Modelle
  - Versorgung
  - Betrieb
  - Projekte
- Weitergabe / Übernahme von Konfigurationen

### 3.1 Konfigurationsbereiche im Detail

#### 3.1.1 Historie

Bis zur Version 3.0 der Kernsoftware wurden die Konfigurationsdaten von der Konfigurationsapplikation in einer einzigen xml-Datei, der `config.xml`, gespeichert und verwaltet.

Seit der Version 3.0 der Kernsoftware werden die Konfigurationsinformationen je Konfigurationsbereich jeweils in eigenen Dateien (Konfigurationsdateien) gespeichert. Diese Dateien sind im Binärformat und haben die Endung `.config`

Das vorliegende Tutorial behandelt ausschließlich die Konfiguration ab Version 3.0.

#### 3.1.2 Konfigurationsverantwortlicher - Konfigurationsbereich

Jedem Konfigurationsbereich ist ein Konfigurationsverantwortlicher zugeordnet. Der Konfigurationsverantwortliche legt fest, welche Konfigurationsapplikation die Daten eines Konfigurationsbereiches ändern darf. Konfigurationsbereiche können nur durch die Konfigurationsapplikation geändert werden, die unter demselben Konfigurationsverantwortlichen gestartet wurde, der als Konfigurationsverantwortlicher für den zu ändernden Konfigurationsbereich eingetragen ist.

#### 3.1.3 Konfigurationsdateien

Je Konfigurationsbereich gibt es genau eine **Konfigurationsdatei**. Der Name der Datei wird dabei aus der PID des Konfigurationsbereichs ergänzt um die Endung `.config` gebildet. Diese Konfigurationsdatei enthält alle Objekte und konfigurierenden Datensätze des Konfigurationsbereichs.

Die Konfigurationsdatei enthält sämtliche Versionen, d. h. die gesamte Historie, des jeweiligen Konfigurationsbereichs. Dadurch ist es im Anwendungsbetrieb der Konfiguration möglich, auch auf Informationen und Objekte zuzugreifen, die in der aktuellen Version modifiziert wurden oder die aktuell nicht mehr existieren. Dadurch wird der Zugriff auf historische Daten (Archivdaten) auch für solche Objekte gewährleistet, die aktuell nicht mehr existieren.

Die Datei ist im Binärformat und hat Indices für den schnellen Zugriff auf die enthaltenen Konfigurationsobjekte über die Objekt-ID und die PID.

Die Konfigurationsapplikation stellt sicher, dass die Objekt-ID's weltweit eindeutig vergeben werden. Dazu wird u. a. die eindeutige Kodierung des Konfigurationsverantwortlichen verwendet.

Durch die Konfigurationsapplikation erfolgt eine automatische Reorganisation der Datei bei Änderungen an der Konfigurationsdatei.

Die Konfigurationsdateien werden ausschließlich durch die Konfigurationsapplikation auf Basis der Daten der zugehörigen Versorgungsdatei erzeugt und geändert.

### 3.1.4 Versorgungsdateien

#### 3.1.4.1 Namensgebung

Je Konfigurationsbereich gibt es neben der Konfigurationsdatei genau eine **Versorgungsdatei**. Der Name der Versorgungsdatei muss dabei aus der PID des Konfigurationsbereichs ergänzt um die Endung `.xml` gebildet werden.

Beispiel: Die nachfolgende Abbildung zeigt einen Ausschnitt aus der Versorgungsdatei des Konfigurationsbereichs „Teilmodell Verkehr Global“.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE konfigurationsBereich PUBLIC "-//K2S//DTD Dokument//DE" "K2S.dtd">
] <konfigurationsBereich pid="kb.tmVerkehrGlobal" name="TeilmodellVerkehrGlobal" verantwortlich="kv.inovat">
]   <info...>
```

Die PID des Konfigurationsbereichs lautet `kb.tmVerkehrGlobal` (gelb hinterlegt). Daraus ergeben sich die Namen der Versorgungsdatei bzw. der Konfigurationsdatei wie folgt:

PID	<code>kb.tmVerkehrGlobal</code>	Wird durch den Anwender festgelegt
-----	---------------------------------	------------------------------------

Versorgungsdatei	<code>kb.tmVerkehrGlobal.xml</code>	Ist durch den Anwender sicherzustellen → sonst Fehlermeldung beim Import durch die Konfigurationsapplikation.
Konfigurationsdatei	<code>kb.tmVerkehrGlobal.config</code>	Wird durch die Konfigurationsapplikation erzeugt, diese stellt Einhaltung der Namenskonvention sicher.

### 3.1.4.2 Inhalt

Die Versorgungsdateien sind XML-Dateien, über der der Anwender Konfigurationen erstellen und pflegen kann. Die Versorgungsdatei enthält dabei immer nur den aktuellen (neusten) Stand eines Konfigurationsbereichs. Des Weiteren sind in den Versorgungsdateien nur änderbare Informationen des Konfigurationsbereichs enthalten. Informationen, die durch die Konfigurationsapplikation bei der Übernahme (Import + Aktivierung) einer Versorgungsdatei in die Konfigurationsdatei erzeugt werden (hier insbesondere die eindeutigen Objekt-ID's und die Historie der Änderungen) sind nicht in der Versorgungsdatei vorhanden.

Der aktuelle Stand eines Konfigurationsbereichs als Versorgungsdatei kann durch aus der Konfigurationsdatei über die Export-Funktion der Konfigurationsapplikation als erzeugt werden. Dabei wird die erzeugte Versorgungsdatei automatisch nach unterschiedlichen Kriterien einheitlich formatiert und sortiert ausgegeben. Zudem werden zusätzliche Informationen, die beim Import nicht unbedingt angegeben werden müssen, automatisch mit ihren Standardwerten ausgegeben.

Beim Export werden zudem die aktuelle Versionsnummer sowie die Abhängigkeitsübersicht zu anderen Konfigurationsbereichen in der erzeugten Versorgungsdatei mit ausgegeben.

### 3.1.4.3 DTD einer Versorgungsdatei

Die Versorgungsdatei ist eine XML-Datei, die den Vorgaben der [K2SDTD] für den `konfigurationsBereich` (→ DOCTYPE) genügen muss. Die nachfolgende Abbildung zeigt eine Übersicht der in der [K2SDTD] definierten Elemente. Für die Erstellung eines Konfigurationsbereichs muss lediglich der grün markierte Bereich Anwendung finden.

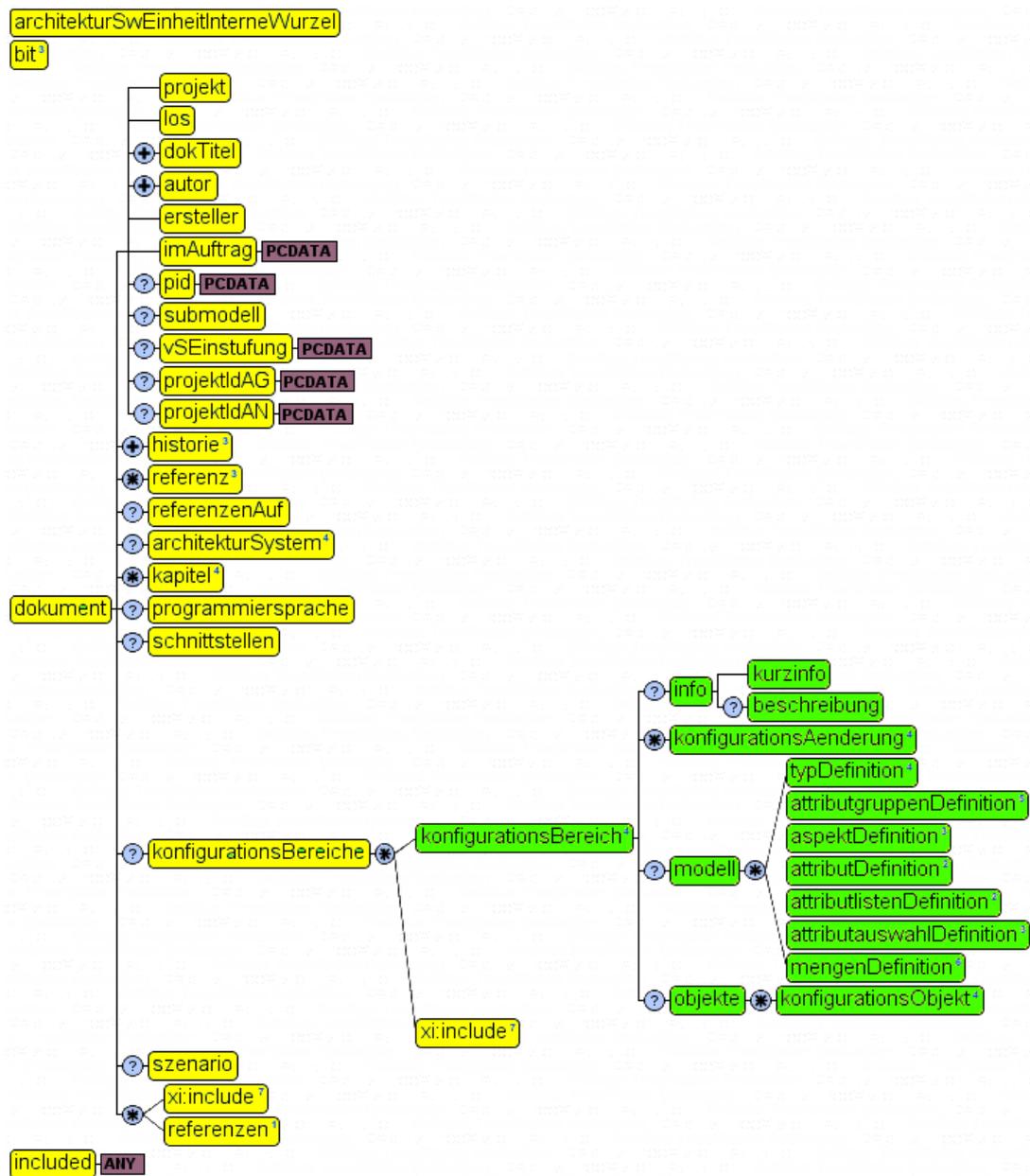


Abbildung 3-1: Zu verwendende Elemente der [K2SDTD]

Der Kopfbereich einer Konfigurationsdatei lautet damit wie folgt (hier am Beispiel des Konfigurationsbereichs des Teilmodells Verkehr Global, zu verwendender DOCTYPE gelb markiert):

```

<?xml version="1.0" encoding="ISO-8859-1" ?>
<DOCTYPE konfigurationsBereich PUBLIC "-//K2S//DTD Dokument//DE" "K2S.dtd">
<konfigurationsBereich pid="kb.tmVerkehrGlobal" name="TeilmodellVerkehrGlobal" verantwortlich="kv.inovat">

```

Abbildung 3-2: Kopfbereich einer Versorgungsdatei

### 3.1.4.4 Aufbau

Ein Konfigurationsbereich besteht aus den in Abbildung 3-3 dargestellten vier Blöcken:

1. Infobereich
2. Änderungsübersicht
3. Modellbereich
4. Objektbereich

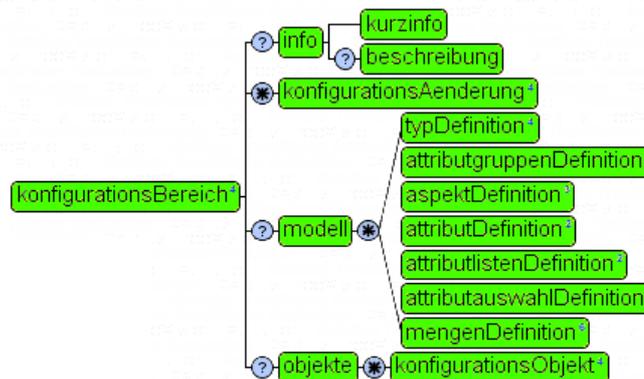


Abbildung 3-3: Aufbau eines Konfigurationsbereichs bzw. einer Versorgungsdatei

Die nachfolgende Abbildung zeigt diesen Aufbau am Beispiel der Versorgungsdatei der Teilmodells Verkehr Global (Info-, Modell- und Objektbereich sind in der Abbildung „zugeklappt“). Der Kommentar zu den Abhängigkeiten zwischen Änderungsübersicht und Modellbereich (grün dargestellt) wird durch die Konfigurationsapplikation automatisch erzeugt → Kapitel 3.1.6.2.

```

<?xml version="1.0" encoding="ISO-8859-1" ?>
<!DOCTYPE konfigurationsBereich PUBLIC "-//K2S/DTD Dokument/DE" "K2S.dtd">
<konfigurationsBereich pid="Kb.taVerkehrGlobal" name="TeilmodellVerkehrGlobal" verantwortlich="kv.inovat">
  <info>
    <konfigurationsaenderung stand="19.03.2007" version="5" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="A:Überarbeitung des Datenkatalogs für
    <konfigurationsaenderung stand="20.03.2007" version="10" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="B:Überarbeitung des Datenkatalogs für
    <konfigurationsaenderung stand="02.05.2007" version="11" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsdokumentation siehe 'Release
    <konfigurationsaenderung stand="01.06.2007" version="12" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsdokumentation siehe 'Release
    <konfigurationsaenderung stand="10.06.2007" version="13" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsdokumentation siehe 'Release
    <konfigurationsaenderung stand="14.06.2007" version="14" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsdokumentation siehe 'Release
    <konfigurationsaenderung stand="09.07.2007" version="15" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsdokumentation siehe 'Release
    <konfigurationsaenderung stand="28.09.2007" version="17" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsdokumentation siehe 'Release
    <konfigurationsaenderung stand="03.10.2007" version="18" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsdokumentation siehe 'Release
    <konfigurationsaenderung stand="23.11.2007" version="19" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsdokumentation siehe 'Release
    <konfigurationsaenderung stand="02.12.2007" version="20" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsdokumentation siehe 'Release
    <konfigurationsaenderung stand="19.12.2007" version="21" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsdokumentation siehe 'Release
    <konfigurationsaenderung stand="20.12.2007" version="22" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsdokumentation siehe 'Release
    <konfigurationsaenderung stand="17.01.2008" version="23" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsdokumentation siehe 'Release
    <konfigurationsaenderung stand="20.02.2008" version="25" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsdokumentation siehe 'Release
  </info>
  <!--
  Abhängigkeiten des Bereichs Kb.taVerkehrGlobal
  Abhängig ab Version Art der Abhängigkeit Version des benötigten Bereichs Benötigter Bereich
  23 notwendig 1 Kb.taKXTlsGlobal
  23 notwendig 1 Kb.kv.inovat
  23 notwendig 1 Kb.systemModellGlobal
  23 notwendig 1 Kb.taGeoReferenzierungGlobal
  23 notwendig 7 Kb.fachModellGlobal
  23 notwendig 4 Kb.metaModellGlobal
  23 notwendig 8 Kb.taGanglinienGlobal
  23 notwendig 2 Kb.taTmcGlobal
  24 notwendig 3 Kb.taFwBetriebGlobal
  24 notwendig 9 Kb.taGanglinienGlobal
  -->
  <modell>
  <objekte>
</konfigurationsBereich>

```

Abbildung 3-4: Aufbau der Versorgungsdatei am Beispiel des Teilmodells Verkehr Global

#### 3.1.4.4.1 Infobereich (→ info)

Enthält die Beschreibung des Konfigurationsbereichs. Hier sollte in jedem Fall eine aussagekräftige Dokumentation des Bereichs, ggf. mit grafischer Darstellung des Modells erfolgen.

#### 3.1.4.4.2 Änderungsübersicht (→ konfigurationsaenderung)

Enthält die Übersicht über die Änderungen an diesem Konfigurationsbereich. Die Änderungsübersicht sollte ebenfalls für jede Änderung am Konfigurationsbereich eingetragen werden. Neben der Angabe von Stand, Autor und Grund der Änderung, die durch den Anwender eingetragen werden, sollte die Versionsangabe immer leer bleiben (Angabe von `version=""`). Wird die Versionsnummer nämlich NICHT durch den Anwender eingetragen, so fügt die Konfigurationsapplikation diese automatisch beim Export ein. Damit ist sichergestellt, dass die in der Versorgungsdatei zugeordnete Versionsnummer auch mit der von der Konfigurationsapplikation verwalteten Versionsnummer übereinstimmt.

#### Tip

- Die Versionsangabe in der Änderungsübersicht leer lassen, d. h. Angabe von `version=""` → Korrekte Versionsnummer wird von der Konfigurationsapplikation beim Export eingetragen.
- Dies funktioniert auch bei mehreren Änderungseinträgen mit leerer Versionsangabe.

#### 3.1.4.4.3 Modellbereich (→ modell)

Enthält die Definitionen für das Modell. Im Modellbereich können dazu **Definitionen** für

- Objekttypen
- Attributgruppen

- Aspekte
- Attribute
- Attributlisten
- Attributauswahlen
- Mengen

festgelegt werden.

#### Hinweis

- Der Modellbereich muss zwar existieren, er kann aber leer bleiben.
- Wird im Modellbereich ein neues Teilmodell definiert, sollte der Objektbereich leer bleiben → Konfigurationsbereich sollte entweder Modell- oder Versorgungsdaten enthalten.

#### 3.1.4.4.4 Objektbereich (→ objekte)

Enthält die Definitionen für konkrete Konfigurationsobjekte. Im Objektbereich können dazu **Definitionen** für

- Konfigurationsobjekte

festgelegt werden.

#### Hinweis

- Der Objektbereich muss zwar existieren, er kann aber leer bleiben.
- Werden im Objektbereich konkrete Objekte einer Anlage konfiguriert, sollte der Modellbereich leer bleiben → Konfigurationsbereich sollte entweder Modell- oder Versorgungsdaten enthalten.

#### 3.1.4.5 Verwaltungsdatei

Damit die Konfigurationsapplikation weiß, welche Konfigurationsbereiche unter welchem Konfigurationsverantwortlich verwaltet werden müssen, verwendet diese die Verwaltungsdatei (Datei: `verwaltungsdaten.xml`)

In der Verwaltungsdatei werden für die jeweilige Umgebung der Konfiguration dokumentiert, welche Versionen der einzelnen Konfigurationsbereich lokal für diese Konfiguration aktuell aktiviert sind und welche Versionen wann aktiv waren.

