



NON-STOP-GOVERNMENT



IT-INVESTITIONS-
PROGRAMM

Wir gestalten Zukunft.

P23R: Spezifikationen zur Infrastruktur

Ein Ergebnisdokument des Projekts P23R | Prozess-Daten-Beschleuniger
im Auftrag des Bundesministeriums des Innern

Simon Dutkowski
Ekaterina Klochkova
Andreas Söllner

Das P23R-Projekt wurde im Rahmen des IT-Investitionsprogramms der Bundesregierung durchgeführt (Fördernummer D4-06-1).

Generalunternehmer



Projektbeteiligte



Projekt

P23R | Prozess-Daten-Beschleuniger

P23R: Spezifikationen zur Infrastruktur

Ergebnisdokument

Januar 2013

Autoren

Simon Dutkowski, Fraunhofer FOKUS

Ekaterina Klochkova, :::tsm total-sourcing-management

Andreas Söllner, :::tsm total-sourcing-management

Zusammenfassung

Die Umsetzung der P23R-Musterimplementierung auf Basis der ausgewählten Basistechnologie ist der zentrale Block der Entwicklungsarbeiten zum Nachweis der Anwendbarkeit. Sie gliedert sich in die folgenden Komponentengruppen, die ihrerseits wiederum jeweils in eine Reihe von Komponenten untergliedert sind:

- Der P23R selbst mit den Komponenten zur Abbildung der Generation-Pipeline vom Nachrichteneingang bis zur Benachrichtigungsübermittlung und den dazu nötigen Support- und Sicherheitskomponenten
- Die P23R-Leitstelle mit den Komponenten zur Bereitstellung von Datenmodell- und Benachrichtigungsregelpaketen und den zugehörigen Verzeichnissen des Zuständigkeitsfinders und den Trusted Service Lists sowie der Administrationsoberfläche
- Der P23R-Client mit den Komponenten zur Darstellung der Benutzeroberfläche sowie den erforderlichen Komponenten der Sicherheitsarchitektur zur Verwaltung von Benutzern und Policies

Darüber hinaus wird die P23R-Musterimplementierung für den Pilotversuch um eine Reihe von Quelldaten- und Kommunikationskonnektoren ergänzt, sprich diese in die P23R-Musterimplementierung integriert. Folgende Konnektoren werden hierzu realisiert:

- Ein Quelldatenkonnektor zur Übertragung von Daten aus einem SAP-HR-System (in diesem Fall für den Pilotversuch bei der BASF SE)
- Ein Quelldatenkonnektor zur Übertragung von Daten aus dem DATEV-HR-System (in diesem Fall für den Pilotversuch bei der DATEV e.G.)
- Ein Kommunikationskonnektor zur Anbindung des statistischen Meldeschreibens eSTATISTIK.core (in diesem Fall für die Abwicklung der vierteljährlichen Verdiensterhebung)
- Ein Kommunikationskonnektor zur Anbindung einer virtuellen Kopfstelle für das DEÜV-Verfahren (in diesem Fall für die Abwicklung der DEÜV-Jahresmeldung)
- Ein Kommunikationskonnektor zur Übermittlung von PDF-Formularen via E-Mail (in diesem Fall für den jährlichen Lohnnachweis an die Berufsgenossenschaften)

Gemeinsam mit den im Pflichtenheft beschriebenen Datenmodell- und Benachrichtigungsregelpaketen bildet diese Lösung das Testbett für die Durchführung des Pilotversuchs in der Domäne Arbeitgebermeldungen. Die Musterimplementierung kommt darüber hinaus – mit anderen Konnektoren integriert – auch im Pilotversuch der Domäne Umwelt/Emissionen zum Einsatz, was die Übertragbarkeit auch praktisch verdeutlicht.

Executive Summary

The realization of the P23R pattern implementation based on a selected basic technology is a central part of the development process needed to prove the P23R-applicability and is divided into the following groups, which are in turn subdivided into a number of components:

- The P23R itself including the components presenting the generation pipeline - from the message reception to the notification transmission as well as the required support and security components.
- The P23R-control point including the components providing the data model rule packages and the notification rule packages as well as the related responsibility finder directories, the Trusted Service Lists and the administration interface components.
- The P23R client including the components displaying the user interface and the required components of the security architecture needed for a users and policies administration.

Aside from that, within the scope of the pilot experiment a number of the source data connectors and communication connectors will be added to the P23R pattern implementation, in other words the following connectors will be integrated into it:

- A data source connector for the data transferring from a SAP HR system (in this case for the pilot experiment for BASF SE)
- A data source connector for the data transferring from DATEV HR system (in this case for the pilot experiment for DATEV)
- A communication connector used for the accessibility to the statistical reporting process eSTATISTIK.core (in this case for the processing a quarterly earnings inquiry)
- A communication connector used for the accessibility to a virtual head office of the DEÜV (Regulation on the Data Registration and Data Transmission) process (in this case for the processing the DEÜV (Regulation on the Data Registration and Data Transmission) process annual report)
- A communication connector for the PDF schedules transmission via e-mail (in this case the annual wage certificate to the professional associations)

Together with the described data model rule packages and the notification rule packages this document provides a basis for the accomplishment of a pilot experiment in the employer reporting obligations domains. The P23R pattern implementation integrated with other connectors will also be employed for the pilot experiment in the environmental domains, demonstrating the P23R applicability in practice

Inhaltsverzeichnis

1	Einleitung.....	1
1.1	Zweck des Dokuments.....	1
1.2	Leserkreis.....	1
1.3	Kontext, Inhalte und Strukturierung	1
2	Gemeinsam genutzte Bibliotheken	3
2.1	Model and Rule Management.....	3
2.1.1	Struktur und Arbeitsweise.....	3
2.1.2	Datenbestand und Persistenz.....	3
2.1.3	Fehlerbehandlung.....	4
2.1.4	Protokollierung.....	4
2.1.5	Schnittstellen.....	4
2.1.6	Artefakte.....	4
2.2	Common Security	4
2.2.1	Struktur und Arbeitsweise.....	4
2.2.2	Schnittstellen.....	4
2.2.3	Fehlerbehandlung.....	5
2.2.4	Protokollierung.....	5
2.2.5	Konfiguration.....	5
2.2.6	Artefakte.....	5
2.3	Common TBRS	5
2.3.1	Struktur und Arbeitsweise.....	6
2.3.2	Schnittstellen.....	6
2.3.3	Fehlerbehandlung.....	7
2.3.4	Protokollierung.....	7
2.3.1	Konfiguration.....	7
2.3.2	Artefakte.....	7
2.4	Common P23R	7
2.4.1	Struktur und Arbeitsweise.....	7
2.4.2	Schnittstellen.....	8
2.4.3	Fehlerbehandlung.....	8
2.4.4	Protokollierung.....	8
2.4.5	Konfiguration.....	8
2.4.6	Artefakte.....	8
3	Spezifikation des P23R.....	9
3.1	Common ESB.....	10

3.1.1	Struktur und Arbeitsweise.....	10
3.1.2	Schnittstellen	10
3.1.3	Fehlerbehandlung.....	11
3.1.4	Protokollierung	11
3.1.5	Konfiguration	11
3.1.6	Artefakte	11
3.2	DataSelCompiler	11
3.2.1	Struktur und Arbeitsweise.....	12
3.2.2	Schnittstellen	12
3.2.3	Fehlerbehandlung.....	12
3.2.4	Protokollierung	12
3.2.5	Konfiguration	12
3.2.6	Artefakte	12
3.3	MessageReceipt.....	13
3.3.1	Struktur und Arbeitsweise.....	13
3.3.2	Fehlerbehandlung.....	13
3.3.3	Protokollierung	13
3.3.4	Schnittstellen	13
3.3.5	Artefakte	14
3.4	NotificationGeneration.....	15
3.4.1	Struktur und Arbeitsweise.....	15
3.4.2	Fehlerbehandlung.....	15
3.4.3	Protokollierung	15
3.4.4	Schnittstellen	16
3.4.5	Artefakte	16
3.5	NotificationDispatch	16
3.5.1	Struktur und Arbeitsweise.....	16
3.5.2	Datenbestand und Persistenz.....	17
3.5.3	Fehlerbehandlung.....	17
3.5.4	Protokollierung	17
3.5.5	Schnittstellen	18
3.5.6	Artefakte	18
3.6	NotificationTransport	19
3.6.1	Struktur und Arbeitsweise.....	19
3.6.2	Fehlerbehandlung.....	19
3.6.3	Protokollierung	19
3.6.4	Konfiguration	20
3.6.5	Schnittstellen	20
3.6.6	Artefakte	21

3.7	DataPool	21
	3.7.1 Struktur und Arbeitsweise	21
	3.7.2 Fehlerbehandlung.....	21
	3.7.3 Protokollierung	22
	3.7.4 Datenbestand, Persistenz.....	22
	3.7.5 Konfiguration	22
	3.7.6 Schnittstellen	23
	3.7.7 Artefakte	24
3.8	ProtocolPool	24
	3.8.1 Struktur und Arbeitsweise	25
	3.8.2 Fehlerbehandlung.....	25
	3.8.3 Datenbestand, Persistenz.....	25
	3.8.4 Schnittstellen	25
	3.8.5 Artefakte	26
3.9	ModelAndRuleManagement	26
	3.9.1 Struktur und Arbeitsweise	26
	3.9.2 Datenbestand und Persistenz.....	28
	3.9.3 Fehlerbehandlung.....	28
	3.9.4 Protokollierung	28
	3.9.5 Schnittstellen	28
	3.9.6 Artefakte	33
3.10	Scheduling.....	34
	3.10.1 Struktur und Arbeitsweise	34
	3.10.2 Fehlerbehandlung.....	34
	3.10.3 Protokollierung	34
	3.10.4 Konfiguration	35
	3.10.5 Schnittstellen	35
	3.10.6 Artefakte	37
3.11	Bootstrapping	37
	3.11.1 Struktur und Arbeitsweise	38
	3.11.2 Konfiguration	38
	3.11.3 Fehlerbehandlung.....	38
	3.11.4 Protokollierung	38
	3.11.5 Schnittstellen	39
	3.11.6 Artefakte	39
3.12	IdentityProvider	39
	3.12.1 Struktur und Arbeitsweise	39
	3.12.2 Fehlerbehandlung.....	39

3.12.3	Protokollierung	39
3.12.4	Konfiguration	40
3.12.5	Schnittstellen	40
3.12.6	Artefakte	40
3.13	Policy Enforcement Point (PEP)	40
3.13.1	Struktur und Arbeitsweise	40
3.13.2	Schnittstellen	41
3.13.3	Fehlerbehandlung.....	41
3.13.4	Protokollierung	41
3.13.5	Konfiguration	41
3.13.6	Artefakte	41
3.14	Extensions	42
3.14.1	Struktur und Arbeitsweise	42
3.14.2	Schnittstellen	42
3.14.3	Fehlerbehandlung.....	42
3.14.4	Protokollierung	43
3.14.5	Konfiguration	43
3.14.6	Artefakte	43
4	Spezifikation der P23R-Leitstelle	45
4.1	Portal und Persistence	45
4.1.1	Struktur und Arbeitsweise	45
4.1.2	Fehlerbehandlung.....	46
4.1.3	Konfiguration	46
4.1.4	Schnittstellen	46
4.1.5	Artefakte	47
4.2	CompetenceFinder	47
4.2.1	Struktur und Arbeitsweise	48
4.2.2	Schnittstellen	48
4.2.3	Datenbestand, Persistenz	49
4.2.4	Fehlerbehandlung.....	49
4.2.5	Protokollierung	49
4.2.6	Konfiguration	49
4.2.7	Artefakte	50
5	Spezifikation des P23R-Client	51
5.1	Common User-Profile	51
5.1.1	Struktur und Arbeitsweise	51
5.1.2	Schnittstellen	51
5.1.3	Fehlerbehandlung.....	52

5.1.4	Konfiguration	52
5.1.5	Artefakte	52
5.2	Portal.....	52
5.2.1	Struktur und Arbeitsweise	53
5.2.2	Schnittstellen	54
5.2.3	Datenbestand, Persistenz.....	54
5.2.4	Fehlerbehandlung.....	55
5.2.5	Protokollierung	55
5.2.6	Konfiguration	55
5.2.7	Artefakte	56
5.3	Guarantor Token Service	56
5.3.1	Arbeitsweise	56
5.3.2	Schnittstellen	56
5.3.3	Sicherheitskonfiguration	57
5.3.4	Fehlerbehandlung.....	57
5.3.5	Protokollierung	57
5.3.6	Konfiguration	57
5.3.7	Artefakte	57
5.4	Policy Administration Point (PAP)	57
5.4.1	Struktur und Arbeitsweise	57
5.4.2	Schnittstellen	58
5.4.3	Fehlerbehandlung.....	62
5.4.4	Protokollierung	63
5.4.5	Konfiguration	63
5.4.6	Artefakte	63
6	Spezifikation der Quelldatenkonnektoren	65
6.1	SourceConnector	65
6.1.1	Struktur und Arbeitsweise	65
6.1.2	Schnittstellen	65
6.1.3	Fehlerbehandlung.....	65
6.1.4	Protokollierung	66
6.1.5	Konfiguration	66
6.1.6	Artefakte	66
6.2	SourceConnector.BASFsap	66
6.2.1	Struktur und Arbeitsweise	66
6.2.2	Datenhaltung	67
6.2.3	Schnittstellen	67
6.2.4	Fehlerbehandlung.....	67

6.2.5	Protokollierung	67
6.2.6	Konfiguration	67
6.2.7	Artefakte	67
6.3	SourceConnector.DATEVraw	68
6.3.1	Struktur und Arbeitsweise	68
6.3.2	Datenhaltung	68
6.3.3	Schnittstellen	68
6.3.4	Fehlerbehandlung.....	68
6.3.5	Protokollierung	68
6.3.6	Konfiguration	69
6.3.7	Artefakte	69
7	Spezifikation der Kommunikationskonnektoren.....	71
7.1	CommunicationConnector	71
7.1.1	Arbeitsweise	71
7.1.2	Schnittstellen	71
7.1.3	Fehlerbehandlung.....	71
7.1.4	Protokollierung	71
7.1.5	Konfiguration	72
7.1.6	Artefakte	72
7.2	CommunicationConnector.eSTATISTIKcore	72
7.2.1	Arbeitsweise	72
7.2.2	Schnittstellen	72
7.2.3	Fehlerbehandlung.....	73
7.2.4	Protokollierung	73
7.2.5	Konfiguration	73
7.2.6	Artefakte	73
7.3	CommunicationConnector.vDEUEV	73
7.3.1	Arbeitsweise	74
7.3.2	Schnittstellen	74
7.3.3	Fehlerbehandlung.....	74
7.3.4	Protokollierung	74
7.3.5	Konfiguration	74
7.3.6	Artefakte	74
7.4	CommunicationConnector.BGviaMail	75
7.4.1	Arbeitsweise	75
7.4.2	Schnittstellen	75
7.4.3	Fehlerbehandlung.....	75
7.4.4	Protokollierung	75

	7.4.5	Konfiguration	76
	7.4.6	Artefakte	76
8		Anhang I: Entwicklungsvorgaben und -richtlinien	77
	8.1	Programmierrichtlinien	77
	8.1.1	Dokumentation der Quellen.....	78
	8.1.2	Versionsschema.....	78
	8.1.3	Namensräume	79
	8.1.4	Namensregeln.....	80
	8.1.5	Logging.....	80
	8.2	Entwicklungstools	81
	8.2.1	Eclipse IDE.....	81
	8.2.2	Versionskontrollsystem Subversion	81
	8.2.3	Maven	81
	8.2.4	Ticket System und Wiki (Trac)	82
	8.2.5	Continuous Integration (Hudson).....	82
	8.2.6	Sonar.....	82
	8.3	Arbeiten mit der Laborumgebung.....	83
9		Anhang II: Bestückung der Leitstelle (TEST und AGM).....	85
	9.1	Bestückung zu Testzwecken	85
	9.1.1	Datenmodellpaket „TEST“	85
	9.1.2	Benachrichtigungsregelpaket „TEST“	86
	9.1.3	Zuständigkeitsinformationen „TEST“	89
	9.2	Bestückung für den Pilotversuch AGM.....	91
	9.2.1	Datenmodellpaket „AGM“	91
	9.2.2	Benachrichtigungsregelpaket „AGM“	92
	9.2.3	Zuständigkeitsinformationen „AGM“	96
10		Anhang III: Testdatensatz und Prüfergebnisse (AGM)	105
	10.1	Testdatensatz der Teildatenmodelle „Unternehmen“ und „Mitarbeiter“.....	105
	10.2	Prüfergebnisse zur vierteljährliche (statistische) Verdiensterhebung.....	113
	10.3	Prüfergebnisse zur DEÜV-Jahresmeldung zur Sozialversicherung.....	116
	10.4	Prüfergebnisse zum Lohnnachweis an die Berufsgenossenschaften.....	126
11		Glossar	129
12		Abkürzungsverzeichnis	151
13		Referenzen.....	153

P23R

Verzeichnis der Abbildungen

Verzeichnis der Abbildungen

Abbildung 1: ESB-Dienste der Musterimplementierung P23R9

Verzeichnis der Tabellen

Tabelle 1:	receiveMessage Message	14
Tabelle 2:	generateNotification Message	16
Tabelle 3:	dispatchNotification Message	18
Tabelle 4:	sendNotification Message	21
Tabelle 5:	storeConfigdata Message	23
Tabelle 6:	remmoveConfigdata Message.....	23
Tabelle 7:	selectData Message	24
Tabelle 8:	writeProtocolEntry Message	25
Tabelle 9:	updateNotificationRulePool Message	29
Tabelle 10:	getNotificationRuleInformationByMessageType Message	29
Tabelle 11:	getNotificationRuleDataSelection Message	30
Tabelle 12:	GetNotificationCoreTransformation Message	30
Tabelle 13:	getNotificationRuleRepresentationTransformation Message	31
Tabelle 14:	GetNotificationRuleTransformationList Message	31
Tabelle 15:	IsNotificationRuleUpToDate Message.....	32
Tabelle 16:	GetNotificationRulePackageStates Message.....	32
Tabelle 17:	SetNotificationRulePackageStates Message	33
Tabelle 18:	scheduleJob Message	35
Tabelle 19:	modifyJob Message	36
Tabelle 20:	cancelJob Message	37
Tabelle 21:	Protokollieren von Fehlermeldungen.....	66
Tabelle 22:	P23R Namensräume	79
Tabelle 23:	Aufbau des Datenmodellpakets „TEST“	85
Tabelle 24:	Aufbau des Benachrichtigungsregelpakets „TEST V 1.0“	86
Tabelle 25:	Aufbau des Benachrichtigungsregelpakets „TEST V 2.0“	88

P23R

Verzeichnis der Tabellen

Tabelle 26:	Testregeln – Leistungskatalog	90
Tabelle 27:	Testregeln – Kriterienkatalog	90
Tabelle 28:	Testregeln – Verzeichnis der Benachrichtigungsempfänger	90
Tabelle 29:	Testregeln – Verzeichnis des technischen Supports	90
Tabelle 30:	Testregeln – Elektronisches Dienstverzeichnis.....	90
Tabelle 31:	Testregeln – Verzeichnis des fachlichen Supports	91
Tabelle 32:	Testregeln – MappingReceiver	91
Tabelle 33:	Testregeln – MappingEservice.....	91
Tabelle 34:	Aufbau des Datenmodellpakets „AGM“	92
Tabelle 35:	Aufbau des Benachrichtigungsregelpakets „AGM“	93
Tabelle 36:	AGM-Pilotdaten – Leistungskatalog	96
Tabelle 37:	AGM-Pilotdaten – Kriterienkatalog	96
Tabelle 38:	AGM-Pilotdaten – Verzeichnis der Benachrichtigungsempfänger	96
Tabelle 39:	AGM-Pilotdaten – Verzeichnis des technischen Supports	98
Tabelle 40:	AGM-Pilotdaten – Elektronisches Dienstverzeichnis.....	98
Tabelle 41:	AGM-Pilotdaten – Verzeichnis des fachlichen Supports	99
Tabelle 42:	AGM-Pilotdaten – MappingReceiver	100
Tabelle 43:	AGM-Pilotdaten – MappingEservice.....	102

1 EINLEITUNG

1.1 ZWECK DES DOKUMENTS

Das vorliegende Dokument dient der Spezifikation der in der Implementierung umgesetzten P23R-Lösung. Diese stellt, basierend auf der P23R-Rahmenarchitektur [1], eine exemplarische Umsetzung des P23R-Prinzips im Kontext der ausgewählten Pilotdomäne Arbeitgebermeldepflichten (AGM) dar. Das Dokument schließt damit nahtlos an die Inhalte des Pflichtenhefts zur Infrastruktur [2] an und spezifiziert dessen Vorgaben.

1.2 LESERKREIS

Dieses Dokument richtet sich an alle Leser, die sich darüber informieren möchten, wie die im Pflichtenheft [2] beschriebene Abbildung der P23R-Rahmenarchitektur für die P23R-Lösung im Kontext der Pilotierung in der Domäne AGM umgesetzt wird.

1.3 KONTEXT, INHALTE UND STRUKTURIERUNG

Die Kapitel 2 bis 7 beinhalten die Spezifikation der jeweils durch den Titel kenntlich gemachten Software-Komponente der P23R-Musterimplementierung bzw. der im Zuge des Piloten AGM entwickelten Quelldaten- und Kommunikationskonnektoren wieder. Um eine leichte Nachvollziehbarkeit der P23R-Musterimplementierung zu ermöglichen, sind die Namen analog zu den jeweiligen Bezeichnungen in der P23R-Rahmenarchitektur [1] gehalten.

Daran anschließend beschreibt Kapitel 8 die geltenden Vorgaben für die Entwicklungsarbeiten, welche während der Entwicklung der P23R-Musterimplementierung eingehalten und im Zuge der Qualitätssicherung geprüft wurden.

Abschließend werden in Kapitel 9 die Bestückung der Leitstelle in Form von Datenmodell- und Benachrichtigungsregelpaketen sowie Zuständigkeitsinformationen sowie in Kapitel 10 die zu verwendenden Testdatensätze zur Erprobung der P23R-Musterimplementierung beschrieben.

P23R

P23R: Spezifikationen zur Infrastruktur

2 GEMEINSAM GENUTZTE BIBLIOTHEKEN

Für den P23R, die P23R-Leitstelle und den P23R-Client sind eine Reihe gemeinsam genutzter Bibliotheken im Einsatz, die im Folgenden beschrieben werden. Sie dienen der Bereitstellung zentraler bzw. übergreifender Funktionen und Klassen zur Abbildung wiederkehrender Aufgaben innerhalb der P23R-Musterimplementierung.

2.1 MODEL AND RULE MANAGEMENT

Die Bibliothek *ModelAndRuleManagement* (im Folgenden *MARM* abgekürzt) implementiert alle Funktionalitäten, die mit dem Zugriff und der Manipulation von Datenmodell- und Benachrichtigungsregelpaket-Informationen zusammenhängen.

Sie wird vorzugsweise von der P23R-Komponente *ModelAndRuleManagement* (siehe Abschnitt 2.1) dem P23R-Client (siehe Kapitel 5) sowie vom Leitstellen-Portal (siehe Kapitel 4) benötigt.

Die Bibliothek *ModelAndRuleManagement* verwendet folgende andere Bibliotheken:

- Common TBRS (siehe Abschnitt 2.3) - Zugriff auf Manifest-Informationen der einzelnen Artefakte

2.1.1 STRUKTUR UND ARBEITSWEISE

Folgende Paketstrukturen mit folgenden Inhalten befinden sich innerhalb der Bibliothek:

- archive
Enthält Zugriffsklassen für die verschiedenen Pakete (Paketliste, Benachrichtigungsregelpaket, Datenmodellpaket) sowie deren Strukturen und Inhalte. Außerdem können aus den Paketstrukturen äquivalente Persistenz-Strukturen erzeugt werden (JPA Entities).
- controlcentre
Enthält Hilfsklassen für den Zugriff auf die REST-Schnittstellen des Datenmodell- und Benachrichtigungsregeldepots der Leitstelle.
- helper
Enthält Hilfsklassen für die Ermittlung des Namespaces verschiedener Artefakt-Typen (Schema, Transformationsscript, Selektionsscript, XML-Instanz).
- persistence
Enthält die Definition der Datenzugriffsschicht (JPA Entities) sowie eine zentrale Datenzugriffs- und Transaktionssteuerungsklasse (PersistenceManager).

2.1.2 DATENBESTAND UND PERSISTENZ

Die Bibliothek *ModelAndRuleManagement* realisiert eine Datenzugriffsschicht auf Basis des Java EE Standards JSR 220 „Java Persistence API“ (JPA) [3]. Somit kann je nach Anforderung an die Zielumgebung ein adäquates Datenbank-Produkt zum Einsatz kommen, welches über diesen Standard angebunden werden kann.

Über die zentrale Klasse *PersistenceManager* erfolgt der Zugriff auf den Datenbestand bzw. dessen Manipulation. Auch die Transaktionssteuerung wird über diese Klasse geregelt, falls sie nicht von außen über einen Container erfolgt.

P23R

P23R: Spezifikationen zur Infrastruktur

Die Konfiguration (DataSource etc.) einer konkreten Datenbank erfolgt nicht in der Bibliothek, sondern in der jeweiligen Komponente, welche die Bibliothek verwendet.

2.1.3 FEHLERBEHANDLUNG

Die Bibliothek *ModelAndRuleManagement* gibt Fehler in der Regel an die aufrufende Komponente weiter und loggt diese über das eingesetzte Logging-Framework.

2.1.4 PROTOKOLLIERUNG

Eine separate Protokollierung außerhalb des Fehler-Loggings erfolgt nicht.

2.1.5 SCHNITTSTELLEN

Die Bibliothek *ModelAndRuleManagement* konsumiert die REST-Schnittstellen des Datenmodell- und Benachrichtigungsregeldepots der Leitstelle. Die Konfiguration der Endpunkte erfolgt durch die aufrufende Komponente.

2.1.6 ARTEFAKTE

Für die Bibliothek *ModelAndRuleManagement* entstehen folgende Artefakte:

- `common.modelandrulemanagement-<version>.jar`
Die Common ModelAndRuleManagement-Bibliothek.

2.2 COMMON SECURITY

Grundlegende Sicherheitsfunktionalitäten werden in einer Bibliothek gekapselt. Mit der Komponente *common.security* werden die Signaturerzeugung und -verifikation sowie der Abruf von Trusted Service Lists (TSLs) von der Leitstelle bewerkstelligt.

2.2.1 STRUKTUR UND ARBEITSWEISE

Die Klasse *SignatureUtil* bietet die zwei Methoden *sign()* und *validate()* an. Für das Signieren muss die Notification in der XML-Repräsentation übergeben werden. Zusätzlich müssen das Zertifikat sowie der private Schlüssel übergeben werden. Als Kanonisierungsalgorithmus für das *Signed-Info*-Element [4] sowie für die zu signierende Notification wird „Exklusives kanonisches XML“ verwendet. Als URI-Referenz auf die Notification wird ein XPath-Ausdruck angegeben. Zum Validieren einer Signatur werden die Signatur (detached) selbst sowie die Notification übergeben. Aus der Signatur kann der öffentliche Schlüssel für die Validierung extrahiert werden.

Trusted Service Lists werden von der Leitstelle bereitgestellt. Dazu steht P23R-seitig die Klasse *TrustServiceListDepot* zum Abruf der TSLs zur Verfügung. Diese Klasse wird im Rahmen des Bootstrappings verwendet, um initiale Konfigurationsdaten und Zertifikate zu registrieren. Im Konstruktor der Klasse *TrustServiceListDepot* kann das Schlüsselmaterial übergeben werden. Dies ist notwendig, um die Signatur einer abgerufenen TSL zu validieren. TSLs werden zwischengespeichert, d. h. liegt noch keine aktuelle TSL vor, so wird diese abgerufen. Ansonsten kann sie direkt aus dem Arbeitsspeicher geladen (ein dateibasierter Zwischenspeicher wird ebenfalls implementiert) und einem Aufrufer übergeben werden.

2.2.2 SCHNITTSTELLEN

Die folgenden Schnittstellen werden in *SignatureUtil* zur Verfügung gestellt.

- `sign(Element, X509Certificate, PrivateKey)`: `SignatureType`
- `validate(SignatureType, Element)`: `boolean`

Die folgenden Schnittstellen werden in `TrustServiceListDepot` zur Verfügung gestellt.

- `get()`: `P23RTrustServiceList`

Als intern erzeugte Tsl-Resource wird der Name „p23r.tsl“ sowie die URI des Bootstrap Distribution Points „`http://leitstelle.p23r.de/p23r.tsl`“ verwendet.

- `get(TslResource)`: `P23RTrustServiceList`

2.2.3 FEHLERBEHANDLUNG

Beim Signieren werden die folgenden Fehler behandelt:

- Das Serialisieren/Deserialisieren der Notification oder der Signatur schlägt fehl.
- Eine Referenz als XPath-Ausdruck auf die Notification konnte nicht erzeugt werden.

Folgende Fehler beim Abruf einer TSL werden von `common.security` abgefangen:

- Die HTTP-Anfrage an die Leitstelle schlägt fehl (wegen der Zeitüberschreitung oder Nichtverfügbarkeit).
- Die Vertrauenswürdigkeit ist aufgrund einer ungültigen Signatur nicht gegeben.

2.2.4 PROTOKOLLIERUNG

Protokolliert werden beim Erzeugen und Validieren einer Signatur aufgetretene Fehler. Für die Verarbeitung von TSLs werden keine Protokolleinträge erzeugt. Stattdessen wird die Ausnahme `TrustServiceListException` mit spezifischen Fehlermeldungen an die Aufrufer von `TrustServiceListDepot` gesendet.

2.2.5 KONFIGURATION

Für die Klasse `SignatureUtil` ist keine weitere Konfiguration vorgesehen. Lediglich das Verarbeiten von TSLs kann mit den folgenden Einstellungen konfiguriert werden:

- Hostname und Port der Leitstelle für die Tsl-Ressource
- Pfad des dateibasierten Zwischenspeichers

2.2.6 ARTEFAKTE

Für das Projekt `common.security` entsteht folgendes Artefakt:

- `common.security-<version>.jar`

Java-Bibliothek mit Signaturklasse `SignatureUtil` sowie der API für TSLs mit der Zugriffsklasse `TrustServiceListDepot`.

2.3 COMMON TBRS

Die Kernfunktionalität des P23R basiert auf dem Umgang mit XML und Schemata, die in der Spezifikation der Technischen Benachrichtigungsregelsprache (T-BRS) [5] festgelegt wurden. Die Bibliothek `common.tbrs` stellt für diese Schemata neben den JAXB generierten Binding-Klassen eine Menge an Hilfsklassen zum Konvertieren, Intsanziieren und Interpretieren zur Verfügung.

P23R

P23R: Spezifikationen zur Infrastruktur

2.3.1 STRUKTUR UND ARBEITSWEISE

Die Klasse *DOMHelper* bietet eine Methode zur Ersetzung des Namespaces des Root-Knotens DOM Dokument. Die Kindelemente behalten dabei ihre Namespaces.

SchemaHelper und *SchemaHelperFactory* erlaubt das Marschallen und Unmarschallen von JAXB Elementen zu DOM Nodes bzw. Strings. Außerdem bietet es die Möglichkeit, Validierungsfunktionen und Methoden einer Instanz aus einem gegebenen Schema zu generieren. Zu setzende Werte werden entweder aus dem Schema (default Attribut) oder einer bereitgestellten Map entnommen.

TransformerHelper transformiert auf Basis der *javax.xml.transform* API XML-Dokumente zwischen DOM Tree und String bzw. Stream.

2.3.2 SCHNITTSTELLEN

Die Klasse *DOMHelper* stellt folgende Methode als Schnittstelle zur Verfügung:

- `changeNamespace(String namespace, Element element): Element`
Ändert den Namespace des Root Elements durch Erzeugen eines neuen DOM Trees, welcher den vorhandenen importiert.

Die Klasse *TransformerHelper* bietet folgende Methoden als Schnittstelle an:

- `elementToString(Element element): String`
Transformiert das Element in einen formatierten String mit Zeilenumbrüchen und Einrückungen.
- `elementToString(Element element, boolean format): String`
Transformiert das Element in einen String (wahlweise formatiert).
- `elementFromString(String xml): Element`
Transformiert ein XML String in einen DOM Tree.
- `elementFromStream(InputStream stream): Element`
Transformiert einen Eingabestrom in einen DOM Tree.
- `elementFromSource(Source source): Element`
Transformiert eine generelle Quelle (*javax.xml.Source*) in einen DOM Tree.

Die Klasse *SchemaHelper*<*SchemaObject*> bietet folgende Schnittstellen an:

- `create(): SchemaObject, create(String xml): SchemaObject, create(InputStream stream): SchemaObject, create(Element element): SchemaObject`
Erzeugen eines JAXB-Elements, entweder ein leeres oder aus einem String, Stream oder DOM Element.
- `marshall(SchemaObject schemaObject): String`
Erzeugt eine String Representation eines JAXB Objekts.

- `validate(String xml): boolean`, `validate(String xml, String [] schemaLocations): boolean`
Validiert ein XML-Dokument oder -Fragment anhand des Schemas bzw. zusätzlicher Schemata.
- `createDefaultInstance(): String`, `createDefaultInstance(String elementName, Map<String, Object>): String`
Erzeugt eine Instanz. Der Rootelementname und Default-Werte können vorgegeben werden.

Die Klasse *SchemaHelperFactory* bietet Getter Methoden zur Instanziierung gängiger *SchemaHelper<>* Objekte. Die Signatur dieser Getter Methoden sieht folgendermaßen aus:

- `getSchemaObjectSchemaHelper(): SchemaHelper<SchemaObject>`
Instanziert ein *SchemaHelper<>* Object für die Klasse *SchemaObject*.

2.3.3 FEHLERBEHANDLUNG

Die Bibliothek gibt Fehler entweder in Form von Exceptions, leeren Rückgabewerten oder *null* als Rückgabewert an die aufrufende Komponente weiter und loggt Fehler über das eingesetzte Logging-Framework mit.

2.3.4 PROTOKOLLIERUNG

Eine separate Protokollierung außerhalb des Fehler-Loggings (siehe Abschnitt 2.3.3) erfolgt nicht. Logging wird gemäß den Vorgaben aus den Programmierrichtlinien durchgeführt.

2.3.1 KONFIGURATION

Für die Bibliothek ist keinerlei Konfiguration erforderlich.

2.3.2 ARTEFAKTE

Für das Projekt *common.tbrs* entsteht folgendes Artefakt:

- `common.tbrs-<version>.jar`
Java-Bibliothek mit JAXB generiert Binding-Klassen sowie diversen Hilfsklassen.

2.4 COMMON P23R

Diese Bibliothek enthält Klassen und Funktionen, die im Umfeld der P23R-Entwicklung grundsätzlich nützlich sind. Sie bietet – neben einer Klasse zum Verwalten von Konfigurationsdateien – eine Enumeration für aus der P23R-Rahmenarchitektur [1] normativ festgelegte Werte der *P23RAppFault*, aus den Webservice Schnittstellen sowie eine Factory zum Erzeugen von *P23RAppFaults* mit Hilfe dieser Enumerationen. Schließlich kann mit Hilfe der *P23RTslHelper* Klasse auf einfache Weise eine Endpunktadresse aus einer TSL anhand eines Namens gelesen werden.

2.4.1 STRUKTUR UND ARBEITSWEISE

Die Klasse *P23RConfig* erlaubt das Anlegen bzw. Ergänzen einer Konfigurationsdatei. Dabei wird die Datei im Home-Verzeichnis des ausführenden Benutzers unter `~/.p23r/<name>` gesucht. Ist sie dort nicht vorhanden, wird sie anhand einer vorgegebenen Ressource mit demselben Namen erzeugt. Ist sie bereits vorhanden, wird sie mit der gleichnamigen Ressource abgeglichen und gegebenenfalls ergänzt. Sie muss zur Verwendung abgeleitet werden.

P23R

P23R: Spezifikationen zur Infrastruktur

Die Klasse *P23RTslHelper* ermittelt für eine gegebene TSL anhand des Namens den Trusted Service und liefert die zugehörige Endpunktadresse.

Die Klassen *P23RXmlDbResource* und *P23RXmlDbResourceContentAccess* bieten eine einfache Verwaltung von XML Ressourcen in der eXist-XML-Datenbank.

2.4.2 SCHNITTSTELLEN

Die Klasse *P23RConfig* bietet folgende Schnittstelle:

- `getStringValue(String name): String`, `getStringValue(String name, String defValue): String`
Liefert den Konfigurationswert für den Namen ‚name‘. Wahlweise kann ein Default-Wert übergeben werden, der zurückgegeben wird, wenn unter dem Namen kein Wert existiert.
- `getBooleanValue(String name): boolean`
Liefert den Konfigurationswert für den Namen ‚name‘.

Die Klasse *P23RTslHelper* bietet folgende Methode als Schnittstelle:

- `discoverServiceEndpointAddress(String name): String`
Ermittelt die Adresse aus der initialisierten TSL anhand des Namens des Trusted Service.

Die Klasse *P23RXmlDbResource* bietet Methoden zum Setzen des Contents und Abfragen von XQuery-Skripten auf Ressourcen. Um eine Ressource als Ganzes aus der Datenbank zu extrahieren, ist die Klasse *P23RXmlDbContentAccess* nötig. Das Fehlen der entsprechenden Methoden in der Klasse *P23RXmlDbResource* verhindert das unbesonnene Halten von großen Mengen an XML-Daten im Speicher.

2.4.3 FEHLERBEHANDLUNG

Die Bibliothek gibt Fehler entweder in Form von Exceptions, leeren Rückgabewerten oder *null* als Rückgabewert an die aufrufende Komponente weiter und loggt Fehler über das eingesetzte Logging-Framework mit. Logausgaben erfolgen gemäß den Programmierrichtlinien.

2.4.4 PROTOKOLLIERUNG

Die Bibliothek führt keine eigene Protokollierung durch.

2.4.5 KONFIGURATION

Für die Bibliothek ist keine Konfiguration notwendig.

2.4.6 ARTEFAKTE

Für das Projekt *common.p23r* entsteht folgendes Artefakt:

- `common.p23r-<version>.jar`
Java-Bibliothek mit P23R-spezifischen Hilfsklassen für Konfiguration, Fault Handling und TSL-Abfragen.

3 SPEZIFIKATION DES P23R

Der P23R umfasst diejenigen Komponenten der P23R-Rahmenarchitektur [1], welche für die Generierung und den Versand von Benachrichtigungen auf Basis aggregierter Unternehmensdaten notwendig sind. Dabei steht insbesondere die Verarbeitung von Benachrichtigungsregelpaketen der P23R-Leitstelle (Kapitel 4) im Fokus der Umsetzung.

Die einzelnen Funktionen, die durch die Abbildung der Rahmenarchitektur (siehe Kapitel 2 des Pflichtenhefts [2]) definiert wurden, werden als ESB-Dienste realisiert (siehe Abbildung 1).

Die darüber hinaus notwendigen Außenschnittstellen, definiert als Webservices, werden mit Hilfe des JAX-WS Frameworks JBossWS [6] umgesetzt bzw. aufgerufen.

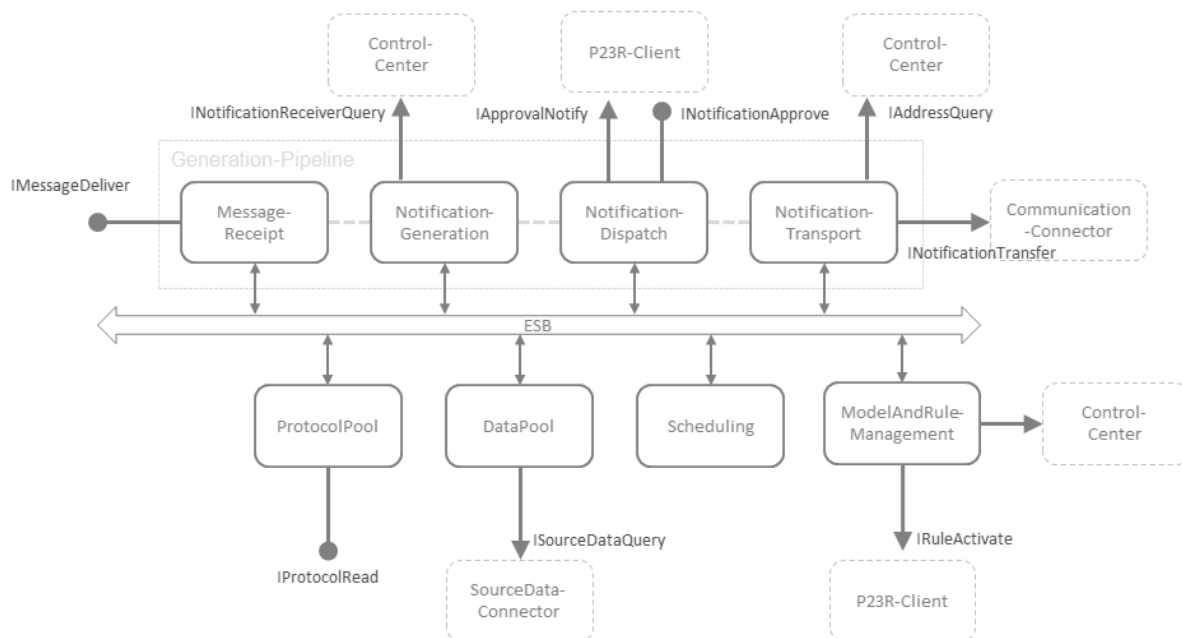


ABBILDUNG 1: ESB-DIENSTE DER MUSTERIMPLEMENTIERUNG P23R

Zur Adressierung wird im JBoss ESB jeder Dienst einer Kategorie zugewiesen und innerhalb dieser Kategorie ein eindeutiger Name. Alle Dienste der Musterimplementierung des P23R werden der Kategorie *P23R* zugeordnet.

Die Kommunikation über den ESB erfolgt mit Hilfe von Messages (siehe Kapitel 3 und die Dokumentation JBoss ESB [7]). Ein Sender einer Message hat dabei im Header die Aktion zu definieren und im Body die Parameter zu setzen. Das JBoss ESB Message Format bietet dabei bereits eine einfache Map als Parameterliste als Teil des Message Body an.

Folgende Pakete aus der P23R-Rahmenarchitektur [1] bilden die Musterimplementierung des P23R und werden durch ESB-Dienste gemäß Pflichtenheft [2] umgesetzt (siehe auch Abbildung 1):

- Nachrichtenempfang (Abschnitt 3.3)
- Benachrichtigungsgenerierung (Abschnitt 3.4)
- Benachrichtigungsversand (Abschnitt 3.5)

P23R

P23R: Spezifikationen zur Infrastruktur

- Benachrichtigungstransport (Abschnitt 3.6)
- Datenpool (Abschnitt 3.7)
- Protokollpool (Abschnitt 3.8)
- Datenmodelle und Benachrichtigungsregeln (Abschnitt 3.9)
- Termine und Zeitüberschreitungen (Abschnitt 3.10)

Die Komponente Bootstrapping (siehe Abschnitt 3.11) findet sich nicht als eigenständiger ESB-Dienst wieder und wird funktionell in davon abhängigen Komponenten umgesetzt. Die Einbindung der Sicherheitsinfrastruktur erfolgt über das Einhängen von Policy Enforcement Points (PEPs) in die Laufzeitumgebung des Application-Servers (siehe Abschnitt 3.13).

3.1 COMMON ESB

Für die ESB-Komponenten werden grundlegende Funktionen in der Bibliothek *p23r.common.esb* zusammengefasst. So sind darin sowohl die Klassen zum Erstellen und Versenden von ESB-Messages enthalten als auch die dafür benötigten Konstanten.

Des Weiteren enthält die Bibliothek eine *ProtocolPoolService* Klasse, die die Verwendung des *ProtocolPool* vereinfacht, ähnlich wie es bei Logging Frameworks geschieht. Schliesslich unterstützt die Klasse *P23RappFaultAdapter* das Weiterleiten von *P23RappFault* Exceptions über den ESB in Richtung externer Webservice Schnittstellen.

3.1.1 STRUKTUR UND ARBEITSWEISE

Die Klasse *P23RServiceInvoker* bietet erweiterte Funktionen zu der vom JBoss ESB bereitgestellten Klasse *ServiceInvoker*. Sie dient als Basis für die Kapselung von konkreten Komponenten. Durch das Implementieren komponentenspezifischer Methoden lassen sich alle ESB-spezifischen Funktionen verstecken.

Der *ProtocolPoolService* kapselt die Verwendung des *ProtocolPool* (siehe Abschnitt 3.8) und bietet Methoden zum einfachen Absetzen von Protokolleinträgen. Die Übertragung geschieht asynchron, so dass Anwendungen nicht durch die Übertragung blockiert werden. Einer Instanz wird ein Name zugeordnet, üblicherweise derjenige der übergeordneten Komponente.

3.1.2 SCHNITTSTELLEN

Die Klasse *P23RServiceInvoker* bietet nur eine protected Schnittstelle, so dass ihre Verwendung nur in abgeleiteten Klassen stattfinden kann:

- `createAction(String uri): Message`, `createAction(URI uri): Message`
Erzeugt ein Message Object, in dem eine bestimmte Aktion gesetzt wurde. Diese Aktion dient dem Empfänger zum Erkennen der gewünschten Funktion und der entsprechend mitgelieferten Parameter.
- `invokeAction(Message action): Message`
Sendet die mit `createAction` erzeugte Nachricht an den Empfänger. Die Übertragung erfolgt synchron. Das zurückgegebene Message Objekt enthält die Antwort der ausgeführten Aktion.

- `sendAction(Message action): void`

Sendet die mit `createAction` erzeugte Nachricht an den Empfänger. Die Übertragung erfolgt asynchron. Eine Antwort gibt es nicht.

Die Klasse `ProtocolPoolService` bietet folgende Methoden als Schnittstelle an:

- `getProtocol(String name): ProtocolPoolService`

Factory zum Erzeugen einer Instanz mit dem Namen `name`. Wenn bereits eine Instanz mit diesem Namen existiert, wird genau diese zurückgegeben und keine neue erzeugt.

- `logInfo(String note): void`, `logInfo(String ruleName, String ruleId, String release, String note): void`, `logInfo(NotificationProfile context, String note): void`

Erzeugt einen Info-Protokolleintrag. Es wird das Default Protokoll Format verwendet.

- `logError(String note, Integer errorCode): void`, `logError(String ruleName, String ruleId, String release, String note, Integer errorCode): void`, `logError(String transactionId, String messageId, String note, Integer errorCode): void`, `logError(NotificationProfile context, String note, Integer errorCode): void`

Erzeugen einen Error Protokolleintrag. Es wird das Default Protokoll Format verwendet.

3.1.3 FEHLERBEHANDLUNG

Die Bibliothek gibt Fehler entweder in Form von Exceptions, leeren Rückgabewerten oder `null` als Rückgabewert an die aufrufende Komponente weiter oder loggt den Fehler über das eingesetzte Logging-Framework mit.

3.1.4 PROTOKOLLIERUNG

Die Bibliothek nimmt keine eigenen Protokollierungen vor. Ein Logging gemäß den Programmierrichtlinien findet statt.

3.1.5 KONFIGURATION

Für die Bibliothek ist keine Konfiguration vorgesehen.

3.1.6 ARTEFAKTE

Für das Projekt `common.tbrs` entsteht folgendes Artefakt:

- `p23r.common.esb-<version>.jar`

Java-Bibliothek mit unterstützenden Klassen für die ESB Komponenten.

3.2 DATASELCOMPILER

Die P23R-Komponente `DataSelCompiler` (`DataSelCompiler`) wird gemäß den Vorgaben des Pflichtenhefts [2] dazu verwendet, etwaig vorhandene `DataSel`-Skripte der Benachrichtigungsregelpakete in `XQuery`-Skripte, die mit der P23R-Musterimplementierung und den einzelnen Konnektoren kompatibel sind, zu übersetzen (vgl. Kapitel 7 der Spezifikation der T-BRS [5]). Die `DataSelCompiler`-Komponente wird durch das Projekt mit der Bezeichnung `p23r.dataselcompiler` als Bibliothek realisiert. Zur Entkopplung und zur optionalen Verwendung ist die Komponente über eine separate Schnittstelle ansprechbar.

P23R

P23R: Spezifikationen zur Infrastruktur

3.2.1 STRUKTUR UND ARBEITSWEISE

Der *DataSelCompiler* basiert auf dem XText-Framework [8] und ist in der Lage, die als Domain Specific Language (DSL) durch die T-BRS [5] vorgegebenen DataSel-Skripte verlustfrei in XQuery-Skripte zu übersetzen. Dabei werden mit Hilfe des Frameworks auf Basis der durch die T-BRS vorgegebenen Grammatik die relevanten Code-Files für den Compiler (Parser, Validierer etc.) erstellt. Die generierten Funktionen zum Parsen, Validieren und Interpretieren werden als Bibliothek zusammengefasst und können zur Verwendung dynamisch eingebunden werden. Die zusätzliche Bibliothek *p23r.dataselcompiler-api* definiert eine abstrakte Factory-Klasse und ein Interface, welche die konkrete Implementierung kapselt. Dadurch kann die Implementierung ausgetauscht werden.

3.2.2 SCHNITTSTELLEN

Um einen *DataSelCompiler* zu verwenden, wird zunächst über die abstrakte Klasse „*DataSelCompilerFactory*“ mit Hilfe der Methode „*newInstance()*“ eine Instanz der verfügbaren Factory Implementierung erzeugt. Ist keine Implementierung verfügbar, wird null zurückgegeben.

Ist eine Factory verfügbar und eine Instanz wird erzeugt, kann über sie mit Hilfe der Methode „*createCompilerInstance()*“ ein *DataSelCompiler* instanziiert werden. Dieser implementiert die Schnittstelle „*DataSelCompiler*“ mit den folgenden beiden Methoden:

- *String compile(String resourceName)* – Der Compiler benötigt einen Namen mit der Endung „.dats“, welches die fest definierte File-Extension für DataSel-Skripte ist. Diese Methode sucht unter diesem Namen eine Resource im Classpath, um sie zu übersetzen.
- *String compile(String resourceName, String resource)* – Bei dieser Methode kann neben dem Namen das Skript als String direkt mit übergeben werden.

3.2.3 FEHLERBEHANDLUNG

Im Fehlerfall, z. B. wenn die die Ressource nicht geladen werden kann oder das Skript nicht valide ist, wird ein leeres Ergebnis zurückgeliefert. Validierungsfehler werden zusätzlich detailliert auf Standard-Error ausgegeben.

3.2.4 PROTOKOLLIERUNG

Der *DataSelCompiler* nimmt keine eigene Protokollierung vor.

3.2.5 KONFIGURATION

Die abstrakte Factory Klasse *DataSelCompilerFactory* verwendet die System Eigenschaft „*de.p23r.datasel.compiler.DataSelCompilerFactoryImpl*“, um die Implementierung zu instanziiieren. Die Anwendung kann darüber eine konkrete *DataSelCompiler* Implementierung referenzieren.

3.2.6 ARTEFAKTE

Für das Projekt *p23r.dataselcompiler* entsteht folgendes Artefakt:

- *p23r.dataselcompiler-api-<version>.jar*
Java-Bibliothek mit der Factory Klasse und der Schnittstelle zum Compiler.
- *p23r.dataselcompiler-<version>.jar*
Java-Bibliothek mit den Klassen der Implementierung für den *DataSelCompiler*.

3.3 MESSAGE RECEIPT

Die P23R-Komponente *MessageReceipt* (Nachrichtempfang) wird gemäß den Vorgaben des Pflichtenhefts [2] umgesetzt. Die *MessageReceipt* Komponente wird durch das Projekt mit der Bezeichnung *p23r.messagereceipt* als ESB-Dienst realisiert.

3.3.1 STRUKTUR UND ARBEITSWEISE

Gemäß den im Pflichtenheft definierten Anforderungen bietet der *MessageReceipt* zwei Möglichkeiten, eine Nachricht an den P23R zur Verarbeitung zu übergeben. Zum einen kann eine Nachricht von außen über die Webservice Schnittstelle *IMessageReceiver* übergeben werden. Zum anderen haben interne Komponenten, wie zum Beispiel der *Scheduling*-Dienst (siehe Abschnitt 3.10) oder der *NotificationTransport*-Dienst (siehe Abschnitt 3.6) die Möglichkeit, die Nachricht über den ESB an die Komponente direkt zu übergeben. Die Verarbeitung findet jeweils innerhalb des ESB-Diensts statt.

Die Nachricht wird verifiziert und bei Bedarf um Standardwerte angereichert. Zusätzlich wird ein initiales Benachrichtigungsprofil (*NotificationProfile*) erstellt. Dieses wird abschließend der *NotificationGeneration* (siehe Abschnitt 3.4) zur weiteren Bearbeitung übergeben.

3.3.2 FEHLERBEHANDLUNG

Folgende Fehler führen zum Abbruch des Nachrichtenempfangs und zu einem entsprechenden Protokolleintrag (siehe Abschnitt 3.3.3).

- Eine fehlerhafte oder unvollständige ESB-Message wurde empfangen.
- Die Nachricht ist ungültig oder nicht zulässig.
- Das Benachrichtigungsprofil konnte nicht erzeugt werden.

3.3.3 PROTOKOLLIERUNG

Der Dienst nutzt zur Protokollierung seiner Aktivitäten die Komponente *ProtocolPool*. Gemäß der Spezifikation für den *ProtocolPool* (siehe Abschnitt 3.8) werden für folgende Ereignisse die entsprechenden Messages an den *ProtocolPool* gesendet:

- Eine valide Message wurde über den ESB empfangen und führt zum Versuch, eine Benachrichtigung zu generieren.
- Ein Fehler ist aufgetreten (siehe Abschnitt 3.3.2).

3.3.4 SCHNITTSTELLEN

Der Nachrichtenempfang erfolgt entweder intern über den ESB oder extern über eine Webservice-Schnittstelle.

3.3.4.1 SCHNITTSTELLE ZUM ESB

Der Dienst ist unter der Adresse *P23R/MessageReceipt* (Category/ServiceName) erreichbar. Folgende Funktion kann über den ESB in Form einer Message aufgerufen werden:

P23R

P23R: Spezifikationen zur Infrastruktur

receiveMessage

Die Message enthält den Aktionswert `receivemessage` im Header und den Parameter `Message` im Payload des Message Bodys (siehe Tabelle 1). Eine Rückmeldung erfolgt nicht.

TABELLE 1: RECEIVEMESSAGE MESSAGE

urn:p23r/main/messagereceiver/action/receivemessage		
Parameter	Typ	Beschreibung
Incoming		
de.p23r.main.param.p23rmessage	String	P23R Nachricht
Outgoing		
-		

3.3.4.2 SCHNITTSTELLE *IMESSAGEDELIVERLOCAL* (EXTERN, EINGEHEND)

Die Schnittstelle *IMessageDeliverLocal* wird als Webservice-Schnittstelle entsprechend der in der P23R-Rahmenarchitektur spezifizierten WSDL implementiert. Die Implementierung erfolgt mit Hilfe von JBossWS und wird als Web Application verpackt.

Die Implementierung verpackt die Nachricht in eine ESB-konforme Message und leitet sie an den MessageReceipt ESB-Dienst weiter, wo sie gemäß den Anforderungen aus dem Pflichtenheft [2] verarbeitet wird. Es wird keine Ausgangsnachricht erzeugt.

Der Webservice erhält eine Eingangsnachricht vom Typ *deliverMessage*. Es wird eine Ausgangsnachricht vom Typ *deliverMessageResponse* erwartet.

3.3.5 ARTEFAKTE

Für die Komponente *MessageReceipt* entstehen folgende Artefakte:

- `p23r.messagereceiverwslib-<version>.jar`
Webservice Bibliothek mit generierten Klassen für die Implementierung eines Service oder eines Consumers.
- `p23r.messagereceiveresb-<version>.esb`
JBoss ESB Service Komponente.
- `p23r.messagereceiverws-<version>.war`
JBoss WS Komponente, enthält `p23r.messagereceiverwslib-<version>.jar` für die Realisierung der WS-Implementierung.

3.4 NOTIFICATIONGENERATION

Die P23R-Komponente *NotificationGeneration* (Benachrichtigungsgenerierung) wird gemäß den Vorgaben des Pflichtenhefts [2] umgesetzt. Die *NotificationGeneration* Komponente wird durch das Projekt mit der Bezeichnung *p23r.notificationgeneration* als ESB-Dienst realisiert.

3.4.1 STRUKTUR UND ARBEITSWEISE

Der ESB-Dienst erhält über den ESB eine Message mit der Aufforderung zur Generierung einer Benachrichtigung. Sie enthält bereits das initiale Benachrichtigungsprofil (*NotificationProfile*) mit allen Informationen, die zur Generierung der gewünschten Benachrichtigung erforderlich sind, insbesondere dem Verweis auf die anzuwendende Benachrichtigungsregel.

Mit Hilfe des *ModelAndRuleManagement* (siehe Abschnitt 3.9) fordert er alle erforderlichen Artefakte der Benachrichtigungsregel an. Er beauftragt den *DataPool* (siehe Abschnitt 3.7) mit der Selektion der benötigten Daten und beginnt auf dem Ergebnis, die Transformation durchzuführen. Nach deren erfolgreichem Abschluss reicht er das Ergebnis an den *NotificationDispatch* (siehe Abschnitt 3.5) weiter.

3.4.2 FEHLERBEHANDLUNG

Folgende Fehler führen zum Abbruch der Generierung und zu einem entsprechenden Protokolleintrag (siehe Abschnitt 3.4.3).

- Das eingehende Benachrichtigungsprofil ist fehlerhaft oder unvollständig.
- Ein benötigtes Artefakt steht nicht zur Verfügung.
- Bei der Anfrage beim DataPool ist ein Fehler aufgetreten.
- Bei einer Transformation ist ein Fehler aufgetreten.

3.4.3 PROTOKOLLIERUNG

Der Dienst nutzt zur Protokollierung seiner Aktivitäten die Komponente *ProtocolPool*. Gemäß der Spezifikation für den *ProtocolPool* (siehe Abschnitt 3.8) werden, wie im Pflichtenheft [2] definiert, für folgende Ereignisse die entsprechenden Protokolleinträge an den *ProtocolPool* gesendet:

- Eine valide Message wurde über den ESB empfangen und führt zum Versuch, eine Benachrichtigung zu generieren.
- Daten wurden erfolgreich selektiert.
- Ein Transformationsschritt wurde erfolgreich durchgeführt.
- Eine Benachrichtigung wurde erfolgreich generiert und wird an den *NotificationDispatch* Dienst übergeben.
- Ein Fehler ist aufgetreten (siehe Abschnitt 3.4.2).

P23R

P23R: Spezifikationen zur Infrastruktur

3.4.4 SCHNITTSTELLEN

Der *NotificationGeneration* Dienst ist nur über den ESB aufrufbar. Er nutzt die ESB-Dienste *ModelAndRuleManagement*, *DataPool* und *ProtocolPool*. Sie selbst rufen bei Bedarf die externe Webservice Schnittstelle *INotificationReceiverQuery* der P23R-Leitstelle (siehe Kapitel 4) auf.

3.4.4.1 SCHNITTSTELLE ZUM ESB

Der Dienst ist unter der Adresse *P23R/NotificationGeneration (Category/ServiceName)* erreichbar. Folgende Funktionen können über den ESB in Form einer Message aufgerufen werden:

generateNotification

Die Message enthält den Aktionswert *generatenotification* im Header und den Parameter *NotificationProfile* im Payload des Message Bodys (siehe Tabelle 2). Es erfolgt keine Rückantwort.

TABELLE 2: GENERATENOTIFICATION MESSAGE

urn:p23r/main/notificationgeneration/action/generatenotification		
Parameter	Typ	Beschreibung
Incoming		
de.p23r.main.param.notificationprofile	String	Benachrichtigungsprofil
Outgoing		
-		

3.4.5 ARTEFAKTE

Für die Komponente *NotificationGeneration* entstehen folgende Artefakte:

- *p23r.notificationgenerator-<version>.esb*
JBoss ESB Service Komponente.

3.5 NOTIFICATIONDISPATCH

Die P23R-Komponente *NotificationDispatch* (Benachrichtigungsversand) wird gemäß den Vorgaben des Pflichtenhefts [2] umgesetzt. Die *NotificationDispatch* Komponente wird durch das Projekt mit der Bezeichnung *p23r.notificationdispatch* als ESB-Dienst realisiert.

3.5.1 STRUKTUR UND ARBEITSWEISE

Der ESB-Dienst erhält über den ESB eine Message mit der Aufforderung zum Versand einer Benachrichtigung. Die übergebene Benachrichtigung wird persistiert und über die Schnittstelle *IApprovalNotify* der P23R-Client (siehe Kapitel 5) darüber benachrichtigt, dass eine freizugebende Benachrichtigung vorliegt.

Die Abarbeitung der Generation-Pipeline wird so lange unterbrochen, bis der P23R-Client über die Schnittstelle *INotificationApprove* die Freigabe der Benachrichtigung erteilt.

Über diese Schnittstelle kann auch eine veränderte Benachrichtigung übergeben werden, die statt der generierten Benachrichtigung an den Empfänger gesendet werden soll. Die veränderte Benachrichtigung wird validiert und ebenfalls persistiert. Danach wird der *NotificationTransport* (siehe Abschnitt 3.6) aufgerufen.

3.5.2 DATENBESTAND UND PERSISTENZ

Zur Speicherung empfangener oder generierter Benachrichtigungen wird eine relationale Datenbank verwendet. Der Zugriff erfolgt über den Java EE Standard JSR 220 JPA [3], wodurch erreicht wird, dass je nach Anforderung an die Zielumgebung ein adäquates Produkt zum Einsatz kommen kann.

Für die Dauer eines Freigabeprozesses wird ebenfalls das Benachrichtigungsprofil mit abgelegt. Wird durch den P23R-Client bei der Freigabe eine geänderte Benachrichtigung übergeben, wird die geänderte Benachrichtigung zusätzlich zur erzeugten Benachrichtigung persistiert.

3.5.3 FEHLERBEHANDLUNG

Folgende Fehler führen zum Abbruch des Nachrichtenversands und zu einem entsprechenden Protokolleintrag (siehe Abschnitt 3.5.4).

- Eine fehlerhafte oder unvollständige Message wurde über den ESB empfangen.
- Der P23R-Client konnte nicht benachrichtigt werden.
- Die empfangene Benachrichtigung ist nicht valide.
- Die empfangene Benachrichtigung konnte nicht gespeichert werden.
- Die Benachrichtigung wurde nicht freigegeben.
- Es konnte keine Empfängeradresse festgestellt werden.

3.5.4 PROTOKOLLIERUNG

Der Dienst nutzt zur Protokollierung seiner Aktivitäten die Komponente *ProtocolPool*. Gemäß der Spezifikation für den *ProtocolPool* (siehe Abschnitt 3.8) werden für folgende Ereignisse die entsprechenden Messages an den *ProtocolPool* gesendet:

- Eine valide Message wurde über den ESB empfangen und führt zum Versuch, eine Benachrichtigung freizugeben.
- Eine Benachrichtigung wurde erfolgreich gespeichert.
- Eine Benachrichtigung wurde freigegeben.
- Der P23R-Client wurde erfolgreich benachrichtigt.
- Die Empfängeradresse konnte festgestellt werden.
- Die Benachrichtigung konnte erfolgreich dem Benachrichtigungstransport übergeben werden.
- Ein Fehler ist aufgetreten (siehe Abschnitt 3.5.3).

P23R

P23R: Spezifikationen zur Infrastruktur

3.5.5 SCHNITTSTELLEN

Die Komponente *NotificationDispatch* kann über den ESB aufgerufen werden. Sie selbst ruft die externen Webservice Schnittstellen *IApprovalNotify* des P23R-Client (siehe Kapitel 5) und *IAddressQuery* der P23R-Leitstelle (siehe Kapitel 4) auf.

3.5.5.1 SCHNITTSTELLE ZUM ESB

Der Dienst ist unter der Adresse *P23R/NotificationDispatch (Category/ServiceName)* erreichbar. Folgende Funktionen können über den ESB aufgerufen werden:

dispatchNotification

Die Message enthält den Aktionswert *dispatchnotification* im Header und die beiden Parameter *Notification* und *NotificationProfile* im Payload des Message Bodys (siehe Tabelle 3). Es erfolgt keine Rückantwort.

TABELLE 3: DISPATCHNOTIFICATION MESSAGE

urn:p23r/main/notificationmanager/action/dispatchnotification		
Parameter	Typ	Beschreibung
Incoming		
de.p23r.main.param.notificationProfile	String	Benachrichtigungsprofil
de.p23r.main.param.notification	String	Benachrichtigung
Outgoing		
-		

3.5.5.2 SCHNITTSTELLE INOTIFICATIONAPPROVE (EXTERN, EINGEHEND)

Die Schnittstelle *INotificationManager* wird als Webservice-Schnittstelle entsprechend der in der P23R-Rahmenarchitektur [1] spezifizierten WSDL implementiert. Der Webservice erhält eine Eingangsnachricht vom Typ *approveNotification*. Es wird eine Ausgangsnachricht vom Typ *approveNotificationResponse* erwartet.

3.5.6 ARTEFAKTE

Für die Komponente *NotificationDispatch* entstehen folgende Artefakte:

- *p23r.notificationmanagerwslib-<version>.jar*
Webservice Bibliothek mit generierten Klassen für die Implementierung eines Service oder eines Consumers.
- *p23r.notificationdispatchesb-<version>.esb*
JBoss ESB Service Komponente, die auch die Persistenz-Schicht zum Abspeichern von Benachrichtigungen enthält.

- `p23r.notificationmanagerws-<version>.war`

JBoss WS Komponente, enthält `p23r.notificationmanagerwslib-<version>.jar` für die Realisierung der WS-Implementierung.

3.6 NOTIFICATIONTRANSPORT

Die P23R-Komponente *NotificationTransport* (Benachrichtigungsversand) wird gemäß den Vorgaben des Pflichtenhefts [2] umgesetzt. Die *NotificationTransport* Komponente wird durch das Projekt mit der Bezeichnung `p23r.notificationtransport` als ESB-Dienst realisiert.

3.6.1 STRUKTUR UND ARBEITSWEISE

Der ESB-Dienst erhält über den ESB eine Message mit der Aufforderung zum Transport einer Benachrichtigung. Nach Eingang der entsprechenden `sendNotification` Message werden im ersten Schritt der oder die Kommunikationskonnektor(en) anhand der im *NotificationProfile* gesetzten Kommunikationskanäle ermittelt.

Dabei kann eine Zuordnung auf zwei Arten stattfinden. Sollte die `id` gesetzt sein, wird versucht, in der Konfiguration den entsprechenden Konnektor zu identifizieren. Im Anschluss findet ein Abgleich anderer vorhandener Werte statt und führt bei Nicht-Übereinstimmung zu einem Fehler. Sollte die `id` nicht gesetzt sein, wird versucht, anhand des Attributs `type` einen passenden Kommunikationskonnektor zu finden.

Eventuell vorhandene Prioritäten bei mehreren gegebenen *channels* werden berücksichtigt. Dabei bedeutet ein niedrigerer Wert eine höhere Priorität und wird entsprechend zuerst berücksichtigt. Bei gleicher Priorität wird keine Angabe über die Reihenfolge gemacht und ist implementierungsabhängig.

Nachdem mindestens ein Kommunikationskonnektor ermittelt werden konnte, werden eventuell benötigte Präsentationsskripte vom *ModelAndRuleManagement* Dienst (siehe Kapitel 3.9) angefordert und mit der Benachrichtigung und den Kommunikationsparametern an den jeweiligen Kommunikationskonnektor (siehe Kapitel 7) übergeben.

3.6.2 FEHLERBEHANDLUNG

Folgende Fehler führen zum Abbruch der Übermittlung an den Empfänger und zu einem entsprechenden Protokolleintrag (siehe Abschnitt 3.6.3):

- Der Kommunikationskonnektor kann nicht ermittelt werden.
- Der Kommunikationskonnektor ist nicht erreichbar.
- Der Kommunikationskonnektor liefert einen Fehler zurück.
- Die Anwendung eines Präsentationsskripts führt zu einem Fehler.
- Eine fehlerhafte oder unvollständige Message wurde über den ESB empfangen.

3.6.3 PROTOKOLLIERUNG

Der Dienst nutzt zur Protokollierung seiner Aktivitäten die Komponente *ProtocolPool*. Gemäß der Spezifikation für den *ProtocolPool* (siehe Abschnitt 3.8) werden für folgende Ereignisse entsprechende Protokolleinträge an den *ProtocolPool* gesendet:

P23R

P23R: Spezifikationen zur Infrastruktur

- Eine valide Message wurde über den ESB empfangen und führt zum Versuch, eine Benachrichtigung zu übermitteln.
- Der ermittelte Kommunikationskonnektor wird aufgerufen.
- Der Kommunikationskonnektor hat die Benachrichtigung erfolgreich übermittelt.
- Ein Fehler ist aufgetreten (siehe Abschnitt 3.6.2).

3.6.4 KONFIGURATION

Die Konfiguration der Registry zur Ermittlung des zu verwendenden Kommunikationskonnektors erfolgt über eine Datei. Sie enthält eine Zuordnung von Kommunikationskanälen zu entsprechenden Kommunikationskonnektoren. Neben einer Identifizierung des Konnektors ist vor allem die Adresse des Webservices relevant.

Ein Kommunikationskanal ist laut P23R-Rahmenarchitektur [1] durch den Datentyp *channelType* definiert und enthält im Wesentlichen einen Typ, einen Identifier sowie eine Priorität.

Eine Konfigurierung der Registry wird folgendermaßen definiert:

```
<CommunicationKonnektorRegistry>
  <CommunicationKonnektor id="0815" type="webservice" priority="1">
    <ServiceName>CommunicationKonnektorService</ServiceName>
    <Address>http://localhost:8080/comkonnektor01/CommunicationAdppterService?wsdl</Address>
  </CommunicationKonnektor>
  <CommunicationKonnektor type="fax">
    <Address> http://connectorhost:8080/comconfax/CommunicationAdppterService?wsdl</Address>
  </CommunicationKonnektor>
</CommunicationKonnektorRegistry>
```

3.6.5 SCHNITTSTELLEN

Die Komponente *NotificationTransport* kann über den ESB aufgerufen werden. Sie selbst ruft die externe Webservice Schnittstelle *INotificationTransfer* des jeweiligen Kommunikationskonnektors (siehe Kapitel 7) auf.

3.6.5.1 SCHNITTSTELLE ZUM ESB

Der Dienst ist unter der Adresse *P23R/NotificationTransport* (Category/ServiceName) erreichbar. Folgende Funktionen können über den ESB aufgerufen werden:

sendNotification

Die Message enthält den Aktionswert *sendnotification* im Header und die beiden Parameter *Notification* und *NotificationProfile* im Payload des Message Bodys (siehe Tabelle 4). Es erfolgt keine Rückmeldung.

TABELLE 4: SENDNOTIFICATION MESSAGE

urn:p23r/main/notificationtransport/action/sendnotification		
Parameter	Typ	Beschreibung
Incoming		
de.p23r.main.param.notificationProfile	String	Benachrichtigungsprofil
de.p23r.main.param.notification	String	Benachrichtigung
Outgoing		
-		

3.6.6 ARTEFAKTE

Für die Komponente *NotificationTransport* entstehen folgende Artefakte:

- p23r.notificationtransport-<version>.esb
JBoss ESB Service Komponente

3.7 DATAPOOL

Die P23R-Komponente *DataPool* (Datenpool) wird gemäß den Vorgaben des Pflichtenhefts [2] umgesetzt. Die *DataPool* Komponente wird durch das Projekt mit der Bezeichnung *p23r.datapool* als ESB-Dienst realisiert.

3.7.1 STRUKTUR UND ARBEITSWEISE

Der *DataPool* Dienst erfüllt zwei wesentliche Aufgaben. Zum einen nimmt er Konfigurationsdaten an und legt sie unter dem Schlüssel eines zugehörigen Namespaces ab. Zum zweiten selektiert er Daten anhand eines übergebenen Namespaces und eines Selektionsskripts.

Je nachdem, ob der übergebene Namespace zu einer vorher abgelegten Konfiguration gehört oder einem Quelldatenkonnektor zugeordnet ist, führt er die Selektion aus oder reicht sie an den Konnektor weiter. Das Ergebnis wird an den Aufrufenden zurückgeliefert.

Im Falle eines Selektionsskripts im DataSel [9] (Kapitel 5) wird das Skript zunächst in XQuery transformiert, bevor es auf die in XML vorliegenden Daten angewandt wird.

Im Falle, dass der Namespace sowohl auf Konfigurationsdaten verweist als auch einem Quelldatenkonnektor zugeordnet sein sollte, wird zuerst versucht, die Selektion auf den Konfigurationsdaten auszuführen. Tritt hierbei ein Fehler auf, z. B. weil das Selektionsskript nicht passt, wird die Selektion an den Quelldatenkonnektor weitergereicht.

3.7.2 FEHLERBEHANDLUNG

Folgende Fehler führen zu einem entsprechenden Protokolleintrag (siehe Abschnitt 3.7.3) und einer Rückmeldung zum aufrufenden Dienst.

P23R

P23R: Spezifikationen zur Infrastruktur

- Für einen Namespace existieren bereits Konfigurationsdaten.
- Eine Selektion auf die gewünschten Konfigurationsdaten schlug fehl.
- Der Quelldatenkonnektor oder eine Konfiguration kann nicht ermittelt werden.
- Der Quelldatenkonnektor ist nicht erreichbar.
- Der Quelldatenkonnektor liefert einen Fehler zurück.
- Eine fehlerhafte oder unvollständige Message wurde über den ESB empfangen.

3.7.3 PROTOKOLLIERUNG

Der Dienst nutzt zur Protokollierung seiner Aktivitäten die Komponente *ProtocolPool*. Gemäß der Spezifikation für den *ProtocolPool* (siehe Abschnitt 3.8) werden für folgende Ereignisse entsprechende Protokolleinträge an den *ProtocolPool* gesendet:

- Eine valide Message wurde über den ESB empfangen und führt zum Versuch, Daten zu selektieren.
- Ein SourceData-Konnektor wurde erfolgreich aufgerufen.
- Eine Selektion auf Konfigurationsdaten wurde erfolgreich durchgeführt.
- Die selektierten Daten wurden erfolgreich an den aufrufenden Dienst zurückgeliefert.
- Ein Fehler ist aufgetreten (siehe Abschnitt 3.7.2).

3.7.4 DATENBESTAND, PERSISTENZ

Alle Konfigurationsdaten werden in einer relationalen Datenbank abgelegt. Sie können anhand eines eindeutig zugeordneten Namespaces ausgelesen werden. Sowohl die eigentlichen Konfigurationsdaten als auch der Namespace werden als String definiert.

3.7.5 KONFIGURATION

Über die *serverconfig.xml* werden die Zugriffsinformationen für die eXist-Datenbank konfiguriert. In ihr legt der Datenpool neben den Systemkonfigurationen die regelpaketspezifischen Configurations ab. Für den Zugriff werden die URI für die Datenbank, der Collection Name, die zugehörigen Credentials für die Collection sowie die Resource-Id benötigt.

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
  <xmldb>
    <uri>xmldb:exist://</uri>
    <create>>true</create>
    <datapool>
      <collection>
        <name>db</name>
        <username>admin</username>
        <password>admin</password>
      </collection>
      <resource>
        <configuration>
```

```

        <id>configuration.xml</id>
    </configuration>
</resource>
</datapool>
</xml:db>
</configuration>
    