```

<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<!DOCTYPE verwaltungsdaten PUBLIC "-//K2S//DTD Verwaltung/DE" "verwaltungsdaten.dtd">
<verwaltungsdaten>
  <konfigurationsverantwortlicher pid="kv.inovat"/>
  <konfigurationsbereich pid="kb.kv.inovat" verzeichnis=".../kv.inovat/config/">
    <version nr="1" zeitpunkt="03.03.2007 17:58:38,000"/>
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.tmEreignisKalenderGlobal" verzeichnis=".../kv.inovat/config/">
    <version nr="1" zeitpunkt="03.03.2007 17:58:38,000"/>
    <version nr="2" zeitpunkt="05.03.2007 20:54:15,698"/>
    <version nr="3" zeitpunkt="08.03.2007 17:04:58,888"/>
    <version nr="4" zeitpunkt="16.03.2007 18:04:47,560"/>
    <version nr="5" zeitpunkt="20.03.2007 22:41:27,023"/>
    <version nr="6" zeitpunkt="04.06.2007 15:55:41,140"/>
    <version nr="7" zeitpunkt="10.06.2007 13:07:30,640"/>
    <version nr="8" zeitpunkt="09.07.2007 08:07:09,515"/>
    <version nr="9" zeitpunkt="17.01.2008 17:22:34,078"/>
    <version nr="10" zeitpunkt="20.02.2008 12:36:37,453"/>
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.tmGanglinienGlobal" verzeichnis=".../kv.inovat/config/">
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.tmGeoReferenzierungGlobal" verzeichnis=".../kv.inovat/config/">
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.tmKEXTlsGlobal" verzeichnis=".../kv.inovat/config/">
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.tmSystemKalenderGlobal" verzeichnis=".../kv.inovat/config/">
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.tmTmcGlobal" verzeichnis=".../kv.inovat/config/">
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.tmUmfeldDatenGlobal" verzeichnis=".../kv.inovat/config/">
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.tmVerkehrGlobal" verzeichnis=".../kv.inovat/config/">
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.kv.bea" verzeichnis=".../kv.bea/config/">
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.tmVewProtokolleGlobal" verzeichnis=".../kv.bea/config/">
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.fachModellGlobal" verzeichnis=".../kv.kappich/config/">
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.kv.kappich" verzeichnis=".../kv.kappich/config/">
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.metaModellGlobal" verzeichnis=".../kv.kappich/config/">
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.systemModellAoe" verzeichnis=".../kv.kappich/config/">
  <konfigurationsbereich pid="kb.systemModellGlobal" verzeichnis=".../kv.kappich/config/">
    <version nr="1" zeitpunkt="03.03.2007 17:58:38,000"/>
    <version nr="2" zeitpunkt="15.03.2007 09:46:01,945"/>
    <version nr="3" zeitpunkt="18.03.2007 16:26:08,806"/>
    <version nr="4" zeitpunkt="20.03.2007 22:41:27,033"/>
    <version nr="5" zeitpunkt="04.06.2007 15:11:13,031"/>
    <version nr="10" zeitpunkt="11.06.2007 21:26:22,546"/>
    <version nr="11" zeitpunkt="26.06.2007 10:15:57,593"/>
    <version nr="12" zeitpunkt="02.07.2007 06:23:51,312"/>
    <version nr="13" zeitpunkt="20.07.2007 06:40:50,843"/>
    <version nr="15" zeitpunkt="23.11.2007 13:14:48,968"/>
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.tmIlseTls" verzeichnis=".../kv.kappich/config/">
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.tmVewBetriebGlobal" verzeichnis=".../kv.kappich/config/">
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.tmVewSimulationGlobal" verzeichnis=".../kv.kappich/config/">
  </konfigurationsbereich>
</verwaltungsdaten>

```

Abbildung 3-5: Verwaltungsdatei

Die wesentlichen Bestandteile der Verwaltungsdatei sind:

konfigurationsverantwortlicher

Konfigurationsverantwortlicher, unter dem die Konfiguration gestartet wird. Der Konfigurationsverantwortliche wird über das Attribut `pid` angegeben.

konfigurationsbereich

Konfigurationsbereich, der zur Umgebung der Konfiguration gehört. Dabei wird die PID des Bereichs angegeben und das Verzeichnis, in dem die zugehörige Konfigurationsdatei steht.

version

Zu den Konfigurationsbereichen wird jede in der Umgebung jemals aktivierte Version dokumentiert. Dabei gibt das Attribut `nr` die entsprechende Versionsnummer der aktivierten Version an und das Attribut `zeitpunkt` den Zeitstempel, wann die Konfiguration mit dieser Version gestartet wurde (lokaler Aktivierungszeitpunkt).

Die Liste der aktivierten Versionen ist in der Regel NICHT lückenfrei; es kann bei der Übernahme von Konfigurationsbereichen von anderen Konfigurationen ohne weiteres der Fall sein, dass nicht jede Version auf dem Zielsystem aktiviert wurde. Es ist **nicht** notwendig und (bei laufenden Systemen) oft auch gar nicht möglich, jeden Versionswechsel des Quellsystems zu übernehmen.

### 3.1.5 Kriterien zur Aufteilung der Konfiguration in Konfigurationsbereiche

Die Aufteilung der gesamten Konfiguration in Konfigurationsbereiche erfolgt nach fachlichen, versorgungstechnischen und verwaltungstechnischen Kriterien.

Innerhalb eines Konfigurationsbereichs wird zwischen dem Modellbereich und dem Objektbereich unterschieden.

In der Regel werden für die Datenmodellierung (Modellbereich in einem Konfigurationsbereich) und die Versorgung (Objektbereich in einem Konfigurationsbereich) unterschiedliche Konfigurationsbereiche angelegt.

Die Aufteilung der Gesamtkonfiguration in einzelne Konfigurationsbereiche erfolgt dabei im Wesentlichen nach folgenden Kriterien:

- **Konfigurationsverantwortlicher**

Jeder Konfigurationsbereich ist einem Konfigurationsverantwortlichen zugeordnet. Nur dieser kann den Konfigurationsbereich ändern und ist für die Pflege zuständig. Die Konfigurationsbereiche müssen also in jedem Fall sinnvoll nach den Konfigurationsverantwortlichen aufgeteilt werden.

- **Trennung Modelldefinitionen und Objektdefinitionen**

In der Regel sollten die Datenmodelle und die konkreten Objektdefinitionen in unterschiedlichen Konfigurationsbereichen erfolgen. Dies ergibt sich i. d. R. bereits aus den unterschiedlichen Verantwortlichkeiten für die Modellpflege (projektübergreifend) und die konkrete Anlagenversorgung (projekt- / hersteller- / oranisationspezifisch).

- **Austausch / Weitergabe von Konfigurationsbereichen**

Die Konfigurationsdaten werden i. d. R. in verschiedenen AOE (Autarken Organisationseinheiten wie VRZ, UZ) benötigt. Konfigurationsbereiche sollten deshalb auch nach potenziell auszutauschenden Konfigurationsinformationen aufgeteilt werden.

- **Quelle von Versorgungsdaten**

Insbesondere bei der Erstellung der Objektdefinitionen für ein Projekt müssen häufig Massendaten (TLS-Versorgungen, Karteninformationen eines Netzes, Messquerschnitte etc.) in einen Konfigurationsbereich eingetragen werden. Um eine automatische Generierung zu ermöglichen, sollten Konfigurationsbereiche auch abhängig von der Informationsquelle aufgeteilt werden<sup>8</sup>.

#### Anmerkung

- Die unterschiedlichen Aufteilungen der Gesamtkonfiguration auf Konfigurationsbereiche dienen lediglich der logischen Strukturierung.
- Die logische Strukturierung der Konfigurationsbereiche kann sich überschneiden. So kann ein Konfigurationsbereich sowohl Modell- als auch Objektbereiche enthalten. Ebenso können sich Objektdefinitionen oder modellspezifische Bereiche über mehrere Konfigurationsbereiche erstrecken.

#### Tipp

- In einem Konfigurationsbereich sollte i. d. R. entweder ein Modell oder

<sup>8</sup> Dies ist auch eine Aufteilung nach der Verantwortlichkeit, hier aber der Verantwortlichkeit für die Erstellung und nicht die der Pflege (→ Konfigurationsverantwortlicher).

konkrete Objekte für eine Versorgung definiert werden.

- Modelldaten und Projekt- /anlagenspezifische Daten sollten jeweils unter eigenen Konfigurationsverantwortlichen erstellt werden.

### 3.1.6 Beschreibung der aktuellen Standard-Konfigurationsbereiche

Nachfolgend sind die Konfigurationsbereiche der Konfigurationsverantwortlichen beschrieben, die zum aktuellen Zeitpunkt Modelle für das offizielle BSVRZ definieren. Darüber hinaus existieren weitere Konfigurationsbereiche anderer Konfigurationsverantwortlicher, die entweder über die ZS oder direkt beim jeweiligen KV verfügbar sind.

In den Tabellen sind folgende Informationen dargestellt:

<b>Konfigurationsbereich</b>	PID des Konfigurationsbereichs (d. h. die zugehörige Versorgungsdatei hat den Namen <code>PID.xml</code> und die zugehörige Konfigurationsdatei den Namen <code>PID.config</code> ).
<b>Bemerkung</b>	Kurze Beschreibung des Inhalts für den Konfigurationsbereich.
<b>KV</b>	Gibt an, ob dieser Konfigurationsbereich für die Neuanlage eines eigenen Konfigurationsverantwortlichen (KV) benötigt wird, gegebenenfalls mit Angabe der notwendigen Versionsnummer. Ist der KV mit ja bzw. der Versionsnummer gekennzeichnet, so muss dieser Konfigurationsbereich in der Verwaltungsdatei eingetragen werden, mit der der neue KV angelegt werden soll. Ist auch die Versionsnummer eingetragen, so muss diese auch in der Verwaltungsdatei eingetragen werden <sup>9</sup> .
<b>AOE</b>	Gibt an, ob dieser Konfigurationsbereich für die Neuanlage eines eigenen AOE benötigt wird, gegebenenfalls mit Angabe der Versionsnummer. Ist der AOE mit ja gekennzeichnet, so muss dieser Konfigurationsbereich in der Verwaltungsdatei eingetragen werden, mit der der neue AOE angelegt werden soll. Ist auch die Versionsnummer eingetragen, so muss diese auch in der Verwaltungsdatei eingetragen werden.

<sup>9</sup> Normalerweise wird bei der Aktivierung automatisch die höchste verfügbare Versionsnummer von der Konfigurationsapplikation ermittelt und automatisch in die Verwaltungsdatei eingetragen. Die angegebenen Versionsnummer sind die aktuell gültigen.

### 3.1.6.1 kv.kappich

Konfigurationsbereich	Bemerkung	KV	AOE
kb.kv.kappich	Konfigurationsverantwortlicher kv.kappich	1	1
Konfigurationsbereiche mit Modelldefinitionen			
kb.metaModellGlobal	Metamodell	12	12
kb.systemModellGlobal	Globale Modelldefinitionen für das System (z. B. Datenverteiler etc.)	19	19
kb.systemModellAoe	Systemmodell für einen AOE	nein	4
kb.fachModellGlobal	Teilmodell mit allgemeinen Festlegungen für andere Fachmodelle	7	7
kb.tmVewBetriebGlobal	Teilmodell für Betriebsmeldungen	5	5
kb.tmVewSimulationGlobal	Teilmodell für die Simulation	7	7
kb.tmBuVGlobal	Teilmodell für Bedienung und Visualisierung	nein	5
kb.tmlIseTIs	Teilmodell für das Inselbus-Lokalbus-Service-Equipment	4	4
Konfigurationsbereiche mit Objektdefinitionen			
kb.ilse	Objektdefinitionen für das Inselbus-Lokalbus-Service-Equipment	nein	nein
kb.objekteIntegrationsTestTools	Objektdefinitionen für Testtools	nein	nein
kb.objekteTestSystem	Objektdefinitionen für Testsystem	nein	nein
kb.objekteTestUnterzentraleK2S_100_MessQuerschnitte	Objektdefinitionen für 100 MQ (Test)	nein	nein
kb.objekteTestUnterzentraleK2S_10_MessQuerschnitte	Objektdefinitionen für 10 MQ (Test)	nein	nein

Tabelle 3-1: Konfigurationsbereiche des kv.kappich

### 3.1.6.2 kv.inovat

Konfigurationsbereich	Bemerkung	KV	AOE
kb.kv.inovat	Konfigurationsverantwortlicher kv.inovat	1	1
Konfigurationsbereiche mit Modelldefinitionen			
kb.tmVerkehrGlobal	Teilmodell mit der Fachmodellierung verkehrlich relevanter Zusammenhänge (Fahrstreifen, Messquerschnitte, Abschnitte etc.), der Modellierung des Netzes (Knoten, Segmente, Teilsegmente, Netze) und wichtiger verkehrsrelevanter dynamischer Objekttypen (Stau, Baustelle, Unfall etc.)	31	31
kb.tmEreignisKalenderGlobal	Teilmodell für den Ereigniskalender	10	10
kb.tmGanglinienGlobal	Teilmodell für die Ganglinien (Automatisches Lernen, Prognose)	10	10
kb.tmGeoReferenzierungGlobal	Teilmodell mit Objekttypen zur geografischen Referenzierung von Objekten in WGS84-Koordinaten (Punkt, Linie, Fläche, komplexe Objekte)	2	2
kb.tmKExTIsGlobal	Teilmodell für die TLS-Anbindung	5	5
kb.tmSystemKalenderGlobal	Teilmodell für den Systemkalender	6	6
kb.tmTmcGlobal	Teilmodell für TMC-Meldungen	4	4
kb.tmUmfeldDatenGlobal	Teilmodell für Umfelddaten (Umfelddatenstationen und Sensoren)	---	---
Konfigurationsbereiche mit Objektdefinitionen			
---	---	---	---

Tabelle 3-2: Konfigurationsbereiche des kv.inovat

### 3.1.6.3 kv.bea

Konfigurationsbereich	Bemerkung	KV	AOE
kb.kv.bea	Konfigurationsverantwortlicher kv.bea	1	1
Konfigurationsbereiche mit Modelldefinitionen			
kb.tmVewProtokolleGlobal	Teilmodell für das Archivsystem und für Protokolle und Auswertungen	2	2
Konfigurationsbereiche mit Objektdefinitionen			
---	---	nein	nein

Tabelle 3-3: Konfigurationsbereiche des kv.bea

### 3.1.6.4 kv.bitctrl

Konfigurationsbereich	Bemerkung	KV	AOE
kb.kv.bitctrl	Konfigurationsverantwortlicher kv.bitctrl	nein	nein
Konfigurationsbereiche mit Modelldefinitionen			
kb.tmlsis	Teilmodell mit Datenmodellierung für ISIS (Baustelleninformationssystem).	nein	nein
kb.tmKExEmailFaxGlobal	Teilmodell mit Datenmodellierung zur Anbindung und zum Datenaus- tausch mit über eMail, FAX und SMS.	nein	nein
kb.tmKExLMStGlobal	Teilmodell mit Datenmodellierung zum Datenaustausch mit der Lan- desmeldestelle (LMS).	nein	nein
kb.tmKExLMSITMCCodes	Abbildung der Codetabellen aus DIN EN ISO 14819-2 auf Attributdefiniti- onen.	nein	nein
kb.tmVewUdaGlobal	Teilmodell mit Datenmodellierung zur Verwaltung der Umfassenden Datenanalyse.	nein	nein
Konfigurationsbereiche mit Objektdefinitionen			
---	---	nein	nein

Tabelle 3-4: Konfigurationsbereiche des kv.bitctrl

### 3.1.6.5 kv.dambach

Konfigurationsbereich	Bemerkung	KV	AOE
kb.kv.dambach	Konfigurationsverantwortlicher kv.dambach	1	1
Konfigurationsbereiche mit Modelldefinitionen			
kb.tmAnzeigenGlobal	Teilmodell für AQ	nein	nein
kb.tmAnzeigenGlobalZwischenschicht	Abstrahiertes Teilmodell für Standardumsetzungen, wird durch konkrete Modelle erweitert	nein	nein
kb.tmAnzeigenGlobalZwischenschichtTls	Konkretes Teilmodell für Standardumsetzungen nach TLS	nein	nein
kb.tmDarstellungAnzeigeQuerschnitt	Teilmodell zur Konfiguration von Darstellungen von AQ für Bedienoberflächen	nein	nein
kb.tmNbaGlobal	Teilmodell für das NBA-Modell Köln-Koblenz	nein	nein
kb.tmRechner	Teilmodell für Rechner für SWE Start/Stop und DCF77	4	4
kb.tmUsv	Teilmodell für SWE USV	nein	nein
kb.tmVewEngstellenGlobal	Teilmodell für SWE Engstellenverwaltung	2	2
Konfigurationsbereiche mit Objektdefinitionen			
---	---	nein	nein

Tabelle 3-5: Konfigurationsbereiche des kv.dambach

### 3.1.6.6 kv.logos

Konfigurationsbereich	Bemerkung	KV	AOE
kb.kv.logos	Konfigurationsverantwortlicher kv.logos	nein	nein
Konfigurationsbereiche mit Modelldefinitionen			
kb.tmLangZeitVerkehrsDaten	Teilmodell für die Langzeitdatenerfassung und Auswertung in das Format der BASt	nein	nein
Konfigurationsbereiche mit Objektdefinitionen			
---	---	nein	nein

Tabelle 3-6: Konfigurationsbereiche des kv.logos

### 3.1.6.7 kv.ptv

Konfigurationsbereich	Bemerkung	KV	AOE
kb.kv.ptv	Konfigurationsverantwortlicher kv.ptv	nein	nein
Konfigurationsbereiche mit Modelldefinitionen			
kb.tmGeraeteEinbauOrte	Teilmodell für die Beschreibung von Geräten mit Angaben zu Einbauorten, Betriebsinformationen etc. Wird z. B. vom AQ-Modell verwendet.	nein	nein
Konfigurationsbereiche mit Objektdefinitionen			
---	---	nein	nein

Tabelle 3-7: Konfigurationsbereiche des kv.ptv

## 3.1.7 Abhängigkeiten zwischen Konfigurationsbereichen

### 3.1.7.1 Was sind Abhängigkeiten?

Durch die Aufteilung der Gesamtkonfiguration auf viele einzelne Konfigurationsbereiche entstehen Abhängigkeiten zwischen Konfigurationsbereichen. Dies ist völlig normal und letztendlich auch beabsichtigt.

Typ-, Attributgruppen-, Attributtypdefinition etc. werden in unterschiedlichen Modellen definiert und bei neuen Modellen wieder verwendet.

Für die konkrete Versorgung von Systemen werden die Objekte der Anlagen auf Basis von Typdefinitionen angelegt. Die Typdefinitionen und die Objektdefinitionen befinden sich dabei in unterschiedlichen Konfigurationsbereichen, i. d. R. werden sie zudem auch von unterschiedlichen Konfigurationsverantwortlichen gepflegt.

Die dadurch entstehenden Abhängigkeiten (Konfigurationsbereich A benötigt Definitionen aus den Bereiche B, C, ...) müssen bei der Zusammenstellung einer Gesamtkonfiguration berücksichtigt werden. Nur wenn alle benötigten Abhängigkeiten innerhalb einer Konfiguration erfüllt sind, kann eine Konfiguration aktiviert werden.

Bei der Neuanlage einer Konfiguration, also bei der Zusammenstellung der notwendigen Konfigurationsbereiche, muss also ermittelt werden, welcher Konfigurationsbereich welche anderen Konfigurationsbereiche benötigt. Dazu unterstützt die Konfigurationsapplikation seit einigen Versionen die automatische Dokumentation der Abhängigkeiten entsprechend der nachfolgenden Beschreibung. Diese Abhängigkeiten zwischen den Konfigurationsbereichen müssen bei der Zusammenstellung einer Konfiguration, also bei der erstmaligen Erstellung der Verwaltungsdatei, berücksichtigt werden.

### 3.1.7.2 Dokumentation der Abhängigkeiten

In der aktuellen Version der Konfigurationsapplikation wird beim Export des Konfigurationsbereichs die Liste der Abhängigkeiten in die Versorgungsdatei als Kommentar automatisch eingefügt (grüner Text).

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE KonfigurationsBereich PUBLIC "-//K2S//DTD Dokument//DE" "K2S.dtd">
<KonfigurationsBereich pid="kb.tmVerkehrGlobal" name="TeilmodellVerkehrGlobal" verantwortlich="kv.inovat">
  <info...>
  <KonfigurationsAenderung stand="19.03.2007" version="5" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="A:Überarbei
  <KonfigurationsAenderung stand="20.03.2007" version="10" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="B:Überarbe
  <KonfigurationsAenderung stand="02.05.2007" version="11" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsc
  <KonfigurationsAenderung stand="01.06.2007" version="12" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsc
  <KonfigurationsAenderung stand="10.06.2007" version="13" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsc
  <KonfigurationsAenderung stand="14.06.2007" version="14" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsc
  <KonfigurationsAenderung stand="09.07.2007" version="15" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsc
  <KonfigurationsAenderung stand="28.09.2007" version="17" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsc
  <KonfigurationsAenderung stand="03.10.2007" version="18" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsc
  <KonfigurationsAenderung stand="23.11.2007" version="18" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsc
  <KonfigurationsAenderung stand="02.12.2007" version="20" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsc
  <KonfigurationsAenderung stand="19.12.2007" version="21" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsc
  <KonfigurationsAenderung stand="20.12.2007" version="22" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsc
  <KonfigurationsAenderung stand="17.01.2008" version="23" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsc
  <KonfigurationsAenderung stand="20.02.2008" version="25" autor="inovat, Dipl.-Ing. H. C. Kniß (HCK)" grund="Änderungsc
  <!--
  Abhängigkeiten des Bereichs kb.tmVerkehrGlobal
  Abhängig ab Version  Art der Abhängigkeit  Version des benötigten Bereichs  Benötigter Bereich
  23  notwendig  1  kb.tmKExtlsGlobal
  23  notwendig  1  kb.kv.inovat
  23  notwendig  1  kb.systemModellGlobal
  23  notwendig  1  kb.tmGeoReferenzierungGlobal
  23  notwendig  7  kb.fachModellGlobal
  23  notwendig  4  kb.metaModellGlobal
  23  notwendig  8  kb.tmGanglinienGlobal
  23  notwendig  2  kb.tmTacGlobal
  24  notwendig  3  kb.tmVewBetriebGlobal
  24  notwendig  9  kb.tmGanglinienGlobal
  -->
  <modell...>
  <objekte...>
</KonfigurationsBereich>
  
```

Abbildung 3-6: Abhängigkeiten des kb.tmVerkehrGlobal in der Version 25  
Eintrag in der Versorgungsdatei nach dem Export

Dokumentiert wird:

- Abhängig ab Version:** Versionsnummer des vorliegenden Konfigurationsbereichs, ab dem eine Abhängigkeit zu einem anderen Bereich (→ **Benötigter Bereich**) besteht.
- Art der Abhängigkeit:** Gibt an, ob der → **Benötigte Bereich** *notwendig* oder *optional* ist.
- Version des benötigten Bereichs:** Gibt die minimale Versionsnummer des → **Benötigten Bereichs** an.
- Benötigter Bereich:** Bereich, der zur Aktivierung des betrachteten Bereichs notwendig ist.

Die benötigten Bereiche haben natürlich selbst wieder Abhängigkeiten, die ebenfalls berücksichtigt werden müssen.

#### Hinweis

- Die Abhängigkeit eines Konfigurationsbereichs von anderen Konfigurationsbereichen wird beim Export automatisch in der **Versorgungsdatei als Kommentar** eingefügt.

### 3.1.8 Kodierungsschemata und Namenskonventionen für KV und AOE.

Bei der Neuanlage eines Konfigurationsverantwortlichen bzw. einer AOE muss eine eindeutige Kodierung für den KV bzw. AOE vergeben werden.

Diese Kodierung wirkt direkt auf die Vergabe der ID's der Konfigurationsobjekte des jeweiligen Konfigurationsbereichs aus und stellt darüber sicher, dass weltweit keine zwei Objekte dieselbe ID erhalten können. Dazu müssen natürlich die Kodierungen ebenfalls eindeutig sein.

### 3.1.8.1 Kodierung für Konfigurationsverantwortlichen / AOE

Um dies zu gewährleisten, vergibt die Zentrale Stelle (ZS) für jedes Bundesland und jede Firma eine eigene kurze Nummer (vorerst zweistellig). Jedes Bundesland und jede Firma hat dann die Möglichkeit für sich bis zu 100 weitere Kodierungen YYXX zu vergeben. Dabei ist YY die von der ZS vergebene zweistellige Nummer und XX eine von der Organisation selbst vergebene Nummer (00-99). Falls alle 100 mögliche Kodierungen YYXX von einer Organisation erschöpft sind, kann sich die Organisation von der Zentralen Stelle eine weitere zweistellige Kodierung (ZZ) zuordnen lassen und dann weitere 100 Kodierungen ZZXX nach dem gleichen Schema vergeben.

### 3.1.8.2 PID von Konfigurationsverantwortlichen / AOE

Konfigurationen werden unterschieden nach extern gelieferten Konfigurationsanteilen und nach Konfigurationsanteilen, die durch den jeweiligen Hersteller bzw. ein Bundesland (im Folgenden als Organisation bezeichnet) gepflegt werden.

Die jeweiligen Konfigurationsverantwortlichen (PID des Konfigurationsverantwortlichen) sollte bei den durch die Organisation *organisation* gepflegten Konfigurationsverantwortlichen nach folgendem Schema benannt<sup>10</sup>:

**kv.[aoe.] organisation [.zusatz][.ort][.auftraggeber][.hierarchie]**

mit

<b>kv</b>	festes Präfix für <b>K</b> onfigurations <b>v</b> erantwortlicher.
<b>aoe</b>	zusätzliches festes Präfix für <b>A</b> utarke <b>O</b> rganisations <b>E</b> inheit, wenn Konfigurationsverantwortlicher auch autarke Organisationseinheit ist. Dies sollte normalerweise nur bei Konfigurationsverantwortlichen der Fall sein, die zum Ausführen einer Konfiguration in einem konkreten Projekt verwendet werden. <u>Allgemeine Konfigurationsbereiche</u> mit Modellen, die von anderen Anwendern verwendet werden, sollten ausschließlich als einfacher Konfigurationsverantwortlicher gepflegt und publiziert werden
<i>organisation</i>	Kürzel der Organisation, Firma (bea, bitctrl, dambach, inovat, kappich, logos, etc.) bzw. Land, siehe Kapitel 3.1.8.3.
<i>zusatz</i>	optional, zusätzliche Angaben der Verwendung (z.B. ModelleXyz, ProjektXyz, etc.).
<i>ort</i>	optional, Sitz des Auftraggebers und/oder Ortsangabe für die zusätzlich angegebene <b>hierarchie</b> , siehe Beschreibung Kapitel 3.1.8.3.
<i>auftraggeber</i>	optional, siehe Beschreibung bei Kapitel 3.1.8.3.
<i>hierarchie</i>	optionale weitere Unterscheidung (fettgedruckt), wie z. B.: <ul style="list-style-type: none"> <li>o kv.aoe.inovat.nw.stnrw.<b>vrz.leverkusen</b></li> <li>o kv.aoe.inovat.nw.stnrw.<b>uz.loehne</b></li> <li>o kv.aoe.inovat.nw.stnrw.<b>uz.weilerswist</b></li> <li>o kv.aoe.inovat.nw.boschung.<b>zentrale.kamen</b></li> </ul>

Die optionalen Zusätze **ort**, **auftraggeber**, und **hierarchie** werden i. d. R. durch den Auftraggeber zentral verwaltet.

<sup>10</sup> Bei dem dargestellten Schema handelt es sich um einen Vorschlag, der gegebenenfalls organisationsabhängig abgeändert werden kann.

Die Konfigurationsbereiche (mit Endung \*.xml bzw. \*.config) sowie die Ordnernamen entsprechen der Bezeichnung des Konfigurationsverantwortlichen, also **z. B.**:

Konfigurationsverantwortlicher / AOE	Bemerkung
<b>kv.bea</b>	Allgemeine Modelle, durch beck et al gepflegt.
<b>kv.bitctrl</b>	Allgemeine Modelle, durch BitCtrl gepflegt.
<b>kv.dambach</b>	Allgemeine Modelle, durch Dambach gepflegt.
<b>kv.inovat</b>	Allgemeine Modelle, durch inovat gepflegt.
<b>kv.kappich</b>	Allgemeine Modelle, durch Kappich Systemberatung gepflegt
<b>kv.logos</b>	Allgemeine Modelle, durch Logos gepflegt
<b>kv.bw</b>	Allgemeine Modelle, durch Baden-Württemberg gepflegt
<b>kv.nw</b>	Allgemeine Modelle, durch Nordrhein-Westfalen gepflegt
<b>kv.sl</b>	Allgemeine Modelle, durch das Saarland gepflegt
...	...weitere...
<b>kv.inovat.nw</b>	Allgemeine Modelle für NRW, durch inovat gepflegt.
...	...weitere...
<b>kv.aoe.inovat.nw.stnrw.vrz.leverkusen</b>	<b>AOE</b> für den Betrieb der <u>speziellen Versorgung</u> der VRZ Leverkusen in NRW, Verantwortlich <b>inovat</b> , Ort NRW und Auftraggeber StraßenNRW.
<b>kv.aoe.inovat.nw.boschung.uz.kamen</b>	<b>AOE</b> für den Betrieb der <u>speziellen Versorgung</u> der Boschung Zentrale in Kamen in NRW, Verantwortlich <b>inovat</b> , Ort NRW und Auftraggeber Boschung Mecatronic.
<b>kv.aoe.nw.vrz.leverkusen</b>	<b>AOE</b> für den Betrieb der <u>speziellen Versorgung</u> der VRZ Leverkusen in NRW, Verantwortlich <b>NW. Ort und Auftraggeber wurden weggelassen, da identisch mit Verantwortlichem.</b>

Tabelle 3-8: Beispiele für Konfigurationsverantwortliche /AOE

### 3.1.8.3 Länderkürzel

Land bzw. Bundesland, in dem der Auftraggeber sitzt. In bestimmten Ausnahmefällen wird hier auch der Ort indirekten Auftraggebers verwendet, wenn z. B. eine Behörde ein Unternehmen beauftragt, welches diesen Auftrag als Unterauftrag weitergibt.

Orts-/Länderkürzel	Land
<b>bb</b>	Brandenburg
<b>be</b>	Berlin

Orts-/Länderkürzel	Land
<b>bw</b>	Baden-Württemberg
<b>by</b>	Bayern
<b>hb</b>	Bremen
<b>he</b>	Hessen
<b>hh</b>	Hamburg
<b>mv</b>	Mecklenburg-Vorpommern
<b>ni</b>	Niedersachsen
<b>nw</b>	Nordrhein-Westfalen
<b>rp</b>	Rheinland-Pfalz
<b>sh</b>	Schleswig-Holstein
<b>sl</b>	Saarland
<b>sn</b>	Sachsen
<b>st</b>	Sachsen-Anhalt
<b>th</b>	Thüringen
<b>de</b>	Deutschlandallgemein
<b>ch</b>	Schweiz
<b>at</b>	Österreich
<b>ia</b>	interne Projekte

Tabelle 3-9: Länderkürzel für Ortsbezüge

### 3.1.8.4 PID von Konfigurationsbereichen

Konfigurationsbereiche sollten folgenden Namenskonventionen folgen<sup>11</sup>:

<b>kb.kv_aoe</b>	Für den Konfigurationsverantwortlichen/AOE
<b>kb.default.kv_aoe</b>	Für den Defaultbereich des KV/AOE
<b>kb.tmName</b>	Für Teilmodelle
<b>kb.objekteName</b>	Für Objektversorgungen

mit

<b>kb</b>	festes Präfix für <b>K</b> onfigurations <b>B</b> ereich.
<b>default</b>	zusätzliches festes Präfix für den <b>Default</b> konfigurationsbereich des zuständigen KV/AOE.
<b>kv_aoe</b>	KV/AOE entsprechend Vorgaben Kapitel 3.1.8.2.
<b>tm</b>	zusätzliches festes Präfix für <b>Teil</b> Modell.
<b>objekte</b>	zusätzliches festes Präfix für Konfigurationsbereiche mit konkreten <b>Objekt</b> definitionen.
<b>Name</b>	Name des Modells bzw. Bereichs.

Beispiele für KV/AOE: kv.aoe.nw.vrz.leverkusen

<sup>11</sup> Hinweis: Einige der bereits seit längerem bestehenden Konfigurationsbereiche folgen (aus historischen Gründen) nicht diesem Namensschemata.

- `kb.kv.aoe.nw.vrz.leverkusen`
- `kb.default.kv.aoe.nw.vrz.leverkusen`

Beispiele für Teilmodell bzw. Objektversorgungen

- `kb.tmFachModellGlobal`
- `kb.objekteNwUzDuerenVerkehrMqFs`

## 3.2 Konfigurationsverantwortliche und AOE

### 3.2.1 Konfigurationsverantwortliche

Jedem Konfigurationsbereich ist ein Konfigurationsverantwortlicher zugeordnet. Nur die Konfigurationsapplikation, die unter dem Konfigurationsverantwortlichen des Konfigurationsbereichs gestartet wurde, hat die Möglichkeit, die enthaltenen Konfigurationsdaten zu ändern oder zu erweitern.

Der Konfigurationsverantwortliche ist seinerseits auch ein Konfigurationsobjekt, das in einem Konfigurationsbereich versorgt werden muss. Da die Konfigurationsapplikation potentiell unter jedem Konfigurationsverantwortlichen gestartet werden kann, muss der Konfigurationsverantwortliche in einem eigenen Konfigurationsbereich definiert werden.

Der Konfigurationsverantwortliche wirkt sich auf die Vergabe der ID's der Konfigurationsobjekte des jeweiligen Konfigurationsbereichs aus. Dazu muss für den Konfigurationsverantwortlichen eine eindeutige Kodierung festgelegt werden.

Dazu vergibt die Zentrale Stelle (ZS) für jedes Bundesland und jede Firma eine eigene kurze Nummer (vorerst zweistellig). Jedes Bundesland und jede Firma hat dann die Möglichkeit für sich bis zu 100 weitere Kodierungen YYXX zu vergeben. Dabei ist YY die von der ZS vergebene zweistellige Nummer und XX eine von der Organisation selbst vergebene Nummer (00-99). Falls alle 100 mögliche Kodierungen YYXX von einer Organisation erschöpft sind, kann sich die Organisation von der ZS eine weitere zweistellige Kodierung (ZZ) zuordnen lassen und dann weitere 100 Kodierungen ZZXX nach dem gleichen Schema vergeben.

#### Anmerkung

- **Konfigurationsverantwortliche** werden für die **Definition** von Modellen und Objekten verwendet.

### 3.2.2 Autarke Organisationseinheit (AOE)

Ein wesentlicher Aspekt der Architektur des Datenverteilersystems ist die Möglichkeit, Datenverteiler miteinander zu koppeln. Jede Verkehrsrechnerzentrale und Unterzentrale stellt ein System dar, welches aus einem oder mehreren Rechnern bestehen kann. Jedes dieser Systeme ist unabhängig (autark) von anderen Systemen, mit denen es über den Datenverteiler verbundenen ist<sup>12</sup>. Ein solches unabhängiges System wird im BSVRZ als **Autarke Organisationseinheit (AOE)** bezeichnet. Jede AOE hat einen eigenen Datenverteiler, eine eigene Konfigurationsapplikation und eine eigene Konfiguration.

Analog zum Konfigurationsverantwortlichen ist auch die AOE ein eigenes Konfigurationsobjekt. Die AOE ist vom Konfigurationsverantwortlichen abgeleitet, d. h. ein

<sup>12</sup> Autark bedeutet in diesem Zusammenhang, dass jedes System für sich gestartet und betrieben werden kann. Das durch die fehlende Kopplung Einschränkungen der Funktionen gegeben seien können, ist natürlich möglich.