```

3.7.6 SCHNITTSTELLEN

Die Komponente *DataPool* wird intern über den ESB angesprochen und führt die übergebenen Selektions- und Transformationskripte aus.

3.7.6.1 SCHNITTSTELLE ZUM ESB

Folgende Funktionen können durch das Senden der entsprechenden ESB-Message an den *DataPool* Dienst aufgerufen werden:

storeConfigdata

Übergibt Konfigurationsdaten zum Abspeichern an den Datenpool (siehe Tabelle 5).

TABELLE 5: STORECONFIGDATA MESSAGE

urn:p23r/main/datapool/action/storeconfigdata		
Parameter	Typ	Beschreibung
Incoming		
de.p23r.main.param.namespace	String	Namespace für die Konfigurationsdaten
de.p23r.main.param.configdata	String	Konfigurationsdaten
Outgoing		
de.p23r.main.param.result	String	Ergebnis der Operation. „success“ wenn erfolgreich, „error“ im Fehlerfall.

removeConfigdata

Löscht Konfigurationsdaten aus dem Datenpool (siehe Tabelle 6).

TABELLE 6: REMMOVECONFIGDATA MESSAGE

urn:p23r/main/datapool/action/removeconfigdata		
Parameter	Typ	Beschreibung
Incoming		
de.p23r.main.param.namespace	String	Namespace der zu entfernenden

P23R

P23R: Spezifikationen zur Infrastruktur

urn:p23r/main/datapool/action/removeconfigdata		
Parameter	Typ	Beschreibung
		Konfigurationsdaten
Outgoing		
de.p23r.main.param.result	String	Ergebnis der operation. „success“ wenn erfolgreich, „error“ im Fehlerfall.

selectData

Selektiert Konfigurationsdaten aus dem Datenpool (siehe Tabelle 7).

TABELLE 7: SELECTDATA MESSAGE

urn:p23r/main/datapool/action/selectdata		
Parameter	Typ	Beschreibung
Incoming		
de.p23r.main.param.namespace	String	Namespace der zu selektierenden Daten
de.p23r.main.param.selectionscript	String	Selektionsbeschreibung (XQuery oder DataSel)
Outgoing		
de.p23r.main.param.selectiondata	String	Die selektierten Daten als XML Dokument

3.7.7 ARTEFAKTE

Für die Komponente *DataPool* entstehen folgende Artefakte:

- p23r.datapool-<version>.esb

JBoss ESB Service Komponente, die auch die Persistenz-Schicht zum Abspeichern von Konfigurationsdaten enthält.

3.8 PROTOCOLPOOL

Die P23R-Komponente *ProtocolPool* (Protokollpool) wird gemäß den Vorgaben des Pflichtenhefts [2] umgesetzt. Die *ProtocolPool* Komponente wird durch das Projekt mit der Bezeichnung *p23r.protocolpool* als ESB-Dienst realisiert.

3.8.1 STRUKTUR UND ARBEITSWEISE

Die internen P23R-Komponenten können durch Aufruf des Diensts einen Protokolleintrag in einer Datenbank-Tabelle persistieren. Für den P23R-Client (siehe Kapitel 5) wird ein Webservice bereitgestellt, über den Protokolleinträge eingesehen bzw. abgerufen werden können.

Für jeden Protokolleintrag werden folgende Attribute persistiert:

- transactionid (eindeutige Vorgangsnummer)
- content (Protokollinformation als XML)
- eventDate (Zeitpunkt des Eintretens des zu protokollierenden Ereignisses)
- type (Typ der zu protokollierenden Information)
- recordID (eindeutiger Identifier des Protokolleintrags)
- recordDate (Zeitpunkt des Schreibens des Protokolleintrags)

3.8.2 FEHLERBEHANDLUNG

Folgende Fehler können auftreten:

- Eine fehlerhafte Search-Anfrage kommt über den Webservice herein und dieser Fehlerfall wird mit dem aus der Rahmenarchitektur spezifizierten Fehlercode beantwortet.
- Ein fehlerhafter Protokolleintrag wird gesendet und wird mit einem entsprechenden Protokolleintrag dokumentiert. Der gesendete Protokolleintrag wird verworfen.

3.8.3 DATENBESTAND, PERSISTENZ

Zur Speicherung der Protokolleinträge wird eine relationale Datenbank verwendet. Der Zugriff erfolgt über den Java EE Standard JSR 220 JPA [3]. Somit kann je nach Anforderung an die Zielumgebung ein adäquates Produkt zum Einsatz gebracht werden.

3.8.4 SCHNITTSTELLEN

Protokolleinträge werden über den ESB eingereicht. Über eine Webservice-Schnittstelle können Protokolleinträge selektiv abgerufen werden.

3.8.4.1 SCHNITTSTELLE ZUM ESB

Der Dienst ist unter der Adresse *P23R/ProtocolPool* (Category/ServiceName) erreichbar. Folgende Funktionen können über den ESB aufgerufen werden:

writeProtocolEntry

Schreibt einen Protokolleintrag in den Protokollpool (siehe Tabelle 8). Es erfolgt keine Rückmeldung.

TABELLE 8: WRITEPROTOCOLENTRY MESSAGE

urn:p23r/main/protocolpool/action/writeprotocoleentry		
Parameter	Typ	Beschreibung
Incoming		

P23R

P23R: Spezifikationen zur Infrastruktur

urn:p23r/main/protocolpool/action/writeprotocolentry		
Parameter	Typ	Beschreibung
de.p23r.main.param.protocolentry	ProtocolEntry	Protokolleintrag
Outgoing		
-		

3.8.4.2 SCHNITTSTELLE IPROTOCOLQUERY (EXTERN, EINGEHEND)

Die Schnittstelle *IProtocolQuery* wird als Webservice-Schnittstelle entsprechend der in der Rahmenarchitektur [1] spezifizierten WSDL implementiert. Der Webservice erhält eine Eingangsnachricht vom Typ *queryProtocol* und liefert im Erfolgsfall eine Ausgangsnachricht vom Typ *queryProtocolResponse*.

3.8.5 ARTEFAKTE

Für die Komponente *ProtocolPool* entstehen folgende Artefakte:

- p23r.protocolpoolwslib-<version>.jar
Webservice Bibliothek mit generierten Klassen für die Implementierung eines Services oder eines Consumers.
- p23r.protocolpooleb-<version>.esb
JBoss ESB Komponente, enthält p23r.protocolpoolpersistence-<version>.jar für die Datenzugriffsschicht.
- p23r.protocolpoolws-<version>.war
JBoss WS Komponente, enthält p23r.protocolpoolwslib-<version>.jar für die Realisierung des WS-Implementierung.

3.9 MODELANDRULEMANAGEMENT

Die P23R-Komponente *ModelAndRuleManagement* (Datenmodell und Benachrichtigungsregeln) wird gemäß den Vorgaben des Pflichtenhefts [2] umgesetzt. Die *ModelAndRuleManagement* Komponente (im Folgenden *MARM* abgekürzt) wird durch das Projekt mit der Bezeichnung *p23r.model-andrulemanagement* als ESB-Dienst realisiert.

Die Komponente *MARM* verwendet dazu im Besonderen folgende zentrale Bibliothek, die direkt mit der Funktionsweise des *MARM* verbunden ist:

- Common Model and Rule Management (siehe Abschnitt 2.1)

3.9.1 STRUKTUR UND ARBEITSWEISE

Die Komponente *MARM* sorgt für den Aufbau sowie für die Aktualisierung des internen Datenmodell- und Benachrichtigungsregelbestands durch Abgleichen mit der P23R-Leitstelle (siehe Kapitel 4), wobei bei Bedarf auch etwaig vorhandene DataSel-Skripte in mit der P23R-Musterimplementierung kompatible XQuery-Skripte übersetzt werden (siehe Abschnitt 3.2).

Sie informiert den P23R-Client (siehe Kapitel 5) über Veränderungen und nimmt Statusänderungen vor (z. B. Aktivierungen bzw. Deaktivierungen von Benachrichtigungsregelgruppen).

Außerdem interagiert sie mit den Komponenten der Generation-Pipeline, indem sie bestimmte Informationen zu den Benachrichtigungsregeln liefert, z. B. Manifest-Informationen, Identifikatoren, aber auch Selektions- oder Transformationsskripte.

Im Folgenden werden einzelne Prozesse in ihrem Ablauf genauer erläutert:

- Abgleich / Aktualisierung der Datenmodell- / Benachrichtigungsregelpakete

Die Komponente *MARM* wird über die entsprechende ESB-Schnittstelle von der Komponente *Scheduling* (siehe Abschnitt 3.10) oder über die entsprechende WS-Schnittstelle (Bestandteil der *IRuleActivate*-Schnittstelle) manuell vom P23R-Client (siehe Kapitel 5) dazu aufgefordert, den internen Bestand an Datenmodell- bzw. Benachrichtigungsregelpaket-Informationen aufzubauen bzw. zu aktualisieren.

Dazu wird die entsprechende REST-Schnittstelle der P23R-Leitstelle (siehe Kapitel 4) abgefragt. Im Ergebnis liegen alle relevanten Informationen aktualisiert vor, inklusive der Berechnung bestimmter Statusinformationen (Empfehlungen etc.).

- Austausch von Statusinformationen über den Datenmodell- / Benachrichtigungsregelpaketbestand

Zwischen der Komponente *MARM* und dem P23R-Client (siehe Kapitel 5) werden über die *IRuleActivate*-Schnittstelle Statusinformationen über den Zustand des internen Datenmodell- bzw. Benachrichtigungsregelpaketbestands ausgetauscht.

Das ermöglicht dem P23R-Client (siehe Kapitel 5) die Visualisierung des Datenbestands sowie die Anforderung von Änderungen an diesem Datenbestand, welche durch die Komponente *MARM* validiert und ausgeführt werden.

- Interaktionen mit *Scheduling*

Über die Komponente *Scheduling* (siehe Abschnitt 3.10) werden bei Aktivierungen bzw. Deaktivierungen die Ausführungszeiten hinterlegt bzw. entfernt.

- Interaktionen mit *DataPool*

Über die Komponente *DataPool* (siehe Abschnitt 3.7) werden bei Aktivierungen bzw. Deaktivierungen die Konfigurationsinformationen hinterlegt bzw. entfernt.

- Interaktionen mit *MessageReceipt*

Der Komponente *MessageReceipt* (siehe Abschnitt 3.3) werden verschiedene Informationen über die jeweilige Benachrichtigungsregel (z. B. Identifikatoren, Manifest, Schemainformationen, ggf. Skripte für das Profile-Splitting etc.) zur Verfügung gestellt.

- Interaktionen mit *NotificationGeneration*.

Der Komponente *NotificationGeneration* (siehe Abschnitt 3.4) werden die Selektions- und Transformationsskripte sowie die Schemainformationen zur Verfügung gestellt.

P23R

P23R: Spezifikationen zur Infrastruktur

3.9.2 DATENBESTAND UND PERSISTENZ

Zur Speicherung der Datenmodell- bzw. Benachrichtigungsregelpaket-Informationen wird eine relationale Datenbank verwendet. Der Zugriff erfolgt über den Java EE Standard JSR 220 JPA [3]. Somit kann je nach Anforderung an die Zielumgebung ein adäquates Produkt zum Einsatz kommen. Entitäten und Persistenzmanager sind in der zentralen Bibliothek Common Model and Rule Management (siehe Abschnitt 2.1) realisiert.

3.9.3 FEHLERBEHANDLUNG

Folgende Fehler führen zu einem entsprechenden Protokolleintrag (siehe Abschnitt 3.9.4):

- Eine fehlerhafte oder unvollständige Message wurde über den ESB empfangen
- Die erfolgreiche Abarbeitung in den jeweiligen ESB-Schnittstellen (siehe Schnittstellen) konnte nicht erfolgreich abgeschlossen werden (z. B. Timeouts)
- Während der Abarbeitung ist ein unvorhergesehener Fehler aufgetreten

3.9.4 PROTOKOLLIERUNG

Der Dienst nutzt zur Protokollierung seiner Aktivitäten die Komponente *ProtocolPool* (siehe Abschnitt 3.8). Gemäß Pflichtenheft [2] werden folgende Aktivitäten protokolliert:

- Herunterladen der Liste von Benachrichtigungsregelpaketen von der Leitstelle
- Herunterladen eines Benachrichtigungsregelpakets von der Leitstelle
- Aktivierung / Deaktivierung einer Benachrichtigungsregelgruppe
- Aktivierung / Deaktivierung einer Benachrichtigungsregel
- Fehler (siehe Abschnitt 3.9.3).

3.9.5 SCHNITTSTELLEN

Die Komponente *MARM* kann über den ESB sowie über die WS-Schnittstelle *IRuleActivate* aufgerufen werden. Sie selbst ruft die REST-Schnittstellen der Leitstelle sowie verschiedene ESB-Schnittstellen anderer Komponenten auf.

3.9.5.1 SCHNITTSTELLE ZUM ESB

Der ESB-Dienst ist unter der Adresse *P23R/ModelAndRuleManagement* (Category/ServiceName) erreichbar. Folgende Funktionen können über den ESB aufgerufen werden:

updateNotificationRulePool

Aktualisiert den NotificationRulePool an Hand der übergebenen Statusinformationen (siehe Tabelle 9). Es erfolgt keine Rückmeldung.

TABELLE 9: UPDATENOTIFICATIONRULEPOOL MESSAGE

urn:p23r/main/notificationrulepoolmanagement/action/updatesnotificationrulepool		
Parameter	Typ	Beschreibung
Eingehend (P23R-Client)		
de.p23r.main.esb.param.updateinformation	String	Update Information
de.p23r.main.esb.param.tenant	String	Tenant
Ausgehend		
-		

getNotificationRuleInformationByMessageType

Gibt die Regelinformationen zu einem bestimmten Nachrichtentyp aus dem NotificationRulePool zurück (siehe Tabelle 10).

TABELLE 10: GETNOTIFICATIONRULEINFORMATIONBYMESSAGETYPE MESSAGE

urn:p23r/main/notificationrulepoolmanagement/action/getnotificationruleinformationbymessagetype		
Parameter	Typ	Beschreibung
Eingehend (MessageReceipt)		
de.p23r.main.esb.param.message.type	String	Message Type (Namespace)
Ausgehend		
de.p23r.main.esb.param.notificationrule.id	String	Notification Rule Identifier
de.p23r.main.esb.param.notificationrule.manifest	String	Notification Rule Manifest XML
de.p23r.main.esb.param.notificationrulegroup.id	String	Notification RuleGroup Identifier
de.p23r.main.esb.param.notificationrulegroup.manifest	String	Notification RuleGroup Manifest XML
de.p23r.main.esb.param.notificationrulepackage.id	String	Notification RulePackage Identifier
de.p23r.main.esb.param.message.schema	String	Schema for the Message of the Notification Rule
de.p23r.main.esb.param.multiprofilegeneration.selection	String	Selection Script for MultiNotificationProfile Generation (optional)
de.p23r.main.esb.param.multiprofilegeneration.transformation	String	Transformation Script for MultiNoti-

P23R

P23R: Spezifikationen zur Infrastruktur

urn:p23r/main/notificationrulepoolmanagement/action/getnotificationruleinformationbymessagetype		
Parameter	Typ	Beschreibung
		ficationProfile Generation (optional)

getNotificationRuleDataSelection

Gibt das Selectionsskript zu einer bestimmten Benachrichtigungsregel aus dem NotificationRulePool zurück (siehe Tabelle 11).

TABELLE 11: GETNOTIFICATIONRULEDATASELECTION MESSAGE

urn:p23r/main/notificationrulepoolmanagement/action/getnotificationruledataselection		
Parameter	Typ	Beschreibung
Eingehend (NotificationGeneration)		
de.p23r.main.esb.param.notificationrule.id	String	Notification Rule Identifier
Ausgehend		
de.p23r.main.esb.param.notificationrule.selection	String	Selection Script
de.p23r.main.esb.param.notificationrule.selection.schema	String	Schema for result of Selection

getNotificationCoreTransformation

Gibt die Core-Transformation zu einer bestimmten Benachrichtigungsregel aus dem NotificationRulePool zurück (siehe Tabelle 12).

TABELLE 12: GETNOTIFICATIONCORETRANSFORMATION MESSAGE

urn:p23r/main/notificationrulepoolmanagement/action/getnotificationcoretransformation		
Parameter	Typ	Beschreibung
Eingehend (NotificationGeneration)		
de.p23r.main.esb.param.notificationrule.id	String	Notification Rule Identifier
Ausgehend		
de.p23r.main.esb.param.notificationrule.notificationcore	String	NotificationCore Transformation Script

getNotificationRuleRepresentationTransformation

Gibt das Repräsentationsscript zu einer bestimmten Benachrichtigungsregel und einem bestimmten Kommunikationskanal sowie Kommunikationsempfänger aus dem NotificationRulePool zurück (siehe Tabelle 13).

TABELLE 13: GETNOTIFICATIONRULEREPRESENTATIONTRANSFORMATION MESSAGE

urn:p23r/main/notificationrulepoolmanagement/action/getnotificationrulerepresentationtransformation		
Parameter	Typ	Beschreibung
Eingehend (NotificationGeneration)		
de.p23r.main.esb.param.notificationrule.id	String	Notification Rule Identifier
de.p23r.main.esb.param.target.id	String	Target Identifier
de.p23r.main.esb.param.channel.id	String	Channel Identifier
Ausgehend		
de.p23r.main.esb.param.notificationrule.notificationcore	String	Representation Transformation Script if found, otherwise empty String

getNotificationRuleTransformationList

Gibt die Transformationskripte zu einer bestimmten Benachrichtigungsregel aus dem NotificationRulePool zurück (siehe Tabelle 14).

TABELLE 14: GETNOTIFICATIONRULETRANSFORMATIONLIST MESSAGE

urn:p23r/main/notificationrulepoolmanagement/action/getnotificationruletransformationlist		
Parameter	Typ	Beschreibung
Eingehend (NotificationGeneration)		
de.p23r.main.esb.param.notificationrule.id	String	Notification Rule Identifier
Ausgehend		
de.p23r.main.esb.param.notificationrule.transformation.list	List <Map stepEntry <String key, String value>	List of Transformation Steps. keys: de.p23r.main.esb.key.notificationrule.transformationlist.notification : a notification transformation script

P23R

P23R: Spezifikationen zur Infrastruktur

urn:p23r/main/notificationrulepoolmanagement/action/getnotificationruletransformationlist		
Parameter	Typ	Beschreibung
		de.p23r.main.esb.key.notificationrule.transformationlist.profile : a profile transformation script
		de.p23r.main.esb.key.notificationrule.transformationlist.schema : the schema for the result of the notification transformation script

isNotificationRuleUpToDate

Prüft, ob bestimmte Benachrichtigungsregel noch aktuell ist (siehe Tabelle 15).

TABELLE 15: ISNOTIFICATIONRULEUPTODATE MESSAGE

urn:p23r/main/notificationrulepoolmanagement/action/isnotificationruleuptodate		
Parameter	Typ	Beschreibung
Eingehend (P23R-Client)		
de.p23r.main.esb.param.notificationrule.id	String	Notification Rule Identifier
de.p23r.main.esb.param.tenant	String	Tenant
Ausgehend		
de.p23r.main.esb.param.notificationrule.updateinformation	JAXB	JAXB UpToDateResultType

getNotificationRulePackageStates

Gibt die Statusinformationen zu den Benachrichtigungsregelpaketen aus dem NotificationRulePool zurück (siehe Tabelle 16).

TABELLE 16: GETNOTIFICATIONRULEPACKAGESTATES MESSAGE

urn:p23r/main/notificationrulepoolmanagement/action/getnotificationrulepackagestates		
Parameter	Typ	Beschreibung
Eingehend (P23R-Client)		
de.p23r.main.esb.param.tenant	String	Tenant
Ausgehend		

urn:p23r/main/notificationrulepoolmanagement/action/getnotificationrulepackagestates		
Parameter	Typ	Beschreibung
de.p23r.main.esb.param.notificationrulepackage.state	JAXB	JAXB RulePackageStateResultType

setNotificationRulePackageStates

Setzt die Statusinformationen zu den Benachrichtigungsregelpaketen im NotificationRulePool auf den übergebenen Wert (siehe Tabelle 17).

TABELLE 17: SETNOTIFICATIONRULEPACKAGESTATES MESSAGE

urn:p23r/main/notificationrulepoolmanagement/action/setnotificationrulepackagestates		
Parameter	Typ	Beschreibung
Eingehend (P23R-Client)		
de.p23r.main.esb.param.notificationrulepackage.state.list	JAXB	List of JAXB RulePackageStateType
de.p23r.main.esb.param.tenant	String	Tenant
Ausgehend		
de.p23r.main.esb.param.notificationrulepackage.state	JAXB	JAXB RulePackageStateResultType

3.9.5.2 SCHNITTSTELLE IRULEACTIVATE (EXTERN, EINGEHEND)

Die Schnittstelle *IRuleActivate* wird als Webservice-Schnittstelle entsprechend der in der P23R-Rahmenarchitektur [1] spezifizierten WSDL implementiert. Der Webservice erhält eine Eingangsnachricht entweder vom Typ *getNotificationRulePackageStates* oder *setNotificationRulePackageStates* oder *isUpToDate*. Es wird eine korrespondierende Ausgangsnachricht entweder vom Typ *getNotificationRulePackageStatesResponse* oder *setNotificationRulePackageStatesResponse* oder *isUpToDateResponse* erwartet.

3.9.6 ARTEFAKTE

Für die Komponente *ModelAndRuleManagement* entstehen folgende Artefakte:

- p23r.iruleactivate.wslib-<version>.jar
Webservice Bibliothek mit generierten Klassen für die Implementierung eines Service oder eines Consumer.
- p23r.modelandrulemanagement-<version>.esb
JBoss ESB Komponente nutzt diverse Common-Bibliotheken, z. B. für Persistenz und REST-Kommunikation.

P23R

P23R: Spezifikationen zur Infrastruktur

- p23r.modelandrulmanagement.ws-<version>.war

JBoss WS Komponente, enthält p23r.iruleactivate.wslib-<version>.jar für die Realisierung der WS-Implementierung

3.10 SCHEDULING

Die P23R-Komponente *Scheduling* (Termine und Zeitüberschreitungen) wird gemäß den Vorgaben des Pflichtenhefts [2] umgesetzt. Die *Scheduling* Komponente wird durch das Projekt mit der Bezeichnung *p23r.scheduling* als ESB-Dienst realisiert.

3.10.1 STRUKTUR UND ARBEITSWEISE

Der Dienst *Scheduling* verwendet zum Verwalten der zeitlichen Vorgaben den Open Source Quartz Enterprise Job Scheduler. Folgende Funktionen stellt er bereit:

- Komponenten können durch das Senden einer ESB Message an den Scheduling Dienst einen Auftrag abgeben. Als Rückgabe erhalten die Komponenten einen eindeutigen Identifier.

Komponenten können durch das Senden einer ESB Message an den Scheduling Dienst einen Auftrag löschen oder aktualisieren.

Der Scheduling Dienst sendet zum Zeitpunkt X eine Nachricht an den gewünschten Empfänger.

Diese kann entweder eine vorgefertigte Message an den *MessageReceipt* (siehe Abschnitt 3.3) sein, oder nur der Hinweis, dass ein Ereignis eingetreten ist. Diese Zeitpunkte können einmalig oder periodisch auftreten. Im letzteren Fall wird der Auftrag automatisch gelöscht.

Das Absetzen einer ESB Message stellt sowohl die Funktionalität eines Call-Backs als auch das Auslösen eines Ereignisses dar.

3.10.2 FEHLERBEHANDLUNG

Folgende Fehler können während der Ausführung des Diensts auftreten und führen zu einem Protokolleintrag (siehe Abschnitt 3.10.3):

- Der eingehende Auftrag kann nicht eingeordnet werden. Neben dem Protokolleintrag wird auch ein Fehler an den Auftraggeber zurückgegeben.
- Eine Nachricht aufgrund eines Auftrags kann nicht gesendet werden. Der Auftrag wird gelöscht, auch wenn es ein periodischer ist.

3.10.3 PROTOKOLLIERUNG

Der Dienst nutzt zur Protokollierung seiner Aktivitäten die Komponente *ProtocolPool*. Gemäß der Spezifikation für den *ProtocolPool* (siehe Abschnitt 3.8) werden für folgende Ereignisse entsprechende Protokolleinträge an den *ProtocolPool* gesendet:

- Annahme eines Auftrags
- Versenden einer Message aufgrund eines Auftrags
- Fehler (siehe Abschnitt 3.10.2)

3.10.4 KONFIGURATION

Das *Scheduling* kann selbst nicht konfiguriert werden. Jedoch kann das zugrunde liegende Quartz Framework konfiguriert werden. Siehe dazu die Dokumentation der Quartz-Bibliothek [10].

3.10.5 SCHNITTSTELLEN

Der Scheduler bietet nur die interne Anbindung an den ESB.

3.10.5.1 SCHNITTSTELLE ESB

Folgende Funktionen können durch das Senden der entsprechenden ESB Message an den Scheduler aufgerufen werden:

scheduleJob

Die Message enthält den Aktionswert *schedulejob* im Header, daraufhin erfolgt eine Rückmeldung. Diese enthält im Erfolgsfall einen eindeutigen Identifier für den Auftrag und im Fehlerfall einen entsprechenden Fehlercode (siehe Tabelle 18).

TABELLE 18: SCHEDULEJOB MESSAGE

urn:p23r/main/scheduling/action/schedulejob		
Parameter	Typ	Beschreibung
Incoming		
de.p23r.main.param.job.name	String	Bezeichner. Kann für Protokollierung verwendet werden
de.p23r.main.param.job.type	String	Definiert die Art der Definition: „quartz“ – Eine Quartz konforme Beschreibung für den Java Quartz-Scheduler. p23r – eine XML Beschreibung wie sie aus der Regel kommt. ... - sonstige
de.p23r.main.param.scheduling.rulemanifest	String	Rulemanifest.
de.p23r.main.param.scheduling.schema	String	Eine XSD, um die Message für den MessageReceipt zu instanziiieren.
de.p23r.main.param.scheduling.servicename	String	Name des Service, der angerufen werden soll.
de.p23r.main.param.scheduling.actionname	String	Name der Aktion, die angerufen werden soll.
de.p23r.main.param.scheduling.parametername	String	Name des Parameters, der übergeben werden soll.

P23R

P23R: Spezifikationen zur Infrastruktur

urn:p23r/main/scheduling/action/schedulejob		
Parameter	Typ	Beschreibung
de.p23r.main.param.scheduling.parameter	String	Werte des Parameters, die übergeben werden sollen.
de.p23r.main.param.scheduling.executiondate	String	Zeit, wann der Aufruf ausgeführt werden soll.
Outgoing		
de.p23r.main.param.job.identifier	String	Im Falle einer erfolgreichen Einordnung des Auftrags.
de.p23r.main.param.job.error.code	int	Fehlercode. 1 – falscher Parameter 2 – fehlender Parameter
de.p23r.main.param.job.error.description	String	Textuelle Beschreibung des Fehlers. (optional)

modifyJob

Die Message enthält den Aktionswert *modifyjob* im Header. Es erfolgt eine Rückmeldung. Diese enthält im Erfolgsfall nichts und im Fehlerfall einen entsprechenden Fehlercode. Diese Funktion ermöglicht nur das Überschreiben der Zeitangaben. Für den Fall, dass die Message oder der Empfänger geändert werden soll, muss der ursprüngliche Job gelöscht und ein neuer angelegt werden (siehe Tabelle 19).

TABELLE 19: MODIFYJOB MESSAGE

urn:p23r/main/scheduling/action/modifyjob		
Parameter	Typ	Beschreibung
Incoming		
de.p23r.main.param.job.identifier	String	Identifiziert den zu modifizierenden Job.
de.p23r.main.param.job.definition	String	Neue Beschreibung. Sie muss vom selben Typ wie die ursprüngliche sein.
Outgoing		
de.p23r.main.param.job.error.code	int	Im Falle eines Fehlers enthält der Parameter einen Fehlercode.

urn:p23r/main/scheduling/action/modifyjob		
Parameter	Typ	Beschreibung
		1 – falscher Parameter 2 – fehlender Parameter
de.p23r.main.param.job.error.description	String	Textuelle Beschreibung des Fehlers. (optional)

cancelJob

Die Message enthält den Aktionswert canceljob im Header. Es erfolgt eine Rückmeldung. Sie enthält im Erfolgsfall nichts und im Fehlerfall einen entsprechenden Fehlercode. Diese Funktion ermöglicht das Löschen eines Auftrags anhand des Identifiers (siehe Tabelle 20).

TABELLE 20: CANCELJOB MESSAGE

urn:p23r/main/scheduling/action/canceljob		
Parameter	Typ	Beschreibung
Incoming		
de.p23r.main.param.job.identifier	String	Identifier des zu löschenden Jobs.
Outgoing		
de.p23r.main.param.job.error.code	int	Im Falle eines Fehlers enthält der Parameter einen Fehlercode. 1 – falscher Parameter 2 – fehlender Parameter
de.p23r.main.param.job.error.description	String	Textuelle Beschreibung des Fehlers. (optional)

3.10.6 ARTEFAKTE

Für die Komponente *Scheduling* entstehen folgende Artefakte:

- p23r.scheduler.esb-<version>.esb
JBoss ESB Service Komponente

3.11 BOOTSTRAPPING

Die P23R-Komponente *Bootstrapping* wird gemäß den Vorgaben des Pflichtenhefts [2] umgesetzt. Bei der *Scheduling* Komponente handelt es sich aus Implementierungssicht jedoch nicht um eine Komponente, sondern um verschiedenen Konfigurationsmöglichkeiten bzw. Prozesse, welche

P23R

P23R: Spezifikationen zur Infrastruktur

bestimmte Deployment-Einheiten des P23R initial und auch bei jedem weiteren Hochfahren innerhalb der Ablaufumgebung (JBoss Application Server) in einen einsatzbereiten Zustand versetzen.

3.11.1 STRUKTUR UND ARBEITSWEISE

Beim Hochfahren bestimmter Deployment-Einheiten des P23R innerhalb der JBoss Applikationsserverumgebung wird ein Bootstrapping Einsprungs-Punkt realisiert. Durch diese Einsprungspunkte werden die im Pflichtenheft [2] beschriebenen Bootstrapping-Prozesse durchgeführt. Dazu zählt insbesondere die Überprüfung der konfigurierten Dienst-Endpunkte.

Derzeit scheint dies für die folgenden Deployment-Einheiten angebracht:

- ModelAndRuleManagement (ESB, siehe Abschnitt 3.9)

Die in der Konfigurationseinheit „Leitstelle“ hinterlegten Dienst-Endpunkte werden überprüft, die Verfügbarkeit wird protokolliert und ein Aufbau / eine Aktualisierung des ModelAndRulePool erfolgt.

- NotificationDispatch (ESB, siehe Abschnitt 3.5)

Die in der Konfigurationseinheit „Quellsysteme“ hinterlegten Dienst-Endpunkte werden überprüft, die Verfügbarkeit wird protokolliert.

Die im Folgenden näher beschriebenen Konfigurationseinheiten werden von der Komponente *Bootstrapping* verarbeitet:

- Konfigurationseinheit „P23R-Leitstelle“

Konfiguration der Dienst-Endpunkte aus der entsprechenden TSL aus dem Trusted-Service-List-Depot der Leitstelle (siehe Kapitel 4).

- Konfigurationseinheit „P23R-Client“

Enthält Informationen zum Dienst-Endpunkte für IApprovalNotify (siehe Kapitel 5).

- Informationseinheit „P23R-Instanz“

Enthält Informationen zur unterstützenden Version der Rahmenarchitektur der installierten P23R-Instanz.

3.11.2 KONFIGURATION

Der URI der P23R-Leitstelle muss in der Konfigurationseinheit „P23R-Leitstelle“ korrekt hinterlegt sein.

3.11.3 FEHLERBEHANDLUNG

Folgende Fehler führen zu einem entsprechenden Protokolleintrag (siehe Abschnitt 3.11.4):

- Konfigurationseinheit „P23R -Leitstelle“ nicht vorhanden bzw. fehlerhaft
- Konfigurationseinheit „P23R-Client“ nicht vorhanden bzw. fehlerhaft

3.11.4 PROTOKOLLIERUNG

Folgende Aktivitäten werden über die Komponente *ProtocolPool* (siehe Abschnitt 3.8) protokolliert:

- Einlesen der Konfigurationseinheit „P23R-Leitstelle“
- Einlesen der Konfigurationseinheit „P23R-Client“
- Aufruf der Schnittstelle P23R-Leitstelle Trusted-Service-List-Depot
- Einlesen der P23R-Instanz-Informationen

3.11.5 SCHNITTSTELLEN

Die Komponente verfügt über keine internen Schnittstellen.

3.11.6 ARTEFAKTE

Es entstehen keine eigenständigen Artefakte.

3.12 IDENTITYPROVIDER

Die Implementierung des P23R-Identity-Providers wird über die Framework-Funktionalität von Metro mit der Sicherheitsbibliothek WSIT [11] realisiert. Das Framework sieht vor, dass an den Schlüsselstellen der Authentifizierung, der Attributabfrage und der Ausstellung der SAML-Assertion eine eigene Implementierung in den Kontrollfluss des Frameworks eingefügt werden kann.

3.12.1 STRUKTUR UND ARBEITSWEISE

In der P23R-Musterimplementierung wird die Authentisierung basierend auf einer SAML-Assertion (Holder-of-Key) in Form eines Garantor-Tokens realisiert.

Der P23R-Identity-Provider überprüft, ob das Garantor-Token noch gültig ist und ob es von einem vertrauten Garantor Token Service ausgestellt wurde. Ist dies erfolgreich, werden in einem weiteren Schritt die Attribute eines Benutzers aus dem Garantor-Token ausgelesen, um sie bei der Ausstellung der P23R-Identity-Assertion einfließen lassen zu können.

Die P23R-Identity-Assertion wird im letzten Schritt gemäß Abschnitt 4.2 in der P23R-Sicherheitsarchitektur [12] mit den entsprechenden Attributen ausgestellt.

3.12.2 FEHLERBEHANDLUNG

Folgende Fehler verhindern das erfolgreiche Ausstellen einer P23R-Identity-Assertion:

- Das Garantor-Token ist nicht mehr gültig.
- Das Garantor-Token wurde nicht von einem vertrauten Garantor Token Service ausgestellt.
- Der Dienst, für den die P23R-Identity-Assertion ausgestellt werden soll, ist nicht als vertrauter Service Provider registriert.
- Die Nachricht wurde nicht gemäß über die WS-SecurityPolicy konfigurierten Sicherheitsmaßnahmen versendet.

3.12.3 PROTOKOLLIERUNG

Die folgenden Aktivitäten werden alternativ während der Verarbeitung der Anfragenachricht zur Ausstellung einer P23R-Identity-Assertion im Server-Log protokolliert:

- Das Garantor-Token ist nicht mehr gültig.
- Attribute konnten für die Ausstellung der P23R-Identity-Assertion nicht beschafft werden.

P23R

P23R: Spezifikationen zur Infrastruktur

3.12.4 KONFIGURATION

Die Konfiguration der Sicherheitsmechanismen für den Serviceaufruf des P23R-Identity-Providers sowie das Verhalten bei der Ausstellung der P23R-Identity-Assertion wird über WS-SecurityPolicy in dem WSDL-Dokument des P23R-Identity-Providers umgesetzt. Die Vertraulichkeit und Integrität der Nachricht werden über die WS-Policy *IdentityProvider_Policy* mit Hilfe von X.509 Zertifikaten sichergestellt. Über das Element *STSConfiguration* lassen sich Gültigkeitsdauer des Tokens sowie die Dienste, für die ein Token ausgestellt werden soll, konfigurieren.

3.12.5 SCHNITTSTELLEN

Der P23R-Identity-Provider stellt für Benutzer die Methode *authenticate* bereit, um sich basierend auf einem Guarantor-Token eine P23R-Identity-Assertion ausstellen zu lassen.

authenticate

Die Anfrage und Antwort zur P23R-Identity-Assertion erfolgt über folgende SOAP Actions:

- RequestSecurityToken:
`http://schemas.xmlsoap.org/ws/2005/02/trust/RST/Issue`
- RequestSecurityTokenResponse:
`http://schemas.xmlsoap.org/ws/2005/02/trust/RSTR/Issue`

3.12.6 ARTEFAKTE

Für die Komponente *IdentityProvider* entstehen folgende Artefakte:

- `p23r.identityprovider.ws-<version>.war`
Enthält die Implementierung des P23R-IdentityProviders zum Deployen auf einen Applikationsserver.

3.13 POLICY ENFORCEMENT POINT (PEP)

Bevor der Webservice-Endpunkt aufgerufen werden kann, werden der vorgelegte Authentisierungsnachweis sowie die Autorisierung für den Dienst überprüft. Dazu wurde das WSDL-Dokument der relevanten Dienste mit einer WS-Policy ausgestattet, welche die Vorlage einer SAML-Assertion des P23R-IdentityProviders sowie die Signierung und Verschlüsselung des SOAP-Bodys vorschreibt.

In der P23R-Musterimplementierung werden folgende Dienste über einen PEP gesichert:

- `IMessageDeliverLocal` aus `p23r.messagereceipt.ws`
- `INotificationApprove` aus `p23r.notificationdispatch`

3.13.1 STRUKTUR UND ARBEITSWEISE

Die Signatur und die Entschlüsselung werden durch WSIT [11] umgesetzt. Der Applikationsserver JBoss nutzt hierzu als JBossWS-Komponente (*jbossws-metro-3.3.1.GA*) die Version 1.3.1 von WSIT.

Die Validierung der SAML-Assertion sowie die Prüfung der Autorisierung erfolgt nach dem Muster der eingehenden Nachrichten der P23R-Sicherheitsarchitektur [12].

In der Implementierung der zu sichernden Webservices werden per Java-Annotation die SOAP-Handler *SAMLAssertionValidityPeriodVerificationHandler* sowie *SecurityHandler* registriert.

Der erste Handler überprüft, ob die Assertion zeitlich gültig ist, der zweite hingegen, ob die Autorisierung für den Zugriff auf die Webservice-Implementierung gegeben ist.

3.13.2 SCHNITTSTELLEN

Die Security-Applikationslogik ist vollständig in der Komponente *p23r.pep* integriert und wird der Implementierung des Webservices als Java-Bibliothek zur Verfügung gestellt. Die Prüfung der Sicherheitsobjekte passiert intern innerhalb des Applikationsservers, allerdings werden XACML-Policies vom Policy Administration Point *ent.pap* (siehe Abschnitt 5.4) abgefragt.

3.13.3 FEHLERBEHANDLUNG

Die folgenden Fehler werden vom Policy Enforcement Point und seinen Subkomponenten verarbeitet:

- Der XACML-Context-Request konnte nicht erstellt werden.
- Der Policy Administration Point ist nicht erreichbar.
- Der SOAP Response von einem Policy Administration Point enthält kein XACML-Policy-Statement.
- Die OpenSAML-Bibliothek [13] konnte nicht initialisiert werden.

3.13.4 PROTOKOLLIERUNG

Die folgenden Ereignisse vom Policy Enforcement Point protokolliert:

- Zugriff gewährt
- Zugriff verweigert für Subjekt ID
 - Anfrageparameter für eine Autorisierungsentscheidung:
 - Subject ID
 - Subject Role
 - Web Service Name
 - Web Service Port
 - Ressourcen ID
 - Action ID

3.13.5 KONFIGURATION

In der Datei „pdp.properties“ kann die URL für den Policy Administration Point konfiguriert werden.

3.13.6 ARTEFAKTE

Für die Komponenten *MessageReceipt* und *NotificationDispatch* entstehen zusätzlich folgende Artefakte:

P23R

P23R: Spezifikationen zur Infrastruktur

- ent.pep-<version>.jar

Diese Java-Bibliothek enthält die Applikationslogik zur Prüfung der *P23R-Identity-Assertion*, zur Durchsetzung einer Autorisierungsentscheidung, zur Evaluierung einer Autorisierungsentscheidung sowie zum Abruf von XACML-Policies.

3.14 EXTENSIONS

Die T-BRS [5] sieht die Möglichkeit vor, P23R-spezifische Funktionen in Selektions- und Transformationskripten zu verwenden. Diese Erweiterungsfunktionen werden in der Bibliothek *p23r.extensions* implementiert.

Die Bibliothek wird insbesondere beim Generieren von Benachrichtigungen zur Laufzeit verwendet, kann aber ebenfalls in XML-Entwicklungstools, die Saxon als Transformationsframework einsetzen, verwendet werden.

3.14.1 STRUKTUR UND ARBEITSWEISE

Die Bibliothek implementiert den Saxon-spezifischen Weg, um die XQuery- und Transformationsengine durch eigene Funktionen zu erweitern. Die Funktionen delegieren den Aufruf an interne Komponenten als auch zu externen Diensten über Webservice-Schnittstellen, um Ergebnisse zu liefern.

3.14.2 SCHNITTSTELLEN

Folgende Methoden werden als Erweiterungsfunktion umgesetzt:

- `selectDataForNamespace(String namespace, String selektion, String parameter): String`
Ruft den dem Namespace zugeordneten *Quelldatenkonnektor* (siehe Kapitel 6) auf.
- `queryReceivers(Criteria criteria, String notificationRuleId): Receivers`
Ermittelt die Empfänger durch Aufrufen des *Zuständigkeitsverzeichnisses* (siehe Abschnitt 4.2).
- `protocolWrite(Enum type, NotificationProfile context, String message, Integer errorCode): void`
Sendet einen Protokolleintrag an den *ProtocolPool* (siehe Abschnitt 3.8).
- `stopGeneration(String transactionId, String notificationId): void`
Signalisiert der *Generation-Pipeline* (siehe Abschnitt 3.4) nach der Abarbeitung des Skripts die Generierung abubrechen.

3.14.3 FEHLERBEHANDLUNG

Fehler werden intern abgefangen und es wird ein leeres Ergebnis geliefert. Die Vorgehensweise zur Erweiterung von XQuery und XSLT durch eigene Methoden bietet keinen adäquaten Mechanismus, um explizit Fehlerfälle zu behandeln. Damit die Ausführung des Skripts aber in jedem Fall fortgeführt werden kann, wird das leere Ergebnis als Fehlerindikation herangezogen. Hierbei hängt es vom Einzelfall ab, ob ein leeres Ergebnis als Fehler erkannt werden kann oder ob auch ein leeres Ergebnis ein gültiger Rückgabewert ist. Fehler werden jedoch mitgelogged.

3.14.4 PROTOKOLLIERUNG

Es wird nicht implizit protokolliert. Protokolleinträge können jedoch über die Funktion `protocolWrite` explizit aus einem Skript heraus vorgenommen werden.

3.14.5 KONFIGURATION

Die Extension-Bibliothek wird über die Konfigurationsdatei `~/.p23r/p23rextconfig.xml` konfiguriert. Sie enthält das Mapping zwischen Namespaces und Quelldatenkonnektor. Für den Aufruf des *Zuständigkeitsverzeichnisses* (siehe Abschnitt 4.2) wird die URL der TSL angegeben sowie der Name, unter dem der Dienst in der TSL zu finden ist. Im Folgenden ist eine `p23rextconfig.xml` abgebildet.

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
  <competencecentre>
    <tsl>
      <url>http://${cc.address}/${tsl.name}</url>
    </tsl>
    <notificationreceiverquery>
      <endpoint>
        <tslservicename>NotificationReceiverQuery</tslservicename>
        <service>INotificationReceiverQueryPorttype</service>
      </endpoint>
    </notificationreceiverquery>
  </competencecentre>
  <connector>
    <namespace>http://leitstelle.p23r.de/NS/TBR/TEST/Pivot/Mitarbeiter</namespace>
    <url>http://${ent.address}/ent.sourceconnector/SourceDataQueryService</url>
    <serviceName>ISourceDataQuery</serviceName>
  </connector>
  <connector>
    <namespace>http://leitstelle.p23r.de/NS/TBR/TEST/Pivot/Unternehmen</namespace>
    <url>http://${ent.address}/ent.sourceconnector/SourceDataQueryService</url>
    <serviceName>ISourceDataQuery</serviceName>
  </connector>
  <connector>
    <namespace>http://leitstelle.p23r.de/NS/p23r/nrl/Configuration1-0</namespace>
    <url>xml:db:exist:///db/configuration.xml</url>
  </connector>
</configuration>
```

3.14.6 ARTEFAKTE

Für das Projekt `p23r.extension` entsteht folgendes Artefakt:

- `p23r.extensions-<version>.jar`

Java-Bibliothek mit Implementierung für die Erweiterungsfunktionen auf Basis von Saxon

P23R

P23R: Spezifikationen zur Infrastruktur

4 SPEZIFIKATION DER P23R-LEITSTELLE

Die P23R-Komponente *ControlCentre* (P23R-Leitstelle) wird gemäß den Vorgaben des Pflichtenhefts [1] umgesetzt und in Form einer Laborleitstelle instanziiert. Die *ControlCentre* Komponente wird durch das Projekt mit der Bezeichnung *cc.controlcentre.portal* realisiert.

Zu den prinzipiell umzusetzenden Funktionsbestandteilen des *ControlCentre* zählen:

- Verwaltung von Datenmodell- und Benachrichtigungsregelpaketen, der TSLs sowie der Kataloge und Verzeichnisse des Zuständigkeitsverzeichnisses
- Bereitstellung eines P23R-Leitstellenportals, welches entsprechend autorisierten Anwendern die Verwaltung der nachfolgend aufgezählten Bereiche erlaubt:
 - P23R-Paket-Depot
 - P23R-Trusted-Service-List-Depot
 - P23R-Zuständigkeitsverzeichnis

Das *ControlCentre* ist zudem in der Lage, eine zu sich passende TSL beim ersten Start des *ControlCentre* an Hand seiner Konfiguration zu erzeugen. Es wird hierzu beim Starten geprüft, ob bereits eine entsprechende Liste vorliegt oder ob eine generiert und im TSL-Depot abgelegt werden muss.

4.1 PORTAL UND PERSISTENCE

Die Implementierung des P23R-Leitstellenportals erfolgt auf Basis des JBoss Seam Frameworks [14], wobei das P23R-Leitstellenportal als Web Applikation auf einem JBoss Application Server [15] bereitgestellt wird.

Im Kern besteht das P23R-Leitstellenportal aus Datenbank-Entities, die auf eine relationale Datenbank über die JPA zugreifen und deren Inhalt auf Java Server Faces (JSF) basierenden Webseiten dargestellt wird, welche mit Hilfe der JSF RichFaces [16] Bibliothek gestaltet werden.

Die Verwaltung der Datenbank-Entities erfolgt im Wesentlichen über die gemeinsam genutzte Bibliothek des Model- und Rulemanagements (siehe Abschnitt 2.1).

Für die Zugriffe auf die im *ControlCentre* verwalteten Datenmodell- und Benachrichtigungsregelpakete werden RESTful Webservices eingesetzt. Diese werden mit Hilfe von JBoss RESTEasy umgesetzt.

Zur Speicherung der Modell- und Regelpakete sowie Paketlisten wird eine relationale Datenbank verwendet. Der Zugriff erfolgt über den Java EE Standard JSR 220 JPA [3]. Somit kann je nach Anforderung an die Zielumgebung ein adäquates Produkt zum Einsatz kommen.

4.1.1 STRUKTUR UND ARBEITSWEISE

Das P23R-Leitstellenportal bietet folgende Funktionalitäten an:

- Verwaltung des P23R-Depots
 - Dient dazu Datenmodell- und Benachrichtigungsregelpakete zum *ControlCentre* hochzuladen und aus den hochgeladenen Paketen eine Paketliste zu erzeugen. Außerdem kann man

P23R

P23R: Spezifikationen zur Infrastruktur

den aktuellen Zustand des P23R-Depots (als Übersicht aller enthaltenen Pakete und Paketlisten) einsehen.

- **Verwaltung des Zuständigkeitsverzeichnis**
Dient dazu die Tabellen des Zuständigkeitsverzeichnisses jeweils einzeln zu bearbeiten bzw. den aktuellen Inhalt der einzelnen Kataloge und Verzeichnisse einzusehen.
- **Benutzerverwaltung**
Dient dazu neue Benutzer anzulegen, bestehende zu modifizieren oder zu löschen. Eine Zuordnung von Benutzern zu Rollen ist nicht vorgesehen. Es existiert aktuell nur die Rolle Administrator.

4.1.2 FEHLERBEHANDLUNG

Da es sich beim P23R-Leistellenportal um eine Oberflächenkomponente handelt, werden Fehler bis zum User-Interface propagiert.