- **AOE ist auch ein Konfigurationsverantwortlicher<sup>13</sup>.**

Es muss unterschieden werden, ob der ein Konfigurationsverantwortlicher für die Daten-Modellierung und Versorgung verwendet werden soll, oder ob der Konfigurationsverantwortliche einer Laufzeitumgebung eines Systems mit eigener Konfiguration für den Betrieb zugeordnet werden soll. Im letzteren Fall wird dann AOE verwendet.

#### Anmerkung

- **AOE** werden für den **Betrieb** einer Konfiguration verwendet.

### 3.3 Konfigurationsapplikation im Detail

#### 3.3.1 Anwendungsbetrieb

Die Beschreibung der Funktionen und die Nutzung der Konfigurationsapplikation im Anwendungsbetrieb (→ Teil A, Kapitel 2.1.3.1) ist nicht Bestandteil dieses Tutorial sondern werden im Tutorial [TutorialBSVRZ] näher erläutert.

#### 3.3.2 Konfigurationsbetrieb

Die nachfolgenden Kapitel beschreiben die unterschiedlichen Arbeitsmodi der Konfiguration im Konfigurationsbetrieb aus technischer Sicht. Einen Überblick über diese Funktionen enthält Teil A, Kapitel 2.1.3.2).

Im Konfigurationsbetrieb kann die Konfigurationsapplikation in unterschiedlichen Arbeitsmodi gestartet werden. Der Arbeitsmodus wird jeweils durch einen speziellen Aufrufparameter ausgewählt. Abhängig vom gewählten Modus sind weitere Aufrufparameter anzugeben.

Nach der Durchführung der entsprechenden Arbeiten beendet sich die Konfigurationsapplikation automatisch.

#### Wichtig

- Während der Arbeit mit der Konfigurationsapplikation im Konfigurationsbetrieb darf keine Konfiguration parallel im Anwendungsbetrieb auf der zu bearbeitenden Konfiguration gestartet sein (→ Fehlermeldung).  
 → eine im Betrieb laufende Konfiguration kann nicht über die hier beschriebenen Mechanismen direkt geändert werden.

#### 3.3.2.1 Übersicht der Modi und zusätzlichen Aufrufparameter

Modus	Aufrufparameter	Bemerkung
Aufrufparameter, die in allen Modi gesetzt werden	<b>-verwaltung</b> ={ <i>verwaltungsdaten</i> }	{ <i>verwaltungsdaten</i> }: Komplette Pfad- und Datei- angabe der für die Da-

<sup>13</sup> Aber nicht umgekehrt! Ein AOE ist also ein Konfigurationsverantwortlicher mit zusätzlichen Eigenschaften. Die Details werden im Teil B beschrieben.

Modus	Aufrufparameter	Bemerkung
müssen		tei <i>verwaltungsda-</i> <i>ten.xml</i>
Import	<b>-import</b> ={ <i>liste-xml-dateien</i> } <b>-verzeichnis</b> ={ <i>dir.xml</i> }	{ <i>liste-xml-</i> <i>dateien</i> }: Komma separierte Liste der zu importierenden XML-Dateien.  { <i>dir.xml</i> }: Pfadangabe zu den XML- Dateien (wo stehen die zu importierenden XML- Dateien), <u>relativ</u> zum Ort der Verwaltungsdatei
Konsistenzprü- fung	<b>-konsistenzprüfung</b>	Führt die Konsistenzprü- fung durch.
Aktivierung	<b>-aktivierung</b>	Aktiviert alle importierten Bereiche. Die Konsis- tenzprüfung wird automa- tisch durchgeführt.
Export	<b>-export</b> ={ <i>liste-config-dateien</i> } <b>-verzeichnis</b> ={ <i>dir.xml</i> }	{ <i>liste-config-</i> <i>dateien</i> }: Komma separierte Liste der zu exportierenden CONFIG-Dateien.  Werden keine CONFIG- Dateien angegeben, wer- den alle Dateien der Kon- figuration exportiert!  { <i>dir.xml</i> }: Pfadangabe zu den XML- Dateien (wohin soll expor- tiert werden), <u>relativ</u> zum Ort der Verwaltungsdatei
Freigabe zur Übernahme	<b>-freigabeuebernahme</b>	
Freigabe zur Aktivierung	<b>-freigabeaktivierung</b>	
Freigabe zur Aktivierung ohne lokale Aktivierung	<b>-freigabeZurAktivierungOhne LokaleAktivierungDurchDenKV=</b> { <i>freigabeBereiche</i> } <sup>14</sup>	{ <i>freigabeBereiche</i> }: Komma separierte Liste der zu freizugebenden CONFIG-Dateien

<sup>14</sup>Aufrufparameter natürlich ohne Leerzeichen und Zeilenumbrüche, also:

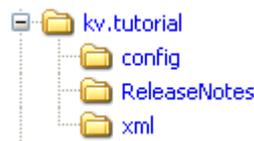
**-freigabeZurAktivierungOhneLokaleAktivierungDurchDenKV**={*freigabeBereiche*}

Tabelle 3-10: Übersicht der relevanten Aufrufparameter der Konfigurationsapplikation

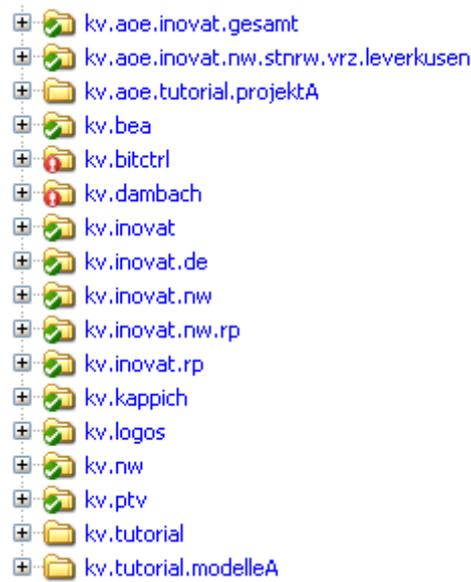
### 3.3.2.2 Konfigurationsumgebung

In den folgenden Kapiteln werden die einzelnen Arbeitsmodi der Konfigurationsapplikation mit ihrer Funktion, den relevanten Aufrufparametern und einem Beispiel-Startscript beschrieben. Die Startscripte basieren auf den mitgelieferten Scripten des Beispielsystems zur Kernsoftware Version 3.4.4. Weitere Details zur Einrichtung sind deshalb dem Tutorial zur „Kernsoftware“ zu entnehmen.

Die in den Scripten verwendeten relativen Pfadangaben und Bezeichnungen von Ordner basieren auf den Ordnerstrukturen, wie sie im Kapitel 3.5.4 auch zur Weitergabe von Konfigurationen beschrieben sind. Für einen Konfigurationsverantwortlichen sieht die Ordnerstruktur (hier am Beispiel des kv.inovat) wie folgt aus:



Die Konfigurationen anderer benötigter Konfigurationsverantwortlicher liegen jeweils mit identischer interner Ordnerstruktur parallel zu diesem Konfigurationsverantwortlichen (Beispielumgebung).



Für ein produktives System zur Konfigurationsverwaltung sollte eine eigene Umgebung zur Verwaltung und Pflege der Konfiguration nebst automatisierten Scripten eingerichtet werden. Ein Vorschlag für eine entsprechende Umgebung wird im Teil C beschrieben.

### 3.3.2.3 Import

#### 3.3.2.3.1 Funktion

Beim Import werden einzelne oder mehrere, neu angelegte oder geänderte, Versorgungsdateien (\*.xml-Dateien) in die Konfiguration übernommen. Dabei werden Änderungen in die bestehenden korrespondierenden Konfigurationsdateien (\*.config-Dateien) übernommen bzw. neu erstellte Versorgungsdateien werden in neue Konfigurationsdateien konvertiert. Dabei erkennt die Konfigurationsapplikation automatisch die Unterschiede zwischen den importierten Versorgungsdateien und den Konfigurationsdateien:

- geänderte Objekte<sup>15</sup>
- gelöschte Objekte
- neue Objekte

Änderungen der Konfigurationsdateien werden versioniert, d. h. es ist nachvollziehbar, bei welcher Version die jeweilige Änderung gültig wurde.

### 3.3.2.3.2 Aufrufparameter

<code>-import={liste-xml-dateien}</code>	<code>{liste-xml-dateien}:</code> Komma separierte Liste der zu importierenden Konfigurationsbereiche (also der XML-Dateien). <b>ACHTUNG:</b> Angegeben werden muss jeweils die <b>PID</b> des Konfigurationsbereichs, also der Dateiname der XML-Datei <b>OHNE</b> die Endung <code>.xml</code>
<code>-verzeichnis={dir.xml}</code>	<code>{dir.xml}:</code> Pfadangabe zu den XML-Dateien (wo stehen die zu importierenden XML-Dateien), <u>relativ</u> zum Ort der Verwaltungsdatei
<code>-verwaltung={verwaltungsdaten}</code>	<code>{verwaltungsdaten}:</code> Komplette Pfad- und Dateiangabe der für die Datei <code>verwaltungsdaten.xml</code>

### 3.3.2.3.3 Startscript für den Import

```
#!/bin/bash
. einstellungen.sh

$java \
-cp ../distributionspakete/de.bsvrz.puk.config/de.bsvrz.puk.config-runtime.jar \
-Xmx300m \
de.bsvrz.puk.config.main.ConfigurationApp \
-import=kb.testModell,kb.testObjekte \
-verzeichnis=../xml \
-verwaltung=../config/verwaltungsdaten.xml \
$debugDefaults \
-debugLevelStdErrText=INFO \
-debugLevelFileText=CONFIG \
```

<sup>15</sup> Objekte bezeichnet hier sowohl Modelldefinitionen (Objektypdefinitionen, Attributgruppendifinitionen etc.) als auch Versorgungsobjekte (konkrete Definitionen von realen Objekten wie MQ, DE etc.).

### 3.3.2.4 Konsistenzprüfung

#### 3.3.2.4.1 Funktion

Die Konsistenzprüfung prüft die neuste Version der eigenen<sup>16</sup> Konfigurationsbereiche und freigegebene Versionen von fremden<sup>17</sup> Konfigurationsbereichen. Dabei werden die Konfigurationsdaten u. a. auf folgende Fehler / Warnungen geprüft:

- **Lokale Fehler** innerhalb eines Konfigurationsbereichs. Hier wird u. a. geprüft, ob
  - eindeutige Objekt-ID's und je Version eindeutige PID's vergeben wurden
  - nur erlaubte Zeichen in PID's verwendet wurden
  - die Beschränkungen und Anforderungen des Datenmodells an Mengen erfüllt sind
  - die Beschränkungen und Anforderungen des Datenmodells an konfigurierende Datensätze erfüllt sind
  - alle aus lokalen Konfigurationsbereichen referenzierten Objekte vorhanden sind
  - spezielle Beschränkungen der verschiedenen Referenzierungsarten erfüllt sind
- **Interferenz-Fehler**, die sich durch die gemeinsame Betrachtung mehrerer Konfigurationsbereiche ergeben. Hier wird u. a. geprüft, ob
  - alle referenzierten Objekte der Konfiguration bekannt sind.
  - alle von gültigen Objekten referenzierten Objekte gültig sind.
  - die Bedingung erfüllt ist, dass Referenzen nur dann undefiniert sein dürfen, wenn dies vom jeweiligen Attributtyp zugelassen ist
  - die PID's der nicht als gelöscht markierten Objekte bereichsübergreifend eindeutig sind. Dieser Sachverhalt ist nicht unbedingt ein Fehler<sup>18</sup>.
- **Warnungen**, die auf eventuelle Fehler hindeuten können. Hier wird u. a. geprüft, ob
  - je Objekttyp und Parameterattributgruppe ein Default-Parametersatz<sup>19</sup> definiert wurde.

#### 3.3.2.4.2 Aufrufparameter

##### -konsistenzprüfung

Führt die Konsistenzprüfung für die angegebene Konfiguration durch. Der Umfang der Konfiguration wird durch die Einträge in der Verwaltungsdatei festgelegt.

**-verwaltung**={*verwaltungsdaten*} {*verwaltungsdaten*}:

Komplette Pfad- und Dateiangabe der für die Datei *verwaltungsdaten.xml*

<sup>16</sup> Konfigurationsbereiche, die unter der Verwaltung des aktuellen Konfigurationsverantwortlichen liegen → Verwaltungsdatei.

<sup>17</sup> Konfigurationsbereiche, die von anderen Konfigurationsverantwortlichen übernommen wurden.

<sup>18</sup> Über einen Parameter der Konfigurationsapplikation kann eingestellt werden, ob doppelte PID's als Fehler oder als Warnung betrachtet werden sollen. Ab der Version 3.4.4. wird dies standardmäßig als Fehler behandelt.

<sup>19</sup> Da bis zum Erscheinen der Version 3.4.4. noch nicht alle Konfigurationsbereiche auf diese Anforderung umgestellt waren, wird dies aktuell noch als Warnung behandelt. In zukünftigen Versionen der Konfigurationsapplikation wird dies jedoch als Fehler betrachtet.

### 3.3.2.4.3 Startscript für die Konsistenzprüfung

```
#!/bin/bash
. einstellungen.sh

$java \
-cp ../distributionspakete/de.bsvrz.puk.config/de.bsvrz.puk.config-runtime.jar \
-Xmx300m \
de.bsvrz.puk.config.main.ConfigurationApp \
-konsistenzprüfung \
-verwaltung=../config/verwaltungsdaten.xml \
$debugDefaults \
-debugLevelStdErrText=INFO \
-debugLevelFileText=CONFIG \
```

### 3.3.2.5 Aktivierung

#### 3.3.2.5.1 Funktion

Damit die importierten Änderungen für die Konfigurationsapplikation im Anwendungsbetrieb verfügbar sind, müssen diese aktiviert werden.

Durch die Aktivierung stehen beim nächsten Start der Konfigurationsapplikation die neusten Versionen der eigenen Bereiche und die neusten zur Aktivierung freigegebenen Versionen von übernommenen Konfigurationsbereichen bereit.

Dazu werden je Bereich in der jeweiligen Konfigurationsdatei folgende Versionsnummern vorgehalten:

- in der eigenen Konfiguration aktivierbare Version
- zur Übernahme in fremden Konfigurationen freigegebene Version
- zur Aktivierung in fremden Konfigurationen freigegebene Version

In der Verwaltungsdatei ist je Bereich die lokal tatsächlich aktivierte Version dokumentiert.

Bei der Funktion Aktivierung wird die Funktion Konsistenzprüfung, die auch alleine aufgerufen werden kann, implizit durchgeführt.

Bei der Aktivierung werden die Versionsnummern in der Verwaltungsdatei und in Konfigurationsdateien von eigenen Bereichen aktualisiert. Die Aktivierung wird mit dem nächsten Systemstart abgeschlossen.

#### 3.3.2.5.2 Aufrufparameter

##### -aktivierung

Aktiviert alle eigenen importierten Bereiche, zur Übernahme in fremden Konfigurationen und zur Aktivierung in fremden Konfigurationen freigegebene Versionen.

Bei einer Aktivierung wird automatisch immer auch die Konsistenzprüfung durchgeführt.

**-verwaltung**={*verwaltungsdaten*} {*verwaltungsdaten*}:

Komplette Pfad- und Dateiangabe der für die Datei  
*verwaltungsdaten.xml*

### Startscript für die Aktivierung

```
#!/bin/bash
. einstellungen.sh

$java \
  -cp ../distributionspakete/de.bsvrz.puk.config/de.bsvrz.puk.config-runtime.jar \
  -Xmx300m \
  de.bsvrz.puk.config.main.ConfigurationApp \
  -aktivierung \
  -verwaltung=../config/verwaltungsdaten.xml \
  $debugDefaults \
  -debugLevelStdErrText=INFO \
  -debugLevelFileText=CONFIG \
```

## 3.3.2.6 Export

### 3.3.2.6.1 Funktion

Die Export-Funktion erzeugt aus einzelnen oder mehreren Konfigurationsdateien entsprechende Versorgungsdateien (Konvertierung der \*.config-Dateien in \*.xml-Dateien). Diese Versorgungsdateien enthalten die Definition von Typen, Attributgruppen etc. sowie die Objekte des Konfigurationsbereichs jeweils nach PID sortiert. Dabei sind nur Informationen enthalten, die von außen vorgegeben werden können (also z. B. keine Objekt-ID's). Die Versorgungsdateien enthalten zudem keine Historie. Exportiert wird also genau die neuste Version des entsprechenden Konfigurationsbereichs.

### 3.3.2.6.2 Aufrufparameter

**-export**={*liste-config-dateien*} {*liste-config-dateien*}:

Komma separierte Liste der zu exportierenden Konfigurationsdateien (also der CONFIG-Dateien).

**ACHTUNG:** Angegeben werden muss jeweils die **PID** des Konfigurationsbereichs, also der Dateiname der CONFIG-Datei **OHNE** die Endung *.config*

Werden keine CONFIG-Dateien angegeben, werden alle Dateien der Konfiguration exportiert, also auch die anderer Konfigurationsverantwortlicher.

**-verzeichnis**={*dir.xml*} {*dir.xml*}:

Pfadangabe zu den XML-Dateien (wohin soll exportiert werden), relativ zum Ort der Verwaltungsdatei

**-verwaltung**={*verwaltungsdaten*} {*verwaltungsdaten*}:

Komplette Pfad- und Dateiangebe der für die Datei  
*verwaltungsdaten.xml*

### 3.3.2.6.3 Startscript für den Export

```
#!/bin/bash
. einstellungen.sh

$java \
-cp ../distributionspakete/de.bsvrz.puk.config/de.bsvrz.puk.config-runtime.jar \
-Xmx300m \
de.bsvrz.puk.config.main.ConfigurationApp \
-export=kv.inovat, kv.tmVerkehrGlobal,kv.tmKexTLSGlobal \
-verzeichnis=../xml \
-verwaltung=../config/verwaltungsdaten.xml \
$debugDefaults \
-debugLevelStdErrText=INFO \
-debugLevelFileText=CONFIG \
```

### 3.3.2.7 Freigabe

Bei der Funktion Freigabe werden die Varianten

- Freigabe zur Übernahme
- Freigabe zur Aktivierung
- Freigabe zur Aktivierung ohne lokale Aktivierung durch den KV

unterschieden.

### 3.3.2.8 Freigabe zur Übernahme

#### 3.3.2.8.1 Funktion

Bei der Freigabe zur Übernahme wird jeweils die neuste Version eigener Bereiche zur Übernahme in anderen Konfigurationen freigegeben und entsprechend in der Konfigurationsdatei vermerkt. Es erfolgt eine implizite Konsistenzprüfung, die sicherstellt, dass die jeweils neuste Version von lokalen Bereichen keine lokalen Fehler enthält.

Die entsprechenden Konfigurationsbereiche können anschließend in anderen Konfigurationen bereits verwendet werden (z. B. für eigene, von diesen Bereichen abhängig Änderungen), die Konfiguration lässt sich aber dort nicht aktivieren.

#### 3.3.2.8.2 Aufrufparameter

**-freigabeuebernahme**

**-verwaltung**={*verwaltungsdaten*} {*verwaltungsdaten*}:

Komplette Pfad- und Dateiangebe der für die Datei  
*verwaltungsdaten.xml*

### 3.3.2.8.3 Startscript für die „Freigabe zur Übernahme“

```
#!/bin/bash
. einstellungen.sh

$java \
-cp ../distributionspakete/de.bsvrz.puk.config/de.bsvrz.puk.config-runtime.jar \
-Xmx300m \
de.bsvrz.puk.config.main.ConfigurationApp \
-freigabeuebernahme \
-verwaltung=../config/verwaltungsdaten.xml \
$debugDefaults \
-debugLevelStdErrText=INFO \
-debugLevelFileText=CONFIG \
```

### 3.3.2.9 Freigabe zur Aktivierung

#### 3.3.2.9.1 Funktion

Bei der Freigabe zur Aktivierung wird jeweils die lokal aktivierte Version eigener Bereiche zur Übernahme und Aktivierung in anderen Konfigurationen freigegeben und entsprechend in der Konfigurationsdatei vermerkt. Damit kann diese Version anschließend in fremden Konfigurationen aktiviert werden.

#### 3.3.2.9.2 Aufrufparameter

**-freigabeaktivierung**

**-verwaltung**={*verwaltungsdaten*} {*verwaltungsdaten*}:

Komplette Pfad- und Dateiangabe der für die Datei  
*verwaltungsdaten.xml*

### 3.3.2.9.3 Startscript für die „Freigabe zur Aktivierung“

```
#!/bin/bash
. einstellungen.sh

$java \
-cp ../distributionspakete/de.bsvrz.puk.config/de.bsvrz.puk.config-runtime.jar \
-Xmx300m \
de.bsvrz.puk.config.main.ConfigurationApp \
-freigabeaktivierung \
-verwaltung=../config/verwaltungsdaten.xml \
$debugDefaults \
-debugLevelStdErrText=INFO \
-debugLevelFileText=CONFIG \
```

### 3.3.2.10 Freigabe zur Aktivierung ohne lokale Aktivierung durch den KV

#### 3.3.2.10.1 Funktion

Spezialvariante für gegenseitig abhängige Konfigurationen → Tutorial Teil C.

Prinzipiell geht es hier um die Möglichkeit zur Weitergabe einer zur Aktivierung freigegebenen Konfiguration an einen anderen Konfigurationsverantwortlichen, obwohl man diese Konfiguration bei sich selbst (auf Grund der gegenseitigen Abhängigkeit) noch gar nicht aktivieren kann.

#### 3.3.2.10.2 Aufrufparameter

**-freigabeZurAktivierungOhne LokaleAktivierungDurchDenKV={freigabeBereiche}** *{freigabeBereiche}*:  
Komma separierte Liste der zu freizugebenden Konfigurationsdateien (also der CONFIG-Dateien).  
**ACHTUNG:** Angegeben werden muss jeweils die **PID** des Konfigurationsbereichs, also der Dateiname der CONFIG-Datei **OHNE** die Endung *.config*

**-verwaltung={verwaltungsdaten}** *{verwaltungsdaten}*:  
Komplette Pfad- und Dateiangabe der für die Datei *verwaltungsdaten.xml*

## Startscript für die Konfigurationsapplikation

```
#!/bin/bash
. einstellungen.sh

$java \
-cp ../distributionspakete/de.bsvrz.puk.config/de.bsvrz.puk.config-runtime.jar \
-Xmx300m \
de.bsvrz.puk.config.main.ConfigurationApp \
-freigabeZurAktivierungOhneLokaleAktivierungDurchDenKV=kv.tmVerkehrGlobal \
-verwaltung=../config/verwaltungsdaten.xml \
$debugDefaults \
-debugLevelStdErrText=INFO \
-debugLevelFileText=CONFIG \
```

## 3.4 Standardworkflows

### 3.4.1 Grundsätzliches

In den nachfolgenden Kapiteln werden die Standardvorgehensweisen für die Erstellung und Pflege von Konfigurationsbereichen unter einem Konfigurationsverantwortlichen bzw. AOE für die Bereiche

- Modelle erstellen und pflegen
- Versorgungsdaten (Objekte) erstellen und pflegen
- Betriebsumgebung erstellen und pflegen
- Projektumgebung erstellen und pflegen

beschrieben.

Ziel ist dabei die grundsätzliche, schematische Darstellung der durchzuführenden Schritte für die jeweilige Aufgabe.

Spezielle Hantierungen, wie das Anlegen eines neuen Konfigurationsverantwortlichen bzw. eines AOE nebst Defaultbereich werden in eigenen Kapiteln detailliert beschrieben.

Die praktische Umsetzung dieser Vorgehensweisen setzt sowohl eine entsprechend eingerichtete Umgebung (siehe Kapitel 3.3.2.2) als auch die Kenntnisse der Anwendung der Konfigurationsapplikation in den unterschiedlichen Betriebsmodi nebst Anpassung der Befehls- und Skriptdateien entsprechend Kapitel 3.3.2 voraus.

In den Darstellungen werden die Namenskonventionen für KV, AOE und Konfigurationsbereiche gemäß Kapitel 3.1.8 angewendet. Dabei wird für die Organisation (normalerweise die Firma oder das Bundesland) in den Beispielen als Name *tutorial* verwendet.

*Kursiv* dargestellte Befehle sind optional und müssen nicht unbedingt ausgeführt werden.

**Hinweis**

- Die einzelnen Schritte des Standardworkflows (Modell, Objekte, Betrieb, Projekte) sind als kompakte Zusammenfassung für die tägliche Arbeit als eigenständige Anlage „BSVRZ Tutorial Konfiguration – Standardworkflow“ vorhanden.

### 3.4.2 Modelle

#### Ziele

- Anlegen eines neuen Arbeitsbereichs für die Modellpflege für den `kv.tutorial.modelleA`.
  - Anlegen der notwendigen Ordnerstruktur (siehe Kapitel 3.3.2.2)
  - Erstellung der Konfigurationsverantwortlichen `kv.tutorial.modelleA`. (siehe Kapitel 3.4.6.1)
  - Erstellung von Modellbereichen (Konfigurationsbereiche mit Modelldefinitionen). (siehe Kapitel 3.4.6.2)
  - Darstellung des Workflow für die Pflege dieses Arbeitsbereichs. (siehe Kapitel 3.4.7)

#### Arbeitspunkte

Die nachfolgende Tabelle enthält die durchzuführenden Arbeitspunkte (AP) für die Modellpflege.

<div style="border: 1px solid black; padding: 5px; width: 40px; margin: 0 auto;"> <p style="font-size: 2em; margin: 0;">M</p> <p style="font-size: 0.8em; margin: 0;">Modelle</p> </div>	Übergreifendes Modell		
	Benötigt		Konfigurationsverantwortlicher für übergreifende Modellierung. I.d.R. Modelle, die über die ZID verbreitet werden.
	kv.tutorial.modelleA		
	Anlegen		
	kb.kv.tutorial.modelleA.xml		
	Arbeitspunkte		
	Arbeits Schritte	mögl. Nächster AP	Bemerkung
	KV erstellen		
	1 Ordner <code>kv.tutorial.modelleA</code> mit Unterordnern etc. anlegen		Ordner Vorlage kopieren und umbenennen, Vorlage umbenennen.
	2 <code>kb.kv.xxx.xml</code> in <code>kb.tutorial.modelleA.xml</code> umbenennen.		
3 Modellierung --> ID etc. anpassen		Pid, Kodierung, Name, Kommentar etc. anpassen.	
4 Verwaltungsdatei anpassen		<b>Pid, KV, etc. anpassen</b>	
5 <b>import</b>	Fehler --> 3	<b>Skript anpassen!</b> Bei Fehler weiter bei 3.	
6 <b>konsistenzpruefung</b>	Fehler --> 3	Bei Fehler weiter bei 3.	
7 <b>aktivierung</b>	Fehler --> 3	Bei Fehler weiter bei 3.	
8 <b>export</b>	Fehler --> 3	<b>Skript anpassen!</b> Bei Fehler weiter bei 3.	
9 <b>freigabe-aktivierung</b>	Fehler --> 3	Bei Fehler weiter bei 3.	
Modelle erstellen			
1 <code>kb.tmTutorialModellA.xml</code> anlegen		Aus Modell-Objekt-Vorlage kopieren und anpassen.	
...			
<code>kb.tmTutorialModellXyz.xml</code> anlegen			
2 Modellierung --> XML-Editor			
3 <b>import</b>	--> 2 (ändern, Fehler)	<b>Skript anpassen!</b> Bei Fehler weiter bei 2.	
4 <b>konsistenzpruefung</b>	--> 2 (ändern, Fehler)	Bei Fehlern oder bei weiteren Änderungen.	
5 <b>export</b>	--> 2 (ändern, Fehler)	<b>Skript anpassen!</b> Bei Fehler weiter bei 2.	
6 <b>aktivierung</b>	--> 2 (ändern, Fehler)	Bei Fehlern oder bei weiteren Änderungen.	
7 <b>export</b>	--> 2 (ändern, Fehler)	<b>Skript anpassen!</b> Bei Fehler weiter bei 2.	
8 <b>freigabe-aktivierung</b>	--> 2 (ändern, Fehler)	Bei Fehlern oder bei weiteren Änderungen.	

Tabelle 3-11: Arbeitsablauf für die Modellpflege

### Erläuterungen zum Bereich „KV erstellen“

- Die Schritte 1 bis 9 (Anlegen einer Konfigurationsverantwortlichen) sind einschließlich der notwendigen Vorlagen für die Versorgungsdatei und Verwaltungsdatei im Kapitel 3.4.6.1 detailliert beschrieben.
- Scriptanpassung im Schritt 5: Hier muss Dateiname der Versorgungsdatei mit dem erstellten Konfigurationsverantwortlichen in das Befehlsscript für den Import eingetragen werden.
- Scriptanpassung im Schritt 8: Hier muss Dateiname der Versorgungsdatei mit dem erstellten Konfigurationsverantwortlichen in das Befehlsscript für den Export eingetragen werden. Alternativ kann hier auch keine einschränkende Angabe erfolgen, es werden dann immer alle Konfigurationsdateien exportiert.

### Erläuterungen zum Bereich „Modelle erstellen“

- Als Vorlage für die Versorgungsdatei mit den Modelldefinitionen kann die Vorlage aus Kapitel 3.4.6.2 verwendet werden.
- Scriptanpassung im Schritt 3: Hier muss Dateiname der Versorgungsdatei mit dem erstellten Modellbereich in das Befehlsscript für den Import eingetragen werden.
- Scriptanpassung im Schritt 5 und 7: Hier muss Dateiname der Versorgungsdatei mit dem erstellten mit dem erstellten Modellbereich in das Befehlsscript für den Export eingetragen werden. Alternativ kann hier auch keine einschränkende Angabe erfolgen, es werden dann immer alle Konfigurationsdateien exportiert.

## 3.4.3 Versorgung (Objekte)

### Ziele

- Anlegen eines neuen Arbeitsbereichs für die Pflege von Versorgungsdaten (Konkrete Objektdefinitionen für eine oder mehrere Anlagen) für den `kv.tutorial.objekteA`.
  - Anlegen der notwendigen Ordnerstruktur (siehe Kapitel 3.3.2.2)
  - Erstellung der Konfigurationsverantwortlichen `kv.tutorial.objekteA`. (siehe Kapitel 3.4.6.1)
  - Erstellung von Objektbereichen (Konfigurationsbereiche mit Objektdefinitionen). (siehe Kapitel 3.4.6.2)
  - Darstellung des Workflow für die Pflege dieses Arbeitsbereichs. (siehe Kapitel 3.4.7)

## Arbeitspunkte

Die nachfolgende Tabelle enthält die durchzuführenden Arbeitspunkte (AP) für die Objektpflege.

Objekte (Modell)		
Benötigt	kv.tutorial.objekteA	Konfigurationsverantwortlicher für projektspezifische/landesspezifische Modellierung und konkrete Objekte.
Anlegen	kb.kv.tutorial.objekteA.xml	
Arbeitsschritte	mögl. Nächster AP	Bemerkung
<b>KV erstellen</b>		
1	Ordner kv.tutorial.objekteA mit Unterordner etc. anlegen	Ordnervorlage kopieren und umbenennen, Vorlage umbenennen.
2	kb.kv.xxx.xml in kb.kv.tutorial.objekteA.xml umbenennen.	
3	Modellierung --> ID etc. anpassen	Pid, Kodierung, Name, Kommentar etc. anpassen.
4	Verwaltungsdatei anpassen	<b>Pid, KV, etc. anpassen</b>
5	<b>import</b>	<b>Skript anpassen !</b> , Bei Fehler weiter bei 3.
6	<b>konsistenzpruefung</b>	Fehler --> 3 Bei Fehler weiter bei Schritt 3.
7	<b>aktivierung</b>	Fehler --> 3 Bei Fehler weiter bei Schritt 3.
8	<b>export</b>	Fehler --> 3 <b>Skript anpassen !</b> , Bei Fehler weiter bei 3.
9	<b>freigabe-aktivierung</b>	Fehler --> 3 Bei Fehler weiter bei Schritt 3.
<b>Objekte (Modelle) erstellen</b>		
1	kb.objekteTutorialObjekteA.xml anlegen	Aus Modell-Objekt-Vorlage kopieren und anpassen.
...		
	kb.objekteTutorialObjekteXyz.xml anlegen	
2	Objekte anlegen --> XML-Editor	
3	Verwaltungsdatei anpassen	<b>Modell und KV aus "M" eintragen !!</b>
4	<b>import</b>	--> 2 (ändern, Fehler) <b>Skript anpassen !</b> , Bei Fehler weiter bei 2.
5	<b>konsistenzpruefung</b>	--> 2 (ändern, Fehler) Bei Fehlern oder bei weiteren Änderungen.
6	<b>export</b>	--> 2 (ändern, Fehler) <b>Skript anpassen!</b> , Bei Fehler weiter bei 2.
7	<b>aktivierung</b>	--> 2 (ändern, Fehler) Bei Fehlern oder bei weiteren Änderungen.
8	<b>export</b>	--> 2 (ändern, Fehler) <b>Skript anpassen!</b> , Bei Fehler weiter bei 2.
9	<b>freigabe-aktivierung</b>	--> 2 (ändern, Fehler) Bei Fehlern oder bei weiteren Änderungen.

Tabelle 3-12: Arbeitsablauf für die Objektpflege

### Erläuterungen zum Bereich „KV erstellen“

- Die Schritte 1 bis 9 (Anlegen einer Konfigurationsverantwortlichen) sind einschließlich der notwendigen Vorlagen für die Versorgungsdatei und Verwaltungsdatei im Kapitel 3.4.6.1 detailliert beschrieben.
- Scriptanpassung im Schritt 5: Hier muss Dateiname der Versorgungsdatei mit dem erstellten Konfigurationsverantwortlichen in das Befehlsskript für den Import eingetragen werden.
- Scriptanpassung im Schritt 8: Hier muss Dateiname der Versorgungsdatei mit dem erstellten Konfigurationsverantwortlichen in das Befehlsskript für den Export eingetragen werden. Alternativ kann hier auch keine einschränkende Angabe erfolgen, es werden dann immer alle Konfigurationsdateien exportiert.

### Erläuterungen zum Bereich „Objekte erstellen“

- Als Vorlage für die Versorgungsdatei mit den Objektdefinitionen kann die Vorlage aus Kapitel 3.4.6.2 verwendet werden.
- Scriptanpassung im Schritt 4: Hier muss Dateiname der Versorgungsdatei mit dem erstellten Objektdefinitionen in das Befehlsskript für den Import eingetragen werden.
- Scriptanpassung im Schritt 6 und 8: Hier muss Dateiname der Versorgungsdatei mit dem erstellten mit dem erstellten Objektdefinitionen in das Befehlsskript für den Export eingetragen werden. Alternativ kann hier auch keine einschränkende Angabe erfolgen, es werden dann immer alle Konfigurationsdateien exportiert.

### 3.4.4 Betrieb

#### Ziele

- Anlegen eines neuen Arbeitsbereichs für die Erstellung eines AOE und Default-Bereichs für den `kv.aoe.tutorial.projektA`.
  - Anlegen der notwendigen Ordnerstruktur (siehe Kapitel 3.3.2.2)
  - Erstellung einer AOE `kv.aoe.tutorial.projektA` (siehe Kapitel 3.4.6.1)
  - Erstellung eines Default-Bereichs für den `kv.aoe.tutorial.projektA` (siehe Kapitel 3.4.6.1.8)

#### Arbeitspunkte

Die nachfolgende Tabelle enthält die durchzuführenden Arbeitspunkte (AP) für den Betrieb (AOE, Defaultbereich).