4.1.3 KONFIGURATION

Konfigurationsinformationen werden in der Konfigurationsdatei `~/.p23r/serverconfig.xml` hinterlegt, die ausschnittsweise wie folgt beschrieben ist:

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
  <depot>
    <url>http://p23rdevcc:8080/portal/modelandruldepot</url>
  </depot>
  <cofi>
    <notificationreceiver-
url>http://p23rdevcc:8080/cc.controlcentre.competencefinder.inquiry/INotificationReceiverQuer
yService</notificationreceiverurl>
    <address-
queryurl>http://p23rdevcc:8080/cc.controlcentre.competencefinder.iadquery/IAddressQueryService
</addressqueryurl>
  </cofi>
  <tsl>
    <location>/opt/jboss-5.1.0.GA/server/default/deploy/ROOT.war</location>
    <name>p23r.tsl</name>
    <address>http://p23rdevcc:8080/p23r.tsl</address>
  </tsl>
  <keystore>
    <resource>TSL_SIGx.pfx</resource>
    <password>Test1234</password>
  </keystore>
</configuration>
```

4.1.4 SCHNITTSTELLEN

Das P23R-Benachrichtigungsregelpaket-Depot stellt RESTful Webservices gemäß den Vorgaben des Pflichtenhefts [2] zur Verfügung.

Im Einzelnen werden folgende Schnittstellen realisiert:

- <https://leitstelle.p23r.de/tsl/<id>>
- <https://leitstelle.p23r.de/portal/modelandrudedepot/rulepackage/<id>>
- <https://leitstelle.p23r.de/portal/modelandrudedepot/modelpackage/<id>>
- <https://leitstelle.p23r.de/portal/modelandrudedepot/packagelist/<release>>
- https://leitstelle.p23r.de/portal/modelandrudedepot/packagelist/latest_release

Diese dienen dazu, die in der Leitstelle verwalteten Regelpakete, Modellpakete und Regelpaketlisten abzufragen. Mit dem letzten Aufruf erhält man die zuletzt angelegte Regelpaketliste zum Download. Bei den anderen Aufrufen muss man den Identifier bzw. den Releasenamen als Parameter übergeben.

4.1.5 ARTEFAKTE

Für die Komponenten *ControlCentre* entstehen folgende Artefakte:

- `cc.controlcentre.portal-<version>.war`
Dabei handelt es sich um eine JBoss Seam Web-Anwendung. Sie enthält die Funktions- und Oberflächenbestandteile des P23R-Depots des TSL-Depots sowie die REST-Schnittstellen.

4.2 COMPETENCEFINDER

Der *CompetenceFinder* (P23R-Zuständigkeitsverzeichnis) ist die Basisdatenquelle für die Verknüpfungsinformationen bei der Ermittlung von Benachrichtigungsempfängern und Kommunikationskanälen. Für den Komplex an Katalogen, Verzeichnissen und Diensten wird in diesem Kontext nachfolgend der Begriff *CompetenceFinder* analog zur Verwendung in Rahmenarchitektur und Pflichtenheft genutzt.

Die Implementierung des *CompetenceFinder* erfolgt auf Basis des JBoss Frameworks. Der *CompetenceFinder* stellt hierzu Webservices auf einem JBoss Application Server [15] bereit.

Im Kern besteht der *CompetenceFinder* aus Zugriffsdiensten auf verschiedene relationale Datenbanken über eine Hibernate Persistenzschicht. Die Webservices verknüpfen die Abfragen bei den verschiedenen Datenbanken basierend auf den Mappinginformationen der Zugriffsdienste auf das Verzeichnis der Zuständigkeiten.

Der *CompetenceFinder* wird durch folgende Maven-Projekte (mit den entsprechenden Unterprojekten) realisiert:

- `cc.controlcentre.competencefinder`
Zentrales Maven-Projekt für den *CompetenceFinder*, bestehend aus den folgenden Maven Unterprojekten:
 - `cc.controlcentre.competencefinder.inquery`
Maven-Projekt mit allen Funktionalitäten für den Webservice zur Ermittlung der Benachrichtigungsempfänger.
 - `cc.controlcentre.competencefinder.iadquery`
Maven-Projekt mit allen Funktionalitäten für den Webservice zur Ermittlung der Kommunikationskanäle.

P23R

P23R: Spezifikationen zur Infrastruktur

- `cc.controlcentre.competencefinder.nrfinder`
Maven-Projekt mit allen Funktionalitäten für den Zugriff auf die Verknüpfungsinformationen in MappingReceiver.
- `cc.controlcentre.competencefinder.nrfinder`
Maven-Projekt mit allen Funktionalitäten für den Zugriff auf die Verknüpfungsinformationen in MappingEservice.
- `cc.controlcentre.competencefinder.scatalogue`
Maven-Projekt mit allen Funktionalitäten für den Zugriff auf den Leistungskatalog.
- `cc.controlcentre.competencefinder.ccatalogue`
Maven-Projekt mit allen Funktionalitäten für den Zugriff auf den Kriterienkatalog.
- `cc.controlcentre.competencefinder.nrdirectory`
Maven-Projekt mit allen Funktionalitäten für den Zugriff auf das Verzeichnis der Benachrichtigungsempfänger.
- `cc.controlcentre.competencefinder.esdirectory`
Maven-Projekt mit allen Funktionalitäten für den Zugriff auf das elektronische Dienstverzeichnis.
- `cc.controlcentre.competencefinder.ccatalogue`
Maven-Projekt mit Unterstützungsfunktionalitäten für den CompetenceFinder.

Im *root* werden jeweils Dienste vorgehalten. Bei den Zugriffsdiensten werden im Hibernate Ordner unter *dao* die Zugriffsklassen für die einzelnen Tabellen und unter *object* die Java Entities, welche die Tabellen repräsentieren, vorgehalten.

4.2.1 STRUKTUR UND ARBEITSWEISE

Die inhaltliche Ausgestaltung findet sich detailliert im Pflichtenheft [2] wieder. Nachfolgend wird die methodische Vorgehensweise dargestellt.

Die beiden Schnittstellen nehmen die Anfragen entgegen und prüfen zuerst auf Fehler, die eventuell in der Anfrage vorkommen könnten. Danach werden die Verknüpfungsinformationen aus dem *CompetenceFinder* über die entsprechenden Zugriffsdienste abgefragt. Zum Abschluss werden die Ergebnisinformationen aus dem Zielverzeichnis über den entsprechenden Zugriffsdienst abgefragt und in das Java Objekt für den Webservice transformiert.

Die Zugriffsdienste nutzen jeweils die *dao*-Klassen und bauen aus den Java Objekten, welche die Tabellen repräsentieren, das Gesamtobjekt für den Katalog bzw. das Verzeichnis auf.

Die Zugriffsdienste bieten darüber hinaus Methoden für das Speichern und Löschen von Objekten in den Katalogen und Verzeichnissen. Dafür nutzen sie ebenfalls die *dao* Klassen.

4.2.2 SCHNITTSTELLEN

Der *CompetenceFinder* stellt eine Reihe von Webservices zur Verfügung, die als externe Webservices realisiert werden.

4.2.2.1 SCHNITTSTELLE INOTIFICATIONRECEIVERQUERY (EXTERN, EINGEHEND)

Die Schnittstelle *INotificationReceiverQuery* muss vom *CompetenceFinder* als Webservice-Schnittstelle entsprechend der in der Rahmenarchitektur spezifizierten WSDL implementiert werden. Der Webservice erhält eine Eingangsnachricht vom Typ *queryReceiver*. Es wird eine Ausgangsnachricht vom Typ *queryReceiverResponse* erwartet.

4.2.2.2 SCHNITTSTELLE IADDRESSQUERY (EXTERN, EINGEHEND)

Die Schnittstelle *IAddressDiscoverer* muss vom *CompetenceFinder* als Webservice-Schnittstelle entsprechend der in der Rahmenarchitektur spezifizierten WSDL implementiert werden. Der Webservice erhält eine Eingangsnachricht vom Typ *queryAddress*. Es wird eine Ausgangsnachricht vom Typ *queryAddressResponse* erwartet.

4.2.3 DATENBESTAND, PERSISTENZ

Zur Speicherung der Inhalte des *CompetenceFinder* werden relationale Datenbanken verwendet. Jeder Katalog und jedes Verzeichnis werden in einzelnen Datenbanken gehalten, um die Nachhaltigkeit zu erhöhen. Für die P23R-Musterimplementierung wird eine MySQL-Datenbank verwendet.

Der Zugriff erfolgt über Hibernate mit entsprechendem Mapping und Konfigurationen. Durch Anpassung der Konfiguration kann somit je nach Anforderung an die Zielumgebung ein adäquates Produkt zum Einsatz kommen.

Folgende Ressourcen werden in den Projekten bereitgestellt:

- sql - SQL Skripte zum Generieren der Datenbank inklusive Beispiel- und Pilotdaten
- hibernate - Hibernate Konfiguration für die Anbindung der Datenbank
- hibernate/mapping – Hibernate Mapping für die Tabellen zu den Java Entities

4.2.4 FEHLERBEHANDLUNG

Folgende Fehler können innerhalb des *CompetenceFinder* auftreten und werden durch Protokolleinträge repräsentiert:

- Fehler beim Zugriff auf die Persistenzschicht
- Unvorhergesehener Fehler zur Laufzeit
- Fehler bei den Anfragen über die Webservices, welche auch zu einem P23RAppFault führen.

4.2.5 PROTOKOLLIERUNG

Der *CompetenceFinder* protokolliert die Zugriffe auf die Kataloge und Verzeichnisse mit Speichern und Löschen sowie das Abfragen von Inhalten. Darüber hinaus werden Fehler protokolliert. Bei der Protokollierung werden keine Informationen über den Anfrager festgehalten.

4.2.6 KONFIGURATION

Konfigurationsinformationen werden zur Datenbankverbindung (Hibernate) und zum Logging bereitgestellt. Zentral werden beim Deployment die Kommunikationsadressen der Webservices konfiguriert. Für die Auswahl der multilingualen Textelemente ist in der deutschen Realisierung Deutsch als Primär- und Englisch als Sekundärsprache fest verankert.

P23R

P23R: Spezifikationen zur Infrastruktur

4.2.7 ARTEFAKTE

Für den *CompetenceFinder* als Bestandteil der P23R-Leitstelle entstehen die folgenden Artefakte:

- cc.controlcentre.competencefinder.inrquery-<version>.war
Webservice Schnittstelle zur Ermittlung der Benachrichtigungsempfänger
- cc.inotificationreceiverquery.wslib-<version>.jar
Webservice Bibliothek mit generierten Klassen für die Implementierung des Services für die Ermittlung der Benachrichtigungsempfänger
- cc.controlcentre.competencefinder.iadquery-<version>.war
Webservice Schnittstelle zur Ermittlung der Kommunikationskanäle
- cc.iaddressquery.wslib-<version>.jar
Webservice Bibliothek mit generierten Klassen für die Implementierung des Services für die Ermittlung der Kommunikationskanäle
- cc.controlcentre.competencefinder.scatalogue-<version>.jar
Bibliothek mit Klassen für den Zugriff auf den Leistungskatalog
- cc.controlcentre.competencefinder.ccatalogue-<version>.jar
Bibliothek mit Klassen für den Zugriff auf den Kriterienkatalog.
- cc.controlcentre.competencefinder.nrdirectory-<version>.jar
Bibliothek mit Klassen für den Zugriff auf das Verzeichnis der Benachrichtigungsempfänger
- cc.controlcentre.competencefinder.esdirectory-<version>.jar
Bibliothek mit Klassen für den Zugriff auf das elektronische Dienstverzeichnis
- cc.controlcentre.competencefinder.common-<version>.jar
Bibliothek mit Utility Klassen für den CompetenceFinder

5 SPEZIFIKATION DES P23R-CLIENT

Das *portal* (P23R-Client) wird gemäß den Vorgaben des Pflichtenhefts [2] als Onlineanwendung umgesetzt. Darüber hinaus gehört noch eine Reihe von Sicherheitskomponenten „logisch“ zum P23R-Client und wird daher im folgenden Kontext aufgeführt.

5.1 COMMON USER-PROFILE

Die Komponente *ent.common.userprofile* (Benutzerprofil) realisiert den gemeinsamen Zugriff auf Benutzerinformationen für das Portal (siehe Abschnitt 5.2) und die Sicherheitskomponenten (siehe Abschnitt 5.3 und Abschnitt 5.4).

5.1.1 STRUKTUR UND ARBEITSWEISE

Das *ent.common.userprofile* persistiert die zu einem Benutzer gehörenden Merkmale, wie Benutzername, Organisationszugehörigkeit in einer Datenbank und ermöglicht den gemeinsamen Zugriff auf diese Benutzerinformationen.

5.1.2 SCHNITTSTELLEN

Die Funktion der Bibliothek wird durch die Klasse *Userprofile* bereitgestellt. Die Klasse bietet folgende Methoden als Schnittstelle:

- `getUsers(): Users`
Liefert die Liste aller User. Die Liste muss jedoch zuerst mit Hilfe von `loadUsers` aus der Datenbank geladen werden.
- `setUsers(Users users): void`
Erlaubt das initialisieren von Benutzern.
- `getUser(String username): User`
Liefert den Benutzer mit bestimmten Namen.
- `setUser(User user): void`
Setzt den aktuellen Benutzer.
- `getSecurity(): Security`
Liefert das Security Objekt für das aktuelle Userprofile. Das Security Objekt enthält Das Zertifikat und den Private Key des Benutzers.
- `verifyPassword(String clearTextPassword): boolean`
Überprüft ein gegebenes Passwort.
- `loadUsers(): void`
Lädt alle Userprofile aus der Datenbank
- `loadUser(String username): boolean`
Lädt ein bestimmtes Userprofile.

P23R

P23R: Spezifikationen zur Infrastruktur

- `removeUser(String username)`: boolean
Löscht ein Userprofile
- `createPassword(String clearTextPassword, String seed)`: String
Generiert ein Hash Wert für ein clear Text Passwort.

5.1.3 FEHLERBEHANDLUNG

Fehler werden soweit möglich intern abgefangen. Manche Fehler werden über Exceptions oder `null` als Rückgabewert an die aufrufende Anwendung weitergeleitet.

5.1.4 KONFIGURATION

Die Bibliothek wird über die Datei `~/p23r/clientconfig.xml` konfiguriert. Sie enthält die Zugriffsdaten für die eXist XML Datenbank. Neben der URL wird dort der Collection Name, die zugehörigen Credentials, sowie die zu verwendende Resource Id gelistet:

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
  <xmldb>
    <uri>xmldb:exist://localhost:8888/exist/xmlrpc/db</uri>
    <collections>
      <users>
        <name>P23R_CLIENT</name>
        <username>admin</username>
        <password>admin</password>
        <document>users.xml</document>
      </users>
    </collections>
  </xmldb>
</configuration>
```

5.1.5 ARTEFAKTE

Für das `ent.common.userprofile` als Bestandteil des P23R-Client entsteht das folgende Artefakt:

- `ent.common.userprofile-<version>.jar`
Bibliothek zum lesenden und schreibenden Zugriff auf die persistierten Benutzerinformationen.

5.2 PORTAL

Die Web Applikation `ent.portal` stellt das `portal` zur Verfügung. Die Implementierung des `portal` basiert auf dem JBoss Seam Framework [14].

Es wird als J2EE konforme Web-Applikation umgesetzt. Im Kern besteht das `portal` aus einer Datenbasis, die in einer eXist [17] XML-Datenbank untergebracht wird und über Java Server Faces (JSF) basierende Webseiten dargestellt und verwaltet werden kann.

Die Gestaltung des `portal` findet mit Hilfe der JSF RichFaces [16] Bibliothek statt und lehnt sich an das CI des Projekt „P23R“ an.

5.2.1 STRUKTUR UND ARBEITSWEISE

Das *portal* bietet verschiedene Interaktionsmöglichkeiten mit der P23R-Infrastruktur, die in den folgenden Kapiteln näher beschrieben werden. Die einzelnen Funktionalitäten basieren dabei direkt auf den Anwendungsfällen und Anforderungen der Lastenhefts [18].

5.2.1.1 BENUTZERVERWALTUNG

Die Benutzerverwaltung wird durch die gemeinsam mit den Sicherheitskomponenten des P23R-Client genutzte Komponente *ent.common.userprofile* (siehe Abschnitt 5.1) implementiert und im *portal* zugänglich gemacht.

Diese stellt folgende Funktionalität zur Verfügung:

- Verwalten von Benutzern (Anlegen, Ändern, Löschen, Aktivieren/Deaktivieren)
- Zuweisung des Benutzers zu einer Organisationseinheit
- Abspeichern eines Zertifikates für eine Dokumentensignatur

Die Zugriffe auf bestimmte Funktionen werden an Hand der definierten Rollen (bei Bedarf auch auf Benutzerebene) festgelegt.

5.2.1.2 NUTZDATENVERWALTUNG

Das *portal* bietet die Möglichkeit, Unternehmensdaten des *ent.sourceconnector* (siehe Kapitel 6.1 „SourceConnector“) zu initialisieren und zu verändern:

- Einspielen eines initialen Datensatzes aus einer dem Schema entsprechenden XML
- Kontinuierliches Verändern des Datensatzes bis dieser vom Quelldatenkonnektor überschrieben wird

5.2.1.3 REGELVERWALTUNG

Für die Regelverwaltung wird die Schnittstelle *IRuleActivate* des P23R implementiert. Das *portal* bietet folgende Funktionalitäten dieser Schnittstelle an:

- Manuelles Aktualisieren der Regelinformationen
- Anzeigen von Regelpaketinformationen, Regelpaketstatuten und des Regelpaket-Manifests.
- Anzeigen von Regelgruppeninformationen, Regelgruppenstatuten und des Regelgruppen-Manifests
- Anzeigen von Regelinformationen, Regelstatuten und des Regel-Manifests
- Selektieren und Deselektieren von Regelpaketen
- Aktivieren und Deaktivieren von Regelgruppen
- Aktivieren und Deaktivieren von Regeln
- Aktivieren und Deaktivieren der Funktion zur Automatischen Freigabe einer Regel

P23R

P23R: Spezifikationen zur Infrastruktur

Durch die Implementierung der Schnittstelle *IMessageDeliverLocal* wird folgende Funktionalität zur Verfügung gestellt:

- Manuelles Auslösen einer Regel

5.2.1.4 BENACHRICHTIGUNGSÜBERMITTLUNG

Für die Steuerung der Benachrichtigungsübermittlung kommen die Webservices *IApprovalNotify* und *INotificationApprove* zum Einsatz.

Dabei wird eine Weboberfläche zur Steuerung implementiert, die die freizugebende Benachrichtigung in Empfang nimmt, diese für den Benutzer zur Nachbearbeitung vorlegt und nach erfolgter Freigabe eine Meldung an den P23R zur Versandfreigabe übermittelt.

Falls der angemeldete Benutzer ein Zertifikat hinterlegt hat, so wird die Benachrichtigung bei der Freigabe zusätzlich noch mittels einer Dokumentensignatur signiert. Zusätzlich wird eine Option für einen Microsoft Excel Export und Import zur Verfügung gestellt.

5.2.1.5 EINSICHT IN DAS PROTOKOLL DES P23R

Das *portal* bietet durch die Implementierung der Schnittstelle *IProtocolQuery* die Möglichkeit das P23R Protokoll einzusehen und optional nach den Protokollfeldern Typ (Info oder Fehler) und Transactions-ID zu filtern.

5.2.2 SCHNITTSTELLEN

Das *portal* implementiert gemäß der Rahmenarchitektur die folgenden Webservices.

5.2.2.1 SCHNITTSTELLE IAPPROVALNOTIFY (EXTERN, AUSGEHEND)

Die Schnittstelle *IApprovalNotify* muss als Webservice-Schnittstelle entsprechend der in der Rahmenarchitektur spezifizierten WSDL implementiert werden. Der Webservice erhält eine Eingangsnachricht vom Typ *notifyApproval*. Es wird eine Ausgangsnachricht vom Typ *notifyApprovalResponse* erwartet.

5.2.3 DATENBESTAND, PERSISTENZ

Das *portal* muss neben den Unternehmensnutzdaten auch die Möglichkeit haben, Benachrichtigungen während des Freigabeprozesses zu speichern.

Durch die Tatsache, dass keine Rücksicht auf vorhandene Datenbestände in existierenden Technologien (z. B. relationalen Datenbanken) genommen werden muss, wird der Einfachheit halber der Datenbestand bereits im Pivot-Datenmodell Format (XML) verwaltet. Da auch die Benachrichtigungen als XML-Dokumente geliefert werden, bietet sich der Einsatz einer XML-Datenbank an.

Als XML-Datenbank kommt eXist-db [17] zum Einsatz. Der Zugriff erfolgt über die XML-DB API, die als Teil der eXist Datenbank zur Verfügung steht. Über die API ist es möglich, die Dokumente nicht nur als Ganzes abzuspeichern, sondern auch zu modifizieren, bzw. Selektionen effektiv auszuführen, auch wenn der Datenbestand sehr groß ist.

eXist strukturiert XML Dokumente hierarchisch in sogenannten „Collections“ und erlaubt eine genaue Zugriffsteuerung inklusive Schreib-, Lese- und Änderungsrechten für Benutzer und Benutzergruppen.

5.2.4 FEHLERBEHANDLUNG

Folgende Fehler können innerhalb des *portal* auftreten und führen zu einem Protokolleintrag (siehe Abschnitt 5.2.5):

- Unvorhergesehener Fehler zur Laufzeit
- Ausnahmen beim Zugriff auf die Datenbank

5.2.5 PROTOKOLLIERUNG

Das *portal* protokolliert bestimmte Ereignisse in eine spezielle Protokoll-Datei. Zu diesen Ereignissen zählen:

- Anmelden am Portal
- Anlegen eines neuen Nutzers
- Löschen eines Nutzers
- Ändern eines Nutzers
- Fehler zur Laufzeit
- Erhalt einer Benachrichtigung zur Freigabe
- Änderungen an einer Benachrichtigung
- Freigeben einer Benachrichtigung
- Ablehnen einer Benachrichtigung
- Wiederherstellen einer Benachrichtigung
- Selektieren/Deselektieren von Regelpacketen
- Aktivieren/Deaktivieren von Regelgruppen
- Deaktivieren/Aktivieren von Regeln
- Manuelles Auslösen einer Regel
- Einrichten und auslösen der Automatischen Freigabe
- Bearbeiten und Neuinstanten von Nutzdaten

5.2.6 KONFIGURATION

Konfigurationsinformationen werden zur Datenbankverbindung (eXistDB) bereitgestellt. Da sich das *portal* die Nutzdaten mit der Komponente *ent.sourceconnector*, sowie die Benutzerverwaltung die Komponente *ent.common.userprofile* mit anderen Komponenten teilt, werden hier ebenfalls entsprechende Einstellungen bereitgestellt. Zentral werden beim Deployment die Kommunikationsadressen der Webservices konfiguriert.

P23R

P23R: Spezifikationen zur Infrastruktur

5.2.7 ARTEFAKTE

Für das *portal* entsteht das folgende Artefakt:

- ent.portal-<version>.war

JBoss Seam Web-Anwendung, enthält auch die Funktions- und Oberflächenbestandteile der Portal-Komponenten.

5.3 GUARANTOR TOKEN SERVICE

Der Guarantor Token Service bildet einen unternehmensinternen Anmeldemechanismus. Das kann zum Beispiel eine passwortbasierte oder eine zertifikatsbasierte Authentisierung sein. Diese wird auf eine SAML Assertion, dem Guarantor-Token abgebildet. Der Guarantor Token Service ist als Webservice realisiert.

Erst mit einem Guarantor Token kann über den P23R-Identity-Provider des P23R eine Identity Assertion für die P23R-Fachdienste ausgestellt werden. Das erfordert ein direktes Vertrauensverhältnis des Guarantor Token Services eines Unternehmens mit dem P23R-Identity-Provider. Ferner hat jeder P23R-Fachdienst ein direktes Vertrauensverhältnis zum P23R-Identity-Provider.

5.3.1 ARBEITSWEISE

Die Implementierung des Guarantor Token Services wird über die Framework-Funktionalität von Metro mit WSIT [11] realisiert. Das Framework sieht vor, dass an den Schlüsselstellen der Anmeldevalidierung, der Attributabfrage und der Ausstellung der SAML-Assertion eine eigene Implementierung in den Kontrollfluss des Frameworks eingefügt werden kann.

In der Musterimplementierung wird der Authentisierungsmechanismus Benutzername/Passwort realisiert. In dem entsprechenden Callback-Handler werden die Credentials entgegengenommen und über das unternehmensinterne Benutzerverwaltungssystem (z. B. ein LDAP) validiert.

Ist dies erfolgreich, werden in einem weiteren Schritt die Attribute eines Benutzers aus der unternehmensinternen Benutzerverwaltung ausgelesen, um sie bei der Ausstellung des Guarantor-Tokens einfließen lassen zu können. Das Guarantor-Token wird schließlich analog zu Abschnitt 4.2 in [2] mit den entsprechenden Attributen ausgestellt.

5.3.2 SCHNITTSTELLEN

Der Guarantor Token Service stellt für Nutzer die Methode *authenticate* bereit, um sich mit seinen Credentials ein Guarantor-Token ausstellen zu lassen.

authenticate

Die Anfrage und Antwort zum Guarantor-Token erfolgt über folgende SOAP Actions:

- RequestSecurityToken:
<http://schemas.xmlsoap.org/ws/2005/02/trust/RST/Issue>
- RequestSecurityTokenResponse:
<http://schemas.xmlsoap.org/ws/2005/02/trust/RSTR/Issue>

5.3.3 SICHERHEITSKONFIGURATION

Die Konfiguration der Sicherheitsmechanismen für den Serviceaufruf des Guarantor Token Services, sowie das Verhalten bei der Ausstellung des Guarantor-Tokens wird über WS-SecurityPolicy in dem WSDL-Dokument des Guarantor Token Services umgesetzt. Die Vertraulichkeit und Integrität der Nachricht wird über die WS-Policy *P23rGuarantorTokenService_Policy* mit Hilfe von X.509 Zertifikaten sichergestellt. Über das Element *STSConfiguration* lassen sich Gültigkeitsdauer des Tokens sowie die Dienste, für die ein Token ausgestellt werden soll, konfigurieren.

5.3.4 FEHLERBEHANDLUNG

Folgende Fehler verhindern das erfolgreiche Ausstellen eines Guarantor-Tokens:

- Credentials können nicht validiert werden bzw. sind ungültig
- Der Dienst, für den das Guarantor-Token ausgestellt werden soll, ist nicht als vertrauter Service Provider registriert
- Die Nachricht wurde nicht gemäß über die WS-SecurityPolicy konfigurierten Sicherheitsmaßnahmen versendet

5.3.5 PROTOKOLLIERUNG

Die folgenden Ereignisse werden während der Verarbeitung einer Authentisierung protokolliert:

- Die Benutzername-/Passwort-Authentifizierung ist fehlgeschlagen
- Es können keine Attribute aus der Datenbank geladen werden (ent.common.userprofile)

5.3.6 KONFIGURATION

Für den Guarantor Token Service werden lediglich Hostname und Port der Sicherheitsdienste auf Unternehmensseite (Ablaufumgebung des Guarantor Token Services) sowie auf P23R-Seite in der Datei „p23r.properties“ konfiguriert.

5.3.7 ARTEFAKTE

Für die Komponente *GuarantorTokenService* entstehen folgende Artefakte:

- ent.guarantortokenservice.ws-<version>.war
Enthält die Implementierung des Guarantor Token Services zum Deployen auf einen Applikationsserver.

5.4 POLICY ADMINISTRATION POINT (PAP)

Der Policy Administration Point (PAP) lädt XACML Policies vom Dateisystem, welche zu einer Anfrage (*XACMLPolicyQuery*) passen und gibt sie dem PDP zurück, welcher in der Musterimplementierung im Dienst *IMessageDeliverLocal* eingesetzt wird. Der PAP wird im Unternehmen betrieben.

5.4.1 STRUKTUR UND ARBEITSWEISE

Technisch gesehen ist der PAP als Java Servlet umgesetzt. Anfragen von einem PDP führen dazu, dass in einem registrierten Verzeichnis (Policy Store) jede darin befindliche XACML Policy bzgl. der Anfrage hin ausgewertet wird, ob es eine Übereinstimmung hinsichtlich der Anfrage-Attribute gibt. Im Erfolgsfall wird die Policy an den Aufrufer zurückgesendet, sodass dieser eine Autorisierungsprüfung vornehmen kann.

P23R

P23R: Spezifikationen zur Infrastruktur

5.4.2 SCHNITTSTELLEN

Der Empfang der Nachrichten erfolgt über die externe Servlet-Schnittstelle mit dem gesonderten Pfad „policyquery“. In der P23R-Sicherheitsarchitektur [12] wurde das SAML-XACML-Protokoll für den Abruf definiert. Daher entsprechen der Versand einer Anfrage einer *XACMLPolicyQuery* und die Antwort eines *XACMLPolicyStatements*. In [12] wird das genutzte Format definiert.

Das folgende Code-Beispiel zeigt eine *XACMLPolicyQuery*.

```
<?xml version="1.0" encoding="UTF-8"?>
<soap11:Envelope
  xmlns:soap11="http://schemas.xmlsoap.org/soap/envelope/">
  <soap11:Body>
    <xacmlprof:XACMLPolicyQuery xmlns:xacmlprof=
"urn:oasis:names:tc:SAML:2.0:profiles:attribute:XACML"
      Destination="http://127.0.0.1:8090/ent.pap/policyquery"
      ID="8fc3ac52-91a0-4371-ab07-600cbd2d98c4"
      IssueInstant="2011-10-11T13:13:07.719Z"
      Version="2.0">
      <saml2:Issuer
        xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion"
        Format="urn:oasis:names:tc:SAML:2.0:nameid-format:entity">
http://p23r.de/policy-cache
      </saml2:Issuer>
      <xacml-context:Request
        xmlns:xacml-context=
"urn:oasis:names:tc:xacml:2.0:context:schema:os">
        <xacml-context:Subject
          SubjectCategory=
"urn:oasis:names:tc:xacml:1.0:subject-category:access-subject">
          <xacml-context:Attribute
            AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
            DataType="http://www.w3.org/2001/XMLSchema#string">
            <xacml-context:AttributeValue>erika.mustermann@p23r.de
            </xacml-context:AttributeValue>
          </xacml-context:Attribute>
          <xacml-context:Attribute
            AttributeId="urn:oasis:names:tc:xacml:2.0:subject:role"
            DataType="http://www.w3.org/2001/XMLSchema#string">
            <xacml-context:AttributeValue>Anwender
            </xacml-context:AttributeValue>
          </xacml-context:Attribute>
        </xacml-context:Subject>
        <xacml-context:Resource>
          <xacml-context:Attribute
            AttributeId="urn:oasis:names:tc:xacml:1.0:resource:resource-id"
            DataType="http://www.w3.org/2001/XMLSchema#string">
            <xacml-context:AttributeValue>1
```

```

        </xacml-context:AttributeValue>
    </xacml-context:Attribute>
    <xacml-context:Attribute
        AttributeId=
"urn:p23r:attributes:web-services:1.0:wSDL:1.1:service"
        DataType="http://www.w3.org/2001/XMLSchema#string">
        <xacml-context:AttributeValue>
{http://leitstelle.p23r.de/NS/P23R/IMessageDeliverLocal1-0}IMessageDeliverLocalService
        </xacml-context:AttributeValue>
    </xacml-context:Attribute>
    <xacml-context:Attribute
        AttributeId=
"urn:p23r:attributes:web-services:1.0:wSDL:1.1:port"
        DataType="http://www.w3.org/2001/XMLSchema#string">
        <xacml-context:AttributeValue>
{http://leitstelle.p23r.de/NS/P23R/IMessageDeliverLocal1-0}IMessageDeliverLocal
        </xacml-context:AttributeValue>
    </xacml-context:Attribute>
</xacml-context:Resource>
<xacml-context:Action>
    <xacml-context:Attribute
        AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id"
        DataType="http://www.w3.org/2001/XMLSchema#string">
        <xacml-context:AttributeValue>deliverMessage
        </xacml-context:AttributeValue>
    </xacml-context:Attribute>
</xacml-context:Action>
<xacml-context:Environment />
</xacml-context:Request>
</xacmlprof:XACMLPolicyQuery>
</soap11:Body>
</soap11:Envelope>

```

Das folgende Code-Beispiel zeigt ein *XACMLPolicyStatement*:

```

<?xml version="1.0" encoding="UTF-8"?>
<soap11:Envelope
    xmlns:soap11="http://schemas.xmlsoap.org/soap/envelope/">
    <soap11:Body>
        <saml2p:Response
            ID="3682ac93-01ba-4856-8899-3e1f96d98a95"
            IssueInstant="2011-10-11T13:13:09.893Z"
            Version="2.0"
            xmlns:saml2p="urn:oasis:names:tc:SAML:2.0:protocol">
            <saml2p:Status>
                <saml2p:StatusCode

```

```

Value=
"urn:oasis:names:tc:SAML:2.0:status:Success"></saml2p:StatusCode>
</saml2p:Status>
<saml2:Assertion
  ID="1782091d-ec59-4ee3-a3a8-64807e1113f5"
  IssueInstant="2011-10-11T13:13:09.892Z"
  Version="2.0"
  xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">
<saml2:Statement
  xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion"
  xmlns:xacml-saml=
"urn:oasis:names:tc:xacml:2.0:profile:saml2.0:v2:schema:assertion"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:type="xacml-saml:XACMLPolicyStatementType">
<xacml:PolicySet
  xmlns:xacml="urn:oasis:names:tc:xacml:2.0:policy:schema:os"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation=
"urn:oasis:names:tc:xacml:2.0:policy:schema:os http://docs.oasis-
open.org/xacml/access_control-xacml-2.0-policy-schema-os.xsd"
  PolicyCombiningAlgId=
"urn:oasis:names:tc:xacml:1.1:policy-combining-algorithm:ordered-permit-overrides"
  PolicySetId=
"urn:p23r:names:1.0:default:policysetid-msgreceiver01"
  Version="1.0">
<xacml:Description>
Sample policy set for P23R Message Receiver Service
</xacml:Description>
<xacml:Target>
<xacml:Resources>
<xacml:Resource>
<xacml:ResourceMatch
  MatchId=
"urn:oasis:names:tc:xacml:1.0:function:string-equal">
<xacml:AttributeValue
  DataType=
"http://www.w3.org/2001/XMLSchema#string">{http://leitstelle.p23r.de/NS/P23R/IMessageDeliverLo
call-0}IMessageDeliverLocalService
</xacml:AttributeValue>
<xacml:ResourceAttributeDesignator
  AttributeId=
"urn:p23r:attributes:web-services:1.0:wSDL:1.1:service"
  DataType="http://www.w3.org/2001/XMLSchema#string" />
</xacml:ResourceMatch>
<xacml:ResourceMatch
  MatchId=
"urn:oasis:names:tc:xacml:1.0:function:string-equal">

```

```

        <xacml:AttributeValue
            DataType=
"http://www.w3.org/2001/XMLSchema#string">{http://leitstelle.p23r.de/NS/P23R/IMessageDeliverLo
call1-0}IMessageDeliverLocal
        </xacml:AttributeValue>
        <xacml:ResourceAttributeDesignator
            AttributeId=
"urn:p23r:attributes:web-services:1.0:w3dl:1.1:port"
            DataType="http://www.w3.org/2001/XMLSchema#string" />
        </xacml:ResourceMatch>
    </xacml:Resource>
</xacml:Resources>
</xacml:Target>
<xacml:Policy
    PolicyId=
"urn:p23r:names:1.0:default:policyid:permit-msgservice01"
    RuleCombiningAlgId=
"urn:oasis:names:tc:xacml:1.0:rule-combining-algorithm:permit-overrides">
    <xacml:Target>
        <xacml:Actions>
            <xacml:Action>
                <xacml:ActionMatch MatchId=
"urn:oasis:names:tc:xacml:1.0:function:string-equal">
                    <xacml:AttributeValue
                        DataType=
"http://www.w3.org/2001/XMLSchema#string">deliverMessage</xacml:AttributeValue>
                    <xacml:ActionAttributeDesignator
                        DataType="http://www.w3.org/2001/XMLSchema#string"
                        AttributeId=
"urn:oasis:names:tc:xacml:1.0:action:action-id" />
                </xacml:ActionMatch>
            </xacml:Action>
        </xacml:Actions>
    </xacml:Target>
    <xacml:Rule
        RuleId="MessageReceiverPermitRule"
        Effect="Permit">
        <xacml:Condition>
            <xacml:Apply
                FunctionId="urn:oasis:names:tc:xacml:1.0:function:or">
                <xacml:Apply
                    FunctionId=
"urn:oasis:names:tc:xacml:1.0:function:string-at-least-one-member-of">
                    <xacml:SubjectAttributeDesignator
                        AttributeId=
"urn:oasis:names:tc:xacml:2.0:subject:role"
                        DataType="http://www.w3.org/2001/XMLSchema#string"

```

```
        SubjectCategory=
"urn:oasis:names:tc:xacml:1.0:subject-category:access-subject" />
        <xacml:Apply
            FunctionId=
"urn:oasis:names:tc:xacml:1.0:function:string-bag">
            <xacml:AttributeValue
                DataType=
"http://www.w3.org/2001/XMLSchema#string">Anwender
            </xacml:AttributeValue>
        </xacml:Apply>
    </xacml:Apply>
</xacml:Condition>
</xacml:Rule>
<!-- A final, "fall-through" rule that always denies -->
<xacml:Rule
    RuleId="MessageReceiverDenyRule"
    Effect="Deny" />
</xacml:Policy>
<!-- A final, "fall-through" policy that always denies -->
<xacml:Policy
    PolicyId="urn:p23r:names:1.0:default:policyid:deny-all"
    RuleCombiningAlgId=
"urn:oasis:names:tc:xacml:1.0:rule-combining-algorithm:deny-overrides">
    <xacml:Target />
    <xacml:Rule
        RuleId="urn:p23r:names:1.0:default:ruleid:deny-rule"
        Effect="Deny" />
    </xacml:Policy>
</xacml:PolicySet>
</saml2:Statement>
</saml2:Assertion>
</saml2p:Response>
</soap11:Body>
</soap11:Envelope>
```

5.4.3 FEHLERBEHANDLUNG

Beim Deployment bzw. Hochfahren des Applikationsservers wird die Bibliothek OpenSAML [13] initialisiert. Im Fehlerfall wird eine *java.lang.SecurityException* geworfen. Damit kann keine Nachricht empfangen werden. Während der Verarbeitung werden die Fehlerfälle folgender Szenarien abgedeckt:

- Dekodieren der HTTP SOAP 11 Nachricht
- Verpacken der Antwortnachricht als XACMLPolicyStatement
- Es können keine Policies aus dem Policy Store gelesen werden

5.4.4 PROTOKOLLIERUNG

Für die Policy-Aktivierung wird die *XACMLPolicyQuery* gegen dem PAP bekannte Policies ausgewertet. Der Versuch wird entsprechend protokolliert („PAP evaluating against PolicyXYZ.xml“). Es werden keine identifizierenden Daten der Anfrage im Server-Log protokolliert.

5.4.5 KONFIGURATION

Für das Aktivieren der Policies müssen mögliche Policies vom Policy Store geladen und geprüft werden. Der Policy Store ist dateibasiert implementiert, d. h. in einem vorgegebenem Verzeichnis sind alle Policies als XML-Dateien abgelegt. Dieses Verzeichnis kann in der Datei „p23r.properties“ in „ent.install“ konfiguriert werden.

5.4.6 ARTEFAKTE

Für das Projekt ent.pap der Komponente PAP entsteht folgendes Artefakt:

- ent.pap-<version>.war

Web-Komponente, enthält einen SAML Responder sowie den Policy Administration Point für die Realisierung der Servlet-Implementierung.

P23R

P23R: Spezifikationen zur Infrastruktur

6 SPEZIFIKATION DER QUELLDATENKONNEKTOREN

Die zu implementierenden Quelldatenkonnektoren sind eigenständig lauffähige Komponenten, die über eine Webservice-Schnittstelle anzusteuern sind. Sie können aus diesem Grunde auch auf andere Rechner ausgelagert werden. Eine weitreichendere Trennung zum P23R ist jedoch nicht ratsam, da die Beziehung nicht weiter in der Rahmenarchitektur abgesichert ist.

6.1 SOURCECONNECTOR

Der SourceConnector *ent.sourceconnector* stellt die Komponente zur P23R Musterimplementierung dar. Der Datenbestand des *sourceconnector* wird mit der Funktionalität des P23R-Clients „Unternehmensnutzdatenverwaltung (Stamm- und Bewegungsdaten)“ geteilt.

6.1.1 STRUKTUR UND ARBEITSWEISE

Durch Zurverfügungstellung der Webservice-Schnittstelle *ISourceDataQuery* wird die Funktion zur Selektierung der Unternehmensdaten aus den einzelnen Namensräumen bereitgestellt. Der *sourceconnector* kann hierbei eine beliebige Anzahl an Namensräumen bereitstellen insofern die entsprechenden Daten in der Datenbank vorhanden sind.

Er stellt aber nicht die Funktionalität bereit, Daten aus einem Vorsystem zu übernehmen. Der *sourceconnector* stellt sowohl einen Unternehmensdatenraum als auch die Funktionalität der Ausführung eines „Vorselektionskripts“ (*preselectionScript*) zur Verfügung.

6.1.1.1 DATENHALTUNG

Durch die Tatsache, nicht auf vorhandene Datenbestände in existierenden Technologien (z. B. relationalen Datenbanken) Rücksicht nehmen zu müssen, wird der Einfachheit halber der Datenbestand bereits im Pivot-Datenmodell Format (XML) verwaltet, da diese Daten in XML-Form vorliegen, weshalb sich der Einsatz einer XML Datenbank anbietet.

Als XML Datenbank kommt eXist-db [17] zum Einsatz. Der Zugriff erfolgt über die XML-DB API, die als Teil der eXist Datenbank zur Verfügung steht. Über die API ist es möglich, die Dokumente nicht nur als Ganzes abzuspeichern, sondern auch zu modifizieren, bzw. Selektionen effektiv auszuführen, auch wenn der Datenbestand sehr groß ist. eXist strukturiert XML-Dokumente hierarchisch in sogenannten „Collections“ und erlaubt eine genaue Zugriffssteuerung inklusive Schreib-, Lese- und Änderungsrechten für Benutzer und Benutzergruppen.

6.1.2 SCHNITTSTELLEN

Der *sourceconnector* implementiert gemäß der Rahmenarchitektur *ISourceDataQuery* Webservice für Selektierung und Lieferung von Unternehmensdaten.

6.1.3 FEHLERBEHANDLUNG

Folgende Fehler können innerhalb des *sourceconnector* auftreten und führen zu einem Protokolleintrag (siehe Abschnitt 6.1.4):

- Unvorhergesehener Fehler zur Laufzeit
- Ausnahmen beim Zugriff auf die Datenbank
- Implementierungsspezifische Fehler

P23R

P23R: Spezifikationen zur Infrastruktur

6.1.4 PROTOKOLLIERUNG

Der *sourceconnector* protokolliert bestimmte Ereignisse in eine spezielle Protokoll-Datei. Zu diesen Ereignissen zählen:

- Aufruf des *sourceconnector*
- Protokollierung von spezifizierten Fehlermeldungen (ungültiger Namensraum, ungültige Selektion)
- Fehler zur Laufzeit

Zusätzlich wird gemäß Rahmenarchitektur für folgende Fehler ein *P23RAppFault* innerhalb des WebServices generiert:

TABELLE 21: PROTOKOLLIEREN VON FEHLERMELDUNGEN

P23RAppFault	
Fehlerkennung	Beschreibung
SI001	Fehler zur Laufzeit
SD001	Fehlender oder ungültiger Parameter
SD002	Fehlende oder ungültige Selektion
SD003	Fehlender oder ungültiger Namensraum
SD004	Bei Aufruf der WebServiceMethode "getMirrorData"
SD005	Bei Aufruf der WebServiceMethod "isDataUpToDate"

6.1.5 KONFIGURATION

Konfigurationsinformationen werden zur Datenbankverbindung (eXistDB) bereitgestellt.

6.1.6 ARTEFAKTE

Für den *sourceconnector* entsteht das folgende Artefakt:

- ent.sourceconnector-<version>.war
JBoss Web-Anwendung, enthält auch die Funktionsbestandteile für WebService-Implementierung gemäß SOAP.

6.2 SOURCECONNECTOR.BASFSAP

Der *SourceConnector.BASFSap* *ent.sourceconnectorBASFSAP* stellt die Komponente für das P23R-Pilotierungsszenario der BASF da. Dieser empfängt einen periodischen Datenexport aus SAP R/3 und wandelt diesen in die Namensräume „Unternehmen“ und „Mitarbeiter“ um.

6.2.1 STRUKTUR UND ARBEITSWEISE

Durch Zurverfügungstellung der WebService-Schnittstelle *ISourceDataQuery* wird die Funktion zur Selektierung der Unternehmensdatennamensräume bereitgestellt.

Der *SourceConnectorBASFSAP* stellt hierbei die Namensräume „Unternehmen“ und „Mitarbeiter“ bereit. Diese Daten werden aus dem SAP/R3 Vorsystem der BASF importiert und in das entsprechende XML-Schema transformiert.

Hierfür wird in periodischen Zeitabständen ein „Transferverzeichnis“ auf Änderungen geprüft und wenn diese vorhanden sind, wird die exportierte SAP-Datenstruktur in den Mitarbeiternutzdatennamensraum transformiert.

Da über das auf SAP-Seite verwendete PU12 Programm keine Unternehmensdaten exportiert werden, wird der Unternehmensnutzdatennamensraum statisch im *SourceConnectorBASFSAP* hinterlegt (vgl. *SourceConnector*).

6.2.2 DATENHALTUNG

Als Zwischenspeicher auf Connectorseite kommt die eXist-db [17] zum Einsatz. Zwischen dem Vorsystem der DATEV und dem *SourceConnectorBASFSAP* - das zuvor erwähnte „Transferverzeichnis“.

6.2.3 SCHNITTSTELLEN

Der *SourceConnectorBASFSAP* implementiert gemäß der Rahmenarchitektur *ISourceDataQuery* Webservice für Selektierung und Lieferung von Unternehmensdaten.

6.2.4 FEHLERBEHANDLUNG

Folgende Fehler können innerhalb des *SourceConnectorBASFSAP* auftreten und führen zu einem Protokolleintrag (siehe Abschnitt 6.2.5):

- Unvorhergesehener Fehler zur Laufzeit
- Ausnahmen beim Zugriff auf die Datenbank
- Implementierungsspezifische Fehler

6.2.5 PROTOKOLLIERUNG

Der *SourceConnectorBASFSAP* protokolliert bestimmte Ereignisse in eine spezielle Protokoll-Datei. Zu diesen Ereignissen zählen:

- Aufruf des *SourceConnectorBASFSAP*
- Protokollierung von spezifizierten Fehlermeldungen (ungültiger Namensraum, ungültige Selektion)
- Fehler zur Laufzeit

Zusätzlich wird gemäß Rahmenarchitektur ein *P23RAppFault* innerhalb des *WebServices* generiert (siehe Abschnitt 6.1.4).

6.2.6 KONFIGURATION

Konfigurationsinformationen werden zur Datenbankverbindung (eXistDB), sowie zur Verbindung zum Transferverzeichnis bereitgestellt.

6.2.7 ARTEFAKTE

Für den *SourceConnectorBASFSAP* entsteht das folgende Artefakt:

P23R

P23R: Spezifikationen zur Infrastruktur

- ent.sourceconnectorBASFSAP-<version>.war

JBoss Web-Anwendung, enthält auch die Funktionsbestandteile für Webservice-Implementierung gemäß SOAP

6.3 SOURCECONNECTOR.DATEVRAW

Der *SourceConnector.DATEVRAW* *ent.sourceconnectorDATEVRAW* stellt die Komponente für das P23R-Pilotierungsszenario der DATEV dar. Dieser empfängt einen periodischen Datenexport aus den DATEV-Vorsystemen und wandelt diesen in die Namensräume „Unternehmen“ und „Mitarbeiter“ um.

6.3.1 STRUKTUR UND ARBEITSWEISE

Durch Zurverfügungstellung der Webservice-Schnittstelle *ISourceDataQuery* wird die Funktion zur Selektierung der Unternehmensdatennamensräume bereitgestellt.

Der *SourceConnectorDATEVRAW* stellt hierbei die Namensräume „Unternehmen“ und „Mitarbeiter“ bereit. Diese Daten werden aus dem DATEV Vorsystem als XML importiert und in das entsprechende Ziel-XML-Schema transformiert.

Hierfür wird in periodischen Zeitabständen ein „Transferverzeichnis“ auf Änderungen geprüft und, falls vorhanden, wird die exportierte DATEV-XML-Datenstruktur in den Mitarbeiternutzdatennamensraum und Unternehmensnutzdatennamensraum transformiert.

6.3.2 DATENHALTUNG

Als Zwischenspeicher auf Connectorseite kommt die eXist-db [17] zum Einsatz. Zwischen dem Vorsystem der DATEV und dem *SourceConnectorDATEVRAW* - das zuvor erwähnte „Transferverzeichnis“.

6.3.3 SCHNITTSTELLEN

Der *SourceConnectorDATEVRAW* implementiert gemäß der Rahmenarchitektur *ISourceDataQuery* Webservice für Selektierung und Lieferung von Unternehmensdaten.

6.3.4 FEHLERBEHANDLUNG

Folgende Fehler können innerhalb des *SourceConnectorDATEVRAW* auftreten und führen zu einem Protokolleintrag (siehe Abschnitt 6.3.5):

- Unvorhergesehener Fehler zur Laufzeit
- Ausnahmen beim Zugriff auf die Datenbank
- Implementierungsspezifische Fehler

6.3.5 PROTOKOLLIERUNG

Der *SourceConnectorDATEVRAW* protokolliert bestimmte Ereignisse in einer speziellen Protokoll-Datei. Zu diesen Ereignissen zählen:

- Aufruf des *SourceConnectorDATEVRAW*
- Protokollierung von spezifizierten Fehlermeldungen (ungültiger Namensraum, ungültige Selektion)
- Fehler zur Laufzeit

Zusätzlich wird gemäß Rahmenarchitektur ein P23RAppFault innerhalb des WebServices generiert (siehe Abschnitt 6.1.4).