Betrieb		
<b>B</b> Betrieb	Benötigt	Autarke Organisationseinheit für ein konkretes Projekt (AOE: VRZ, UZ, etc.). Enthält i.d.R. nur den AOE und den notwendigen Defaultbereich für dynamische Objekte.
	Anlegen	
	<code>kv.aoe.tutorial.projektA</code>	
	<code>kb.kv.aoe.tutorial.projektA.xml</code>	
	<code>kb.default.kv.aoe.tutorial.projektA.xml</code>	
Arbeitspunkte		
Arbeitspunkte	mögl. Nächster AP	Bemerkung
<b>AOE erstellen</b>		
1 Ordner <code>kv.aoe.tutorial.projektA</code> mit Unterordnern anlegen		Ordnervorlage kopieren und umbenennen, Vorlage umbenennen.
2 <code>kb.kv.aoe.xxx.xml</code> in <code>kb.kv.aoe.tutorial.projektA.xml</code> umbenennen.		
3 Modellierung --> ID etc. anpassen		Pid, Kodierung, Name, Kommentar etc. anpassen.
4 Verwaltungsdatei anpassen		<b>Pid, KV, etc. anpassen</b>
5 <code>import</code>	Fehler --> 3	<b>Skript anpassen !</b> , Bei Fehler weiter bei 3.
6 <code>konsistenzpruefung</code>	Fehler --> 3	Bei Fehler weiter bei Schritt 3.
7 <code>aktivierung</code>	Fehler --> 3	
<b>Default-Bereich erstellen</b>		
8 <code>kb.default.kv.aoe.xxx.xml</code> in <code>kb.default.kv.aoe.tutorial.projektA.xml</code> umbenennen.		Aus anderem Bereich Datei als Vorlage kopieren.
9 Modellierung --> ID etc. anpassen		Pid, ID-KV, Name, Kommentar etc. anpassen.
10 <code>import</code>	Fehler --> 9	<b>Skript anpassen !</b> , Bei Fehler weiter bei 9.
11 <code>konsistenzpruefung</code>	Fehler --> 9	Bei Fehler weiter bei Schritt 9.
12 <code>aktivierung</code>	Fehler --> 9	Bei Fehler weiter bei Schritt 9.
13 <code>export</code>	Fehler --> 9	<b>Skript anpassen !</b> , Bei Fehler weiter bei 9.
14 <code>freigabe-aktivierung</code>	Fehler --> 9	Bei Fehler weiter bei Schritt 9.

Tabelle 3-13: Arbeitsablauf für den Betrieb (AOE, Defaultbereich)

#### Erläuterungen zum Bereich „AOE erstellen“

- Die Schritte 1 bis 7 (Anlegen einer Konfigurationsverantwortlichen) sind einschließlich der notwendigen Vorlagen für die Versorgungsdatei und Verwaltungsdatei im Kapitel 3.4.6.1 detailliert beschrieben.
- Skriptanpassung im Schritt 5: Hier muss Dateiname der Versorgungsdatei mit dem erstellten AOE in das Befehlsskript für den Import eingetragen werden.

### Erläuterungen zum Bereich „Defaultbereich erstellen“

- Die Schritte 8 bis 14 (Anlegen eines Defaultbereichs) sind einschließlich der notwendigen Vorlagen für die Versorgungsdatei und Verwaltungsdatei im Kapitel 3.4.6.1 detailliert beschrieben.
- Scriptanpassung im Schritt 10: Hier muss Dateiname der Versorgungsdatei mit dem erstellten Defaultbereich in das Befehlsscript für den Import eingetragen werden.
- Scriptanpassung im Schritt 13: Hier muss Dateiname der Versorgungsdatei mit dem erstellten mit dem erstellten Objektdefinitionen in das Befehlsscript für den Export eingetragen werden. Alternativ kann hier auch keine einschränkende Angabe erfolgen, es werden dann immer alle Konfigurationsdateien exportiert.

### 3.4.5 Projekt

#### Ziele

- Anlegen eines neuen Arbeitsbereichs mit einer kompletten Konfiguration im Rahmen eines Projekts (Zusammenstellung von Modellen und Versorgungsdaten unterschiedlicher Konfigurationsverantwortlicher), die unter einem AOE betrieben werden sollen.
  - Anlegen der notwendigen Ordnerstruktur (siehe Kapitel 3.3.2.2)
  - Zusammenstellung der benötigten Konfigurationsdateien.
  - Darstellung des Workflow für die Pflege dieses Arbeitsbereichs. (siehe Kapitel 3.4.7)

## Arbeitspunkte

Die nachfolgende Tabelle enthält die durchzuführenden Arbeitspunkte (AP) für Projektpflege.

Projekt			
<b>P</b> Projekte	<b>Benötigt</b> Projektkordner diverse Konfigurationsbereiche	neu anlegen zusammen kopieren	Konkrete Projektumgebung für eine AOE. Alle benötigten Konfigurationsbereiche werden ins Projekt kopiert und dort aktiviert. Im Projekt selbst werden keine KB importiert oder geändert.
	<b>Arbeitsschritte</b>	mögl. Nächster AP	Bemerkung
	Neue erstellte Konfigurationsbereiche Kopieren		
	1 kb.kv.tutorial.modellA.config		
	2 kb.tmTutorialModellA.config		Eigene Modelle übernehmen
	...		
	kb.tmTutorialModellXyz.config		
	3 ...		Diverse Modelle anderer KV übernehmen
	4 kb.kv.tutorial.objekteA.config		
	5 kb.objekteTutorialObjekteA.config		Eigene Versorgungen übernehmen
	...		
	kb.objekteTutorialObjekteXyz.config		
	6 ...		Diverse Versorgungen anderer KV übernehmen
	7 kb.kv.aoe.tutorial.projektA.config		<b>AOE</b> für dieses Projekt mit Defaultbereich für dynamische Objekte.
	8 kb.default.kv.aoe.tutorial.projektA.config		
	Konfiguration aktivieren		
	9 Verwaltungsdatei anpassen		<b>Bereiche aus "M", "O" und "B" eintragen!</b>
	11 <i>konsistenzpruefung</i>	Fehler --> M, O, B	Bei Fehlern zurück zu den Schritten M, O oder B
	12 <i>aktivierung</i>	Fehler --> M, O, B	
	Projekt starten		Siehe Tutorial "Kernsoftware"
	1 <i>Einstellungen testen</i>		<i>Projekteinstellung (Java, Classpath etc.) testen</i>
	2 <i>Kernsoftware starten</i>		System starten
	3 <i>GTM starten</i>		Neue Modelle und Objekte über GTM prüfen
	4 <i>KonfigBrowserOffline starten</i>		<i>Neue Modelle und Objekte prüfen</i>
	5 ...		

Tabelle 3-14: Arbeitsablauf für die Projektpflege

## Erläuterungen

- Schritte 1 bis 8ff: Hier sind jeweils die für die Zusammenstellung benötigten zur **Aktivierung freigegebenen** Konfigurationsdateien der unterschiedlichen Konfigurationsverantwortlichen nebst den Konfigurationsdateien des jeweiligen Konfigurationsverantwortlichen in die eigene Projektumgebung zu kopieren (benötigt werden hier nur die CONFIG-Dateien).
- Schritt 9: In die Verwaltungsdatei müssen alle Konfigurationsbereiche aus Schritt 1-8ff eingetragen werden. Als Konfigurationsverantwortlicher ist der AOE aus Schritt B einzutragen. Für diesen sind ebenfalls die Konfigurationsdatei für den AOE und den Defaultbereich (in Schritt 7 und 8) dargestellt in die Projektumgebung zu kopieren (und natürlich in die Verwaltungsdatei einzutragen).
- Schritt 12: Mit der Aktivierung werden die aktuellen Versionsnummern der zu übernehmenden Konfigurationsbereiche in die Verwaltungsdatei neu eingetragen bzw. aktualisiert.
- Projekt starten: Für den Test der erstellten Konfiguration im Anwendungsbetrieb siehe Tutorial „Kernsoftware“.

### 3.4.6 Beschreibung einzelner Teilschritte für die Workflow

Für die Aufgaben

- KV erstellen
- AOE erstellen
- Default-Bereich erstellen

sind in den nachfolgenden Kapiteln detaillierte Anleitungen nebst Dateivorlagen beschrieben.

### 3.4.6.1 Konfigurationsverantwortlichen oder AOE erstellen

Die Erstellung eines Konfigurationsverantwortlichen oder eines AOE sind hinsichtlich der durchzuführenden Arbeitsschritte sehr ähnlich.

Folgende Schritte sind dazu notwendig (*Schritt 8 und 9 nur für AOE*):

1. Beantragung einer Kodierungsbasis bei der ZS für die Organisation
2. Festlegung der Kodierung für diesen KV oder die AOE
3. Festlegung des Typs des Konfigurationsverantwortlichen („einfacher“ KV oder „Autarke Organisations-Einheit“ AOE)
4. Festlegung der PID für den KV oder die AOE
5. Erstellung des Konfigurationsbereichs für den KV oder die AOE (Anpassung der Vorlage)
6. Verwaltungsdatei erstellen (Anpassung Vorlage)
7. Konfigurationsdatei für KV oder AOE erstellen (Import – Aktivierung)
8. *Erstellung des Defaultbereichs (Anpassung Vorlage)*
9. *Konfigurationsdatei für Defaultbereich erstellen (Import – Aktivierung – Freigabe Aktivierung)*
10. Konfigurationsbereiche zur Aktivierung freigeben (Freigabe Aktivierung)

#### 3.4.6.1.1 Schritt 1: Beantragung einer Kodierungsbasis bei der ZS für die Organisation

Hinweis: Dieser Schritt muss je Organisation NUR einmal durchgeführt werden!

Zum Einrichten eines neuen Konfigurationsverantwortlichen oder einer AOE muss als erstes eine eindeutige PID und eine eindeutige Kodierung festgelegt werden.

Die Vergabe der PID und die Vergabe der Kodierung unterliegen der Verantwortung der durchführenden Organisation, wobei allerdings die Kodierung auf Basis einer Kodierungsbasis erfolgt, die zentral bei der ZS beantragt werden muss.

Die ZS vergibt für jede Organisation (Firma/Verwaltung/Bundesland) eine eindeutige, zweistellige Nummer (Kodierungsbasis).

Aus dieser Kodierungsbasis kann dann die Organisation durch hinzufügen zweier zusätzlicher Ziffern eine 4-stellige Zahl, die so genannte Kodierung bilden (insgesamt also 100 mögliche Kodierungen je Organisation).

→ Kodierung XXYY

mit

XX = Kodierungsbasis von ZS vorgegeben

YY = Ziffernkombination durch Organisation frei zu vergeben

Wenn alle 100 mögliche Kodierungen XXYY von der Organisation erschöpft sind, kann sich die Organisation von der ZS eine weitere zweistellige Kodierung (ZZ) zuordnen lassen und dann weitere 100 Kodierungen ZZYY nach dem selben Schema vergeben.

#### 3.4.6.1.2 Schritt 2: Festlegung der Kodierung für diesen KV / diese AOE

Hat man von der ZS für seine Organisation eine Kodierungsbasis erhalten, so kann die Kodierung nach den unter Schritt 1. beschriebenen Regeln frei vergeben werden.

Für die folgenden Beispiele wird davon ausgegangen, dass die Organisation **tutorial** von der ZS die Kodierungsbasis **99** zugeteilt bekommen hat.

Die möglichen Kodierungen liegen dann im Bereich von **9900** bis **9999**.

**WICHTIG**

- PID und Kodierung des KV bzw. AOE müssen sowohl einzeln für sich (und damit auch in der Kombination) eindeutig sein.
- Die Verantwortung dafür und die Verwaltung liegen in der Verantwortung der jeweiligen Organisation.
- Die ZS sorgt dafür, dass die Vergabe der Kodierungsbasis eindeutig ist.

### 3.4.6.1.3 Schritt 3: Festlegung des Typs des Konfigurationsverantwortlichen („einfacher“ KV oder „Autarke Organisations-Einheit“ AOE)

Die Unterschiede zwischen einem einfachen KV und einer AOE wurden in den Kapiteln 2.1.2.4 und 2.1.2.5 detailliert beschrieben.

Als Merkregel gilt:

- **Konfigurationsverantwortliche** werden für die **Definition** von Modellen und Objekten verwendet → Typ ist **typ.konfigurationsVerantwortlicher**
- **AOE** werden für den **Betrieb** einer Konfiguration verwendet → Typ ist **typ.autarkeOrganisationsEinheit**

D. h., für die Standardworkflow „Modelle“ und „Objekte“ werden KV verwendet, für den „Betrieb“ und „Projekte“ werden AOE angelegt.

### 3.4.6.1.4 Schritt 4: Festlegung der PID für den KV / die AOE

Die PID für den KV / die AOE muss ebenfalls „weltweit“ eindeutig vergeben werden, ist aber prinzipiell durch die Organisation frei festlegbar.

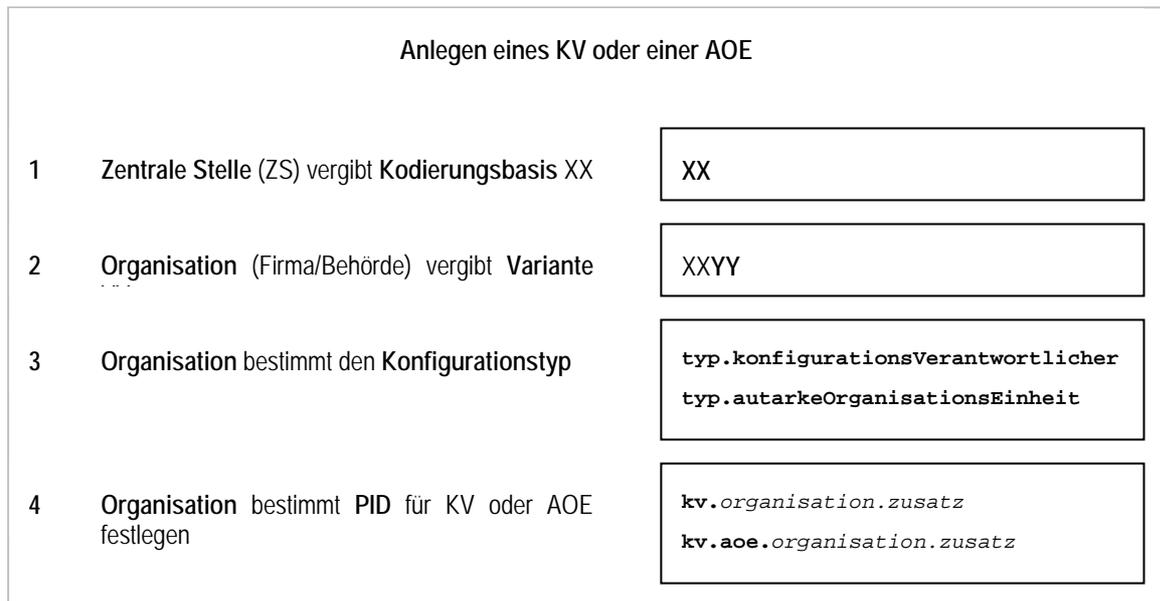
Werden die Namenskonventionen für KV, AOE und Konfigurationsbereiche gemäß Kapitel 3.1.8 angewendet, so ergibt sich für die PID (mit Organisation = **tutorial**):

für den KV : `kv.tutorial.XXX`

für die AOE: `kv.aoe.tutorial.XXX`

wobei .XXX für die insgesamt 100 möglichen Varianten frei vergeben werden kann.

Die Schritte 1. bis 4. sind in der nachfolgenden Abbildung zusammengefasst.



*Abbildung 3-7: Einrichten von Konfigurationsverantwortlichen(KV oder AOE)*

#### 3.4.6.1.5 Schritt 5: Erstellung des Konfigurationsbereichs für den KV oder die AOE (Anpassung der Vorlage)

Abhängig vom gewählten Typ des Konfigurationsverantwortlichen (`typ.konfigurationsVerantwortlicher` oder `typ.autarkeOrganisationsEinheit`) ist die Versorgungsdatei (XML-Datei) für den Konfigurationsbereich des KV bzw. AOE anzulegen. Dazu können die nachfolgend beschriebenen Vorlagen verwendet werden, die an den entsprechend gekennzeichneten Stellen mit den Daten aus den Schritten 1.- 4. gefüllt werden müssen.

#### Beispiel für die Daten aus Schritt 1. – 4. für einen KV:

<b>Kodierungsbasis</b>	99	von der ZS für die Organisation eindeutig vergeben
<b>Kodierung</b>	9901	lfd. Nr. 01 durch die Organisation eindeutig vergeben
<b>Konfigurationstyp</b>	<code>typ.konfigurationsVerantwortlicher</code>	KV ist „einfacher“ Konfigurationsverantwortlicher
<b>Organisation</b>	tutorial	Name der Organisation
<b>Zusatz</b>	modelleA	Ergänzende Angabe, dass dieser KV die ModelleA verwaltet.
<b>PID</b>	<code>kv.tutorial.modelleA</code>	PID des KV

### Vorlage für die Versorgungsdatei eines KV

Die **gelb** markierten Stellen **müssen** entsprechend den Daten angepasst werden, die **grün** markierten Stellen sind Kommentare und **sollten** korrekt angepasst werden.