6.3.6 KONFIGURATION

Konfigurationsinformationen werden zur Datenbankverbindung (eXistDB), sowie zur Verbindung zum Transferverzeichnis bereitgestellt.

6.3.7 ARTEFAKTE

Für den *SourceConnectorDATEVRAW* entsteht das folgende Artefakt:

- ent.sourceconnectorDATEVRAW-<version>.war
- JBoss Web-Anwendung, enthält auch die Funktionsbestandteile für Webservice-Implementierung gemäß SOAP.

P23R

P23R: Spezifikationen zur Infrastruktur

7 SPEZIFIKATION DER KOMMUNIKATIONSKONNEKTOREN

Die zu implementierenden Kommunikationskonnektoren sind eigenständig lauffähige Komponenten, die über eine Webservice-Schnittstelle anzusteuern sind. Sie können aus diesem Grunde auch auf andere Rechner ausgelagert werden. Eine weitreichendere Trennung zum P23R ist jedoch nicht ratsam, da die Beziehung nicht weiter in der Rahmenarchitektur abgesichert wurde.

Neben einem grundlegenden Kommunikationskonnektor für die Musterimplementierung, der im Wesentlichen zur Nachweisführung spezifischer P23R-Funktionalitäten – wie der korrekten Anwendung von Signaturen und Repräsentationsskripten – dient, werden auch die für das Pilotszenario AGM benötigten Konnektoren definiert, welche allesamt auf eine Übergabe von Repräsentationsskripten und Verarbeitung von Signaturfunktionen verzichten, da diese in den jeweils statisch geforderten Szenarien nicht erforderlich sind.

7.1 COMMUNICATIONCONNECTOR

Der `CommunicationConnector` *p23r.communicationconnector* stellt die Komponente zur P23R-Musterimplementierung dar. Seine Funktionalität beschränkt sich auf die Anwendung des passend übergebenen Repräsentationsskripts, das protokollieren von Aufrufen und eventuellen Fehlern sowie den Nachweis der korrekten Anwendung Signaturen auf die Benachrichtigung.

7.1.1 ARBEITSWEISE

Der `communicationconnector` implementiert als zentrale Komponente die Webservice-Schnittstelle `INotificationTransfer` der Rahmenarchitektur. Als Teil der Musterimplementierung implementiert dieser keine spezifischen Funktionen, bildet jedoch folgende Funktionalitäten ab:

- Protokollierung der übergebenen Parameter inkl. Repräsentationsskript
- Validierung und Protokollierung einer optional übergebenen Dokumentensignatur
- Validierung und Protokollierung einer optional übergebenen CoreSignatur
- Rückgabe des Webservice-Rückgabewertes `SendProtocol`

7.1.2 SCHNITTSTELLEN

Der `communicationconnector` implementiert gemäß Rahmenarchitektur `INotificationTransfer` Webservice zur Annahme einer Benachrichtigung und Parameter aus dem Zuständigkeitsverzeichnis, sowie Rückgabe von Werten für das P23R-Protokoll.

7.1.3 FEHLERBEHANDLUNG

Folgende Fehler können innerhalb des `communicationconnector` auftreten und führen zu einem Protokolleintrag:

- Unvorhergesehener Fehler zur Laufzeit

7.1.4 PROTOKOLLIERUNG

Der `communicationconnector` protokolliert bestimmte Ereignisse in einer speziellen Protokoll-Datei. Zu diesen Ereignissen zählen:

P23R

P23R: Spezifikationen zur Infrastruktur

- Aufruf des `CommunicationConnector` und der übergebenen Parameter
- Validierung einer optional vorhandenen Dokumentensignatur
- Validierung einer optional vorhandenen CoreSignatur
- Protokollierung von Fehlern zur Laufzeit

Zusätzlich liefert gemäß der Rahmenarchitektur der `WebService` `INotificationTransfer` bei Erfolg oder aufgetretenen Fehlern einen Eintrag im Rückmeldetyp `SendProtocol` an den `NotificationTransport` (Benachrichtigungstransport) zurück.

7.1.5 KONFIGURATION

Da der `communicationconnector` keine weitere Funktionalität außer einer Protokollierung zur Verfügung stellt, sind keine spezifischen Konfigurationen vorhanden.

7.1.6 ARTEFAKTE

Für den `CommunicationConnector` entsteht das folgende Artefakt:

- `p23r.communicationconnector-<version>.war`
JBoss Web-Anwendung, enthält auch die Funktionsbestandteile für `WebService`-Implementierung gemäß SOAP

7.2 COMMUNICATIONCONNECTOR.ESTATISTIKCORE

Der `eSTATISTIK.core`-Konnektor bereitet eine Benachrichtigung für die Übermittlung an das namensgebende Verfahren des statistischen Bundesamtes vor und übermittelt dieses über den verfahrensspezifischen Testkanal.

Die Implementierung erfolgt in Form der Komponente `p23r.estatistikcoreconnector` und implementiert hierzu die `WebService` Schnittstelle `INotificationTransfer` der Rahmenarchitektur.

7.2.1 ARBEITSWEISE

Der `estatistikcoreconnector` wird vom `NotificationTransport` (siehe Abschnitt 3.6) via SOAP aufgerufen. Als einziger Parameter wird hier die Benachrichtigung selber ausgewertet, andere Parameter werden ignoriert.

Die vom P23R übergebene Benachrichtigung wird mittels einer XSL-Transformation und einem im Konnektor hinterlegtem XSLT-Skript in das für `eSTATISTIK.core` verwendete Meldungsschema transformiert.

Als Meldungsschema wird gemäß der Bestückung der Leitstelle (siehe Abschnitt 9.2) der Typ „Datensatzkennung 1000107300199“ verwendet. Das daraus resultierende Meldungsschema wird via HTTP-POST an den allgemeinen Dateneingang des `eSTATISTIK.core`-Verfahrens gesendet.

Damit es nicht zu Fehlmeldungen kommt, wird in der Meldung die Option „test kennung=100“ verwendet, welches ein Verwerfen nach Dateneingang zur Folge hat.

7.2.2 SCHNITTSTELLEN

Der `estatistikcoreconnector` implementiert gemäß Rahmenarchitektur die folgenden `WebServices`, sowie folgende verwaltungsspezifische Schnittstellen:

- `INotificationTransfer` – zur Annahme einer Benachrichtigung und Parameter aus dem Zuständigkeitsverzeichnis, sowie Rückgabe von Werten für das P23R-Protokoll.
- HTTP-POST Dateneingang des `eSTATISTIK.core` Meldeverfahrens

7.2.3 FEHLERBEHANDLUNG

Folgende Fehler können innerhalb des `estatistikcoreconnector` auftreten und führen zu einem Protokolleintrag:

- Unvorhergesehener Fehler zur Laufzeit
- Fehler bei der Transformation der Meldung in das Meldeschema `eSTATISTIK.core`.
- Fehler bei der Übertragung der Meldung an den Dateneingang von `eSTATISTIK.core`.
- Fehler bei der Datenanahme der Meldung von `eSTATISTIK.core`.

7.2.4 PROTOKOLLIERUNG

Der `estatistikcoreconnector` protokolliert bestimmte Ereignisse in eine spezielle Protokoll-Datei. Zu diesen Ereignissen zählen:

- Aufruf des `eSTATISTIK.core`-Konnektors und der übergebenen Parameter
- Ergebnis und Rückgabewerte der Datenannahmestelle des `eSTATISTIK.core`-Verfahrens
- Protokollierung von Fehlern zur Laufzeit

Zusätzlich liefert gemäß Rahmenarchitektur der `WebService INotificationTransfer` bei Erfolg oder aufgetretenen Fehlern einen Eintrag im Rückmeldetyp `SendProtocol` an den `NotificationTransport` (Benachrichtigungstransport) zurück.

7.2.5 KONFIGURATION

Der `estatistikcoreconnector` benötigt keine spezifischen Konfigurationen.

7.2.6 ARTEFAKTE

Für den `eSTATISTIK.core`-Konnektor entsteht das folgende Artefakt:

- `p23r.estatistikcoreconnector-<version>.war`
JBoss Web-Anwendung, enthält auch die Funktionsbestandteile für `WebService`-Implementierung gemäß SOAP.

7.3 COMMUNICATIONCONNECTOR.VDEUEV

Der `vDEÜV`-Konnektor transformiert eine Benachrichtigung in den zugehörigen `DEÜV`-Datensatz und überträgt diesen via `SFTP` an eine virtuelle (sprich simulierte) Kopfstelle, wo diese in einen der jeweils echten Kopfstelle entsprechenden Ordner abgelegt wird.

Die Implementierung erfolgt in Form der Komponente `p23r.vdeuevconnector` und implementiert hierzu die `WebService` Schnittstelle `INotificationTransfer` der Rahmenarchitektur.

P23R

P23R: Spezifikationen zur Infrastruktur

7.3.1 ARBEITSWEISE

Der *vdeuevconnector* wird vom *NotificationTransport* (siehe Abschnitt 3.6) via SOAP aufgerufen. Als Parameter wird hier zum einen die Benachrichtigung selber ausgewertet, sowie die Zieladresse der virtuellen Kopfstelle - andere Parameter werden ignoriert.

Die vom P23R übergebene Benachrichtigung wird mittels eines Parser in das DEÜV-Meldungsformat transformiert. Dieses Meldungsformat besteht aus jeweils einem Vorlaufdatensatz, den Meldedatensätzen und einem Nachlaufdatensatz pro Mitarbeiter.

Die daraus erstellten Datendateien werden via SFTP (FTP über SSH) auf einen Server übertragen. Dieser Server stellt die virtuelle Kopfstelle des DEÜV-Verfahrens dar.

Optional kann für die Übertragung ein SSH-Tunnel konfiguriert werden.

7.3.2 SCHNITTSTELLEN

Der *vdeuevconnector* implementiert gemäß Rahmenarchitektur die folgenden WebServices, sowie folgende verwaltungsspezifische Schnittstellen:

- INotificationTransfer – zur Annahme einer Benachrichtigung und Parameter aus dem Zuständigkeitsverzeichnis, sowie Rückgabe von Werten für das P23R-Protokoll.
- SFTP Dateneingang des simulierten Meldeverfahrens

7.3.3 FEHLERBEHANDLUNG

Folgende Fehler können innerhalb des *vdeuevconnector* auftreten und führen zu einem Protokolleintrag:

- Unvorhergesehener Fehler zur Laufzeit
- Fehler bei der Transformation der Meldung in das Zielmeldeschema
- Fehler bei der Übertragung der Meldung an den Dateneingang.

7.3.4 PROTOKOLLIERUNG

Der *vdeuevconnector* protokolliert bestimmte Ereignisse in eine spezielle Protokoll-Datei. Zu diesen Ereignissen zählen:

- Aufruf des vDEÜV-Konnektors und der übergebenen Parameter
- Ergebnis der Übertragung zur Datenannahmestelle (SFTP-Servers)
- Protokollierung von Fehlern zur Laufzeit

Zusätzlich liefert gemäß Rahmenarchitektur der Webservice INotificationTransfer bei Erfolg oder aufgetretenen Fehlern einen Eintrag im Rückmeldetyp SendProtocol an den *NotificationTransport* (Benachrichtigungstransport) zurück.

7.3.5 KONFIGURATION

Konfigurationsinformationen werden zur Datenübermittlung via SFTP und einem optionalen SSH-Tunnel bereitgestellt.

7.3.6 ARTEFAKTE

Für den vDEÜV -Konnektor entsteht das folgende Artefakt:

- `p23r.vdeuevconnector-<version>.war`

JBoss Web-Anwendung, enthält auch die Funktionsbestandteile für Webservice-Implementierung gemäß SOAP.

7.4 COMMUNICATIONCONNECTOR.BGVIAEMAIL

Der BGviaMail-Konnektor bereitet eine Benachrichtigung mittels eines hinterlegten Transformationskripts in Form eines PDF-Formulars auf und sendet dieses als Anhang via E-Mail an eine über das Zuständigkeitsverzeichnis ermittelte Zieladresse.

Die Implementierung erfolgt in Form der Komponente `p23r.bgviamailconnector`, und implementiert hierzu die Webservice Schnittstelle `INotificationTransfer` der Rahmenarchitektur.

7.4.1 ARBEITSWEISE

Der `bgviamailconnector` wird vom `NotificationTransport` (siehe Abschnitt 3.6) via SOAP aufgerufen. Als Parameter wird hier zum einen die Benachrichtigung selber ausgewertet, aber auch die Zieladresse des Empfängers - andere Parameter werden ignoriert.

Die vom P23R übergebene Benachrichtigung wird mittels einer Transformation des im Konnektor hinterlegten XSL-FO-Skriptes in ein PDF-Dokument umgewandelt. Hierbei kommt die Open Source Bibliothek Apache FOP 1.0 [19] zum Einsatz. Das erstellte Dokument wird anschließend als Emailanhang an die vom Zuständigkeitsverzeichnis ermittelte Adresse versendet.

7.4.2 SCHNITTSTELLEN

Der `bgviamailconnector` implementiert gemäß Rahmenarchitektur die folgenden WebServices, sowie folgende verwaltungsspezifische Schnittstellen:

- `INotificationTransfer` – zur Annahme einer Benachrichtigung und Parameter aus dem Zuständigkeitsverzeichnisses, sowie Rückgabe von Werten für das P23R-Protokoll.
- Datenversand via SMTP und MIME-Email.

7.4.3 FEHLERBEHANDLUNG

Folgende Fehler können innerhalb des `bgviamailconnector` auftreten und führen zu einem Protokolleintrag:

- Unvorhergesehener Fehler zur Laufzeit
- Fehler bei der Transformation der Meldung in das Zielmeldeschema oder in das PDF-Dokument.
- Fehler beim Datenversand der Meldung.

7.4.4 PROTOKOLLIERUNG

Der `bgviamailconnector` protokolliert bestimmte Ereignisse in eine spezielle Protokoll-Datei. Zu diesen Ereignissen zählen:

- Aufruf des BGviaMail-Konnektors und der übergebenen Parameter
- Ergebnis des Datenversandes der Meldung
- Protokollierung von Fehlern zur Laufzeit

P23R

P23R: Spezifikationen zur Infrastruktur

Zusätzlich liefert gemäß P23R-Rahmenarchitektur [1] der `WebService INotificationTransfer` bei Erfolg oder aufgetretenen Fehlern einen Eintrag im Rückmeldetyp `SendProtocol` an den *NotificationTransport* (Benachrichtigungstransport) zurück.

7.4.5 KONFIGURATION

Konfigurationsinformationen werden zur Datenübermittlung via Email bereitgestellt.

7.4.6 ARTEFAKTE

Für den `BGviaMail`-Konnektor entsteht das folgende Artefakt:

- `p23r.bgviamailconnector-<version>.war`

JBoss Web-Anwendung enthält auch die Funktionsbestandteile für `WebService`-Implementierung gemäß SOAP.

8 ANHANG I: ENTWICKLUNGSVORGABEN UND -RICHTLINIEN

In diesem Kapitel werden die Richtlinien für die Implementierung festgelegt. Dies beinhaltet Vorgaben und Methoden für die Erstellung von Quellen, zu verwendende Tools bei der Entwicklung, Dokumentation der Software, das Erstellen und Durchführen von Tests, sowie Art und Weise der Zusammenstellung und Auslieferung der Software.

8.1 PROGRAMMIERRICHTLINIEN

Die Entwickler haben Software nach den allgemeingültigen Qualitätsmerkmalen zu erstellen. Insbesondere haben sie den Code auch für andere lesbar und wartungsfreundlich zu halten.

„There are in fact two target platforms for all code: the runtime hardware, and the human cerebral cortex. All code 'runs' in the human brain, in the sense that all code needs to be understood by a human being. Coding style is always concerned with this second target platform, not with the first.“

Dieses Zitat aus [20] beschreibt, welche Arbeitsweise dem Projekt zugrunde liegt.

Zu den üblichen Fehlern in der Java Welt gehören:

- Zu lange (große) Klassen
- Zu lange (große) Methoden
- Zu wenig Javadoc
- Leere Catch Blöcke zur Unterdrückung von Exceptions
- Unnötige Multiple Return Statements
- Missbrauch von Exceptions um den Programmfluss zu steuern
- Übermäßiger Gebrauch von „instanceof“
- Verwendung von Arrays anstelle von Collections
- Nicht Gruppieren von Methoden bezüglich ihres Scopes

Dies sind die wesentlichen Aspekte, die bei der Erstellung von Codes berücksichtigt werden müssen, damit die Lesbarkeit und Verständlichkeit nicht darunter leidet. Jedoch muss deutlich gemacht werden, dass ein Verletzen dieser Regeln nicht zwangsläufig zu schlechter Software führt, aber dadurch das Arbeiten im Team deutlich erschwert wird.

Zur Umsetzung von Funktionalität sind die anerkannten Vorgaben, wie sie auf den „Java Practices“ [21] und „Secure Coding Guidelines for the Java Programming Language“ [22] beschrieben werden, einzuhalten.

Die Formatierung der Quellen entspricht den Editor-Vorgaben der Eclipse Umgebung[23]. Jede textbasierte Datei muss mit einem projektspezifischem Kopf (Header) beginnen.

Adressen und Referenzen zur Kommunikation dürfen nicht hart-codiert werden (z. B. müssen IP-Adressen zur Laufzeit konfiguriert werden können).

P23R

P23R: Spezifikationen zur Infrastruktur

Alle weiteren Regeln werden durch den Regelsatz in Sonar (siehe Abschnitt 8.2.6) vorgegeben. Durch den vorgeschriebenen Einsatz dieses Entwicklungstools wird die Einhaltung überprüfbar.

8.1.1 DOKUMENTATION DER QUELLEN

Java Quellen sind mit Hilfe von Javadoc [24] zu kommentieren. Bei der Verwendung des Javadoc-Tools gelten die Konventionen, wie sie von Sun beschrieben wurden [25].

Schnittstellen, die als WSDL-Definitionen vorliegen, müssen dokumentiert werden. Da sie elementare Bestandteile der Rahmenarchitektur bilden, werden ausführliche semantische Beschreibungen in der Rahmenarchitektur erstellt. Deshalb wird es als ausreichend angesehen, wenn eine entsprechende Dokumentation im zugehörigen Quellcode mit Hilfe von Javadoc erstellt wird.

Sie muss aber als solche klar erkennbar sein, das heißt, es muss explizit darauf hingewiesen werden, dass diese Methode oder Klasse als Web Service Schnittstelle nach außen exportiert wird.

8.1.2 VERSIONSSCHEMA

Versionierung dient nicht nur zur reinen Archivierung und Nachverfolgung sondern spielt insbesondere auch bei der Verwaltung von Abhängigkeiten eine wichtige Rolle. Durch den Einsatz von Maven [26] bietet sich das folgende Standard-Versionsschema als Basis in diesem Umfeld an (siehe auch [27]) an:

`<MajorVersion [> . <MinorVersion [> . <IncrementalVersion]] [> - <BuildNumber | Qualifier] >`

Dabei gilt, dass der Qualifier als String abgebildet ist und alle anderen numerisch (Integer) hinterlegt sind. Im Kontext der Umsetzung wird festgelegt, dass mindestens Major, Minor und Incremental vorhanden sein müssen.

Damit wird grundsätzlich dem semantischen Ansatz, wie er in der SemVer Spezifikation [28] beschrieben ist, gefolgt.

Konkret bedeutet dies:

- MajorVersion orientiert sich an den Release-Versionen, festgelegt durch den Release Plan innerhalb der agilen Softwareentwicklung. (z. B. 0, 1 und 2)
- MinorVersion zugehörig zu einer Major Version dürfen nur Rückwärtskompatible Funktionen beinhalten. Sie werden zumindest teilweise vom Release Plan vorgegeben (z. B. 0.9, 1.0, 1.5, 2.0). Zusätzliche Zwischenversionen können bei Bedarf hinzugefügt werden, sollten aber bei allen Komponenten bis zu einem gewissen Grade harmonisiert werden.
- IncrementalVersion kann vom Entwickler vergeben werden und markiert Bug-Fixes von fehlerhaftem Verhalten. Sie müssen nicht mit anderen Komponenten übereinstimmen.

Entwickler sollten lokal und in der Entwicklungsumgebung den Qualifier „SNAPSHOT“ verwenden, andere übliche Qualifier sind momentan nicht vorgesehen (GA, CR, Alpha, Beta, SP, ...). Abweichungen dieses Schemas werden von Maven als reinen Qualifier interpretiert und sind deshalb nicht zugelassen.

Es ist zu überdenken, inwieweit Versionen von einzelnen Komponenten abweichen dürfen und ob es eine übergreifende Version für komplette Releases geben muss.

Als Ergänzung soll nicht unerwähnt bleiben, dass auch Maven seine Versionsschemas einer Überarbeitung unterzieht [29].

8.1.3 NAMENSRÄUME

Namensräume spielen sowohl bei den Quellen als auch bei den verwendeten Datenmodellen eine Rolle. Bei den Quellen reden wir von Paketen, während wir bei den Datenmodellen die XML-Namespaces definieren müssen.

Die Bezeichnung der Namensräume lehnt sich am endgültigen Projektnamen und den Vorgaben des CI an:

- Java Packages
 - Alle Klassen liegen im Namensraum „de.p23r“
 - Zusätzlich gibt es Subpakete für die einzelnen Domänen:
 - de.p23r.main – Der eigentliche P23R
 - de.p23r.cc – Leitstelle (P23R-Control-Centre)
 - de.p23r.client – P23R-Client
 - de.p23r.common – Komponentenübergreifende Definitionen und Funktionen
- XML Namespaces
 - Es gelten die Projektvorgaben, die auch in der Rahmenarchitektur Verwendung finden und in Tabelle 22: P23R Namensräume zusammengefasst werden.

TABELLE 22: P23R NAMENSRÄUME

URL	Erläuterung
http(s)://www.p23r.de/	Die offizielle Webseite zum P23R-Prinzip (Methodenleitfaden, Veröffentlichungen, Kontakt, etc.).
https://leitstelle.p23r.de/	Webserver für die operativen Artefakte der Leitstelle, d. h. Schema, Regelpakete, TSLs, Services, etc..
http(s)://www.leitstelle.p23r.de/	Die offizielle Webseite der Leitstelle (für Informationen an Entwickler, Hersteller, Regelexperten, etc.).
https://leitstelle.p23r.de/p23r.tsl	Das Initialisierungsdokument, das jeder P23R laden muss, um alle anderen Dienste und Dokumente (Benachrichtigungsregelpakete) zu finden.
https://leitstelle.p23r.de/TSL/	Der offizielle Rootpfad für TSLs der Öffentlichen Leitstelle.
https://leitstelle.p23r.de/NRP/	Der offizielle Rootpfad für Benachrichtigungsregelpakete der Öffentlichen Leitstelle.
https://leitstelle.p23r.de/NS/P23R/	Der offizielle Rootpfad für Schemata, WSDL, etc. der Öffentlichen Leitstelle.
https://leitstelle.p23r.de/NS/P23R/NRL/	Der offizielle Rootpfad für Schemata, die im P23R zur Implementierung der Benachrichtigungsregelpakete gebraucht werden.

P23R

P23R: Spezifikationen zur Infrastruktur

URL	Erläuterung
https://leitstelle.p23r.de/NS/P23R/Security/	Der offizielle Rootpfad für Schemata, die im P23R zur Implementierung der Sicherheitsinfrastruktur gebraucht werden.
http(s)://demonstrator.p23r.de	Der Demonstrator des P23R auf Basis der Musterimplementierung.
https://demonstrator.p23r.de/NS	Der offizielle Rootpfad für Schemata, die für die Musterimplementierung des P23R gebraucht werden.

8.1.4 NAMENSREGELN

Komponenten müssen mit einem Präfix beginnen, der sie einer der drei Domänen zuweist. Die drei Domänen sind

- p23r – P23R Komponenten
- cc – ControlCentre (Leitstelle)
- ent – Enterprise (Unternehmen, bzw. P23R-Client Komponenten)

Daraus ergeben sich auch die Regeln für die Maven Identifier:

- groupId
Alle Artefakte haben die groupId „de.p23r“
- artifactId
Eine artifactId beginnt mit p23r, cc oder ent, entsprechend der Zugehörigkeit der Komponente, gefolgt von einem Punkt und dem Projektnamen.

Beispiele: *p23r.datakonnektor*, *ent.portal*, *cc.controlcentre*

8.1.5 LOGGING

Grundsätzlich ist jede Ausgabe über System.out nicht gestattet. Das betrifft insbesondere auch die Ausgabe von Exceptions mit Hilfe der .printStackTrace()-Methodenfamilie.

Stattdessen ist die Verwendung eines Logging-Frameworks zwingend. Als Logging-Framework ist Slf4J [30] zu verwenden, es sei denn die verwendete Technology stellt bereits ein anderes Framework zur Verfügung.

Üblicherweise wird dabei pro Klasse eine Logger Instanz verwendet. Dabei können auf folgende Log-Levels zurückgegriffen werden:

- FATAL – Eine Fortführung der Anwendung ist nicht mehr möglich.
- ERROR – Wenn definitiv ein Problem aufgetreten ist, das die weitere Funktion beeinträchtigt.
- WARNING – Wenn etwas Ungewöhnliches passiert, aber die Funktion der Anwendung nicht gefährdet ist.
- INFO – Informationen, die informativ während des Betriebs sind.

- DEBUG – Informationen die der Entwicklung dienen und in Produktionsumgebungen ausgeschaltet werden können.
- TRACE – Dient nur zum verfolgen des Programflusses.

8.2 ENTWICKLUNGSTOOLS

8.2.1 ECLIPSE IDE

Als integrierte Entwicklungsumgebung (IDE) wird Eclipse [23] verwendet. Für das Entwicklerteam werden vorkonfigurierte Profile bereitgestellt, über die auch Anpassungen schnell an alle weitergegeben werden können.

So ist es möglich, über diese Profile die Formatierungsregeln, Klassentemplates und Kommentierungsvorgaben zu definieren und an alle jederzeit weiterzugeben. Die Verwaltung dieser Profile geschieht mit Hilfe der Pulse [31] Eclipse Verwaltungssoftware.

8.2.2 VERSIONSKONTROLLSYSTEM SUBVERSION

Zur Versionskontrolle wird SVN [32] eingesetzt. Das Repository für das Projekt hat folgende Struktur:

- pbbdev->development->{category}->{component}
 - ->trunk
 - ->tags
 - ->branches

Werte für Kategorien ({category}) sind:

- common – Projekte, die Gemeinsamkeiten abbilden.
- p23r – Projekte, die Komponenten des eigentlichen P23R definieren.
- controlcentre - Projekte, die Komponenten der Leitstelle definieren.
- enterprise - Projekte, die Komponenten des Unternehmens bzw. des P23R-Client definieren.
- examples – Beispiele, als Erfahrungsaustausch zwischen den Entwicklern
- demo – Projekte, die zur Steuerung des Demonstrators dienen.

Grundsätzlich gelten die üblichen Vorgehensweisen, wie sie in der Dokumentation zu SVN [33] beschrieben werden. Dies betrifft insbesondere die oben beschriebene Aufteilung in trunk, tags und branches, die für jedes einzelne Projekt zu verwenden ist.

Es dürfen nur projektspezifische Artefakte eingebracht werden. Artefakte, die sich auf die lokale Entwicklungsumgebung beziehen, sind von der Versionskontrolle auszuschließen, z. B. durch Verwendung des svn:ignore Mechanismus.

Revisionsnummern stehen in keinem Zusammenhang mit der Version einer Komponente, wie sie in Abschnitt 8.1.2 beschrieben wird.

8.2.3 MAVEN

Für die Erstellung der Komponenten ist Maven [26] zu benutzen. Soweit nicht anders erforderlich, ist der gesamte Lebenszyklus der Softwareerstellung mit Maven abzudecken. Das beginnt beim Generie-

P23R

P23R: Spezifikationen zur Infrastruktur

ren von Quellen, Kompilieren, Zusammenstellen von Paketen bis hin zum Deployment des Ergebnisses.

Lediglich die Versionskontrolle liegt außerhalb der Zuständigkeit dieses Werkzeugs. Das heißt nicht, dass nicht auch andere Werkzeuge verwendet werden können. Prinzipiell ist aber zu prüfen, ob nicht ebenfalls eine Einbettung dieses Werkzeuges in Form eines Maven Plugins zur Verfügung steht.

8.2.3.1 MAVEN REPOSITORY (NEXUS)

Hauptaufgabe des Werkzeugs ist die Verwaltung von Abhängigkeiten. Damit dies auch innerhalb des Projektes uneingeschränkt funktioniert, ist der Einsatz eines eigenen „Remote“ Repositories [34] zwingend erforderlich.

Neben dem Spiegeln von öffentlichen Repositories erlaubt es deren Zusammenfassen über einen Repository Server. Zusätzlich können beliebige Repositories definiert werden, z. B. für Third Party Libraries, die nicht über ein öffentliches Repository zur Verfügung stehen.

Somit kann komplett auf mitgelieferte Abhängigkeiten (Libraries) verzichtet werden. Letztlich stellt es die eigenen Entwicklungen für alle bereit, so dass darüber auch projektinterne Abhängigkeiten aufgelöst werden können.

8.2.4 TICKET SYSTEM UND WIKI (TRAC)

Für die Entwickler stehen ein Wiki und ein auf Trac [35] basierendes Ticketsystem zur Verfügung. Das Wiki dokumentiert aktuelle Entwicklungszustände und dient in erster Linie als Austauschplattform für die Entwickler untereinander.

Des Weiteren beinhaltet das Trac ein Ticketsystem um Fehler, Änderungswünsche und andere Punkte nachvollziehbar zu verwalten. Das Ticket System ist dem Projekt entsprechend zu konfigurieren und muss die geplanten Releases und einzelnen Komponenten explizit berücksichtigen.

8.2.5 CONTINUOUS INTEGRATION (HUDSON)

Als Continuous Integration System kommt Hudson [36] zum Einsatz. Es ermöglicht eine einfache und intuitive Darstellung über den Zustand einzelner Komponenten.

Durch die direkte Unterstützung von SVN, Maven und Sonar können schnell und praktisch alle wesentlichen Komponenten automatisiert überwacht und deren Zustand bezüglich Kompilierbarkeit jederzeit abgerufen werden.

Hudson stellt auch direkt die Ergebnisse der durchgeführten Unit Tests dar, sofern sie als Teil des Erstellungsprozesses definiert sind. So kann auch über den funktionalen Zustand informiert werden. Hudson übernimmt auch die Durchführung von Sonar basierten Code Analysen und überstellt sie dem Sonar Server zu aufbereiteten Darstellung über eine Web Seite.

8.2.6 SONAR

Sonar [37] dient der Qualitätssicherung auf Code Ebene. Es analysiert die Quellen und nimmt neben reinen statistischen Informationen auch Bezug auf festgelegte Programmierregeln und den Grad der Verträglichkeit. Es wird der standardmäßig vorgegebene Regelsatz als Basis verwendet und gegebenenfalls angepasst.

Die Entwickler sind aufgefordert, ihren Code streng an diesen Regeln auszurichten und gegebenenfalls von Zeit zu Zeit nachzubessern. Trotzdem sei darauf hingewiesen, dass Fälle existieren, bei denen diese Regeln nicht einzuhalten sind oder gar mit Absicht gebrochen werden müssen.

Es ist ebenfalls nicht unüblich, dass sogar generierte Quellen wesentliche Regeln der Softwareerstellung nicht einhalten. In diesem Fall können die Abweichungen ignoriert werden. Wenn es als störend empfunden wird, weil es z. B. die Ergebnisse verfälscht, müssen diese Quellen von der Analyse ausgeschlossen werden.

8.3 ARBEITEN MIT DER LABORUMGEBUNG

Softwareverwaltung findet mit Hilfe des Software Versionskontrollsystem SVN statt. Jede Komponente muss aus diesem System mit wenigen Handgriffen extrahierbar, kompilierbar und zusammensetzbar sein. Diese Vorgänge sollten höchstens in Ausnahmefällen auf dem lokalen Entwicklerrechner durchgeführt werden. Die üblichen Schritte im P23R Kontext sind:

- SVN Checkout (z. B. `svn co https://.../<component>/trunk <component>`)
- Maven basiertes Erstellen des deployable Artefakts (z. B. `mvn package`)
- Maven basiertes Deployment – Wobei hier nicht Deployment in das Maven Repository gemeint ist, sondern in die Laufzeitumgebung. (z. B. `mvn jboss:hard-deploy`)

Wünschenswert wäre die Bereitstellung von vereinfachten Batch Scripts, die die gängigsten Befehlsfolgen zusammenfassen (z. B. `update.sh`), ist aber optional.

P23R

P23R: Spezifikationen zur Infrastruktur

9 ANHANG II: BESTÜCKUNG DER LEITSTELLE (TEST UND AGM)

Die einzelnen Datenmodell- und Benachrichtigungsregelpakete werden gemäß den Vorgaben des Pflichtenhefts [2] auf Grundlage der T-BRS umgesetzt, wobei anstelle von DataSel-Skripten in den Pilotpaketen AGM (im Gegensatz zu den Testpaketen) jedoch XQuery-Skripte zum Einsatz kommen.

Hinweis: Bei der Darstellung der Pakete wird im Folgenden auf die einfassenden Verzeichnisse „Paket-Name“ und „Release“ verzichtet, um die Lesbarkeit nicht unnötig zu erschweren.

9.1 BESTÜCKUNG ZU TESTZWECKEN

Die folgenden Artefakte dienen dem Test der P23R-Musterimplementierung mit Hilfe eines fiktiven Regelpakets.

9.1.1 DATENMODELLPAKET „TEST“

Der vollständige Aufbau des Datenmodellpakets kann der folgenden Tabelle entnommen werden:

TABELLE 23: AUFBAU DES DATENMODELLPAKETS „TEST“

Name	Datei (D) / Ver- zeichnis (V)	Beschreibung
Manifest.xml	D	Enthält Informationen über die eindeutige ID des Datenmodellpakets, die Versionsnummer und das Veröffentlichungsdatum.
Unternehmen	V	Teildatenmodell „Unternehmen“
Manifest.xml	D	Enthält Informationen über die eindeutige ID des Teildatenmodells, die Versionsnummer und das Veröffentlichungsdatum.
Source.txt	D	Enthält eine Zusammenfassung der Inhalte des Teildatenmodells.
TEST_Pivot_Unternehmen.xsd	D	Enthält die eigentliche Beschreibung des Teildatenmodells „Unternehmen“ zur Beschreibung fiktiver Unternehmensinformationen für die Testmeldungen des P23R.
Mitarbeiter	V	Teildatenmodell „Mitarbeiter“
Manifest.xml	D	Enthält Informationen über die eindeutige ID des Teildatenmodells, die Versionsnummer und das Veröffentlichungsdatum.
Source.txt	D	Enthält eine Zusammenfassung der Inhalte des Teildatenmodells.
TEST_Pivot_Mitarbeiter.xsd	D	Enthält die eigentliche Beschreibung des Teildatenmodells „Mitarbeiter“ zur Beschreibung fiktiver Mitarbeiterinformationen für die Testmeldungen des P23R.

P23R

P23R: Spezifikationen zur Infrastruktur

9.1.2 BENACHRICHTIGUNGSREGELPAKET „TEST“

Der vollständige Aufbau des Benachrichtigungsregelpaketes in den Version 1.0 und 2.0 kann den folgenden Tabellen entnommen werden:

TABELLE 24: AUFBAU DES BENACHRICHTIGUNGSREGELPAKETS „TEST V 1.0“

Name	Datei (D) / Ver- zeichnis (V)	Beschreibung
Manifest.xml	D	Neben den Abhängigkeiten zu den zugehörigen Datenmodellpaketen werden in den Metadaten Informationen erfasst, dass dieses Benachrichtigungsregelpaket als Testpakets aus dem Projekt „P23R“ zu betrachten ist.
Recommendation.xquery	D	Als Voraussetzung gilt das Vorhandensein von mindestens einem Mitarbeiter innerhalb der Mitarbeiterdaten der Quelldaten.
Recommendation.dats	D	
Recommendation.xsl	D	
TEST_Statistik	V	Benachrichtigungsregelgruppe „Statistikmeldungen“
Manifest.xml	D	Enthält Informationen über die eindeutige ID der Benachrichtigungsregelgruppe, die Versionsnummer und das Veröffentlichungsdatum.
Recommendation.xquery	D	Als Voraussetzung gilt das Vorhandensein von mindestens einem Mitarbeiter innerhalb der Mitarbeiterdaten der Quelldaten.
Recommendation.dats	D	
Recommendation.xsl	D	
Configuration.xquery	D	Konfiguriert die zu meldenden Unternehmen für den Fall, dass kein spezifisches Unternehmen in der Nachricht als Parameter angegeben wird.
Configuration.dats		
Configuration.xsl	D	
TEST_Statistik_Jahreserhebung	V	Benachrichtigungsregel „Statistik“
Manifest.xml	D	Enthält Informationen über die eindeutige ID der Benachrichtigungsregel, die Versionsnummer und das Veröffentlichungsdatum.
Message.xsd	D	Als Parameter kann das Unternehmen übergeben werden, für das diese Meldung zu generieren ist.
MultiNotificationProfile.xquery	D	Um eine Generierung der Benachrichtigungsübermittlung je Unternehmen realisieren zu können, wird die Funktion des Multi-Messaging der T-BRS ein-

Anhang II: Bestückung der Leitstelle (TEST und AGM)

Name		Datei (D) / Ver- zeichnis (V)	Beschreibung
	MultiNotificationPro- file.dats	D	gesetzt, welche je Unternehmen die Generierung einer eigenen Benachrichtigung auslöst. Darüber hinaus werden die Kriterien für den späteren Aufruf des Zuständigkeitsverzeichnisses eingetragen.
	MultiNotificationPro- file.xsl	D	
	Notification.xsd	D	Die Generierung wird über ein Selektionsskript mit anschließendem zweistufigem Transformationsprozess (zwei Transformationskripte) durchgeführt. Dabei wird auch das Profil zweimalig mit transformiert. Als Ergebnis wird eine Statistik über die Anzahl der im Unternehmen tätigen Mitarbeiter an den fiktiven Empfänger übermittelt.
	Notificati- on_Transformation_S tep1.xsd	D	
	Notificati- on_Transformation_S tep1.xsl	D	
	Notificati- on_Transformation_S tep2.xsl	D	
	NotificationCore.xsl	D	Die Core-Transformation extrahiert alle Elemente der Benachrichtigung mit Ausnahme der Kontaktinformationen und Begründungsübermittlung.
	Profi- le_Transformation_St ep1.xsl	D	Die Generierung wird über ein Selektionsskript mit anschließendem zweistufigem Transformationsprozess (zwei Transformationskripte) durchgeführt. Dabei wird auch das Profil zweimalig mit transformiert.
	Profi- le_Transformation_St ep2.xsl	D	
	representation- generic-bund.xsl	D	Das Repräsentationsskript bewirkt keinerlei Veränderung an der Benachrichtigung sondern dient lediglich der Nachweisführung der Auswahl des korrekten Transformationskripts.
	Selection.xquery	D	Die Generierung wird über ein Selektionsskript mit anschließendem zweistufigem Transformationsprozess (zwei Transformationskripte) durchgeführt. Dabei wird auch das Profil zweimalig mit transformiert.
	Selection.dats	D	
	Selection.xsd	D	
	Source.txt	D	Enthält eine Zusammenfassung der Inhalte der Benachrichtigungsregel.

P23R

P23R: Spezifikationen zur Infrastruktur

Die folgende Tabelle 25 enthält die Darstellung des Regelpakets in Version 2.0, wobei die Änderungen zur Version 1.0 jeweils fett hervorgehoben sind:

TABELLE 25: AUFBAU DES BENACHRICHTIGUNGSREGELPAKETS „TEST V 2.0“

Name	Datei (D) / Ver- zeichnis (V)	Beschreibung
Manifest.xml	D	Neben den Abhängigkeiten zu den zugehörigen Datenmodellpaketen werden in den Metadaten Informationen erfasst, dass dieses Benachrichtigungsregel-paket als Testpakets aus dem Projekt „P23R“ zu betrachten ist.
Recommendation.xquery	D	Als Voraussetzung gilt das Vorhandensein von mindestens einem Mitarbeiter innerhalb der Mitarbeiterdaten der Quelldaten.
Recommendation.dats	D	
Recommendation.xsl	D	
TEST_Statistik	V	Benachrichtigungsregelgruppe „Statistikmeldungen“
Manifest.xml	D	Enthält Informationen über die eindeutige ID der Benachrichtigungsregel-gruppe, die Versionsnummer und das Veröffentlichungsdatum.
Recommendation.xquery	D	Als Voraussetzung gilt das Vorhandensein von mindestens zehn Mitarbeitern innerhalb der Mitarbeiterdaten der Quelldaten.
Recommendation.dats	D	
Recommendation.xsl	D	
Configuration.xquery	D	Konfiguriert die zu meldenden Unternehmen für den Fall, dass kein spezifi-sches Unternehmen in der Nachricht als Parameter angegeben wird.
Configuration.dats	D	
Configuration.xsl	D	
TEST_Statistik_Jahreserh ebung	V	Benachrichtigungsregel „Statistik“
Manifest.xml	D	Enthält Informationen über die eindeutige ID der Benachrichtigungsregel, die Versionsnummer und das Veröffentlichungsdatum.
Message.xsd	D	Als Parameter kann das Unternehmen übergeben werden, für das diese Mel-dung zu generieren ist.
MultiNotificationPro- file.xquery	D	Um eine Generierung der Benachrichtigungsübermittlung je Unternehmen realisieren zu können, wird die Funktion des Multi-Messaging der T-BRS ein-

Anhang II: Bestückung der Leitstelle (TEST und AGM)

	MultiNotificationProfile.dats	D	gesetzt, welche je Unternehmen die Generierung einer eigenen Benachrichtigung auslöst. Darüber hinaus werden die Kriterien für den späteren Aufruf des Zuständigkeitsverzeichnisses eingetragen.
	MultiNotificationProfile.xsl	D	
	Notification.xsd	D	Die Generierung wird über ein Selektionsskript mit anschließendem zweistufigem Transformationsprozess (zwei Transformationskripte) durchgeführt. Dabei wird auch das Profil zweimalig mit transformiert. Als Ergebnis wird eine Statistik über die Anzahl der im Unternehmen tätigen Mitarbeiter nach Geschlecht gegliedert an den fiktiven Empfänger übermittelt.
	Notification_Transformation_Step1.xsd	D	
	Notification_Transformation_Step1.xsl	D	
	Notification_Transformation_Step2.xsl	D	
	NotificationCore.xsl	D	Die Core-Transformation extrahiert alle Elemente der Benachrichtigung mit Ausnahme der Kontaktinformationen und Begründungsübermittlung.
	Profile_Transformation_Step1.xsl	D	Die Generierung wird über ein Selektionsskript mit anschließendem zweistufigem Transformationsprozess (zwei Transformationskripte) durchgeführt. Dabei wird auch das Profil zweimalig mit transformiert.
	Profile_Transformation_Step2.xsl	D	
	representation-generic-bund.xsl	D	Das Repräsentationsskript bewirkt keinerlei Veränderung an der Benachrichtigung sondern dient lediglich der Nachweisführung der Auswahl des korrekten Transformationskripts.
	Selection.xquery	D	Die Generierung wird über ein Selektionsskript mit anschließendem zweistufigem Transformationsprozess (zwei Transformationskripte) durchgeführt. Dabei wird auch das Profil zweimalig mit transformiert.
	Selection.dats	D	
	Selection.xsd	D	
	Source.txt	D	Enthält eine Zusammenfassung der Inhalte der Benachrichtigungsregel.

9.1.3 ZUSTÄNDIGKEITSINFORMATIONEN „TEST“

Im Leistungskatalog (siehe Tabelle 26) werden die jeweilige Benachrichtigungsregel und die korrespondierende Leistung in Relation gesetzt:

P23R

P23R: Spezifikationen zur Infrastruktur

TABELLE 26: TESTREGELN – LEISTUNGSKATALOG

ruleName	Idservice	name	legalbasis
TEST_Statistik_Jahreserhebung	p23r:service:de.p23r.cc:TEST_Statistik_Jahreserhebung	Testweise statistische Jahresmeldung.	TestG

Im Kriterienkatalog (siehe Tabelle 27) werden die verfügbaren Zuständigkeitskriterien definiert:

TABELLE 27: TESTREGELN – KRITERIENKATALOG

idcriteria	Name
p23r:criteria:de.p23r.cc:test	Test-Kriterium für festen Benachrichtigungsempfänger.

Im Verzeichnis der Benachrichtigungsempfänger (siehe Tabelle 28) werden die fachlichen Empfänger der Benachrichtigung definiert:

TABELLE 28: TESTREGELN – VERZEICHNIS DER BENACHRICHTIGUNGSEMPFÄNGER

idorg	name
p23r:org:de.bund:test	Testweises Bundesamt.

Weiterhin werden im Verzeichnis der Supportadressen (siehe Tabelle 29) die Ansprechpartner für den technischen Support festgehalten:

TABELLE 29: TESTREGELN – VERZEICHNIS DES TECHNISCHEN SUPPORTS

idorg	Name
p23r:org:de.p23r.cc:oeLSt	Die Öffentliche P23R-Leitstelle (im Laborbetrieb).

Im elektronischen Dienstverzeichnis (siehe Tabelle 30) werden die Kommunikationsendpunkte für die Kanäle der Benachrichtigungsübermittlung definiert:

TABELLE 30: TESTREGELN – ELEKTRONISCHES DIENSTEVERZEICHNIS

ideservice	address	type
p23r:eservice:de.bund.test:connect	`` (empty, not NULL)	p23r:estypes:de.p23r.cc.estypes:generic

Weiterhin werden im Verzeichnis (siehe Tabelle 31) die Ansprechpartner für den fachlichen Support festgehalten:

TABELLE 31: TESTREGELN – VERZEICHNIS DES FACHLICHEN SUPPORTS

ideservice	address	type
p23r:eservice:de.p23r.cc:oeLSt.postal	Fraunhofer FOKUS, Kaiserin August Allee, Berlin	Postadresse
p23r:eservice:de.p23r.cc:oeLSt.webform	http://leitstelle.p23r.de/	URL
p23r:eservice:de.p23r.cc:oeLSt.email	mailto:info@leitstelle.p23r.de	E-Mail

Darüber hinaus werden die Verknüpfungen für die Ermittlung der Benachrichtigungsempfänger (siehe Tabelle 32) abgebildet (MappingReceiver):

TABELLE 32: TESTREGELN – MAPPINGRECEIVER

idservice (siehe Tabelle 26)	Suchkriterium		idorg (siehe Tabelle 28)
	idcriteria (siehe Tabelle 27)	value	
p23r:service:de.p23r.cc:TEST_Statistik_Jahreseerhebung	p23r:criteria:de.p23r.cc:test	'' (empty, not NULL)	p23r:org:de.bund:test

Schlussendlich werden die Verknüpfungen für die Ermittlung der Kommunikationsendpunkte (siehe Tabelle 33) abgebildet (MappingEservice):

TABELLE 33: TESTREGELN – MAPPINGESERVICE

Idservice (siehe Tabelle 26)	Idorg (siehe Tabelle 28)	ideservice (siehe Tabelle 30)	Prio
p23r:service:de.p23r.cc:TEST_Statistik_Jahreseerhebung	p23r:org:de.bund:test	p23r:eservice:de.bund.test:connect	1

9.2 BESTÜCKUNG FÜR DEN PILOTVERSUCH AGM

Die folgenden Artefakte dienen der Bestückung für den Pilotversuch in der Domäne Arbeitgebermeldepflichten.

9.2.1 DATENMODELLPAKET „AGM“

Der vollständige Aufbau des Datenmodellpakets kann der folgenden Tabelle 34 entnommen werden:

P23R

P23R: Spezifikationen zur Infrastruktur

TABELLE 34: AUFBAU DES DATENMODELLPAKETS „AGM“

Name	Datei (D) / Ver- zeichnis (V)	Beschreibung
Manifest.xml	D	Neben dem Namen des Modellpakets werden in den Metadaten Informationen erfasst, dass dieses Datenmodellpaket zum Pilotversuch in der Domäne AGM aus dem Projekt „P23R“ gehört.
Unternehmen	V	Teildatenmodell „Unternehmen“
Manifest.xml	D	Neben dem Namen des Teildatenmodells werden in den Metadaten Informationen erfasst, dass dieses Datenmodell zum Pilotversuch in der Domäne AGM aus dem Projekt „P23R“ gehört.
Source.txt	D	Enthält eine Zusammenfassung der Inhalte des Teildatenmodells.
AGM_Pivot_Unternehmen.xsd	D	Enthält die eigentliche Beschreibung des Teildatenmodells auf Basis des im Lastenheft [18] beschriebenen Teildatenmodells „Unternehmen“.
Mitarbeiter	V	Teildatenmodell „Mitarbeiter“
Manifest.xml	D	Neben dem Namen des Teildatenmodells werden in den Metadaten Informationen erfasst, dass dieses Datenmodell zum Pilotversuch in der Domäne AGM aus dem Projekt „P23R“ gehört.
Source.txt	D	Enthält eine Zusammenfassung der Inhalte des Teildatenmodells.
AGM_Pivot_Mitarbeiter.xsd	D	Enthält die eigentliche Beschreibung des Teildatenmodells auf Basis des im Lastenheft [18] beschriebenen Teildatenmodells „Mitarbeiter“.

9.2.2 BENACHRICHTIGUNGSREGELPAKET „AGM“

Der vollständige Aufbau des Benachrichtigungsregelpakets kann der folgenden Tabelle 35 entnommen werden:

TABELLE 35: AUFBAU DES BENACHRICHTIGUNGSREGELPAKETS „AGM“

Name	Datei (D) / Ver- zeichnis (V)	Beschreibung
Manifest.xml	D	Neben den Abhängigkeiten zu den zugehörigen Datenmodellpaketen werden in den Metadaten Informationen erfasst, dass dieses Benachrichtigungsregelpaket zum Pilotversuch in der Domäne AGM aus dem Projekt „P23R“ gehört.
Recommendation.xquery	D	Als Voraussetzung gilt das Vorhandensein von mindestens einem Mitarbeiter zu mindestens einer Arbeitsstätte einer Betriebsstätte eines Unternehmens.
Recommendation.xsl	D	
AGM_SOZ	V	Benachrichtigungsregelgruppe „Sozialversicherungsmeldungen“
Manifest.xml	D	Neben den Abhängigkeiten zu den zugehörigen Datenmodellpaketen und der Festlegung der Recommendation-Namespaces werden in den Metadaten Informationen erfasst, dass dieses Benachrichtigungsregelpaket zum Pilotversuch in der Domäne AGM aus dem Projekt „P23R“ gehört.
Recommendation.xquery	D	Als Voraussetzung gilt das Vorhandensein von mindestens einem Mitarbeiter zu mindestens einer Arbeitsstätte.
Recommendation.xsl	D	
AGM_SOZ_BG- JAHRESMELDUNG	V	Benachrichtigungsregel „Jährlicher Lohnnachweis“
Manifest.xml	D	Neben den Metadaten, dass diese Benachrichtigungsregel zum Pilotversuch in der Domäne AGM aus dem Projekt „P23R“ gehört, beschreibt das Manifest die auszuführenden Generierungsschritte. Als Übermittlungszeitpunkt wird der 15. Tag nach Ablauf des Meldejahres angegeben, wobei die Generierung 5 Tage zuvor angestoßen und die Übermittlung spätestens 12 Stunden vor Ablauf der Meldefrist angemahnt wird. Darüber hinaus wird vermerkt, dass für die Signierung keine qualifizierte elektronische Signatur erforderlich ist.
Message.xsd	D	Als Parameter können über die zugehörige Message sowohl das Meldejahr (z. B. 2010) als auch das Unternehmen übergeben werden, für das diese Meldung zu generieren ist.
MultiNotificationProfile.xquery	D	Um eine Generierung der Benachrichtigungsübermittlung je Unternehmen realisieren zu können, wird die Funktion des Multi-Messaging der T-BRS eingesetzt, welche je Unternehmen die Generierung einer eigenen Benachrichtigung auslöst. Darüber hinaus werden die Kriterien für den späteren Aufruf des Zuständigkeitsverzeichnisses eingetragen.
MultiNotificationProfile.xsl	D	
Notification.xsd	D	Die Generierung wird über ein Selektionsskript mit anschließendem Transformationsprozess realisiert.

P23R

P23R: Spezifikationen zur Infrastruktur

	Notification.xml	D	mationsskript durchgeführt. Dabei wird auch das Profil einmalig mit transformiert.
	NotificationCore.xml	D	Der Core beinhaltet in dieser Benachrichtigungsregel alle Merkmale der Benachrichtigung.
	Profile.xml	D	In der Profile-Transformation erfolgt der Abruf der Informationen aus dem Zuständigkeitsverzeichnis.
	Selection.xquery	D	Die Generierung wird über ein Selektionsskript mit anschließendem Transformationsskript durchgeführt. Dabei wird auch das Profil einmalig mit transformiert.
	Selection.xsd	D	
	Source.txt	D	Enthält eine Zusammenfassung der Inhalte der Benachrichtigungsregel.
	AGM_SOZ_DEUEV-JAHRESMELDUNG	V	Benachrichtigungsregel „DEÜV-Jahresmeldung“
	Manifest.xml	D	Neben den Metadaten, dass diese Benachrichtigungsregel zum Pilotversuch in der Domäne AGM aus dem Projekt „P23R“ gehört, beschreibt das Manifest die auszuführenden Generierungsschritte. Als Übermittlungszeitpunkt wird der 15. Tag nach Ablauf des Meldejahres angegeben, wobei die Generierung 5 Tage zuvor angestoßen und die Übermittlung spätestens 12 Stunden vor Ablauf der Meldefrist angemahnt wird. Darüber hinaus wird vermerkt, dass für die Signierung keine qualifizierte elektronische Signatur erforderlich ist.
	Message.xsd	D	Als Parameter können über die zugehörige Message sowohl das Meldejahr (z. B. 2010) als auch das Unternehmen und optional die Krankenkasse übergeben werden, für das diese Meldung zu generieren ist.
	MultiNotificationProfile.xquery	D	Um eine Gliederung der Benachrichtigungsübermittlung je Annahmestelle realisieren zu können, wird die Funktion des Multi-Messaging der T-BRS eingesetzt, welche je Annahmestelle die Generierung einer eigenen Benachrichtigung auslöst. Darüber hinaus werden die Kriterien für den späteren Aufruf des Zuständigkeitsverzeichnisses eingetragen.
	MultiNotificationProfile.xml	D	
	Notification.xsd	D	Die Generierung wird über ein Selektionsskript mit anschließendem Transformationsskript durchgeführt. Dabei wird auch das Profil einmalig mit transformiert.
	Notification.xml	D	
	NotificationCore.xml	D	Der Core beinhaltet in dieser Benachrichtigungsregel alle Merkmale der Benachrichtigung.
	Profile.xml	D	In der Profile-Transformation erfolgt der Abruf der Informationen aus dem Zuständigkeitsverzeichnis.
	Selection.xquery	D	Die Generierung wird über ein Selektionsskript mit anschließendem Transform-

Anhang II: Bestückung der Leitstelle (TEST und AGM)

	Selection.xsd	D	mationsskript durchgeführt. Dabei wird auch das Profil einmalig mit transformiert.
	Source.txt	D	Enthält eine Zusammenfassung der Inhalte der Benachrichtigungsregel.
AGM_STAT		V	Benachrichtigungsregelgruppe „Statistikmeldungen“
	Manifest.xml	D	Neben den Abhängigkeiten zu den zugehörigen Datenmodellpaketen und der Festlegung der Recommendation-Namespaces werden in den Metadaten Informationen erfasst, dass dieses Benachrichtigungsregelpaket zum Pilotversuch in der Domäne AGM aus dem Projekt „P23R“ gehört.
	Recommendation.xquery	D	Als Voraussetzung gilt das Vorhandensein von mindestens einem Mitarbeiter zu mindestens einer Arbeitsstätte.
	Recommendation.xsl	D	
AGM_STAT_VERDIENSTERHEBUNG		V	Benachrichtigungsregel „Statistische Verdiensterhebung“
	Manifest.xml	D	Neben den Metadaten, dass diese Benachrichtigungsregel zum Pilotversuch in der Domäne AGM aus dem Projekt „P23R“ gehört, beschreibt das Manifest die auszuführenden Generierungsschritte. Als Übermittlungszeitpunkt wird der jeweils 5. Tag nach Ablauf eines Meldequartals angegeben, wobei die Generierung 5 Tage zuvor angestoßen und die Übermittlung spätestens 12 Stunden vor Ablauf der Meldefrist angemahnt wird. Darüber hinaus wird vermerkt, dass für die Signierung keine qualifizierte elektronische Signatur erforderlich ist.
	Message.xsd	D	Als Parameter können über die zugehörige Message sowohl das Meldequartal (z. B. Q1-2010) als auch das Unternehmen übergeben werden, für das diese Meldung zu generieren ist.
	MultiNotificationProfile.xquery	D	Um je Meldesubjekt (Arbeitsstätte mit gesetzter statistischer Identifikationsnummer) eine Meldung / Benachrichtigung zu erzeugen, wird die Funktion des Multi-Messaging der T-BRS eingesetzt, welche für jede einzubeziehende Arbeitsstätte die Generierung einer eigenen Benachrichtigung auslöst. Darüber hinaus werden die Kriterien für den späteren Aufruf des Zuständigkeitsverzeichnisses eingetragen.
	MultiNotificationProfile.xsl	D	
	Notification.xsd	D	Die Generierung wird über ein Selektionsskript mit anschließendem Transformationsskript durchgeführt. Dabei wird auch das Profil einmalig mit transformiert.
	Notification.xsl	D	
	NotificationCore.xsl	D	Der Core beinhaltet in dieser Benachrichtigungsregel alle Merkmale mit Ausnahme der Felder zur Abweichungsbegründung.
	Profile.xsl	D	In der Profile-Transformation erfolgt der Abruf der Informationen aus dem Zuständigkeitsverzeichnis.

P23R

P23R: Spezifikationen zur Infrastruktur

	Selection.xquery	D	Die Generierung wird über ein Selektionsskript mit anschließendem Transformationskript durchgeführt. Dabei wird auch das Profil einmalig mit transformiert.
	Selection.xsd	D	
	Source.txt	D	Enthält eine Zusammenfassung der Inhalte der Benachrichtigungsregel.

9.2.3 ZUSTÄNDIGKEITSINFORMATIONEN „AGM“

Im Leistungskatalog (siehe Tabelle 36) werden die jeweilige Benachrichtigungsregel und die korrespondierende Leistung in Relation gesetzt:

TABELLE 36: AGM-PILOTDATEN – LEISTUNGSKATALOG

ruleName	Idservice	name	legalbasis
AGM_STAT_VERDIENSTERHEBUNG	p23r:service:de.p23r.cc:AGM_STAT_VERDIENSTERHEBUNG	Abgabe vierteljährliche Verdiensterhebung	VerdStatG
AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	Abgabe DEÜV-Jahresmeldung zur Sozialversicherung	DEÜV
AGM_SOZ_BG-JAHRESMELDUNG	p23r:service:de.p23r.cc:AGM_SOZ_BG-JAHRESMELDUNG	Abgabe jährlicher Lohnnachweis an die Berufsgenossenschaft	SGB IV und VII

Im Kriterienkatalog (siehe Tabelle 37) werden die verfügbaren Zuständigkeitskriterien definiert:

TABELLE 37: AGM-PILOTDATEN – KRITERIENKATALOG

idcriteria	Name
p23r:criteria:de.p23r.cc:fixed	Default Kriterium für feste Benachrichtigungsempfänger.
p23r:criteria:de.gkv.krankenkassen.bezeichnung	Die jeweilige Bezeichnung der gesetzlichen Krankenkassen.
p23r:criteria:de.dguv.bg.betriebsnummer	Die jeweilige Betriebsnummer der Berufsgenossenschaft.

Im Verzeichnis der Benachrichtigungsempfänger (siehe Tabelle 38) werden die fachlichen Empfänger der Benachrichtigung definiert:

TABELLE 38: AGM-PILOTDATEN – VERZEICHNIS DER BENACHRICHTIGUNGSEMPFÄNGER

idorg	name
p23r:org:de.bund:StBA	Statistisches Bundesamt
p23r:org:de.gkv:AOK_BW	AOK Baden-Württemberg
p23r:org:de.gkv:AOK_BY	AOK Bayern
p23r:org:de.gkv:AOK_BB	AOK Berlin Brandenburg

Anhang II: Bestückung der Leitstelle (TEST und AGM)

idorg	name
p23r:org:de.gkv:AOK_BH	AOK Bremen / Bremerhaven
p23r:org:de.gkv:AOK_HH	AOK Hamburg
p23r:org:de.gkv:AOK_HE	AOK Hessen
p23r:org:de.gkv:AOK_MV	AOK Mecklenburg-Vorpommern
p23r:org:de.gkv:AOK_NS	AOK Niedersachsen
p23r:org:de.gkv:AOK_RH	AOK Rheinland / Hamburg
p23r:org:de.gkv:AOK_RP	AOK Rheinland-Pfalz / Saarland
p23r:org:de.gkv:AOK_SS	AOK Sachsen
p23r:org:de.gkv:AOK_SA	AOK Sachsen-Anhalt
p23r:org:de.gkv:AOK_SH	AOK Schleswig-Holstein
p23r:org:de.gkv:AOK_TH	AOK Thüringen
p23r:org:de.gkv:AOK_WL	AOK Westfalen-Lippe
p23r:org:de.gkv:BEK	Barmer Ersatzkasse
p23r:org:de.gkv:DAK	Deutsche Angestellten-Krankenkasse
p23r:org:de.gkv:KNAPP	Knappschaft
p23r:org:de.gkv:SEE	See-Krankenkasse
p23r:org:de.gkv:TK	Techniker Krankenkasse
p23r:org:de.dguv:BGRCI	BG Rohstoffe und Chemische Industrie
p23r:org:de.dguv:BGETE	BG Energie Textil Elektro
p23r:org:de.dguv:BGV	BG Transport und Verkehrswirtschaft
p23r:org:de.dguv:BGHM	BG Holz und Metall
p23r:org:de.dguv:BGW	BG Gesundheitsdienste und Wohlfahrtspflege
p23r:org:de.dguv:VBG	BG Verwaltung

Weiterhin werden im Verzeichnis der Supportadressen (siehe Tabelle 39) die Ansprechpartner für den technischen Support festgehalten:

P23R

P23R: Spezifikationen zur Infrastruktur

TABELLE 39: AGM-PILOTDATEN – VERZEICHNIS DES TECHNISCHEN SUPPORTS

idorg	Name
p23r:org:de.p23r.cc:oeLSt	Die Öffentliche P23R-Leitstelle (im Laborbetrieb).

Im elektronischen Dienstverzeichnis (siehe Tabelle 40) werden die Kommunikationsendpunkte für die Kanäle der Benachrichtigungsübermittlung definiert:

TABELLE 40: AGM-PILOTDATEN – ELEKTRONISCHES DIENSTEVERZEICHNIS

ideservice	address	type
p23r:eservice:de.bund.stba:connect	https://www-idev.destatis.de/idev/OnlineMeldung	p23r:estypes:de.p23r.cc.estypes:eSTATISTIKcore
p23r:eservice:de.gkv.66993824:dav	sftp://vdu-ev.leitstelle.p23r.de/dav_66993824	p23r:estypes:de.p23r.cc.estypes:VDEUEV
p23r:eservice:de.gkv.01000251:dav	sftp://vdu-ev.leitstelle.p23r.de/dav_01000251	p23r:estypes:de.p23r.cc.estypes:VDEUEV
p23r:eservice:de.gkv.01000262:dav	sftp://vdu-ev.leitstelle.p23r.de/dav_01000262	p23r:estypes:de.p23r.cc.estypes:VDEUEV
p23r:eservice:de.gkv.05174740:dav	sftp://vdu-ev.leitstelle.p23r.de/dav_05174740	p23r:estypes:de.p23r.cc.estypes:VDEUEV
p23r:eservice:de.gkv.20158137:dav	sftp://vdu-ev.leitstelle.p23r.de/dav_20158137	p23r:estypes:de.p23r.cc.estypes:VDEUEV
p23r:eservice:de.gkv.29720865:dav	sftp://vdu-ev.leitstelle.p23r.de/dav_29720865	p23r:estypes:de.p23r.cc.estypes:VDEUEV
p23r:eservice:de.gkv.33526082:dav	sftp://vdu-ev.leitstelle.p23r.de/dav_33526082	p23r:estypes:de.p23r.cc.estypes:VDEUEV
p23r:eservice:de.gkv.34364249:dav	sftp://vdu-ev.leitstelle.p23r.de/dav_34364249	p23r:estypes:de.p23r.cc.estypes:VDEUEV
p23r:eservice:de.gkv.47860681:dav	sftp://vdu-ev.leitstelle.p23r.de/dav_47860681	p23r:estypes:de.p23r.cc.estypes:VDEUEV
p23r:eservice:de.gkv.64672791:dav	sftp://vdu-ev.leitstelle.p23r.de/dav_64672791	p23r:estypes:de.p23r.cc.estypes:VDEUEV
p23r:eservice:de.gkv.87880235:dav	sftp://vdu-ev.leitstelle.p23r.de/dav_87880235	p23r:estypes:de.p23r.cc.estypes:VDEUEV

Anhang II: Bestückung der Leitstelle (TEST und AGM)

ideservice	address	type
p23r:eservice:de.gkv.99086875:dav	sftp://vdu-ev. leitstelle.p23r.de/dav_99086875	p23r:estypes:de.p23r.cc.estypes:VDEUEV
p23r:eservice:de.dguv.bgrci:mail	mailto:bgrci@leitstelle.p23r.de	p23r:estypes:de.p23r.cc.estypes:BGviaMail
p23r:eservice:de.dguv.bgete:mail	mailto:bgete@leitstelle.p23r.de	p23r:estypes:de.p23r.cc.estypes:BGviaMail
p23r:eservice:de.dguv.bgv:mail	mailto:bgv@leitstelle.p23r.de	p23r:estypes:de.p23r.cc.estypes:BGviaMail
p23r:eservice:de.dguv.bghm:mail	mailto:bghm@leitstelle.p23r.de	p23r:estypes:de.p23r.cc.estypes:BGviaMail
p23r:eservice:de.dguv.bgw:mail	mailto:bgw@leitstelle.p23r.de	p23r:estypes:de.p23r.cc.estypes:BGviaMail
p23r:eservice:de.dguv.vbg:mail	mailto:vbg@leitstelle.p23r.de	p23r:estypes:de.p23r.cc.estypes:BGviaMail

Weiterhin werden im Verzeichnis (siehe Tabelle 41) die Ansprechpartner für den fachlichen Support festgehalten:

TABELLE 41: AGM-PILOTDATEN – VERZEICHNIS DES FACHLICHEN SUPPORTS

ideservice	address	type
p23r:eservice:de.p23r.cc:oeLSt.postal	Fraunhofer FOKUS, Kaiserin August Allee, Berlin	Postadresse
p23r:eservice:de.p23r.cc:oeLSt.webform	http://leitstelle.p23r.de/	URL
p23r:eservice:de.p23r.cc:oeLSt.email	mailto:info@leitstelle.p23r.de	E-Mail

Darüber hinaus werden die Verknüpfungen für die Ermittlung der Benachrichtigungsempfänger (siehe Tabelle 42) abgebildet (MappingReceiver):

P23R

P23R: Spezifikationen zur Infrastruktur

TABELLE 42: AGM-PILOTDATEN – MAPPINGRECEIVER

idservice (siehe Tabelle 36)	Suchkriterium		idorg (siehe Tabelle 38)
	idcriteria (siehe Tabelle 37)	value	
p23r:service:de.p23r.cc:AGM_STAT_VERDIENST ERHEBUNG	p23r:criteria:de.p23r.cc:fixed	`` (empty, not NULL)	p23r:org:de.bund:StBA
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:criteria:de.gkv:krankenkassen.bezeichnung	AOK	p23r:org:de.gkv:AOK_BW
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:criteria:de.gkv:krankenkassen.bezeichnung	AOK_BW	p23r:org:de.gkv:AOK_BW
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:criteria:de.gkv:krankenkassen.bezeichnung	AOK_BY	p23r:org:de.gkv:AOK_BY
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:criteria:de.gkv:krankenkassen.bezeichnung	AOK_BB	p23r:org:de.gkv:AOK_BB
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:criteria:de.gkv:krankenkassen.bezeichnung	AOK_BH	p23r:org:de.gkv:AOK_BH
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:criteria:de.gkv:krankenkassen.bezeichnung	AOK_HH	p23r:org:de.gkv:AOK_HH
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:criteria:de.gkv:krankenkassen.bezeichnung	AOK_HE	p23r:org:de.gkv:AOK_HE
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:criteria:de.gkv:krankenkassen.bezeichnung	AOK_MV	p23r:org:de.gkv:AOK_MV
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:criteria:de.gkv:krankenkassen.bezeichnung	AOK_NS	p23r:org:de.gkv:AOK_NS
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:criteria:de.gkv:krankenkassen.bezeichnung	AOK_RH	p23r:org:de.gkv:AOK_RH

Anhang II: Bestückung der Leitstelle (TEST und AGM)

idservice (siehe Tabelle 36)	Suchkriterium		idorg (siehe Tabelle 38)
	idcriteria (siehe Tabelle 37)	value	
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:criteria:de.gkv:krankenkassen.bezeichnung	AOK_RP	p23r:org:de.gkv:AOK_RP
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:criteria:de.gkv:krankenkassen.bezeichnung	AOK_SS	p23r:org:de.gkv:AOK_SS
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:criteria:de.gkv:krankenkassen.bezeichnung	AOK_SA	p23r:org:de.gkv:AOK_SA
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:criteria:de.gkv:krankenkassen.bezeichnung	AOK_SH	p23r:org:de.gkv:AOK_SH
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:criteria:de.gkv:krankenkassen.bezeichnung	AOK_TH	p23r:org:de.gkv:AOK_TH
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:criteria:de.gkv:krankenkassen.bezeichnung	AOK_WL	p23r:org:de.gkv:AOK_WL
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:criteria:de.gkv:krankenkassen.bezeichnung	Barmer	p23r:org:de.gkv:BEK
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:criteria:de.gkv:krankenkassen.bezeichnung	DAK	p23r:org:de.gkv:DAK
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:criteria:de.gkv:krankenkassen.bezeichnung	Knappschaft	p23r:org:de.gkv:KNAPP
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:criteria:de.gkv:krankenkassen.bezeichnung	Seekasse	p23r:org:de.gkv:SEE
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:criteria:de.gkv:krankenkassen.bezeichnung	TK	p23r:org:de.gkv:TK
p23r:service:de.p23r.cc:AGM_SOZ_BG-JAHRESMELDUNG	p23r:criteria:de.dguv:bg.betriebsnummer	52742028	p23r:org:de.dguv:BGR_CI
p23r:service:de.p23r.cc:AGM_SOZ_BG	p23r:criteria:de.dguv:bg.betriebsnummer	37916971	p23r:org:de.dguv:BGETE
p23r:service:de.p23r.cc:AGM_SOZ_BG	p23r:criteria:de.dguv:bg.betriebsnummer	15141364	p23r:org:de.dguv:BGV
p23r:service:de.p23r.cc:AGM_SOZ_BG	p23r:criteria:de.dguv:bg.betriebsnummer	63886548	p23r:org:de.dguv:BGHM

P23R

P23R: Spezifikationen zur Infrastruktur

idservice (siehe Tabelle 36)	Suchkriterium		idorg (siehe Tabelle 38)
	idcriteria (siehe Tabelle 37)	value	
p23r:service:de.p23r.cc:AGM_SOZ_BG	p23r:criteria:de.dguv:bg.betriebsnummer	32064004	p23r:org:de.dguv:BGHM
p23r:service:de.p23r.cc:AGM_SOZ_BG	p23r:criteria:de.dguv:bg.betriebsnummer	15186676	p23r:org:de.dguv:BGW
p23r:service:de.p23r.cc:AGM_SOZ_BG	p23r:criteria:de.dguv:bg.betriebsnummer	15250094	p23r:org:de.dguv:VBG

Schlussendlich werden die Verknüpfungen für die Ermittlung der Kommunikationsendpunkte (siehe Tabelle 43) abgebildet (MappingEservice):

TABELLE 43: AGM-PILOTDATEN – MAPPINGESERVICE

Idservice (siehe Tabelle 36)	Idorg (siehe Tabelle 38)	ideservice (siehe Tabelle 40)	Prio
p23r:service:de.p23r.cc:AGM_STAT_VERDIENSTERHEBUNG	p23r:org:de.bund:StBA	p23r:eservice:de.bund.stba:connect	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:org:de.gkv:AOK_BW	p23r:eservice:de.gkv.64672791:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:org:de.gkv:AOK_BY	p23r:eservice:de.gkv.87880235:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:org:de.gkv:AOK_BB	p23r:eservice:de.gkv.01000262:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:org:de.gkv:AOK_BH	p23r:eservice:de.gkv.20158137:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:org:de.gkv:AOK_HH	p23r:eservice:de.gkv.01000251:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:org:de.gkv:AOK_HE	p23r:eservice:de.gkv.47860681:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:org:de.gkv:AOK_MV	p23r:eservice:de.gkv.01000251:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:org:de.gkv:AOK_NS	p23r:eservice:de.gkv.29720865:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-	p23r:org:de.gkv:AOK_RH	p23r:eservice:de.gkv.343642	1

Anhang II: Bestückung der Leitstelle (TEST und AGM)

Idservice (siehe Tabelle 36)	Idorg (siehe Tabelle 38)	ideservice (siehe Tabelle 40)	Prio
JAHRESMELDUNG		49:dav	
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:org:de.gkv:AOK_RP	p23r:eservice:de.gkv.47860681:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:org:de.gkv:AOK_SS	p23r:eservice:de.gkv.05174740:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV-JAHRESMELDUNG	p23r:org:de.gkv:AOK_SA	p23r:eservice:de.gkv.01000262:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:org:de.gkv:AOK_SH	p23r:eservice:de.gkv.01000251:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:org:de.gkv:AOK_TH	p23r:eservice:de.gkv.47860681:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:org:de.gkv:AOK_WL	p23r:eservice:de.gkv.33526082:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:org:de.gkv:BEK	p23r:eservice:de.gkv.66993824:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:org:de.gkv:DAK	p23r:eservice:de.gkv.66993824:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:org:de.gkv:KNAPP	p23r:eservice:de.gkv.66993824:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:org:de.gkv:SEE	p23r:eservice:de.gkv.99086875:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_DEUEV	p23r:org:de.gkv:TK	p23r:eservice:de.gkv.66993824:dav	1
p23r:service:de.p23r.cc:AGM_SOZ_BG	p23r:org:de.dguv:BGRCI	p23r:eservice:de.dguv.bgrci:mail	1
p23r:service:de.p23r.cc:AGM_SOZ_BG	p23r:org:de.dguv:BGETE	p23r:eservice:de.dguv.bgete:mail	1
p23r:service:de.p23r.cc:AGM_SOZ_BG	p23r:org:de.dguv:BGV	p23r:eservice:de.dguv.bgv:mail	1
p23r:service:de.p23r.cc:AGM_SOZ_BG	p23r:org:de.dguv:BGHM	p23r:eservice:de.dguv.bghm:mail	1

P23R

P23R: Spezifikationen zur Infrastruktur

Idservice (siehe Tabelle 36)	Idorg (siehe Tabelle 38)	ideservice (siehe Tabelle 40)	Prio
p23r:service:de.p23r.cc:AGM_SOZ_BG	p23r:org:de.dguv:BGW	p23r:eservice:de.dguv.bgw:mail	1
p23r:service:de.p23r.cc:AGM_SOZ_BG	p23r:org:de.dguv:VBG	p23r:eservice:de.dguv.vbg:mail	1

10 ANHANG III: TESTDATENSATZ UND PRÜFERGEBNISSE (AGM)

Zur Prüfung der korrekten Abarbeitung der Benachrichtigungsregeln kommt ein Testdatensatz nach den in Kapitel 10 des Lastenhefts [18] enthaltenen Anforderungen zum Einsatz. Dieser Testdatensatz, der im Folgenden spezifiziert ist, dient als Grundlage der testweisen Abarbeitung der Benachrichtigungsregeln, wobei die jeweils gegen zu prüfende Generierungsergebnisse ebenfalls im Folgenden aufgeführt sind.

10.1 TESTDATENSATZ DER TEILDATENMODELLE „UNTERNEHMEN“ UND „MITARBEITER“

Der Testdatensatz für das Teildatenmodell „Unternehmen“ des Pivotdatenmodells in der für das Szenario AGM definierten Datensatzbeschreibung ist wie folgt ausgeprägt:

```
<?xml version="1.0" encoding="UTF-8"?>
<unternehmensdaten>
  <unternehmen name="Autohaus Hartmann GmbH und Co. KG" taetigkeitsschwerpunkt="Handel mit KFZ und Werkstattbetrieb">
    <anschrift strasse="Hauptstraße" ort="Musterstadt" plz="12345" hausnummer="83"/>
    <kontakt telefon="+4911123456" vorname="Horst" nachname="Hartmann" email="horst.hartmann@mailprovider.de"/>
    <uvtraeger mitgliedsnummer="12345" kennung="52742028"/>
    <betriebsstaette taetigkeitsschwerpunkt="Werkstatt" bezeichnung="BS_001" betriebsnummer="123456789">
      <anschrift strasse="Werkstattstraße" ort="Obermusterstadt" plz="12345" hausnummer="99"/>
      <arbeitsstaette taetigkeitsschwerpunkt="Verkauf" bezeichnung="AS_001" statistikidentnummer="12345">
        <anschrift strasse="Werkstattstraße" ort="Obermusterstadt" plz="12345" hausnummer="99"/>
      </arbeitsstaette>
    </betriebsstaette>
    <betriebsstaette taetigkeitsschwerpunkt="Verkauf" bezeichnung="BS_002" betriebsnummer="987654321">
      <anschrift strasse="Handelsstraße" ort="Untermusterstadt" plz="12345" hausnummer="99"/>
      <arbeitsstaette taetigkeitsschwerpunkt="Verkauf" bezeichnung="AS_002">
        <anschrift strasse="Handelsstraße" ort="Untermusterstadt" plz="12345" hausnummer="99"/>
      </arbeitsstaette>
    </betriebsstaette>
  </unternehmen>
  <unternehmen name="Autohaus Hartmann Verwaltungs GmbH" taetigkeitsschwerpunkt="Bereitstellung von Verwaltungsdiensten für die KFZ-Branche">
    <anschrift strasse="Hauptstraße" ort="Musterstadt" plz="12345" hausnummer="83"/>
    <kontakt telefon="+4911123456" vorname="Horst" nachname="Hartmann" email="horst.hartmann@mailprovider.de"/>
    <uvtraeger mitgliedsnummer="54321" kennung="52742028"/>
  </unternehmen>
</unternehmensdaten>
```

P23R

P23R: Spezifikationen zur Infrastruktur

```
<betriebsstaette taetigkeitsschwerpunkt="Verwaltung" bezeichnung="BS_003" be-
triebsnummer="123459876">
  <anschrift strasse="Hauptstraße" ort="Musterstadt" plz="12345" hausnum-
mer="99"/>
  <arbeitsstaette taetigkeitsschwerpunkt="Verkauf" bezeichnung="AS_003">
    <anschrift strasse="Hauptstraße" ort="Musterstadt" plz="12345"
hausnummer="83"/>
  </arbeitsstaette>
</betriebsstaette>
</unternehmen>
</unternehmensdaten>
```

Der Testdatensatz für das Teildatenmodell „Mitarbeiter“ des Pivotdatenmodells in der für das Szenario AGM definierten Datensatzbeschreibung ist wie folgt ausgeprägt:

```
<?xml version="1.0" encoding="UTF-8"?>
<mitarbeiterdaten>
  <mitarbeiter taetigkeit="1234" bildungsabschluss="5" geschlecht="maennlich" id="MA_001">
    <name nachname="Hartmann" vorname="Horst" von="2010-01-01" bis="2010-12-31"></name>
    <staatsangehoerigkeit wert="Deutsch" von="2010-01-01" bis="2010-12-31"></staatsangehoerigkeit>
    <ostWestMerkmal wert="west" von="2010-01-01" bis="2010-12-31"></ostWestMerkmal>
    <sozialversicherungsnummer wert="123456789"></sozialversicherungsnummer>
    <adresse strasse="Hauptstraße" ort="Musterstadt" plz="12345" hausnummer="83" von="2010-01-01" bis="2010-12-31"></adresse>
    <arbeitsstaette bezeichnung="AS_003" von="2010-01-01" bis="2010-12-31"></arbeitsstaette>
    <charakteristik>
      <status schluessel="28" von="2010-01-01" bis="2010-12-31"></status>
      <leistungsgruppe bezeichnung="LG1" von="2010-01-01" bis="2010-12-31"></leistungsgruppe>
      <krankenkasse bezeichnung="TK" von="2010-01-01" bis="2010-12-31"></krankenkasse>
      <beitragseinordnung alv="0" kv="0" rv="0" pv="0" von="2010-01-01" bis="2010-12-31"></beitragseinordnung>
      <gefahrenklasse bezeichnung="2" von="2010-01-01" bis="2010-12-31"></gefahrenklasse>
      <arbeitszeit umfang="183" von="2010-01-01" bis="2010-12-31"></arbeitszeit>
    </charakteristik>
    <verguetung am="2010-01-31" hoehe="3500" art="grundgehalt" />
    <verguetung am="2010-02-28" hoehe="3500" art="grundgehalt" />
    <verguetung am="2010-03-31" hoehe="3500" art="grundgehalt" />
    <verguetung am="2010-04-30" hoehe="3500" art="grundgehalt" />
    <verguetung am="2010-05-31" hoehe="3500" art="grundgehalt" />
    <verguetung am="2010-06-30" hoehe="3500" art="grundgehalt" />
    <verguetung am="2010-07-31" hoehe="3500" art="grundgehalt" />
    <verguetung am="2010-08-31" hoehe="3500" art="grundgehalt" />
    <verguetung am="2010-09-30" hoehe="3500" art="grundgehalt" />
    <verguetung am="2010-10-31" hoehe="3500" art="grundgehalt" />
```

Anhang III: Testdatensatz und Prüfergebnisse (AGM)

```

        <verguetung am="2010-11-30" hoehe="3500" art="grundgehalt" />
        <verguetung am="2010-12-31" hoehe="3500" art="grundgehalt" />
    </mitarbeiter>
    <mitarbeiter taetigkeit="1234" ausbildungsabschluss="3" geschlecht="maennlich"
id="MA_002">
        <name nachname="Müller" vorname="Christian" von="2010-01-01" bis="2010-12-
31"></name>
        <staatsangehoerigkeit wert="Deutsch" von="2010-01-01" bis="2010-12-
31"></staatsangehoerigkeit>
        <ostWestMerkmal wert="ost" von="2010-01-01" bis="2010-12-31"></ostWestMerkmal>
        <sozialversicherungsnummer wert="123456789"></sozialversicherungsnummer>
        <adresse strasse="Bahnhofstraße" ort="Musterstadt" plz="12345" hausnummer="13"
von="2010-01-01" bis="2010-12-31"></adresse>
        <arbeitsstaette bezeichnung="AS_001" von="2010-01-01" bis="2010-12-
31"></arbeitsstaette>
        <charakteristik>
            <status schluessel="5" von="2010-01-01" bis="2010-12-31"></status>
            <leistungsgruppe bezeichnung="LG2" von="2010-01-01" bis="2010-12-
31"></leistungsgruppe>
            <krankenkasse bezeichnung="AOK" von="2010-01-01" bis="2010-07-
31"></krankenkasse>
            <krankenkasse bezeichnung="BEK" von="2010-08-01" bis="2010-12-
31"></krankenkasse>
            <beitragseinordnung alv="1" kv="1" rv="1" pv="1" von="2010-01-01"
bis="2010-12-31"></beitragseinordnung>
            <gefahrenklasse bezeichnung="3" von="2010-01-01" bis="2010-12-
31"></gefahrenklasse>
            <arbeitszeit umfang="175" von="2010-01-01" bis="2010-12-
31"></arbeitszeit>
        </charakteristik>
        <verguetung am="2010-01-31" hoehe="1925" art="grundgehalt" />
        <verguetung am="2010-02-28" hoehe="1925" art="grundgehalt" />
        <verguetung am="2010-03-31" hoehe="1925" art="grundgehalt" />
        <verguetung am="2010-04-30" hoehe="1925" art="grundgehalt" />
        <verguetung am="2010-05-31" hoehe="1925" art="grundgehalt" />
        <verguetung am="2010-06-30" hoehe="1925" art="grundgehalt" />
        <verguetung am="2010-07-31" hoehe="1925" art="grundgehalt" />
        <verguetung am="2010-08-31" hoehe="545" art="sonderzahlung" />
        <verguetung am="2010-08-31" hoehe="1925" art="grundgehalt" />
        <verguetung am="2010-09-30" hoehe="1925" art="grundgehalt" />
        <verguetung am="2010-10-31" hoehe="1925" art="grundgehalt" />
        <verguetung am="2010-11-30" hoehe="1925" art="grundgehalt" />
        <verguetung am="2010-12-31" hoehe="1925" art="grundgehalt" />
    </mitarbeiter>
    <mitarbeiter taetigkeit="1234" ausbildungsabschluss="3" geschlecht="weiblich"
id="MA_003">
        <name nachname="Schröder" vorname="Julia" von="2010-01-01" bis="2010-07-
31"></name>
        <name nachname="Schmidt" vorname="Julia" von="2010-08-01" bis="2010-12-
31"></name>