Die Versionsnummer (**rot** markiert) sollte immer als Leerstring eingegeben werden. Die korrekte Nummer wird dann automatisch von der Konfigurationsapplikation beim Export eingetragen.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE konfigurationsBereich PUBLIC "-//K2S//DTD Dokument//DE" "K2S.dtd">
<konfigurationsBereich pid="kb.kv. ORGA.ZUSATZ" name="KV. ORGA.ZUSATZ"
  verantwortlich="kv. ORGA.ZUSATZ">
  <info>
    <kurzinfo>
      Bereich mit dem Konfigurationsverantwortlichen für Firma ORGA.
    </kurzinfo>
  </info>
  <konfigurationsAenderung
    stand="xx.xx.2009"
    version=""
    autor="XXX"
    grund="Neuanlage">
  </konfigurationsAenderung>
  <modell>
  </modell>
  <objekte>
    <konfigurationsObjekt
      pid="kv. ORGA.ZUSATZ"
      name="KV. ORGA.ZUSATZ"
      typ="typ.konfigurationsVerantwortlicher">
      <info>
        <kurzinfo>Konfigurationsverantwortlicher von ORGA für XXX.</kurzinfo>
      </info>
      <datensatz
        attributgruppe="atg.konfigurationsVerantwortlicherEigenschaften"
        aspekt="asp.eigenschaften">
        <datum name="kodierung" wert="XXYY"/>
        <datenfeld name="defaultBereich"/>
      </datensatz>
    </konfigurationsObjekt>
  </objekte>
</konfigurationsBereich>
```

Abbildung 3-8: Vorlage einer Versorgungsdatei für einen KV (kb.kv. ORGA.ZUSATZ.xml)

### Beispiel für die Daten aus Schritt 1. – 4. für eine AOE:

<b>Kodierungsbasis</b>	99	von der ZS für die Organisation eindeutig vergeben
<b>Kodierung</b>	9951	lfd. Nr. 51 durch die Organisation eindeutig vergeben
<b>Konfigurationstyp</b>	typ.autarkeOrganisationsEinheit	KV ist Autarke Organisations-Einheit
<b>Organisation</b>	tutorial	Name der Organisation
<b>Zusatz</b>	projektA	Ergänzende Angabe, dass dieser KV für das ProjektA verwendet wird.
<b>PID</b>	kv.aoe.tutorial.projektA	PID des AOE

### Vorlage für die Versorgungsdatei eines AOE

Die **gelb** markierten Stellen **müssen** entsprechend den Daten angepasst werden, die **grün** markierten Stellen sind Kommentare und **sollten** korrekt angepasst werden.

Die Versionsnummer (**rot** markiert) sollte immer als Leerstring eingegeben werden. Die korrekte Nummer wird dann automatisch von der Konfigurationsapplikation beim Export eingetragen.

Wesentliche Unterschiede zur KV-Vorlage sind **schwarz-fett** markiert.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE konfigurationsBereich PUBLIC "-//K2S//DTD Dokument//DE" "K2S.dtd">
<konfigurationsBereich pid="kb.kv.aoe.ORG.A.ZUSATZ" name="KV.AOE.ORG.A.ZUSATZ"
verantwortlich="kv.aoe.ORG.A.ZUSATZ">
  <info>
    <kurzinfo>
      Bereich mit der Autarken Organisationseinheit für Firma ORGA.
    </kurzinfo>
  </info>
  <konfigurationsAenderung
stand="xx.xx.2009"
version=""
autor="XXX"
grund="Neuanlage">
  </konfigurationsAenderung>
  <modell>
  </modell>
  <objekte>
    <konfigurationsObjekt
pid="kv.aoe.ORG.A.ZUSATZ"
name="KV.AOE.ORG.A.ZUSATZ"
typ="typ.autarkeOrganisationsEinheit">
      <info>
        <kurzinfo>Konfigurationsverantwortlicher von ORGA für XXX.</kurzinfo>
      </info>
      <datensatz
attributgruppe="atg.konfigurationsVerantwortlicherEigenschaften"
aspekt="asp.eigenschaften">
        <datum name="kodierung" wert="XXYY"/>
        <datenfeld name="defaultBereich">
          <datum name="0" wert="kb.default.kv.aoe.ORG.A.ZUSATZ"/>
        </datenfeld>
      </datensatz>
      <objektMenge name="Ereignisse"/>
      <objektMenge name="SimulationsStrecken"/>
      <objektMenge name="PuaSkripte"/>
      <objektMenge name="Simulationen"/>
      <objektMenge name="SystemKalenderEinträge"/>
      <objektMenge name="EreignisTypen"/>
      <objektMenge name="Meldungen"/>
    </konfigurationsObjekt>
  </objekte>
</konfigurationsBereich>
```

Abbildung 3-9: Vorlage einer Versorgungsdatei für eine AOE (kb.kv.aoe.ORG.A.ZUSATZ.xml)

#### WICHTIG

- Die Einträge mit den notwendigen Objektmengen beziehen sich auf den Stand vom 07. Januar 2009. Bei Änderungen und Ergänzungen des Datenmodells müssen hier zukünftig gegebenenfalls zusätzliche Mengen eingetragen werden!

### 3.4.6.1.6 Schritt 6: Verwaltungsdatei erstellen (Anpassung Vorlage)

#### Vorlage für die Verwaltungsdatei eines KV und eines AOE

Die **gelb** markierten Stellen **müssen** entsprechend den Daten angepasst werden

Die **grün** markierte Zeile kann bei der Erstellung eines KV weggelassen werden.

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<!--
~ Vorlage für die Verwaltungsdatei eines KV und AOE.
~
~ Aus: BSVRZ-Tutorial Konfiguration, erstellt 2008 durch inovat, Dipl.-Ing. H. C. Kniß
~
~ Durchzuführende Anpassungen:
~
~ 1. Konfigurationsverantwortlicher: PID anpassen
~ 2. Eintrag "<konfigurationsbereich pid="kb.systemModellAoe" verzeichnis=".....">
~ kann bei KV entfallen.
~ 3. Ggf. Verzeichnispfade an eigene Umgebung anpassen
~ 4. Ggf. Versionsnummern für auf aktuellen Stand anpassen (Vorlage Stand 07.01.2009)
~
-->

<!DOCTYPE verwaltungsdaten PUBLIC "-//K2S//DTD Verwaltung//DE" "verwaltungsdaten.dtd">
<verwaltungsdaten>
  <konfigurationsverantwortlicher pid="kv. ORGA.ZUSATZ" />
  <konfigurationsbereich pid="kb.kv.kappich" verzeichnis="../../kv.kappich/config/">
    <version nr="1" />
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.metaModellGlobal" verzeichnis="../../kv.kappich/config/">
    <version nr="11" />
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.systemModellGlobal"
    verzeichnis="../../kv.kappich/config/">
    <version nr="19" />
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.fachModellGlobal" verzeichnis="../../kv.kappich/config/">
    <version nr="7" />
  </konfigurationsbereich>

  <konfigurationsbereich pid="kb.tmVewBetriebGlobal"
    verzeichnis="../../kv.kappich/config/">
    <version nr="5" />
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.tmVewSimulationGlobal"
    verzeichnis="../../kv.kappich/config/">
    <version nr="7" />
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.tmIlseTls" verzeichnis="../../kv.kappich/config/">
    <version nr="4" />
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.systemModellAoe"
    verzeichnis="../../kv.kappich/config/">
    <version nr="4" />
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.tmBuVGGlobal"
    verzeichnis="../../kv.kappich/config/">
    <version nr="5" />
  </konfigurationsbereich>

  <konfigurationsbereich pid="kb.kv.bea" verzeichnis="../../kv.bea/config/">
    <version nr="1" />
  </konfigurationsbereich>
  <konfigurationsbereich pid="kb.tmVewProtokolleGlobal"
    verzeichnis="../../kv.bea/config/">
    <version nr="2" />
  </konfigurationsbereich>

```

```

<konfigurationsbereich pid="kb.kv.inovat" verzeichnis=".../kv.inovat/config/">
  <version nr="1"/>
</konfigurationsbereich>
<konfigurationsbereich pid="kb.tmGanglinienGlobal"
  verzeichnis=".../kv.inovat/config/">
  <version nr="10"/>
</konfigurationsbereich>
<konfigurationsbereich pid="kb.tmVerkehrGlobal" verzeichnis=".../kv.inovat/config/">
  <version nr="31"/>
</konfigurationsbereich>
<konfigurationsbereich pid="kb.tmEreignisKalenderGlobal"
  verzeichnis=".../kv.inovat/config/">
  <version nr="10"/>
</konfigurationsbereich>
<konfigurationsbereich pid="kb.tmSystemKalenderGlobal"
  verzeichnis=".../kv.inovat/config/">
  <version nr="6"/>
</konfigurationsbereich>
<konfigurationsbereich pid="kb.tmGeoReferenzierungGlobal"
  verzeichnis=".../kv.inovat/config/">
  <version nr="2"/>
</konfigurationsbereich>
<konfigurationsbereich pid="kb.tmKExtlsGlobal" verzeichnis=".../kv.inovat/config/">
  <version nr="5"/>
</konfigurationsbereich>
<konfigurationsbereich pid="kb.tmTmcGlobal" verzeichnis=".../kv.inovat/config/">
  <version nr="4"/>
</konfigurationsbereich>

<konfigurationsbereich pid="kb.kv.dambach" verzeichnis=".../kv.dambach/config/">
  <version nr="1"/>
</konfigurationsbereich>
<konfigurationsbereich pid="kb.tmRechner" verzeichnis=".../kv.dambach/config/">
  <version nr="4"/>
</konfigurationsbereich>
<konfigurationsbereich pid="kb.tmVewEngstellenGlobal"
  verzeichnis=".../kv.dambach/config/">
  <version nr="2"/>
</konfigurationsbereich>
</verwaltungsdaten>

```

Abbildung 3-10: Vorlage der Verswaltungsdatei für KV und AOE (verwaltungsdaten.xml)

#### WICHTIG

- Die Einträge mit den Versionsnummern beziehen sich auf den Stand vom 07.01.2009. Bei Änderungen und Ergänzungen des Datenmodells müssen hier zukünftig gegebenenfalls Versionsnummern angepasst werden!

#### 3.4.6.1.7 Schritt 7: Konfigurationsdatei für KV oder AOE erstellen (Import – Aktivierung)

Der neu erstellte Konfigurationsbereich für den KV bzw. die AOE muss mit der Konfigurationsapplikation (und den angepassten Scripten für den Import) in die Konfiguration übernommen werden (→ Erzeugung der Konfigurationsdatei `kb.kv.xxx.config`). Dazu sind die Befehle

- **Importieren**
- **Aktivieren**

- Exportieren (bei Bedarf, die Konfigurationsapplikation trägt dann z. B. die aktuelle Versionsnummer in die Änderungsübersicht und den Kommentar mit Abhängigkeiten zu anderen Bereich in die Versorgungsdatei ein)

durchzuführen.

### 3.4.6.1.8 Schritt 8 (nur bei Erstellung AOE): Erstellung des Defaultbereichs (Anpassung Vorlage)

Im Unterschied zur Erstellung eines "einfachen" Konfigurationsverantwortlichen muss ein Konfigurationsverantwortlicher, der als Autarke Organisationseinheit eingesetzt wird, einen Default-Bereich haben, in dem z.B. dynamische Objekte der unterschiedlichen Objektmengen (wie z. B. für Simulationen etc.) verwaltet werden können.

Dieser Default-Bereich muss vorhanden sein und wird immer dann verwendet, wenn eine Applikation ein dynamisches Objekt anlegt und keinen zu verwendenden Konfigurationsbereich für die Speicherung angibt.

Für den Default-Bereich muss genau die AOE (also die PID des Konfigurationsverantwortlichen) verwendet werden, die auch für den AOE selbst verwendet wurde.

Für den Defaultbereich kann die nachfolgende Vorlage verwendet werden.

Die **gelb** markierten Stellen **müssen** entsprechend den Daten angepasst werden, die **grün** markierten Stellen sind Kommentare und **schwarz** korrekt angepasst werden.

Die Versionsnummer (**rot** markiert) sollte immer als Leerstring eingegeben werden. Die korrekte Nummer wird dann automatisch von der Konfigurationsapplikation beim Export eingetragen.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE konfigurationsBereich PUBLIC "-//K2S//DTD Dokument//DE" "K2S.dtd">
<konfigurationsBereich pid="kb.default.kv.aoe.ORGANISATION.ZUSATZ" name="DEAULT.KV.AOE.ORGANISATION.ZUSATZ"
verantwortlich="kv.aoe.ORGANISATION.ZUSATZ">
  <info>
    <kurzinfo>
      Default-Bereich der Autarken Organisationseinheit für Firma ORGANISATION.
    </kurzinfo>
  </info>
  <konfigurationsAenderung
stand="xx.xx.2009"
version=""
autor="XXX"
grund="Neuanlage">
  </konfigurationsAenderung>
  <modell>
  </modell>
  <objekte>
  </objekte>
</konfigurationsBereich>
```

Abbildung 3-11: Versorgungsdatei Default-Bereich für eine AOE  
(kb.default.kv.aoe.ORGANISATION.ZUSATZ.xml)

### 3.4.6.1.9 Schritt 9 (nur bei Erstellung AOE): Konfigurationsdatei für Defaultbereich erstellen (Import – Aktivierung)

Der neu erstellte Konfigurationsbereich für die AOE muss mit der Konfigurationsapplikation (und den angepassten Scripten für den Import) in die Konfiguration übernommen werden (→ Erzeugung der Konfigurationsdatei kb.default.kv.aoe.xxx.config). Dazu sind die Befehle

- Importieren
- Aktivieren

- Exportieren (bei Bedarf, die Konfigurationsapplikation trägt dann z. B. die aktuelle Versionsnummer in die Änderungsübersicht und den Kommentar mit Abhängigkeiten zu anderen Bereich in die Versorgungsdatei ein)

durchzuführen.

### 3.4.6.1.10 Schritt 10: Konfigurationsbereiche zur Aktivierung freigeben (Freigabe Aktivierung)

Damit die erzeugten Konfigurationsbereiche weitergegeben werden können, muss spätestens vor der Weitergabe der Befehl

- **Freigabe Aktivierung**

durchgeführt werden (Details siehe Kapitel 3.5).

### 3.4.6.2 Vorlage für eine Versorgungsdatei für Modell- und/oder Objektdefinitionen

Für die Erstellung einer Versorgungsdatei für Modell- und/oder Objektdefinitionen kann die nachfolgende Vorlage verwendet werden.

Die **gelb** markierten Stellen **müssen** entsprechend den Daten angepasst werden, die **grün** markierten Stellen sind Kommentare und **sollten** korrekt angepasst werden.

Die Versionsnummer (**rot** markiert) sollte immer als Leerstring eingegeben werden. Die korrekte Nummer wird dann automatisch von der Konfigurationsapplikation beim Export eingetragen.