```

P23R

P23R: Spezifikationen zur Infrastruktur

```
<staatsangehoerigkeit wert="Deutsch" von="2010-01-01" bis="2010-12-31"></staatsangehoerigkeit>
<ostWestMerkmal wert="ost" von="2010-01-01" bis="2010-07-31"></ostWestMerkmal>
<ostWestMerkmal wert="west" von="2010-08-01" bis="2010-12-31"></ostWestMerkmal>
<sozialversicherungsnummer wert="123456789"></sozialversicherungsnummer>
<adresse strasse="Dorfstraße" ort="Musterstadt" plz="12345" hausnummer="7" von="2010-01-01" bis="2010-12-31"></adresse>
<arbeitsstaette bezeichnung="AS_001" von="2010-01-01" bis="2010-12-31"></arbeitsstaette>
<charakteristik>
  <status schluesssel="5" von="2010-01-01" bis="2010-12-31"></status>
  <leistungsgruppe bezeichnung="LG2" von="2010-01-01" bis="2010-12-31"></leistungsgruppe>
  <krankenkasse bezeichnung="AOK" von="2010-01-01" bis="2010-07-31"></krankenkasse>
  <krankenkasse bezeichnung="BEK" von="2010-08-01" bis="2010-12-31"></krankenkasse>
  <beitragseinordnung alv="1" kv="1" rv="1" pv="1" von="2010-01-01" bis="2010-12-31"></beitragseinordnung>
  <gefahrenklasse bezeichnung="3" von="2010-01-01" bis="2010-12-31"></gefahrenklasse>
  <arbeitszeit umfang="175" von="2010-01-01" bis="2010-12-31"></arbeitszeit>
</charakteristik>
<verguetung am="2010-01-31" hoehe="1540" art="grundgehalt" />
<verguetung am="2010-02-28" hoehe="1540" art="grundgehalt" />
<verguetung am="2010-03-31" hoehe="1540" art="grundgehalt" />
<verguetung am="2010-04-30" hoehe="1540" art="grundgehalt" />
<verguetung am="2010-05-31" hoehe="1540" art="grundgehalt" />
<verguetung am="2010-06-30" hoehe="1540" art="grundgehalt" />
<verguetung am="2010-06-30" hoehe="250" art="sonderzahlung" />
<verguetung am="2010-07-31" hoehe="1540" art="grundgehalt" />
<verguetung am="2010-08-31" hoehe="1540" art="grundgehalt" />
<verguetung am="2010-09-30" hoehe="1540" art="grundgehalt" />
<verguetung am="2010-10-31" hoehe="1540" art="grundgehalt" />
<verguetung am="2010-11-30" hoehe="1540" art="grundgehalt" />
<verguetung am="2010-12-31" hoehe="1540" art="grundgehalt" />
<verguetung am="2010-12-31" hoehe="250" art="sonderzahlung" />
</mitarbeiter>
<mitarbeiter taetigkeit="1234" ausbildungsabschluss="2" geschlecht="maennlich" id="MA_004">
  <name vorname="Sebastian" nachname="Fischer" von="2010-01-01" bis="2010-12-31"></name>
  <staatsangehoerigkeit wert="Deutsch" von="2010-01-01" bis="2010-12-31"></staatsangehoerigkeit>
  <ostWestMerkmal wert="west" von="2010-01-01" bis="2010-12-31"></ostWestMerkmal>
  <sozialversicherungsnummer wert="123456789"></sozialversicherungsnummer>
  <adresse strasse="Kaiserstraße" ort="Musterstadt" plz="12345" hausnummer="113" von="2010-01-01" bis="2010-12-31"></adresse>
```

Anhang III: Testdatensatz und Prüfergebnisse (AGM)

```

        <arbeitsstaette    bezeichnung="AS_001"    von="2010-01-01"    bis="2010-08-
31"></arbeitsstaette>
        <arbeitsstaette    bezeichnung="AS_002"    von="2010-09-01"    bis="2010-12-
31"></arbeitsstaette>
        <charakteristik>
            <status schluessel="5" von="2010-01-01" bis="2010-12-31"></status>
            <leistungsgruppe    bezeichnung="LG3"    von="2010-01-01"    bis="2010-12-
31"></leistungsgruppe>
            <krankenkasse    bezeichnung="AOK"    von="2010-01-01"    bis="2010-12-
31"></krankenkasse>
            <beitragseinordnung    alv="1"    kv="1"    rv="1"    pv="1"    von="2010-01-01"
bis="2010-12-31"></beitragseinordnung>
            <gefahrenklasse    bezeichnung="3"    von="2010-01-01"    bis="2010-12-
31"></gefahrenklasse>
            <arbeitszeit    umfang="163"    von="2010-01-01"    bis="2010-12-
31"></arbeitszeit>
        </charakteristik>
        <verguetung am="2010-01-31" hoehe="1540" art="grundgehalt" />
        <verguetung am="2010-02-28" hoehe="1540" art="grundgehalt" />
        <verguetung am="2010-03-31" hoehe="1540" art="grundgehalt" />
        <verguetung am="2010-04-30" hoehe="1540" art="grundgehalt" />
        <verguetung am="2010-05-31" hoehe="1540" art="grundgehalt" />
        <verguetung am="2010-06-30" hoehe="1540" art="grundgehalt" />
        <verguetung am="2010-07-31" hoehe="1540" art="grundgehalt" />
        <verguetung am="2010-08-31" hoehe="1540" art="grundgehalt" />
        <verguetung am="2010-09-30" hoehe="1540" art="grundgehalt" />
        <verguetung am="2010-10-31" hoehe="1540" art="grundgehalt" />
        <verguetung am="2010-11-30" hoehe="1540" art="grundgehalt" />
        <verguetung am="2010-12-31" hoehe="1540" art="grundgehalt" />
    </mitarbeiter>
    <mitarbeiter    taetigkeit="1234"    ausbildungsabschluss="2"    geschlecht="weiblich"
id="MA_005">
        <name    vorname="Sandra"    nachname="Wagner"    von="2010-01-01"    bis="2010-12-
31"></name>
        <staatsangehoerigkeit    wert="Deutsch"    von="2010-01-01"    bis="2010-12-
31"></staatsangehoerigkeit>
        <ostWestMerkmal wert="ost" von="2010-01-01" bis="2010-12-31"></ostWestMerkmal>
        <sozialversicherungsnummer wert="123456789"></sozialversicherungsnummer>
        <adresse strasse="Schillerstraße" ort="Musterstadt" plz="12345" hausnummer="3"
von="2010-01-01" bis="2010-12-31"></adresse>
        <arbeitsstaette    bezeichnung="AS_002"    von="2010-01-01"    bis="2010-08-
31"></arbeitsstaette>
        <arbeitsstaette    bezeichnung="AS_001"    von="2010-09-01"    bis="2010-12-
31"></arbeitsstaette>
        <charakteristik>
            <status schluessel="5" von="2010-01-01" bis="2010-12-31"></status>
            <leistungsgruppe    bezeichnung="LG3"    von="2010-01-01"    bis="2010-12-
31"></leistungsgruppe>
            <krankenkasse    bezeichnung="AOK"    von="2010-01-01"    bis="2010-12-
31"></krankenkasse>

```

P23R

P23R: Spezifikationen zur Infrastruktur

```
<beitragseinordnung alv="1" kv="1" rv="1" pv="1" von="2010-01-01"
bis="2010-12-31"></beitragseinordnung>
<gefahrenklasse bezeichnung="3" von="2010-01-01" bis="2010-12-
31"></gefahrenklasse>
<arbeitszeit umfang="163" von="2010-01-01" bis="2010-12-
31"></arbeitszeit>
</charakteristik>
<verguetung am="2010-01-31" hoehe="1230" art="grundgehalt" />
<verguetung am="2010-02-28" hoehe="1230" art="grundgehalt" />
<verguetung am="2010-03-31" hoehe="1230" art="grundgehalt" />
<verguetung am="2010-04-30" hoehe="1230" art="grundgehalt" />
<verguetung am="2010-05-31" hoehe="1230" art="grundgehalt" />
<verguetung am="2010-06-30" hoehe="1230" art="grundgehalt" />
<verguetung am="2010-07-31" hoehe="1230" art="grundgehalt" />
<verguetung am="2010-08-31" hoehe="1230" art="grundgehalt" />
<verguetung am="2010-08-31" hoehe="250" art="sonderzahlung" />
<verguetung am="2010-09-30" hoehe="1230" art="grundgehalt" />
<verguetung am="2010-10-31" hoehe="1230" art="grundgehalt" />
<verguetung am="2010-11-30" hoehe="1230" art="grundgehalt" />
<verguetung am="2010-12-31" hoehe="1230" art="grundgehalt" />
</mitarbeiter>
<mitarbeiter taetigkeit="1234" ausbildungsabschluss="2" geschlecht="maennlich"
id="MA_006">
<name nachname="Richter" vorname="Benjamin" von="2010-01-01" bis="2010-12-
31"></name>
<staatsangehoerigkeit wert="Deutsch" von="2010-01-01" bis="2010-12-
31"></staatsangehoerigkeit>
<ostWestMerkmal wert="west" von="2010-01-01" bis="2010-12-31"></ostWestMerkmal>
<sozialversicherungsnummer wert="123456789"></sozialversicherungsnummer>
<adresse strasse="Ludwigstraße" ort="Musterstadt" plz="12345" hausnummer="94"
von="2010-01-01" bis="2010-12-31"></adresse>
<arbeitsstaette bezeichnung="AS_002" von="2010-05-01" bis="2010-12-
31"></arbeitsstaette>
<charakteristik>
<status schluessel="5" von="2010-05-01" bis="2010-12-31"></status>
<leistungsgruppe bezeichnung="LG3" von="2010-05-01" bis="2010-12-
31"></leistungsgruppe>
<krankenkasse bezeichnung="BEK" von="2010-05-01" bis="2010-07-
31"></krankenkasse>
<krankenkasse bezeichnung="AOK" von="2010-08-01" bis="2010-12-
31"></krankenkasse>
<beitragseinordnung alv="1" kv="1" rv="1" pv="1" von="2010-05-01"
bis="2010-12-31"></beitragseinordnung>
<gefahrenklasse bezeichnung="3" von="2010-05-01" bis="2010-12-
31"></gefahrenklasse>
<arbeitszeit umfang="163" von="2010-05-01" bis="2010-12-
31"></arbeitszeit>
<mehrfachbeschaeftigung von="2010-05-01" bis="2010-07-
31"></mehrfachbeschaeftigung>
</charakteristik>
<verguetung am="2010-05-31" hoehe="1230" art="grundgehalt" />
```

Anhang III: Testdatensatz und Prüfergebnisse (AGM)

```

    <verguetung am="2010-06-30" hoehe="1230" art="grundgehalt" />
    <verguetung am="2010-07-31" hoehe="1230" art="grundgehalt" />
    <verguetung am="2010-08-31" hoehe="1230" art="grundgehalt" />
    <verguetung am="2010-09-30" hoehe="1230" art="grundgehalt" />
    <verguetung am="2010-10-31" hoehe="1230" art="grundgehalt" />
    <verguetung am="2010-10-31" hoehe="175" art="sonderzahlung" />
    <verguetung am="2010-11-30" hoehe="1230" art="grundgehalt" />
    <verguetung am="2010-12-31" hoehe="1230" art="grundgehalt" />
  </mitarbeiter>
  <mitarbeiter taetigkeit="1234" ausbildungsabschluss="1" geschlecht="maennlich"
id="MA_007">
    <name nachname="Hofmann" vorname="Matthias" von="2010-01-01" bis="2010-12-
31"></name>
    <staatsangehoerigkeit wert="Deutsch" von="2010-01-01" bis="2010-12-
31"></staatsangehoerigkeit>
    <ostWestMerkmal wert="west" von="2010-01-01" bis="2010-12-31"></ostWestMerkmal>
    <sozialversicherungsnummer wert="123456789"></sozialversicherungsnummer>
    <adresse strasse="Goethestraße" ort="Musterstadt" plz="12345" hausnummer="203"
von="2010-01-01" bis="2010-12-31"></adresse>
    <arbeitsstaette bezeichnung="AS_001" von="2010-01-01" bis="2010-09-
30"></arbeitsstaette>
    <charakteristik>
      <status schluessel="3" von="2010-01-01" bis="2010-09-30"></status>
      <leistungsgruppe bezeichnung="LG3" von="2010-01-01" bis="2010-09-
30"></leistungsgruppe>
      <krankenkasse bezeichnung="AOK" von="2010-01-01" bis="2010-09-
30"></krankenkasse>
      <beitragseinordnung alv="1" kv="1" rv="1" pv="1" von="2010-01-01"
bis="2010-09-30"></beitragseinordnung>
      <gefahrenklasse bezeichnung="3" von="2010-01-01" bis="2010-09-
30"></gefahrenklasse>
      <arbeitszeit umfang="163" von="2010-01-01" bis="2010-09-
30"></arbeitszeit>
    </charakteristik>
    <verguetung am="2010-01-31" hoehe="615" art="grundgehalt" />
    <verguetung am="2010-02-28" hoehe="615" art="grundgehalt" />
    <verguetung am="2010-03-31" hoehe="615" art="grundgehalt" />
    <verguetung am="2010-04-30" hoehe="615" art="grundgehalt" />
    <verguetung am="2010-05-31" hoehe="615" art="grundgehalt" />
    <verguetung am="2010-06-30" hoehe="615" art="grundgehalt" />
    <verguetung am="2010-07-31" hoehe="615" art="grundgehalt" />
    <verguetung am="2010-08-31" hoehe="615" art="grundgehalt" />
    <verguetung am="2010-09-30" hoehe="615" art="grundgehalt" />
  </mitarbeiter>
  <mitarbeiter taetigkeit="1234" ausbildungsabschluss="1" geschlecht="weiblich"
id="MA_008">
    <name nachname="Wolf" vorname="Franziska" von="2010-01-01" bis="2010-12-
31"></name>
    <staatsangehoerigkeit wert="Deutsch" von="2010-01-01" bis="2010-12-
31"></staatsangehoerigkeit>

```

P23R

P23R: Spezifikationen zur Infrastruktur

```
<ostWestMerkmal wert="west" von="2010-01-01" bis="2010-12-31"></ostWestMerkmal>
<sozialversicherungsnummer wert="123456789"></sozialversicherungsnummer>
<adresse strasse="Parkstraße" ort="Musterstadt" plz="12345" hausnummer="17"
von="2010-01-01" bis="2010-12-31"></adresse>
<arbeitsstaette bezeichnung="AS_001" von="2010-03-01" bis="2010-04-
30"></arbeitsstaette>
<arbeitsstaette bezeichnung="AS_002" von="2010-05-01" bis="2010-09-
30"></arbeitsstaette>
<arbeitsstaette bezeichnung="AS_001" von="2010-10-01" bis="2010-12-
31"></arbeitsstaette>
<charakteristik>
<status schluesssel="3" von="2010-03-01" bis="2010-12-31"></status>
<leistungsgruppe bezeichnung="LG3" von="2010-03-01" bis="2010-12-
31"></leistungsgruppe>
<krankenkasse bezeichnung="AOK" von="2010-03-01" bis="2010-12-
31"></krankenkasse>
<beitragseinordnung alv="1" kv="1" rv="1" pv="1" von="2010-03-01"
bis="2010-12-31"></beitragseinordnung>
<gefahrenklasse bezeichnung="3" von="2010-03-01" bis="2010-12-
31"></gefahrenklasse>
<arbeitszeit umfang="163" von="2010-03-01" bis="2010-12-
31"></arbeitszeit>
</charakteristik>
<verguetung am="2010-03-31" hoehe="615" art="grundgehalt" />
<verguetung am="2010-04-30" hoehe="615" art="grundgehalt" />
<verguetung am="2010-05-31" hoehe="615" art="grundgehalt" />
<verguetung am="2010-06-30" hoehe="615" art="grundgehalt" />
<verguetung am="2010-07-31" hoehe="615" art="grundgehalt" />
<verguetung am="2010-07-31" hoehe="150" art="sonderzahlung" />
<verguetung am="2010-08-31" hoehe="615" art="grundgehalt" />
<verguetung am="2010-09-30" hoehe="615" art="grundgehalt" />
<verguetung am="2010-10-31" hoehe="615" art="grundgehalt" />
<verguetung am="2010-11-30" hoehe="615" art="grundgehalt" />
<verguetung am="2010-12-31" hoehe="615" art="grundgehalt" />
<verguetung am="2010-12-31" hoehe="75" art="sonderzahlung" />
</mitarbeiter>
<mitarbeiter taetigkeit="1234" ausbildungsabschluss="4" geschlecht="weiblich"
id="MA_009">
<name nachname="Fuchs" vorname="Kathrin" von="2010-01-01" bis="2010-12-
31"></name>
<staatsangehoerigkeit wert="Deutsch" von="2010-01-01" bis="2010-12-
31"></staatsangehoerigkeit>
<ostWestMerkmal wert="ost" von="2010-01-01" bis="2010-05-31"></ostWestMerkmal>
<ostWestMerkmal wert="west" von="2010-06-01" bis="2010-12-31"></ostWestMerkmal>
<sozialversicherungsnummer wert="123456789"></sozialversicherungsnummer>
<adresse strasse="Alte Dorfstraße" ort="Musterstadt" plz="12345" hausnum-
mer="19" von="2010-01-01" bis="2010-12-31"></adresse>
<arbeitsstaette bezeichnung="AS_001" von="2010-01-01" bis="2010-03-
31"></arbeitsstaette>
<arbeitsstaette bezeichnung="AS_003" von="2010-08-01" bis="2010-12-
31"></arbeitsstaette>
```


Anhang III: Testdatensatz und Prüfergebnisse (AGM)

```

    <charakteristik>
      <status schluesssel="5" von="2010-01-01" bis="2010-03-31"></status>
      <status schluesssel="5" von="2010-08-01" bis="2010-12-31"></status>
      <leistungsgruppe bezeichnung="LG5" von="2010-01-01" bis="2010-03-31"></leistungsgruppe>
      <leistungsgruppe bezeichnung="LG4" von="2010-08-01" bis="2010-12-31"></leistungsgruppe>
      <krankenkasse bezeichnung="AOK" von="2010-01-01" bis="2010-03-31"></krankenkasse>
      <krankenkasse bezeichnung="AOK" von="2010-08-01" bis="2010-12-31"></krankenkasse>
      <beitragseinordnung alv="1" kv="1" rv="1" pv="1" von="2010-01-01" bis="2010-03-31"></beitragseinordnung>
      <beitragseinordnung alv="1" kv="1" rv="1" pv="1" von="2010-08-01" bis="2010-12-31"></beitragseinordnung>
      <gefahrenklasse bezeichnung="1" von="2010-01-01" bis="2010-03-31"></gefahrenklasse>
      <gefahrenklasse bezeichnung="1" von="2010-08-01" bis="2010-12-31"></gefahrenklasse>
      <arbeitszeit umfang="75" von="2010-01-01" bis="2010-03-31"></arbeitszeit>
      <arbeitszeit umfang="75" von="2010-08-01" bis="2010-12-31"></arbeitszeit>
      <mehrfachbeschaeftigung von="2010-08-01" bis="2010-12-31"></mehrfachbeschaeftigung>
    </charakteristik>
    <verguetung am="2010-01-31" hoehe="375" art="grundgehalt" />
    <verguetung am="2010-02-28" hoehe="375" art="grundgehalt" />
    <verguetung am="2010-03-31" hoehe="375" art="grundgehalt" />
    <verguetung am="2010-08-31" hoehe="375" art="grundgehalt" />
    <verguetung am="2010-09-30" hoehe="375" art="grundgehalt" />
    <verguetung am="2010-10-31" hoehe="375" art="grundgehalt" />
    <verguetung am="2010-11-30" hoehe="375" art="grundgehalt" />
    <verguetung am="2010-12-31" hoehe="375" art="grundgehalt" />
  </mitarbeiter>
</mitarbeiterdaten>

```

10.2 PRÜFERGEBNISSE ZUR VIERTELJÄHRLICHE (STATISTISCHE) VERDIENSTERHEBUNG

Basierend auf dem zuvor dargestellten Testdatensatz für die Teildatenmodelle des Pivotdatenmodells erfolgt jeweils eine Benachrichtigungsgenerierung für die Quartal 1 – 4 des Jahres 2010.

Für das erste Quartal ergibt sich folgendes Prüfergebnis:

```

<?xml version="1.0" encoding="UTF-8"?>
<notification jahr="2010" quartal="Q1">
  <erhebungseinheit name="Autohaus Hartmann GmbH und Co. KG" identnummer="12345" wirtschaftlichetaetigkeit="Werkstatt">
    <anschrift strasse="Werkstattstraße" ort="Obermusterstadt" hausnummer="99" plz="12345"></anschrift>
    <ansprechpartner telefon="+4911123456" vorname="Horst" zuname="" nachname="Hartmann" email="horst.hartmann@mailprovider.de" />
  </erhebungseinheit>
</notification>

```

P23R

P23R: Spezifikationen zur Infrastruktur

```
</erhebungseinheit>
<vollzeitbeschaeftigte>
  <maennlich>
    <LG1 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
    <LG2 sonderzahlungen="0" personenmonate="3" arbeitsstunden="525" bruttoverdienst="5775" />
    <LG3 sonderzahlungen="0" personenmonate="6" arbeitsstunden="978" bruttoverdienst="6465" />
    <LG4 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
    <LG5 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
  </maennlich>
  <weiblich>
    <LG1 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
    <LG2 sonderzahlungen="0" personenmonate="3" arbeitsstunden="525" bruttoverdienst="4620" />
    <LG3 sonderzahlungen="0" personenmonate="1" arbeitsstunden="163" bruttoverdienst="615" />
    <LG4 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
    <LG5 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
  </weiblich>
</vollzeitbeschaeftigte>
<geringfuegigbeschaeftigte>
  <weiblich personenmonate="3" bruttoverdienst="1125"></weiblich>
</geringfuegigbeschaeftigte>
<abweichungsbegrueundung sonstigeGruende="false" schlechtwettergeld="false" kurzarbeit="false"
streik="false">
  <bemerkung></bemerkung>
</abweichungsbegrueundung>
</notification>
```

Für das zweite Quartal ergibt sich folgendes Prüfergebnis:

```
<?xml version="1.0" encoding="UTF-8"?>
<notification jahr="2010" quartal="Q2">
  <erhebungseinheit name="Autohaus Hartmann GmbH und Co. KG" identnummer="12345" wirtschaftli-
chetaetigkeit="Werkstatt">
    <anschrift strasse="Werkstattstraße" ort="Obermusterstadt" hausnummer="99"
plz="12345"></anschrift>
    <ansprechpartner telefon="+4911123456" vorname="Horst" zuname="" nachname="Hartmann"
email="horst.hartmann@mailprovider.de" />
  </erhebungseinheit>
  <vollzeitbeschaeftigte>
    <maennlich>
      <LG1 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
      <LG2 sonderzahlungen="250" personenmonate="3" arbeitsstunden="525" bruttoverdienst="5775"
/>
      <LG3 sonderzahlungen="0" personenmonate="6" arbeitsstunden="978" bruttoverdienst="6465" />
      <LG4 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
      <LG5 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
    </maennlich>
    <weiblich>
      <LG1 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
      <LG2 sonderzahlungen="0" personenmonate="3" arbeitsstunden="525" bruttoverdienst="4620" />
    </weiblich>
  </vollzeitbeschaeftigte>
</notification>
```

Anhang III: Testdatensatz und Prüfergebnisse (AGM)

```

<LG3 sonderzahlungen="0" personenmonate="1" arbeitsstunden="163" bruttoverdienst="615" />
<LG4 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
<LG5 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
</weiblich>
</vollzeitbeschaeftigte>
<abweichungsbegrueundung sonstigeGruende="false" schlechtwettergeld="false" kurzarbeit="false"
streik="false">
  <bemerkung></bemerkung>
</abweichungsbegrueundung>
</notification>

```

Für das dritte Quartal ergibt sich folgendes Prüfergebnis:

```

<?xml version="1.0" encoding="UTF-8"?>
<notification jahr="2010" quartal="Q3">
  <erhebungseinheit name="Autohaus Hartmann GmbH und Co. KG" identnummer="12345" wirtschaftli-
chetaetigkeit="Werkstatt">
    <anschrift strasse="Werkstattstraße" ort="Obermusterstadt" hausnummer="99"
plz="12345"></anschrift>
    <ansprechpartner telefon="+4911123456" vorname="Horst" zuname="" nachname="Hartmann"
email="horst.hartmann@mailprovider.de" />
  </erhebungseinheit>
  <vollzeitbeschaeftigte>
    <maennlich>
      <LG1 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
      <LG2 sonderzahlungen="545" personenmonate="3" arbeitsstunden="525" bruttoverdienst="5775"
/>
      <LG3 sonderzahlungen="0" personenmonate="5" arbeitsstunden="1014" bruttoverdienst="6465" />
      <LG4 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
      <LG5 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
    </maennlich>
    <weiblich>
      <LG1 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
      <LG2 sonderzahlungen="0" personenmonate="3" arbeitsstunden="525" bruttoverdienst="4620" />
      <LG3 sonderzahlungen="0" personenmonate="1" arbeitsstunden="163" bruttoverdienst="1230" />
      <LG4 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
      <LG5 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
    </weiblich>
  </vollzeitbeschaeftigte>
  <abweichungsbegrueundung sonstigeGruende="false" schlechtwettergeld="false" kurzarbeit="false"
streik="false">
    <bemerkung></bemerkung>
  </abweichungsbegrueundung>
</notification>

```

Für das vierte Quartal ergibt sich folgendes Prüfergebnis:

```

<?xml version="1.0" encoding="UTF-8"?>
<notification jahr="2010" quartal="Q4">
  <erhebungseinheit name="Autohaus Hartmann GmbH und Co. KG" identnummer="12345" wirtschaftli-
chetaetigkeit="Werkstatt">

```

P23R

P23R: Spezifikationen zur Infrastruktur

```
<anschrift strasse="Werkstattstraße" ort="Obermusterstadt" hausnummer="99"
plz="12345"></anschrift>
<ansprechpartner telefon="+4911123456" vorname="Horst" zuname="" nachname="Hartmann"
email="horst.hartmann@mailprovider.de" />
</erhebungseinheit>
<vollzeitbeschaeftigte>
<maennlich>
<LG1 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
<LG2 sonderzahlungen="0" personenmonate="3" arbeitsstunden="525" bruttoverdienst="5775" />
<LG3 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
<LG4 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
<LG5 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
</maennlich>
<weiblich>
<LG1 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
<LG2 sonderzahlungen="250" personenmonate="3" arbeitsstunden="525" bruttoverdienst="4620"
/>
<LG3 sonderzahlungen="75" personenmonate="6" arbeitsstunden="978" bruttoverdienst="5535" />
<LG4 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
<LG5 sonderzahlungen="0" personenmonate="0" arbeitsstunden="0" bruttoverdienst="0" />
</weiblich>
</vollzeitbeschaeftigte>
<abweichungsbegruendung sonstigeGruende="false" schlechtwettergeld="false" kurzarbeit="false"
streik="false">
<bemerkung></bemerkung>
</abweichungsbegruendung>
</notification>
```

10.3 PRÜFERGEBNISSE ZUR DEÜV-JAHRESMELDUNG ZUR SOZIALVERSICHERUNG

Basierend auf dem zuvor dargestellten Testdatensatz für die Teildatenmodelle des Pivotdatenmodells erfolgt eine Benachrichtigungsgenerierung für alle Unternehmen und Mitarbeiter.

Für die im Unternehmen „Autohaus Hartmann GmbH und Co. KG“ bei der „AOK“ versichern Mitarbeiter ergibt sich folgendes Prüfergebnis:

```
<?xml version="1.0" encoding="UTF-8"?>
<notification berichtsjaehr="2010">
  <mitarbeiter personalnummer="MA_004" versicherungsnummer="123456789">
    <personaldaten>
      <name>Fischer</name>
      <vorname>Sebastian</vorname>
      <vorsatz></vorsatz>
      <zusatz></zusatz>
      <titel></titel>
      <strasseUndHausnummer>Kaiserstraße 113</strasseUndHausnummer>
      <land>Deutschland</land>
      <plz>12345</plz>
      <ort>Musterstadt</ort>
```

Anhang III: Testdatensatz und Prüfergebnisse (AGM)

```

        <entgeltInGleitzone>0</entgeltInGleitzone>
        <namensaenderung>>false</namensaenderung>
    </personaldaten>
    <beschaeftigung>
        <beschaeftigungszeit>
            <von>2010-09-01</von>
            <bis>2010-12-31</bis>
        </beschaeftigungszeit>
        <betriebsnummer>987654321</betriebsnummer>
        <personengruppe>101</personengruppe>
        <mehrfachbeschaeftigung>>false</mehrfachbeschaeftigung>
        <ostWestMerkmal>west</ostWestMerkmal>
        <beitragsgruppen>
            <kv>1</kv>
            <rv>1</rv>
            <alv>1</alv>
            <pv>1</pv>
        </beitragsgruppen>
        <taetigkeitsschluessel>12342</taetigkeitsschluessel>
        <staatsangehoerigkeit>000</staatsangehoerigkeit>
        <beitragspflichtigesBruttoarbeitsentgelt>
            <Euro>6160</Euro>
        </beitragspflichtigesBruttoarbeitsentgelt>
        <statuskennzeichen></statuskennzeichen>
    </beschaeftigung>
    <unfallversicherung>
        <betriebsnummerUVTraeger>52742028</betriebsnummerUVTraeger>
        <mitgliedsnummerBeiUVTraeger>12345</mitgliedsnummerBeiUVTraeger>
        <gefahrenstellen>
            <betriebsnummer>987654321</betriebsnummer>
            <tarif>3</tarif>
            <entgelt>6160</entgelt>
            <arbeitsstunden>652</arbeitsstunden>
        </gefahrenstellen>
    </unfallversicherung>
    <einzugsstelle bezeichnung="AOK"></einzugsstelle>
    <absender>
        <firma>Autohaus Hartmann GmbH und Co. KG</firma>
        <strasse>Hauptstraße 83</strasse>
        <plz>12345</plz>
        <ort>Musterstadt</ort>
    </absender>
</mitarbeiter>
<mitarbeiter personalnummer="MA_005" versicherungsnummer="123456789">
    <personaldaten>
        <name>Wagner</name>

```

P23R

P23R: Spezifikationen zur Infrastruktur

```
<vorname>Sandra</vorname>
<vorsatz></vorsatz>
<zusatz></zusatz>
<titel></titel>
<strasseUndHausnummer>Schillerstraße 3</strasseUndHausnummer>
<land>Deutschland</land>
<plz>12345</plz>
<ort>Musterstadt</ort>
<entgeltInGleitzone>0</entgeltInGleitzone>
<namensaenderung>false</namensaenderung>
</personaldaten>
<beschaeftigung>
  <beschaeftigungszeit>
    <von>2010-09-01</von>
    <bis>2010-12-31</bis>
  </beschaeftigungszeit>
  <betriebsnummer>123456789</betriebsnummer>
  <personengruppe>101</personengruppe>
  <mehrfachbeschaeftigung>false</mehrfachbeschaeftigung>
  <ostWestMerkmal>ost</ostWestMerkmal>
  <beitragsgruppen>
    <kv>1</kv>
    <rv>1</rv>
    <alv>1</alv>
    <pv>1</pv>
  </beitragsgruppen>
  <taetigkeitsschluessel>12342</taetigkeitsschluessel>
  <staatsangehoerigkeit>000</staatsangehoerigkeit>
  <beitragspflichtigesBruttoarbeitsentgelt>
    <Euro>4920</Euro>
  </beitragspflichtigesBruttoarbeitsentgelt>
  <statuskennzeichen></statuskennzeichen>
</beschaeftigung>
<unfallversicherung>
  <betriebsnummerUVTraeger>52742028</betriebsnummerUVTraeger>
  <mitgliedsnummerBeiUVTraeger>12345</mitgliedsnummerBeiUVTraeger>
  <gefahrenstellen>
    <betriebsnummer>123456789</betriebsnummer>
    <tarif>3</tarif>
    <entgelt>4920</entgelt>
    <arbeitsstunden>652</arbeitsstunden>
  </gefahrenstellen>
</unfallversicherung>
<einzugsstelle bezeichnung="AOK"></einzugsstelle>
<absender>
  <firma>Autohaus Hartmann GmbH und Co. KG</firma>
```

Anhang III: Testdatensatz und Prüfergebnisse (AGM)

```

        <strasse>Hauptstraße 83</strasse>
        <plz>12345</plz>
        <ort>Musterstadt</ort>
    </absender>
</mitarbeiter>
<mitarbeiter personalnummer="MA_006" versicherungsnummer="123456789">
    <personaldaten>
        <name>Richter</name>
        <vorname>Benjamin</vorname>
        <vorsatz></vorsatz>
        <zusatz></zusatz>
        <titel></titel>
        <strasseUndHausnummer>Ludwigstraße 94</strasseUndHausnummer>
        <land>Deutschland</land>
        <plz>12345</plz>
        <ort>Musterstadt</ort>
        <entgeltInGleitzone>0</entgeltInGleitzone>
        <namensaenderung>false</namensaenderung>
    </personaldaten>
    <beschaeftigung>
        <beschaeftigungszeit>
            <von>2010-08-01</von>
            <bis>2010-12-31</bis>
        </beschaeftigungszeit>
        <betriebsnummer>987654321</betriebsnummer>
        <personengruppe>101</personengruppe>
        <mehrfachbeschaeftigung>false</mehrfachbeschaeftigung>
        <ostWestMerkmal>west</ostWestMerkmal>
        <beitragsgruppen>
            <kv>1</kv>
            <rv>1</rv>
            <alv>1</alv>
            <pv>1</pv>
        </beitragsgruppen>
        <taetigkeitsschluessel>12342</taetigkeitsschluessel>
        <staatsangehoerigkeit>000</staatsangehoerigkeit>
        <beitragspflichtigesBruttoarbeitsentgelt>
            <Euro>6325</Euro>
        </beitragspflichtigesBruttoarbeitsentgelt>
        <statuskennzeichen></statuskennzeichen>
    </beschaeftigung>
    <unfallversicherung>
        <betriebsnummerUVTraeger>52742028</betriebsnummerUVTraeger>
        <mitgliedsnummerBeiUVTraeger>12345</mitgliedsnummerBeiUVTraeger>
        <gefahrenstellen>
            <betriebsnummer>987654321</betriebsnummer>

```

P23R

P23R: Spezifikationen zur Infrastruktur

```
<tarif>3</tarif>
<entgelt>6325</entgelt>
<arbeitsstunden>815</arbeitsstunden>
</gefahrenstellen>
</unfallversicherung>
<einzugsstelle bezeichnung="AOK"></einzugsstelle>
<absender>
  <firma>Autohaus Hartmann GmbH und Co. KG</firma>
  <strasse>Hauptstraße 83</strasse>
  <plz>12345</plz>
  <ort>Musterstadt</ort>
</absender>
</mitarbeiter>
<mitarbeiter personalnummer="MA_008" versicherungsnummer="123456789">
  <personaldaten>
    <name>Wolf</name>
    <vorname>Franziska</vorname>
    <vorsatz></vorsatz>
    <zusatz></zusatz>
    <titel></titel>
    <strasseUndHausnummer>Parkstraße 17</strasseUndHausnummer>
    <land>Deutschland</land>
    <plz>12345</plz>
    <ort>Musterstadt</ort>
    <entgeltInGleitzone>2</entgeltInGleitzone>
    <namensaenderung>false</namensaenderung>
  </personaldaten>
  <beschaeftigung>
    <beschaeftigungszeit>
      <von>2010-10-01</von>
      <bis>2010-12-31</bis>
    </beschaeftigungszeit>
    <betriebsnummer>123456789</betriebsnummer>
    <personengruppe>102</personengruppe>
    <mehrfachbeschaeftigung>false</mehrfachbeschaeftigung>
    <ostWestMerkmal>west</ostWestMerkmal>
    <beitragsgruppen>
      <kv>1</kv>
      <rv>1</rv>
      <alv>1</alv>
      <pv>1</pv>
    </beitragsgruppen>
    <taetigkeitsschluessel>12341</taetigkeitsschluessel>
    <staatsangehoerigkeit>000</staatsangehoerigkeit>
    <beitragspflichtigesBruttoarbeitsentgelt>
      <Euro>1920</Euro>
```


Anhang III: Testdatensatz und Prüfergebnisse (AGM)

```

        </beitragspflichtigesBruttoarbeitsentgelt>
        <statuskennzeichen></statuskennzeichen>
    </beschaeftigung>
    <unfallversicherung>
        <betriebsnummerUVTraeger>52742028</betriebsnummerUVTraeger>
        <mitgliedsnummerBeiUVTraeger>12345</mitgliedsnummerBeiUVTraeger>
        <gefahrenstellen>
            <betriebsnummer>123456789</betriebsnummer>
            <tarif>3</tarif>
            <entgelt>1920</entgelt>
            <arbeitsstunden>489</arbeitsstunden>
        </gefahrenstellen>
    </unfallversicherung>
    <einzugsstelle bezeichnung="AOK"></einzugsstelle>
    <absender>
        <firma>Autohaus Hartmann GmbH und Co. KG</firma>
        <strasse>Hauptstraße 83</strasse>
        <plz>12345</plz>
        <ort>Musterstadt</ort>
    </absender>
</mitarbeiter>
</notification>

```

Für die im Unternehmen „Autohaus Hartmann GmbH und Co. KG“ bei der „BEK“ versichern Mitarbeiter ergibt sich folgendes Prüfergebnis:

```

<?xml version="1.0" encoding="UTF-8"?>
<notification berichtsjahr="2010">
    <mitarbeiter personalnummer="MA_002" versicherungsnummer="123456789">
        <personaldaten>
            <name>Müller</name>
            <vorname>Christian</vorname>
            <vorsatz></vorsatz>
            <zusatz></zusatz>
            <titel></titel>
            <strasseUndHausnummer>Bahnhofstraße 13</strasseUndHausnummer>
            <land>Deutschland</land>
            <plz>12345</plz>
            <ort>Musterstadt</ort>
            <entgeltInGleitzone>0</entgeltInGleitzone>
            <namensaenderung>false</namensaenderung>
        </personaldaten>
        <beschaeftigung>
            <beschaeftigungszeit>
                <von>2010-08-01</von>
                <bis>2010-12-31</bis>
            </beschaeftigungszeit>
            <betriebsnummer>123456789</betriebsnummer>
        </beschaeftigung>
    </mitarbeiter>
</notification>

```

P23R

P23R: Spezifikationen zur Infrastruktur

```
<personengruppe>101</personengruppe>
<mehrfachbeschaeftigung>false</mehrfachbeschaeftigung>
<ostWestMerkmal>west</ostWestMerkmal>
<beitragsgruppen>
  <kv>1</kv>
  <rv>1</rv>
  <alv>1</alv>
  <pv>1</pv>
</beitragsgruppen>
<taetigkeitsschluesel>12343</taetigkeitsschluesel>
<staatsangehoerigkeit>000</staatsangehoerigkeit>
<beitragspflichtigesBruttoarbeitsentgelt>
  <Euro>10170</Euro>
</beitragspflichtigesBruttoarbeitsentgelt>
<statuskennzeichen></statuskennzeichen>
</beschaeftigung>
<unfallversicherung>
  <betriebsnummerUVTraeger>52742028</betriebsnummerUVTraeger>
  <mitgliedsnummerBeiUVTraeger>12345</mitgliedsnummerBeiUVTraeger>
  <gefahrenstellen>
    <betriebsnummer>123456789</betriebsnummer>
    <tarif>3</tarif>
    <entgelt>10170</entgelt>
    <arbeitsstunden>875</arbeitsstunden>
  </gefahrenstellen>
</unfallversicherung>
<einzugsstelle bezeichnung="BKK"></einzugsstelle>
<absender>
  <firma>Autohaus Hartmann GmbH und Co. KG</firma>
  <strasse>Hauptstraße 83</strasse>
  <plz>12345</plz>
  <ort>Musterstadt</ort>
</absender>
</mitarbeiter>
<mitarbeiter personalnummer="MA_003" versicherungsnummer="123456789">
  <personaldaten>
    <name>Schmidt</name>
    <vorname>Julia</vorname>
    <vorsatz></vorsatz>
    <zusatz></zusatz>
    <titel></titel>
    <strasseUndHausnummer>Dorfstraße 7</strasseUndHausnummer>
    <land>Deutschland</land>
    <plz>12345</plz>
    <ort>Musterstadt</ort>
    <entgeltInGleitzone>0</entgeltInGleitzone>
```

Anhang III: Testdatensatz und Prüfergebnisse (AGM)

```

        <namensaenderung>false</namensaenderung>
    </personaldaten>
    <beschaeftigung>
        <beschaeftigungszeit>
            <von>2010-08-01</von>
            <bis>2010-12-31</bis>
        </beschaeftigungszeit>
        <betriebsnummer>123456789</betriebsnummer>
        <personengruppe>101</personengruppe>
        <mehrfachbeschaeftigung>false</mehrfachbeschaeftigung>
        <ostWestMerkmal>west</ostWestMerkmal>
        <beitragsgruppen>
            <kv>1</kv>
            <rv>1</rv>
            <alv>1</alv>
            <pv>1</pv>
        </beitragsgruppen>
        <taetigkeitsschluessel>12343</taetigkeitsschluessel>
        <staatsangehoerigkeit>000</staatsangehoerigkeit>
        <beitragspflichtigesBruttoarbeitsentgelt>
            <Euro>7950</Euro>
        </beitragspflichtigesBruttoarbeitsentgelt>
        <statuskennzeichen></statuskennzeichen>
    </beschaeftigung>
    <unfallversicherung>
        <betriebsnummerUVTraeger>52742028</betriebsnummerUVTraeger>
        <mitgliedsnummerBeiUVTraeger>12345</mitgliedsnummerBeiUVTraeger>
        <gefahrenstellen>
            <betriebsnummer>123456789</betriebsnummer>
            <tarif>3</tarif>
            <entgelt>7950</entgelt>
            <arbeitsstunden>875</arbeitsstunden>
        </gefahrenstellen>
    </unfallversicherung>
    <einzugsstelle bezeichnung="BKK"></einzugsstelle>
    <absender>
        <firma>Autohaus Hartmann GmbH und Co. KG</firma>
        <strasse>Hauptstraße 83</strasse>
        <plz>12345</plz>
        <ort>Musterstadt</ort>
    </absender>
</mitarbeiter>
</notification>

```

Für die im Unternehmen „Autohaus Hartmann Verwaltungs GmbH“ bei der „AOK“ versichern Mitarbeiter ergibt sich folgendes Prüfergebnis:

```
<?xml version="1.0" encoding="UTF-8"?>
```

P23R

P23R: Spezifikationen zur Infrastruktur

```
<notification berichtsjahr="2010">
  <mitarbeiter personalnummer="MA_009" versicherungsnummer="123456789">
    <personaldaten>
      <name>Fuchs</name>
      <vorname>Kathrin</vorname>
      <vorsatz></vorsatz>
      <zusatz></zusatz>
      <titel></titel>
      <strasseUndHausnummer>Alte Dorfstraße 19</strasseUndHausnummer>
      <land>Deutschland</land>
      <plz>12345</plz>
      <ort>Musterstadt</ort>
      <entgeltInGleitzone>2</entgeltInGleitzone>
      <namensaenderung>false</namensaenderung>
    </personaldaten>
    <beschaeftigung>
      <beschaeftigungszeit>
        <von>2010-08-01</von>
        <bis>2010-12-31</bis>
      </beschaeftigungszeit>
      <betriebsnummer>123459876</betriebsnummer>
      <personengruppe>101</personengruppe>
      <mehrfachbeschaeftigung>true</mehrfachbeschaeftigung>
      <ostWestMerkmal>west</ostWestMerkmal>
      <beitragsgruppen>
        <kv>1</kv>
        <rv>1</rv>
        <alv>1</alv>
        <pv>1</pv>
      </beitragsgruppen>
      <taetigkeitsschluessel>12344</taetigkeitsschluessel>
      <staatsangehoerigkeit>000</staatsangehoerigkeit>
      <beitragspflichtigesBruttoarbeitsentgelt>
        <Euro>1875</Euro>
      </beitragspflichtigesBruttoarbeitsentgelt>
      <statuskennzeichen></statuskennzeichen>
    </beschaeftigung>
    <unfallversicherung>
      <betriebsnummerUVTraeger>52742028</betriebsnummerUVTraeger>
      <mitgliedsnummerBeiUVTraeger>54321</mitgliedsnummerBeiUVTraeger>
      <gefahrenstellen>
        <betriebsnummer>123459876</betriebsnummer>
        <tarif>1</tarif>
        <entgelt>1875</entgelt>
        <arbeitsstunden>375</arbeitsstunden>
      </gefahrenstellen>
    </unfallversicherung>
  </mitarbeiter>
</notification>
```

Anhang III: Testdatensatz und Prüfergebnisse (AGM)

```

</unfallversicherung>
<einzugsstelle bezeichnung="AOK"></einzugsstelle>
<absender>
  <firma>Autohaus Hartmann Verwaltungs GmbH</firma>
  <strasse>Hauptstraße 83</strasse>
  <plz>12345</plz>
  <ort>Musterstadt</ort>
</absender>
</mitarbeiter>
</notification>

```

Für die im Unternehmen „Autohaus Hartmann Verwaltungs GmbH“ bei der „TK“ versicherten Mitarbeiter ergibt sich folgendes Prüfergebnis:

```

<?xml version="1.0" encoding="UTF-8"?>
<notification berichtsjaehr="2010">
  <mitarbeiter personalnummer="MA_001" versicherungsnummer="123456789">
    <personaldaten>
      <name>Hartmann</name>
      <vorname>Horst</vorname>
      <vorsatz></vorsatz>
      <zusatz></zusatz>
      <titel></titel>
      <strasseUndHausnummer>Hauptstraße 83</strasseUndHausnummer>
      <land>Deutschland</land>
      <plz>12345</plz>
      <ort>Musterstadt</ort>
      <entgeltInGleitzone>0</entgeltInGleitzone>
      <namensaenderung>>false</namensaenderung>
    </personaldaten>
    <beschaeftigung>
      <beschaeftigungszeit>
        <von>2010-01-01</von>
        <bis>2010-12-31</bis>
      </beschaeftigungszeit>
      <betriebsnummer>123459876</betriebsnummer>
      <personengruppe>101</personengruppe>
      <mehrfachbeschaeftigung>>false</mehrfachbeschaeftigung>
      <ostWestMerkmal>west</ostWestMerkmal>
      <beitragsgruppen>
        <kv>0</kv>
        <rv>0</rv>
        <alv>0</alv>
        <pv>0</pv>
      </beitragsgruppen>
      <taetigkeitsschluessel>12345</taetigkeitsschluessel>
      <staatsangehoerigkeit>000</staatsangehoerigkeit>
      <beitragspflichtigesBruttoarbeitsentgelt>

```

P23R

P23R: Spezifikationen zur Infrastruktur

```
        <Euro>42000</Euro>
        </beitragspflichtigesBruttoarbeitsentgelt>
        <statuskennzeichen>2</statuskennzeichen>
    </beschaeftigung>
    <unfallversicherung>
        <betriebsnummerUVTraeger>52742028</betriebsnummerUVTraeger>
        <mitgliedsnummerBeiUVTraeger>54321</mitgliedsnummerBeiUVTraeger>
        <gefahrenstellen>
            <betriebsnummer>123459876</betriebsnummer>
            <tarif>2</tarif>
            <entgelt>42000</entgelt>
            <arbeitsstunden>2196</arbeitsstunden>
        </gefahrenstellen>
    </unfallversicherung>
    <einzugsstelle bezeichnung="TK"></einzugsstelle>
    <absender>
        <firma>Autohaus Hartmann Verwaltungs GmbH</firma>
        <strasse>Hauptstraße 83</strasse>
        <plz>12345</plz>
        <ort>Musterstadt</ort>
    </absender>
</mitarbeiter>
</notification>
```

10.4 PRÜFERGEBNISSE ZUM LOHNNACHWEIS AN DIE BERUFGENOSSENSCHAFTEN

Basierend auf dem zuvor dargestellten Testdatensatz für die Teildatenmodelle des Pivotdatenmodells erfolgt eine Benachrichtigungsgenerierung für alle Unternehmen.