Der Platzhalter **TEILMODELL-OBJEKTE** ist je nach Verwendung der Vorlage z. B. durch **tmBezeichnungDesModells** oder **objekteBezeichnungDerObjekte** zu ersetzen.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE konfigurationsBereich PUBLIC "-//K2S//DTD Dokument//DE" "K2S.dtd">
<konfigurationsBereich pid="kb.TEILMODELL-OBJEKTE" name="Teilmodell/Objekte xxx"
verantwortlich="kv. ORGA.ZUSATZ ">
  <info>
    <kurzinfo>
      Teilmodell / Objektdefinitionen für XXX.
    </kurzinfo>
  </info>
  <konfigurationsAenderung
stand="xx.xx.2009"
version=""
autor="XXX"
grund="Neuanlage.">
  </konfigurationsAenderung>
  <modell>
  </modell>
  <objekte>
  </objekte>
</konfigurationsBereich>
```

Abbildung 3-12: Versorgungsdatei Vorlage für Modell- und/oder Objektdefinitionen(kb.tmModellXyz.xml bzw. kb.objekteXyz.xml)

### 3.4.6.3 Modellierungsmöglichkeiten (Datenmodell, Metamodell)

Zum Ändern der Konfiguration, insbesondere bei der Neuerstellung oder Änderung des Modellbereichs eines Konfigurationsbereichs, ist das grundsätzliche Verständnis der vorhandenen Möglichkeiten zur Modellierung notwendig.

Eine kompakte Einführung in die Datenmodellierung des BSVRZ enthält das Kapitel 2.1 des Tutorials „Kernsoftware“ [TutorialBSVRZ].

### 3.4.7 Übernahme der Änderungen

Nach Änderungen an den Modellen oder Versorgungen sind die üblichen Schritte zur Übernahme der Änderungen in die Konfiguration durchzuführen (Schritte in Klammern sind optional):

1. Import
2. Konsistenzprüfung
3. (Export)
4. weiter bei 1.

oder

1. Import
2. (Konsistenzprüfung)
3. Aktivierung
4. (Freigabe zur XXX)
5. Export (→ Eintrag der aktuellen Version in die Versorgungsdatei)
6. weiter bei 1.

#### **Tipp**

- Der Zyklus Import, Konsistenzprüfung (ggf. Export) sollte möglichst oft mit nach nur kleineren Änderungen an der Versorgungsdatei durchgeführt werden. Nur so lassen auftretende Fehler einfach nachvollziehen und korrigieren.

**WICHTIG ■ WICHTIG ■ WICHTIG ■ WICHTIG ■ WICHTIG ■ WICHTIG**

- **Nach jedem Zyklus, in jedem Fall aber spätestens nach jeder Aktivierung, sowohl die Versorgungsdateien aber insbesondere die Konfigurationsdateien (\*.config-Dateien) in einer Versionsverwaltung einchecken. Nur so kann man bei schwerwiegenden Fehlern auf einen funktionierenden Stand zurückgehen.**
- **Sicherungen immer mit einer Versionsverwaltung durchführen, damit verlustfrei beliebig viele Änderungsschritte komplett rückgängig gemacht werden können. Schwerwiegende Probleme, wie beschädigte CONFIG-Dateien, können z. T. erst relativ spät erkannt werden!**
- Eine – warum auch immer – zerstörte Konfigurationsdatei lässt sich wegen der Versionisierungsinformationen **NICHT** mehr aus der Versorgungsdatei herstellen.
- Muss eine solche zerstörte Datei über den Import neu erstellt werden, sind alle archivierten Daten, die auf diesem Konfigurationsbereich aufbauen, nicht mehr nutzbar. Wegen der neuen Vergabe der ID's für Objekte kann dies zu erheblichen Problemen bis zum vollständigen Datenverlust historischer Daten in abhängigen Bereichen bzw. abhängigen Konfigurationen führen.

### 3.5 Weitergeben von Konfigurationen

Unter der Weitergabe einer Konfiguration wird im Folgenden die Weitergabe aller<sup>20</sup> Konfigurationsdateien eines Konfigurationsverantwortlichen an einen anderen Konfigurationsverantwortlichen verstanden.

Die Weitergabe von Konfigurationen macht allerdings nur dann Sinn, wenn die weitergegebene Konfigurationsdateien zuvor „freigegeben“ wurden.

Die Freigabe bedeutet dabei eigentlich nur, dass der für diese Konfiguration Verantwortliche die Konfiguration zur Verwendung durch andere Konfigurationsverantwortlich entsprechend markiert. Die Konfigurationsapplikation des übernehmenden Konfigurationsverantwortlichen prüft bei der Übernahme, ob eine solche Markierung (Freigabe) vorliegt. Durch die unterschiedlichen Freigabestufen (zur Übernahme, zur Aktivierung, zur Aktivierung ohne lokale Aktivierung) kann der Verantwortliche damit die Nutzung seiner Konfigurationsbereiche durch andere Konfigurationsverantwortliche gezielt einschränken.

---

<sup>20</sup> In der Regel wird man immer einen „kompletten Konfigurationsverantwortlichen“ (also alle Konfigurationsdateien eines Konfigurationsverantwortlichen) weitergeben. Im Einzelfall kann man natürlich auch nur die geänderten Konfigurationsbereiche weitergeben. Da man aber normalerweise nicht weiß, welchen Stände (also welche Versionen) der einzelnen Konfigurationsdateien beim Empfänger bisher übernommen wurden, sollten immer alle Konfigurationsdateien des Konfigurationsverantwortlichen weitergeben werden.

### 3.5.1 Weitergabe mit „Freigabe zur Übernahme“

Mit „Freigabe zur Übernahme“ gekennzeichnete Konfigurationsbereiche können von einem anderen Konfigurationsverantwortlichen in dessen Konfiguration übernommen werden. Dieser kann für eigene Konfigurationsänderungen in seinen Bereichen auf Modell- bzw. Objektdefinitionen von Objekten aus dem übernommenen Bereich referenzieren bzw. diese Definitionen bei sich verwenden. Er kann seine Bereiche importieren und auf Korrektheit prüfen. Er kann allerdings die so geänderte Konfiguration nicht aktivieren, weil die übernommenen Bereich nur zur Übernahme, nicht aber zur Aktivierung freigegeben sind.

Diese Art der Weitergabe dient insbesondere dazu, bereits zu einem frühen Modellierungszeitpunkt anderen Konfigurationsverantwortlichen „relevante“ Definitionen der eigenen Konfigurationsbereiche zur Verfügung zu stellen, damit diese darauf aufbauend bereits mit der Anpassung ihrer Bereiche beginnen zu können.

### 3.5.2 Weitergabe mit „Freigabe zur Aktivierung“

Mit „Freigabe zur Aktivierung“ gekennzeichnete Konfigurationsbereiche können zusätzlich zu den lediglich zur Übernahme freigegeben Bereichen in der eigenen Konfiguration aktiviert werden, so dass die so erstellte Konfiguration auch im Anwendungsbetrieb verwandt werden kann. Ist die eigene Konfiguration von anderen Konfigurationsbereichen eines fremden Konfigurationsverantwortlichen abhängig, was die Regel ist, so benötigt man diese Konfigurationsbereich mit „Freigabe zur Aktivierung“, damit die eigenen Konfigurationsbereiche ebenfalls nicht nur aktiviert sondern auch zur Weitergabe an Dritte mit „Freigabe zur Aktivierung“ freigegeben werden können.

### 3.5.3 Weitergabe mit „Freigabe zur Aktivierung ohne lokale Aktivierung durch den KV“

Spezialvariante für gegenseitig abhängige Konfigurationen → Tutorial Teil C.

Prinzipiell geht es hier um die Möglichkeit zur Weitergabe einer zur Aktivierung freigegebenen Konfiguration an einen anderen Konfigurationsverantwortlichen, obwohl man diese Konfiguration bei sich selbst (auf Grund der gegenseitigen Abhängigkeit) noch gar nicht aktivieren kann.

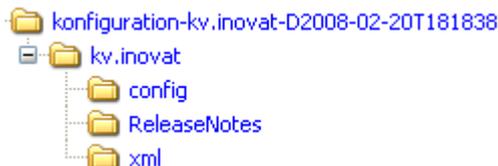
D. h., die eigene Konfiguration kann nicht aktiviert werden (und damit auch nicht freigegeben), weil vorher Konfigurationsbereiche eines anderen Konfigurationsverantwortlichen, von denen man selbst abhängig ist, aktiviert, zur Aktivierung freigegeben und übernommen werden müssten. Der andere Konfigurationsverantwortliche kann dies aber auch nicht durchführen, da er dazu erst den freigegebenen Bereich benötigt (also den, der sich nicht aktivieren lässt). Es liegt also ein klassischer „Deadlock“ vor. Um dieses Problem zu lösen, kann man seinen eigenen, nicht aktivierbaren Bereich mit der Option „Freigabe zur Aktivierung ohne lokale Aktivierung durch den KV“ zur Weitergabe freigeben. Mit diesem Bereich kann dann die andere Seite ihrerseits die notwendigen Bereiche freigeben, so dass man selbst wieder nach Übernahme dieser Bereiche seine eigenen Bereich aktivieren und freigeben kann.

#### Hinweis

- **Freigabe zur Übernahme** und **Freigabe zur Aktivierung** lassen sich nur durchführen, wenn die entsprechenden Konfigurationsbereiche zuvor aktiviert wurden
- **Freigabe zur Aktivierung ohne lokale Aktivierung durch den KV** lässt sich OHNE Aktivierung des Bereichs durchführen. Diese Option ist nur sinnvoll anwendbar, wenn zirkulare Abhängigkeiten zwischen Konfigurationsbereichen unterschiedlicher Konfigurationsverantwortlicher bestehen, die auf beiden Seiten eine Aktivierung verhindern.

### 3.5.4 Zusammenstellung eines KV zur Weitergabe an die ZS

Zur Weitergabe an Dritte (hier insbesondere die ZS) sollten die Konfigurationsbereiche eines Konfigurationsverantwortlichen immer vollständig mit folgenden Dateien und in folgender Struktur weitergegeben werden (Zusammenstellung hier beispielhaft für den Konfigurationsverantwortlichen „kv.inovat“).



- Weitergabe eines (gezippten) Ordners mit dem Namen des KV und Datumsangabe (hier konfiguration-kv.inovat-D2008-02-20T181838)
  - Dieser Ordner enthält dann den Ordner mit dem publizierten Konfigurationsverantwortlichen. Der Name des Ordners ist die PID des Konfigurationsverantwortlichen (hier kv.inovat).
    - Innerhalb dieses Ordners die Ordner
      - config
      - xml
      - ReleaseNotes
      - *gegebenenfalls weitere Ordner mit zusätzlicher Dokumentation etc.*

#### Ordner config

Dieser Ordner mit den entsprechenden Konfigurationsdateien (\*.config-Dateien) ist zwingend notwendig, da dieser die zu publizierenden (und zur Aktivierung freigegebenen) Konfigurationsdateien enthält.

Die folgende Abbildung zeigt beispielhaft den Inhalt für die abgebildete Distribution:

Name	Größe	Typ
kb.kv.inovat.config	21 KB	CONFIG-Datei
kb.tmEreignisKalenderGlobal.config	53 KB	CONFIG-Datei
kb.tmGanglinienGlobal.config	72 KB	CONFIG-Datei
kb.tmGeoReferenzierungGlobal.config	23 KB	CONFIG-Datei
kb.tmKEXTsGlobal.config	370 KB	CONFIG-Datei
kb.tmSystemKalenderGlobal.config	9 KB	CONFIG-Datei
kb.tmTmcGlobal.config	68 KB	CONFIG-Datei
kb.tmUmfeldDatenGlobal.config	1.056 KB	CONFIG-Datei
kb.tmVerkehrGlobal.config	480 KB	CONFIG-Datei
verwaltungsdaten.xml	10 KB	XML Document

Wenngleich die Verwaltungsdatei (verwaltungsdaten.xml) von der übernehmenden Seite NICHT übernommen wird, sollte diese mitgeliefert werden, da diese einen direkten Überblick über die aktivierte Versionsnummer der einzelnen Bereiche liefert und zudem erkennbar ist, welche weiteren Konfigurationsbereiche anderer Konfigurationsverantwortlicher bei der Aktivierung vorhanden waren.

#### Ordner xml

Dieser Ordner mit den entsprechenden Versorgungsdateien (\*.xml-Dateien) ist für die Übernahme nicht zwingend notwendig, sollten aber dennoch immer mitgeliefert werden, da

- über die XML-Versorgungsdateien sich die Inhalte der Konfiguration analysieren lassen
- sich die Abhängigkeiten zwischen den Konfigurationsdateien aus den beim Export automatisch erzeugten Einträgen der XML-Datei ermitteln lassen
- der Empfänger sich nicht extra eine Umgebung nur für den Export zusammenstellen muss (i. d. R. wird man nämlich die XML-Dateien betrachten wollen)
- die XML-Dateien für die Erzeugung einer lesbaren HTML-Darstellung benötigt werden
- die Bereitstellung für den Ersteller des Distributionspakets keine zusätzliche Arbeit erfordert, da die aktuellen XML-Dateien bei im ohnehin vorliegen

Die folgende Abbildung zeigt beispielhaft den Inhalt für die abgebildete Distribution:

Name ▲	Größe	Typ
kb.kv.inovat.xml	2 KB	XML Document
kb.tmEreignisKalenderGlobal.xml	25 KB	XML Document
kb.tmGanglinienGlobal.xml	24 KB	XML Document
kb.tmGeoReferenzierungGlobal.xml	20 KB	XML Document
kb.tmKEXtIsGlobal.xml	393 KB	XML Document
kb.tmSystemKalenderGlobal.xml	8 KB	XML Document
kb.tmTmcGlobal.xml	45 KB	XML Document
kb.tmUmfeldDatenGlobal.xml	366 KB	XML Document
kb.tmVerkehrGlobal.xml	314 KB	XML Document

Wenngleich die Verwaltungsdatei (verwaltungsdaten.xml) von der übernehmenden Seite NICHT übernommen wird, sollte diese mitgeliefert werden, da diese einen direkten Überblick über die aktivierten Versionsnummer der einzelnen Bereich liefert und zudem erkennbar ist, welche weiteren Konfigurationsbereiche anderer Konfigurationsverantwortlicher bei der Aktivierung vorhanden waren.

#### Ordner ReleaseNotes

Dieser Ordner enthält die Dokumentation für die vorliegende Distribution und ist für die Übernahme zwar nicht zwingend notwendig, sollten aber dennoch immer mitgeliefert werden. Alternativ kann die Änderungsdokumentation auch direkt in den Versorgungsdateien bei den entsprechenden XML-Tags vorgenommen werden.

### 3.6 Übernehmen von Konfigurationen

Unter der Übernahme einer Konfiguration wird im Folgenden die Übernahme aller<sup>21</sup> Konfigurationsdateien eines Konfigurationsverantwortlichen von einem anderen Konfigurationsverantwortlichen verstanden.

Voraussetzung für die Übernahme von Konfigurationen ist, das die zu übernehmenden Konfigurationsbereiche freigeben<sup>22</sup> sind.

<sup>21</sup> In der Regel wird man immer einen „kompletten Konfigurationsverantwortlichen“ (also alle Konfigurationsdateien eines Konfigurationsverantwortlichen) übernehmen. Im Einzelfall kann man natürlich auch einzelne geänderte Konfigurationsbereiche übernehmen. Bei der Übernahme einzelner Konfigurationsbereiche ist besonderes Augenmerk auf die Abhängigkeiten zu anderen Konfigurationsbereichen und der jeweils notwendigen Version des entsprechenden Bereichs zu richten.

Bei der Übernahme sind zwei Fälle zu unterscheiden:

- **Aktualisieren:** die zu übernehmenden Konfigurationsbereiche sind in der übernehmenden Konfiguration bereits vorhanden
- **Hinzufügen:** die zu übernehmenden Konfigurationsbereiche sind in der übernehmenden Konfiguration bisher noch nicht vorhanden

Beim **Aktualisieren** sind folgende Schritte durchzuführen:

- Ersetzen der bestehenden Konfigurationsdateien durch die zu übernehmenden Konfigurationsdateien (→ Dateien kopieren)
- Befehle „Aktivieren“ auf der Konfiguration durchführen.

Beim **Hinzufügen** sind folgende Schritte durchzuführen:

- Zu übernehmende Konfigurationsdateien in die eigene Konfiguration kopieren (→ Dateien kopieren)
- Verwaltungsdatei `verwaltungsdaten.xml` anpassen
- Befehle „Aktivieren“ auf der Konfiguration durchführen.

### 3.6.1 Anpassung der Verwaltungsdatei

Für jeden Konfigurationsbereich in der eigenen Konfiguration muss ein Eintrag in der Verwaltungsdatei `verwaltungsdaten.xml` existieren.

Beim Start der Konfiguration liest diese die Verwaltungsdatei ein und lädt die dort eingetragenen Konfigurationsbereiche in der dort eingetragenen Version.

Werden eine oder mehrere neue Bereiche in die Verwaltungsdatei eingetragen, so muss nur der Eintrag für die PID des Konfigurationsbereichs und der Ort der Datei relativ zum Ort der Verwaltungsdatei eingetragen werden.

#### Beispiel

Einer Konfiguration soll der Konfigurationsbereich mit der PID `kb.tmVerkehrGlobal` (Konfigurationsverantwortlicher `kv.inovat`)<sup>23</sup> hinzugefügt werden. Folgender Eintrag muss in der Verwaltungsdatei hinzugefügt werden:

```
<konfigurationsbereich pid="kb.tmVerkehrGlobal" verzeichnis="../../kv.inovat/config/">
</konfigurationsbereich>
```

<sup>22</sup> D. h., dass die Konfigurationsbereiche von dem Konfigurationsverantwortlichen, der die Bereiche zur Verfügung stellt, mit einer der drei Funktionen „Freigabe zur Übernahme“, „Freigabe zur Aktivierung“ oder „Freigabe zur Aktivierung ohne lokale Aktivierung durch den KV“ markiert wurden.

<sup>23</sup> Bei der Übernahme einzelner Konfigurationsbereiche sind die Abhängigkeiten zu anderen Konfigurationsbereichen zu berücksichtigen. Abhängige Konfigurationsbereiche müssen ebenfalls mit übernommen werden. Der im Beispiel verwandte Bereich benötigt neben dem `kb.kv.inovat` noch mehrere andere Bereiche des `kv.inovat` und des `kv.kappich`. Das Beispiel soll lediglich den notwendigen Eintrag in der Verwaltungsdatei für einen zusätzlichen Konfigurationsbereich verdeutlichen.

Der Eintrag der aktuellen freigegeben Version erfolgt bei der Aktivierung durch die Konfiguration automatisch. Ebenso wird nach einem weiteren Start der Konfiguration nach der Aktivierung der Eintrag für den Zeitpunkt automatisch hinzugefügt.

### 3.6.2 Aktivierung einer übernommenen Konfiguration

Zur Aktivierung der übernommenen Bereich muss lediglich die Konfigurationsapplikation mit dem Befehl „Aktivieren“ für die aktuelle Konfiguration aufgerufen werden.

Durch die Aktivierung wird in der Verwaltungsdatei der Eintrag für die aktuelle Version des übernommenen Bereichs eingetragen. Für das obige Beispiel sähe der automatisch erzeugte Eintrag in der Verwaltungsdatei jetzt wie folgt aus:

```
<konfigurationsbereich pid="kb.tmVerkehrGlobal" verzeichnis="../../kv.inovat/config/">
  <version nr="25"/>
</konfigurationsbereich>
```

Erst beim nächsten Start der Konfigurationsapplikation wird diese Version tatsächlich lokal „gültig“. Die Konfigurationsapplikation trägt dann automatisch den Aktivierungszeitpunkt in der Verwaltungsdatei ein, so dass der entsprechende Eintrag dann wie folgt aussieht:

```
<konfigurationsbereich pid="kb.tmVerkehrGlobal" verzeichnis="../../kv.inovat/config/">
  <version nr="25" zeitpunkt="20.02.2008 17:42:21,296"/>
</konfigurationsbereich>
```

Wurde der Bereich nicht neu hinzugefügt sondern nur aktualisiert (d. h., es existierten bereits vorher aktivierte Versionen des Bereichs, so sieht die Verwaltungsdatei nach der Aktivierung wie folgt aus:

```
<konfigurationsbereich pid="kb.tmVerkehrGlobal" verzeichnis="../../kv.inovat/config/">
  <version nr="1" zeitpunkt="03.03.2007 17:58:38,000"/>
  <version nr="2" zeitpunkt="05.03.2007 20:54:15,698"/>
  <version nr="7" zeitpunkt="20.03.2007 08:15:50,141"/>
  <version nr="8" zeitpunkt="20.03.2007 09:45:45,509"/>
  <version nr="10" zeitpunkt="20.03.2007 22:41:27,033"/>
  <version nr="11" zeitpunkt="02.05.2007 12:13:32,062"/>
  <version nr="12" zeitpunkt="04.06.2007 15:55:41,140"/>
  <version nr="13" zeitpunkt="10.06.2007 13:07:30,640"/>
  <version nr="14" zeitpunkt="14.06.2007 09:54:44,703"/>
  <version nr="15" zeitpunkt="09.07.2007 08:07:09,515"/>
  <version nr="16" zeitpunkt="20.07.2007 06:40:50,843"/>
  <version nr="17" zeitpunkt="28.09.2007 19:30:59,046"/>
  <version nr="18" zeitpunkt="03.10.2007 15:46:14,015"/>
  <version nr="19" zeitpunkt="23.11.2007 13:25:37,718"/>
  <version nr="20" zeitpunkt="02.12.2007 14:11:33,640"/>
  <version nr="25" zeitpunkt="20.02.2008 17:42:21,296"/>
</konfigurationsbereich>
```

In diesem Beispiel sieht man zudem, dass nicht alle Versionen eines Konfigurationsbereichs in dieser lokalen Konfiguration übernommen wurden (was auch nicht notwendig ist, eventuell wurden auch gar nicht alle Versionen vom Verantwortlichen freigegeben bzw. weitergegeben).

## 4 Teil C – Workflow bei der Konfiguration

### 4.1 Organisation einer Konfigurationsumgebung

### 4.2 Komplexe Workflow

### 4.3 Sonderfälle beim Workflow

## 5 Teil D – Richtlinien, Fehlersituationen und Lösungen

### 5.1 Richtlinien

### 5.2 Fehlersituationen

## 6 Anhang

### 6.1 Übersichtsdarstellung „BSVRZ Tutorial Konfiguration - Standard-workflow“.