Für das Unternehmen „Autohaus Hartmann GmbH und Co. KG“ ergibt sich folgendes Prüfergebnis:

```
<?xml version="1.0" encoding="UTF-8"?>
<notification berichtsjaehr="2010">
    <unternehmen mitgliedsnummerBG="12345" name="Autohaus Hartmann GmbH und Co. KG">
        <anschrift>
            <strasse>Hauptstraße 83</strasse>
            <plz>12345</plz>
            <ort>Musterstadt</ort>
        </anschrift>
    </unternehmen>
    <ansprechpartner vorname="Horst" zuname="" nachname="Hartmann" telefon="+4911123456"
email="horst.hartmann@mailprovider.de" />
    <entgeltnachweis unternehmenszweig="Werkstatt">
        <gefahrenklasse bezeichnung="3">
            <anzahlBeschaeftigte>6</anzahlBeschaeftigte>
            <anzahlArbeitsstunden>8438</anzahlArbeitsstunden>
            <entgelt>68005</entgelt>
        </gefahrenklasse>
    </entgeltnachweis>
```

Anhang III: Testdatensatz und Prüfergebnisse (AGM)

```

<entgeltnachweis unternehmenszweig="Verkauf">
  <gefahrenklasse bezeichnung="3">
    <anzahlBeschaeftigte>5</anzahlBeschaeftigte>
    <anzahlArbeitsstunden>4450</anzahlArbeitsstunden>
    <entgelt>31365</entgelt>
  </gefahrenklasse>
</entgeltnachweis>
</notification>

```

Für das Unternehmen „Autohaus Hartmann Verwaltungs GmbH“ ergibt sich folgendes Prüfergebnis:

```

<?xml version="1.0" encoding="UTF-8"?>
<notification berichtsjahr="2010">
  <unternehmen mitgliedsnummerBG="54321" name="Autohaus Hartmann Verwaltungs GmbH">
    <anschrift>
      <strasse>Hauptstraße 83</strasse>
      <plz>12345</plz>
      <ort>Musterstadt</ort>
    </anschrift>
  </unternehmen>
  <ansprechpartner vorname="Horst" zuname="" nachname="Hartmann" telefon="+4911123456"
email="horst.hartmann@mailprovider.de" />
  <entgeltnachweis unternehmenszweig="Werkstatt">
    <gefahrenklasse bezeichnung="2">
      <anzahlBeschaeftigte>1</anzahlBeschaeftigte>
      <anzahlArbeitsstunden>2196</anzahlArbeitsstunden>
      <entgelt>42000</entgelt>
    </gefahrenklasse>
  </entgeltnachweis>
  <entgeltnachweis unternehmenszweig="Verwaltung">
    <gefahrenklasse bezeichnung="1">
      <anzahlBeschaeftigte>1</anzahlBeschaeftigte>
      <anzahlArbeitsstunden>375</anzahlArbeitsstunden>
      <entgelt>1875</entgelt>
    </gefahrenklasse>
  </entgeltnachweis>
</notification>

```

P23R

P23R: Spezifikationen zur Infrastruktur

11 GLOSSAR

Adapter

Adapter sind interne Komponenten, um unterschiedliche interne oder externe Implementierungen einer Schnittstelle zu nutzen, bspw. um Datenformate oder Übertragungsprotokolle anzupassen.

Antrag

Ein Antragsprozess stellt einen Typ von Prozessketten zwischen Wirtschaft und Verwaltung dar, der dadurch gekennzeichnet ist, dass ein Antragsteller bei der zuständigen Behörde eine Genehmigung für eine bestimmte Tätigkeit oder auch eine Unterstützungsleistung einholt bzw. nachfragt. Die verschiedenen Typen von Prozessketten zwischen Wirtschaft und Verwaltung werden durch die Merkmale Auslöser und Richtung des Informationsflusses unterschieden. Anträge werden durch ein bestimmtes Anliegen des Antragstellers (eine bestimmte Tätigkeit bspw. ein Bau einer Fabrikhalle soll durchgeführt werden oder Unterstützungsleistungen bspw. in Form von Subventionen sollen in Anspruch genommen werden) ausgelöst. Im Lauf des Antragsprozesses oder Antragsverfahrens werden Informationen zwischen Antragsteller und Genehmigungsbehörde in beide Richtungen ausgetauscht, d. h. der Informationsfluss ist bidirektional.

Arbeitgebermeldepflichten

Der Sammelbegriff Arbeitgebermeldepflichten (kurz AGM) umfasst alle Informations- und Meldepflichten, die ein Unternehmen in seiner Funktion als Arbeitgeber erfüllen muss.

Assertion

Eine Assertion ist eine Zusicherung über einen durchgeführten Prozess und / oder Eigenschaften eines Objekts. Im Rahmen der P23R-Sicherheitsarchitektur werden sog. Identity Assertions genutzt, um von einem vertrauenswürdigen Dienst beglaubigte Zusicherungen über die Identität von Nutzern auszutauschen.

Attribut

Ein Attribut ist ein beschreibendes Merkmal einer Entität, das über einen Namen, eine Bedeutung, eine Struktur und einen Definitionsbereich verfügt. Im Rahmen der P23R-Sicherheitsarchitektur werden Attribute z. B. für Nutzer, Regeln und Ressourcen definiert.

Audit

Protokollierung von fachlichen Ereignissen, z. B. zum Zweck des Datenschutzes oder zur Wahrung der Betroffenenrechte.

Authentifizierung

Unter einer Authentifizierung versteht man die Prüfung einer Authentisierung, d. h. die Überprüfung, dass ein Kommunikationspartner tatsächlich derjenige ist, der er vorgibt zu sein.

P23R

P23R: Spezifikationen zur Infrastruktur

Authentizität

Unter dem Begriff Authentizität (engl. authenticity) versteht man die Eigenschaft, die gewährleistet, dass der Kommunikationspartner tatsächlich derjenige ist, der er vorgibt zu sein, bzw. dass die vorliegenden Informationen von der angegebenen Quelle erstellt wurden.

Quelle: [38]

Autorisierung

Eine Autorisierung ist eine Einräumung von Rechten. Rechte können dabei sowohl an Individuen und abgegrenzte Gruppen vergeben werden als auch an offene Gruppen, die lediglich über Eigenschaften ihrer Mitglieder beschrieben sind (z. B. rollenbasierte Berechtigungsvergabe).

Benachrichtigung (Notification)

Eine Benachrichtigung ist ein Sammelbegriff für die technische Darstellung im P23R für einen Antrag, einen Bericht, eine Meldung oder eine Statistik, der bzw. die an einen Benachrichtigungsempfänger gesendet wird. Eine Benachrichtigung, die ein Benachrichtigungssender an den Benachrichtigungsempfänger übermittelt, ergibt sich bspw. aus juristischer Sicht aus den Benachrichtigungspflichten der Unternehmen gegenüber der Verwaltung.

Benachrichtigung, Öffentliche (Legal Notification)

Eine Öffentliche Benachrichtigung ist der Spezialfall einer Benachrichtigung, deren zugeordnete Öffentliche Benachrichtigungsregel ausdrücklich vom Vorschriftengeber freigegeben ist. Öffentliche Benachrichtigungen sind insbesondere:

- Meldungen (periodisch und anlassbezogen),
- Berichte,
- Anträge.

Benachrichtigungsempfänger (Notification Receiver)

Der Benachrichtigungsempfänger (beispielsweise ein Unternehmen, eine Organisation oder eine Verwaltung) benötigt von einem Benachrichtigungssender (beispielsweise ein Unternehmen, eine Organisation oder eine Verwaltung) Informationen, die er in Form von Benachrichtigungen erhält.

Benachrichtigungsempfänger für eine Öffentliche Benachrichtigung bezeichnet eine Behörde oder eine andere Stelle auf Vollzugsebene mit einem gesetzlichen Auftrag, dessen Rahmen eine Öffentliche Benachrichtigung zu empfangen oder anzufordern ist.

Benachrichtigungsregel (Notification Rule)

Eine Benachrichtigungsregel (BR) beschreibt, wie technisch aus den Daten des Benachrichtigungssenders (z. B. ein Unternehmen) genau eine Benachrichtigung für den Benachrichtigungsempfänger (z. B. eine Verwaltung) generiert wird. Eine Benachrichtigungsregel enthält vor allem verschiedene Berechnungen zur Selektion, Aggregation, Transformation, Validierung und Repräsentation sowie weitere vom P23R benötigte Metainformationen. Für die technische Umsetzung werden die Benachrichtigungsregeln aus den rechtlichen Vorgaben durch den Gesetzgeber bzw. die Verwaltung abgeleitet.

Dabei wird zwischen Technischen und Fachlichen Benachrichtigungsregeln unterschieden. Technische Benachrichtigungsregeln werden in einer Technischen Benachrichtigungsregelsprache (T-BRS) definiert, direkt durch den P23R verstanden und sind auf allen P23Rs ausführbar. Um die Entwicklung und Überprüfung der Benachrichtigungsregeln für Fachleute zu vereinfachen, gibt es Fachliche Benachrichtigungsregeln, die in einer Fachlichen Benachrichtigungsregelsprache (F-BRS) definiert werden, durch Fachleute relativ einfach verstanden und geschrieben werden können sowie automatisch in die Technischen Benachrichtigungsregeln übersetzbar sind.

Benachrichtigungsregel, Öffentliche (Legal Notification Rule)

Eine Öffentliche Benachrichtigungsregel ist ein Spezialfall der Benachrichtigungsregel. Sie basiert auf der Modellierung einer gesetzlichen Vorgabe (der Benachrichtigungspflicht). Die Öffentliche Benachrichtigungsregel wird vom Vorschriftengeber geprüft und als korrekt freigegeben. Während software-technisch keine Unterschiede zur (allgemeinen) Benachrichtigungsregel bestehen, unterscheidet sich die rechtliche Beurteilung der Öffentlichen Benachrichtigungsregel von derjenigen der allgemeinen Benachrichtigungsregel.

Die unveränderte Anwendung der Öffentlichen Benachrichtigungsregel im P23R begründet z. B. eine ausreichende Ausübung der Sorgfaltspflicht bei der Erzeugung bzw. Zusammenstellung einer Benachrichtigung mit Hilfe des P23R. Die Richtigkeit der verwendeten Daten bleibt davon unberührt.

Benachrichtigungsregelgruppe (Notification Rule Group)

Eine Benachrichtigungsregelgruppe (BRG) enthält alle diejenigen Benachrichtigungsregeln, die zur Unterstützung einer Meldung, einer Statistik, eines Berichts usw. benötigt werden. Die Benachrichtigungssender können nur Benachrichtigungsregelgruppen in Benachrichtigungsregelpaketen von einer P23R-Leitstelle beziehen. Die Aktivierung von Benachrichtigungsregeln im P23R erfolgt immer im Rahmen einer Benachrichtigungsregelgruppe. Welche Benachrichtigungsregelgruppen für einen Benachrichtigungssender tatsächlich erforderlich bzw. sinnvoll sind, wird bei der Aktualisierung von Benachrichtigungsregelpaketen mittels spezifischer Entscheidungskriterien für eine Benachrichtigungsregelgruppe überprüft.

Es kann für eine Meldepflicht innerhalb einer Benachrichtigungsregelgruppe zum einem verschiedene Varianten einer Benachrichtigung geben, beispielsweise bedingt durch unterschiedliche Unternehmensgrößen. Zum anderen kann es auch mehrere verschiedene, aber zusammengehörende Benachrichtigungen geben, die zur Umsetzung der Meldepflicht benötigt werden, beispielsweise neben der eigentlichen Meldung auch die Anmeldung bei einer Behörde bzgl. der Meldepflicht.

Benachrichtigungsregelpaket (Notification Rule Package)

Ein Benachrichtigungsregelpaket ist eine Menge von Technischen Benachrichtigungsregelgruppen sowie den dazugehörigen Teildatenmodellen, wie sie technisch durch eine Leitstelle bereitgestellt werden. Ein Benachrichtigungsregelpaket könnte beispielsweise alle benötigten Benachrichtigungsregelgruppen für eine Fachdomäne enthalten. Darüber hinaus gibt es ein Basis-Benachrichtigungsregelpaket, das die Benachrichtigungsregelgruppen enthält, die grundsätzlich jeder P23R insbesondere für seine Initialisierung benötigt.

P23R

P23R: Spezifikationen zur Infrastruktur

Die Benachrichtigungsregelgruppen in einem Benachrichtigungsregelpaket werden nach Gesichtspunkten der technischen Verwandtschaft und des Anwendernutzens zusammengestellt. Sie sind in der Regel nicht deckungsgleich mit der Gruppierung in einem Benachrichtigungsregelwerk.

Benachrichtigungsregelsprache (Notification Rule Language)

Eine Benachrichtigungsregelsprache (BRS) beschreibt die Rechtschreibung und Grammatik, wie Benachrichtigungsregeln, -gruppen und -pakete sowie Datenmodellpakete zu spezifizieren sind. Die Technische Benachrichtigungsregelsprache (T-BRS) wird für die Verteilung der Benachrichtigungsregelpakete und der Datenmodellpakete genutzt, um sicherzustellen, dass jeder P23R unabhängig vom Hersteller die Benachrichtigungsregeln identisch interpretiert.

Die Fachliche Benachrichtigungsregelsprache (F-BRS) soll dagegen den Fachexperten, die die Fachlichen Benachrichtigungsregelwerke entwickeln und spezifizieren müssen, eine möglichst einfach zu erstellende, leicht verständliche und fachlich angepasste Beschreibungsform zur Verfügung stellen, die dann letztlich aber automatisch in die T-BRS übersetzt wird.

Siehe auch *Benachrichtigungsregel*.

Benachrichtigungsregelwerk (Notification Rule Set)

Ein Benachrichtigungsregelwerk (BRW) ist eine logisch oder fachlich abgeschlossene Menge von Benachrichtigungsregeln. Dies könnten beispielsweise alle Benachrichtigungsregeln zu einem Gesetz, einem Rechtsgebiet, einer Fachdomäne oder einer Organisationseinheit sein. Die Kriterien der Zusammenfassung sind rein fachlicher Art. Es gibt keine zwingende Deckungsgleichheit mit den Begriffen „Benachrichtigungsregelpaket“ oder „Benachrichtigungsregelgruppe“.

Benachrichtigungssender (Notification Sender)

Der Benachrichtigungssender (beispielsweise ein Unternehmen, eine Organisation oder eine Verwaltung) sendet Informationen, in Form von Benachrichtigungen, an einen Benachrichtigungsempfänger (beispielsweise ein Unternehmen, eine Organisation oder eine Verwaltung).

Benachrichtigungstyp (Notification Type)

Die Benachrichtigungsregeln generieren eine Benachrichtigung in einem internen, empfängerunabhängigen XML-Format. Jedem dieser XML-Formate kann ein Benachrichtigungstyp zugeordnet werden. Ein Benachrichtigungstyp wird durch einen eindeutigen Namen in Form des standardmäßigen XML-Namensraums identifiziert, der in dem XML-Dokument verwendet wird.

Berechtigung

Siehe Autorisierung.

Bericht

Ein Bericht stellt einen Typ von Prozessketten zwischen Wirtschaft und Verwaltung dar, der dadurch gekennzeichnet ist, dass ein Unternehmen vorgegebene Informationen über eine bestimmte Tätigkeit bspw. die mit der Verbrennung von Abfällen verbundenen Emissionen abgeben muss. Die verschiedenen Typen von Prozessketten zwischen Wirtschaft und Verwaltung werden durch die Merkmale Auslöser und Richtung des Informationsflusses unterschieden.

Berichte sind dadurch charakterisiert, dass sie neben festgelegten Inhalten einen vorgegebenen Fälligkeitstermin und eine vorgegebene Frequenz haben, d. h. sie werden durch das Eintreffen des Fälligkeitstermins ausgelöst. Informationen fließen im Wesentlichen in eine Richtung, vom Unternehmen zur zuständigen Überwachungsbehörde.

Betreibermodell

Ein Betreibermodell ist ein Geschäftskonzept für die Bereitstellung von Gütern und Dienstleistungen, bei dem diese nicht mehr an Kunden verkauft, sondern gegen ein leistungsabhängiges Entgelt zur Nutzung angeboten werden. Betreibermodelle können somit für die Bereitstellung von physischen Produkten und / oder immateriellen Dienstleistungen gestaltet und etabliert werden. Betreibermodelle können gemäß den folgenden Kriterien klassifiziert, beschrieben und gestaltet werden: Leistungsfokus, Organisationsform, Koordinationsform, Kundenfokus, Gegenstand, Leistungsverrechnung, Preismodell, Absatzmarkt, Kontrahierungsform, Center-Konzept und Mitarbeiter.

Betreiber- und Geschäftsmodell

Ein Betreibermodell im Kontext des P23R ist ein Geschäftskonzept für die Bereitstellung von Gütern und Dienstleistungen gegen ein leistungsabhängiges Entgelt.

Betreibermodelle können somit für die Bereitstellung von physischen Produkten und / oder immateriellen Dienstleistungen gestaltet und etabliert werden. Betreibermodelle können gemäß der folgenden Kriterien klassifiziert, beschrieben und gestaltet werden: Zielgruppe, Zielbranche, Anbieter / Provider, Geschäftsfelder, P23R-Lösung, Musterimplementierung, Make-or-Buy-Entscheidung, Betrieb, Preismodell, Abrechnungsmöglichkeiten.

Datenmodell

Als Datenmodell wird das in den Benachrichtigungsregeln verwendete logische, von der konkreten Implementierung unabhängige Pivot-Datenmodell bezeichnet, um auf die Daten des Benachrichtigungssenders (z. B. eines Unternehmens), die im Datenpool zugänglich sind, zuzugreifen. Aus technischen Gründen wird das Datenmodell noch in Teildatenmodelle untergliedert, gepflegt und verteilt. Ein Teildatenmodell entspricht technisch einem XML-Schema mit einem spezifischen XML-Namensraum.

Das Mapping eines logischen Teildatenmodells in ein konkretes Datenmodell des Quellsystems erfolgt beim zugehörigen SourceConnector.

Datenmodellpaket (Model Package)

Ein Datenmodellpaket (MP) ist eine Menge von Teildatenmodellen, wie sie technisch durch eine P23R-Leitstelle bereitgestellt werden. Ein Datenmodellpaket könnte beispielsweise alle benötigten Teildatenmodelle für eine Fachdomäne enthalten. Darüber hinaus gibt es ein Basis-Datenmodellpaket, das die Teildatenmodelle enthält, die grundsätzlich jeder P23R insbesondere für seine Initialisierung benötigt.

Die Teildatenmodelle in einem Datenmodellpaket werden nach Gesichtspunkten der technischen Verwandtschaft und des Anwendernutzens zusammengestellt. Sie sind in der Regel nicht deckungsgleich mit der Gruppierung in einem Fachlichen Benachrichtigungsregelwerk.

P23R

P23R: Spezifikationen zur Infrastruktur

Datenpool

Der Datenpool ist die logische Komponente im P23R, die das Abfragen und Zwischenspeichern der Quelldaten (Unternehmensdaten) sowie den Zugriff auf diese regelt. Dazu kann ein Cache genutzt werden, der die Anfragen mit ihren Antworten zwischenspeichert. Alternativ können die Quelldaten im P23R teilweise gespiegelt werden.

Datenschutz

Datenschutz soll den Einzelnen davor schützen, dass er durch den Umgang mit seinen personenbezogenen Daten in seinem Persönlichkeitsrecht beeinträchtigt wird. Mit Datenschutz wird daher der Schutz personenbezogener Daten vor etwaigem Missbrauch bezeichnet (nicht zu verwechseln mit Datensicherheit).

Datensicherheit

Datensicherheit beziehungsweise IT-Sicherheit bedeutet „die Durchführung aller organisatorischen und technischen Maßnahmen, um das in der Organisation von Unternehmen und Behörden benötigte Niveau an Vertraulichkeit, Verfügbarkeit, Integrität“ und Prüfbarkeit aller verarbeiteten Daten, einschließlich der Programme, sicherzustellen.

Für den Bereich des Datenschutzes sind die korrespondierenden Pflichten in der Anlage zu § 9 Satz 1 BDSG konkretisiert.

Quelle: [39]

Domäne (Fachdomäne)

Die Abgrenzung eines Themenbereiches für die Regelerstellung wird im Kontext des P23R-Prinzips als „Fachdomäne“ (kurz: „Domäne“) bezeichnet. Die Abgrenzungskriterien sind unterschiedlicher Art; sie können auf Rechtsgebieten, Verwandtschaft durch Nutzung stark überschneidender Datenmengen, gleichen oder verwandten Überwachungsgegenständen, verwandten Geschäftsprozessen, dem spezifischen Bedarf einer bestimmten Branche oder anderen rationalen Kriterien basieren.

Elektronische Signatur

Die elektronische Signatur beschreibt ein asymmetrisches Verschlüsselungsverfahren zur Gewährleistung der Authentizität und Integrität elektronischer Daten und zur Überprüfung der Identität des Benutzers. Sie ist praktisch mit einer handschriftlichen Unterschrift oder der Möglichkeit, sich eindeutig gegen Vorlage einer Unterschrift auszuweisen, zu vergleichen. Die Rechtswirksamkeit der elektronischen Signaturen wird in Deutschland durch das Signaturgesetz (SigG) geregelt. Hiernach gibt es drei Arten: Die (allgemeine) elektronische Signatur, die fortgeschrittene elektronische Signatur und die qualifizierte elektronische Signatur. Hierbei hat jede Signatur eine bestimmte Qualitätsstufe. Je höherwertiger die Signatur, desto größer ist ihre Bedeutung für den Rechtsverkehr und desto größer ist ihre Funktionalität.

Empfänger

Empfänger bezeichnet eine Behörde oder eine andere Stelle auf Vollzugsebene mit einem gesetzlichen Auftrag, in dessen Rahmen eine Benachrichtigung zu empfangen oder anzufordern ist.

eSTATISTIK.core

eSTATISTIK.core ist ein innovatives Online-Meldeverfahren, das von den Statistischen Ämtern als eGovernment-Projekt zur Entlastung der Auskunftspflichtigen bei der Datenlieferung entwickelt wurde. Es ermöglicht den meldepflichtigen Unternehmen und öffentlichen Stellen, die erfragten Statistikdaten direkt aus ihrem jeweiligen Softwaresystem elektronisch zu gewinnen und via Internet an den zentralen Dateneingang der amtlichen Statistik zu übermitteln. Hierzu kann ein von zahlreichen Softwareherstellern in die betriebswirtschaftliche Software integriertes Statistikmodul genutzt oder in vielen Fällen die unabhängige PC-Anwendung CORE.reporter angewendet werden.

Quelle: [40]

Fachliche Beratungsstellen

Fachliche Beratungsstellen können die Ersteller von Benachrichtigungsregeln methodisch unterstützen sowie bei Bedarf die Entwicklung Fachlicher Benachrichtigungsregelsprachen (F-BRS) betreuen. Die Einrichtung und der Betrieb Fachlicher Beratungsstellen liegen in der Verantwortung interessierter Vorschriftengeber bzw. Vorschriftengebergruppen.

Fachübergreifende Koordinierungsaufgaben

Durch das Konzept der Autonomie für die einzelnen P23R-Installationen besteht in einigen Bereichen ein übergreifender Koordinierungsbedarf. Dies kann bspw. folgende Koordinierungsaufgaben betreffen:

- Betrieb des P23R-Depots bei einer Öffentlichen Leitstelle für die Bereitstellung von Benachrichtigungsregel- und Datenmodellpaketen
- Prüfung von Benachrichtigungsregeln und Pflege des Pivot-Datenmodells
- Weiterentwicklung der Technischen Benachrichtigungsregelsprache (T-BRS) bei Bedarf
- Einbindung externer Verzeichnisdienste, wie z. B. „Leistungsverzeichnisse“ und „Zuständigkeitsverzeichnisse“, und weiterer Quellen zur Bereitstellung von Zuständigkeitsinformationen über das P23R-Zuständigkeitsverzeichnis
- Angebot eines Online-Entwickler-Portals, um die Entwicklung der Fachlichen Benachrichtigungsregeln zu unterstützen
- Angebot eines Online-Service-Portals, um die Kommunikation mit den P23R-Anbietern und P23R-Betreibern zu unterstützen
- Organisation von Präsenzveranstaltungen zum fachlichen Austausch in und zwischen Interessengruppen, wie z. B. Stakeholdergremien, Communities, fachliche Arbeitsgruppen
- Organisation von Kontakten zu anderen Gremien, die für das P23R-Konzept von Interesse sind.

Fachübergreifende Koordinierungsstelle

Die Fachübergreifende Koordinierungsstelle ist in ihrer Rolle als Dienstleister für den P23R für die Umsetzung der Ziele und Anforderungen des P23R-Prinzips verantwortlich.

P23R

P23R: Spezifikationen zur Infrastruktur

Sie übernimmt die zentrale Koordination aller Aufgaben, die über die Erstellung einzelner Benachrichtigungsregeln und -regelgruppen hinausgehen.

Siehe auch *Fachübergreifende Koordinierungsaufgaben*.

Identity Provider

Ein Identity Provider ist ein Infrastrukturdienst, der Zusicherungen über die Authentizität und Identität von Entitäten (i. Allg. Systemnutzern) in einem standardisierten Format bereitstellt.

Informations- und Meldepflichten

Informations- und Meldepflichten sind der Sammelterm für die unterschiedlichen Typen von Prozessketten zwischen Wirtschaft und Verwaltung. Sie umfassen Antragsprozesse, Archivpflichten, Berichte, Meldungen.

Integrität

Bei der elektronischen Kommunikation ist damit die Unversehrtheit von Informationen und Daten gemeint, d. h. dass die Daten bei der Übertragung nicht verändert wurden.

Quelle: [38]

Intermediär

Ein Intermediär ist ein vom Unternehmen beauftragter Dienstleister, der Prozesse für das Unternehmen ganz oder teilweise durchführt. Er wird im P23R repräsentiert durch die Rolle Intermediär. Der Intermediär ist keine globale Rolle (z. B. Steuerberater, Buchhaltungsservice).

IT-Grundschutz

IT-Grundschutz bezeichnet eine Methodik zum Aufbau eines Sicherheitsmanagementsystems sowie zur Absicherung von Informationsverbänden über Standard-Sicherheitsmaßnahmen. Außerdem wird mit IT-Grundschutz der Zustand bezeichnet, in dem die vom BSI empfohlenen Standard-Sicherheitsmaßnahmen umgesetzt sind, die als Gesamtheit von infrastrukturellen, organisatorischen, personellen und technischen Sicherheitsmaßnahmen, Institutionen mit normalem Schutzbedarf hinreichend absichern.

Quelle: [41]

IT-Sicherheit

IT-Sicherheit bezeichnet einen Zustand, in dem die Risiken, die beim Einsatz von Informationstechnik aufgrund von Bedrohungen und Schwachstellen vorhanden sind, durch angemessene Maßnahmen auf ein tragbares Maß reduziert sind. IT-Sicherheit ist also der Zustand, in dem Vertraulichkeit, Integrität und Verfügbarkeit von Informationen und Informationstechnik durch angemessene Maßnahmen geschützt sind.

Quelle: [41]

Kommunikationskanal

Als Kommunikationskanal bezeichnet man die physische Kommunikationsverbindung zwischen dem P23R und dem Benachrichtigungsempfänger, über die Benachrichtigungen versendet werden. Die physische Kommunikation erfolgt im P23R durch die verschiedenen Kommunikationsadapter, die die Protokolle, wie Webservices, E-Mail, Fax usw., implementieren.

Kommunikationsmaßnahmen

Kommunikationsmaßnahmen sind als Aktivitäten definiert, die von einem kommunikations-treibenden Unternehmen bewusst zur Erreichung kommunikativer Zielsetzungen eingesetzt werden.

Quelle: [42]

Kommunikationsmatrix

Die Kommunikationsmatrix ist eine Darstellungsform der Kommunikationsstrategie bei der Kommunikationsinstrumente und -maßnahmen zeitlich integriert und auf konkrete Zielgruppen abgestimmt werden.

Kommunikationsstrategie

Unter einer Kommunikationsstrategie werden Maßnahmen grundsätzlicher Art zur Erreichung von Kommunikationszielen verstanden. Kommunikationsstrategien können sich in Verwendung einzelner, als auch in Kombination mehrerer Kommunikationsinstrumente niederschlagen.

Quelle: [43]

Komponente

Als Komponenten werden im IT-Grundschutz technische Zielobjekte (siehe dort) oder Teile von Zielobjekten bezeichnet.

Quelle: [41]

Konnektor

Ein Konnektor ist eine Komponente ohne eigene Geschäftslogik, die in die Kommunikation zwischen zwei Anwendungen (Systemkomponenten) eingefügt wird, um Datenformate oder Übertragungsprotokolle zwischen unterschiedlichen Schnittstellen anzupassen.

Leitstelle, Öffentliche

Eine Öffentliche Leitstelle ist eine P23R-Leitstelle, die für die Bereitstellung von Öffentlichen Benachrichtigungsregel- und Datenmodellpaketen sowie des Öffentlichen P23R-Zuständigkeitsverzeichnisses zuständig ist. Im Idealfall gibt es genau eine Öffentliche Leitstelle.

Siehe auch *P23R-Leitstelle*.

Manifest

Mit Manifest wird eine Datei bezeichnet, die Metainformationen enthält. Im P23R werden Manifeste in der Technischen Benachrichtigungsregelsprache (T-BRS) verwendet, bspw. um die Metainformationen von Benachrichtigungspaketen, Benachrichtigungsregelgruppen, Benachrichtigungsregeln u. v. m. zu beschreiben.

Meldung

Eine Meldung ist eine Informationsübermittlung von einem Unternehmen an einen Meldungsempfänger im Rahmen einer Prozesskette. Das sind im juristischen Sinne Meldungen, die sich aus den Meldepflichten des Unternehmens ergeben.

P23R

P23R: Spezifikationen zur Infrastruktur

Methodenleitfaden (MLF)

Der Methodenleitfaden bildet ein Kompendium aus unterschiedlichen Modulen. Die einzelnen Module des Methodenleitfadens richten sich an Entscheider und Experten, die an der Schnittstelle zwischen Wirtschaft und Verwaltung wirken. Sie unterstützen diese in ihren fachlichen, IT-architektonischen, sicherheitstechnischen, wirtschaftlichen und juristischen Analyse- und Gestaltungsaufgaben. In digitaler Form gibt es einen Methodenleitfaden-Online.

Methodenleitfaden-Online (MLF-Online)

Der Methodenleitfaden Online ist die webbasierte Variante des Methodenleitfadens, der im Projekt Prozess-Daten-Beschleuniger entsteht. Der Methodenleitfaden kann von den Nutzern aus der Öffentlichkeit und Fachöffentlichkeit in rollengeführter Anwendung eingesetzt werden.

Modellierung

Bei der Vorgehensweise nach IT-Grundschutz wird bei der Modellierung der betrachtete Informationsverbund eines Unternehmens oder einer Behörde mit Hilfe der Bausteine aus den IT-Grundschutz-Katalogen nachgebildet. Hierzu enthält Kapitel 2.2 der IT-Grundschutz-Kataloge für jeden Baustein einen Hinweis, auf welche Zielobjekte er anzuwenden ist und welche Voraussetzungen dabei gegebenenfalls zu beachten sind.

Quelle: [41]

Nachricht (Message)

Eine Nachricht löst die Generierung einer Benachrichtigung aus. Im Standardfall werden interne Nachrichten innerhalb des P23R zeitgesteuert erzeugt, z. B. um gesetzlichen Meldepflichten fristgerecht nachzukommen. Daneben kann auch ein Benachrichtigungsempfänger eine externe (Öffentliche) Nachricht an den P23R senden und damit gezielt eine Benachrichtigung anfordern.

Eine Nachricht bezieht sich immer auf eine Benachrichtigung. Eine Nachricht kann aus zwei Gründen erzeugt werden:

- Die Nachricht fordert ein Unternehmen auf, eine Benachrichtigung zu erzeugen.
- Die Nachricht ist eine Reaktion auf eine vorhergehende Benachrichtigung. Die Nachricht kann eine Eingangsbestätigung, eine Rückfrage, eine Aufforderung zur Korrektur, eine Genehmigung oder Ablehnung eines Antrags oder Ähnliches enthalten.

Art und Form einer Nachricht werden im Rahmen der Benachrichtigungsregel definiert.

Notfall

Ein Notfall ist ein Schadensereignis, bei dem Prozesse oder Ressourcen einer Institution nicht wie vorgesehen funktionieren. Die Verfügbarkeit der entsprechenden Prozesse oder Ressourcen kann innerhalb einer geforderten Zeit nicht wieder hergestellt werden. Der Geschäftsbetrieb ist stark beeinträchtigt. Eventuell vorhandene Service Level Agreements können nicht eingehalten werden. Es entstehen hohe bis sehr hohe Schäden, die sich signifikant und in nicht akzeptablem Rahmen auf das Gesamtjahresergebnis eines Unternehmens oder die Aufgaben-

erfüllung einer Behörde auswirken. Notfälle können nicht mehr im allgemeinen Tagesgeschäft abgewickelt werden, sondern erfordern eine gesonderte Notfallbewältigungsorganisation.

Quelle: [44]

Notfallkonzept

Das Notfallkonzept umfasst das Notfallvorsorgekonzept und das Notfallhandbuch.

Quelle: [44]

Öffentliche Benachrichtigung (Legal Notification)

Siehe *Benachrichtigung, Öffentliche (Legal Notification)*.

Öffentliche Benachrichtigungsregel (Legal Notification Rule)

Siehe *Benachrichtigungsregel, Öffentliche (Legal Notification Rule)*.

Öffentliche Leitstelle

Siehe *Leitstelle, Öffentliche*.

P23R

Unter der Bezeichnung „P23R“ ist derjenige Teil einer P23R-Lösung zu verstehen, der die Generierung und den Versand von Benachrichtigungen über die von der Leitstelle bereitgestellten Regelwerke realisiert. Der Name „P23R“ leitet sich von „Prozess-Daten-Beschleuniger“ ab. P steht dabei für den ersten Buchstaben, R für den letzten Buchstaben. Dazwischen befinden sich 23 Buchstaben.

P23R-Anwender

Ein P23R-Anwender ist jede natürliche oder juristische Person, die eine P23R-Lösung zur Abwicklung von Informations- und Meldepflichten einsetzt.

P23R-Client

Der P23R selbst stellt ausschließlich Dienstschnittstellen (SOA) zur Verfügung, über die auf seine Funktionalität zugegriffen werden kann. Eine ggf. ergänzende Komponente, die als P23R-Client bezeichnet wird, stellt eine grafische Oberfläche zur Bedienung des P23R bereit. Diese Funktionalität des P23R-Client kann auch direkt in der Unternehmenssoftware integriert sein.

P23R-Depot

Das P23R-Depot stellt Benachrichtigungsregelpakete, die Liste aller Benachrichtigungsregelpakete und die Trusted Service Lists den P23R-Instanzen zur Verfügung. Die Öffentliche Leitstelle hält (mindestens) alle Öffentlichen Benachrichtigungsregeln auf einem Server zum anonymen Download bereit.

P23R-Infrastruktur

Die P23R-Infrastruktur umfasst neben der zentralen Systemkomponente P23R auch den optionalen P23R-Client, die P23R-Leitstelle und den optionalen P23R-TrustedProxy sowie die Definition des P23R-Protokolls.

P23R

P23R: Spezifikationen zur Infrastruktur

P23R-inside

Unter einer P23R-inside-Lösung versteht man eine P23R-Lösung, bei der relevante P23R-Architekturelementen in eine bestehende IT-Lösung integriert werden. Solche Lösungen setzen ein gutes Verständnis der Rahmenarchitektur voraus und können diese in unterschiedlichen Ausprägungen implementieren.

P23R-Instanz

Die P23R-Instanz ist eine in Betrieb befindliche Instanziierung des P23R.

P23R-Leitstelle

Eine P23R-Leitstelle ist eine Organisationseinheit, die den Betrieb der P23Rs technisch unterstützt. Sie generiert die Benachrichtigungsregelpakete sowie die Datenmodellpakete und weitere technische Artefakte und stellt diese für den P23R bereit. Darüber hinaus betreibt sie noch weitere Dienste für den P23R, z. B. das P23R-Zuständigkeitsverzeichnis.

Neben einer oder mehreren Öffentlichen Leitstellen kann es in jedem Unternehmen eigene Unternehmensleitstellen geben, die eigene Benachrichtigungsregelpakete und Datenmodellpakete sowie weitere Dienste für das Unternehmen bereitstellen.

Siehe auch *Leitstelle, Öffentliche*.

P23R-Lösung

Eine P23R-Lösung ist eine mögliche Umsetzung der P23R-Rahmenarchitektur durch einen Softwareanbieter oder IT-Dienstleister im Rahmen eines Betreiber- und Geschäftsmodells. Wie diese jeweils ausgestaltet ist, darf im Rahmen der Architektur frei entschieden werden und basiert in der Regel auf einem der beiden Lösungskonzepte P23R-inside und P23R-standalone.

P23R-Lösungsanbieter

Ein P23R-Lösungsanbieter ist ein Softwareanbieter oder IT-Dienstleister, der eine spezifische P23R-Lösung auf Basis der P23R-Rahmenarchitektur einem definierten Kundenkreis im Rahmen eines Betreiber- und Geschäftsmodells anbietet. Der P23R-Provider stellt hierbei eine Sonderform des P23R-Lösungsanbieters dar.

P23R-Mandant

Der P23R-Mandant, in der Regel eine Organisation oder ein Unternehmen (juristische Person) oder eine natürliche Person, ist eine Rolle im organisatorischen und juristischen Verhältnis zwischen dem Nutzer eines P23R und einem P23R-Provider. Der P23R-Mandant nutzt eine vom Provider bereitgestellte P23R-Instanz, genauer gesagt eine P23R-Mandanteninstanz eines P23R.

P23R-Mandanteninstanz

Die P23R-Mandanteninstanz ist ein Nutzer einer P23R-Instanz, der eigene getrennte Ressourcen besitzt, bspw. eigene Datenhaltung und eigene aktivierte Benachrichtigungsregeln. Die P23R-Mandanteninstanz fasst alle Aspekte einer P23R-Instanz zusammen, die genau einen Mandanten betreffen. Man kann sie so betrachten, als ob die P23R-Instanz einzeln genau nur für diesen Mandanten betrieben würde. Das betrifft vor allem die getrennte Datenhaltung und die Unabhängigkeit der Verarbeitungsprozesse.

P23R-Musterimplementierung

Die P23R-Musterimplementierung ist die im Rahmen des Projekts „Pilotierung und Realisierung eines Prozess-Daten-Beschleunigers (P23R) für den Datenaustausch zwischen Wirtschaft und Verwaltung“ entstandene Open-Source-Musterimplementierung einer P23R-standalone-Lösung. Diese umfasst eine Umsetzung des P23R (inkl. pilotrelevanter Kommunikationskonnektoren), des P23R-Client, sowie einer Laborleitstelle mit Zuständigkeitsverzeichnis.

P23R-Prinzip

Das Prinzip Prozess-Daten-Beschleuniger (P23R-Prinzip) beschreibt Methoden und Architekturkonzepte zur effizienten Gestaltung von Prozessen zwischen Wirtschaft und Verwaltung. Es zielt darauf ab, Prozessketten zwischen Wirtschaft und Verwaltung sinnvoll zu bündeln und zentral bereitgestellte Regelwerke für die automatisierte Abwicklung von Informations- und Meldepflichten zu nutzen.

P23R-Provider

Ein P23R-Provider stellt einem P23R-Mandanten die technische Infrastruktur zur Verfügung, mit der der Mandant in der Lage ist, die Funktionalität des P23R zu nutzen. Der P23R-Provider hat keinen Einblick in die im P23R enthaltenen Daten und Profile.

Es wird nicht zwischen internen und externen P23R-Providern unterschieden, da beide als Dienstleister gemäß IT Infrastructure Library (ITIL) zu betrachten sind.

P23R-Rahmenarchitektur

P23R-Rahmenarchitektur ist ein Dokument, das einen konzeptionellen Überblick über die vollständige Infrastruktur des Prozess-Daten-Beschleunigers (P23R) und deren Systemkomponenten, die Schnittstellen und die verwendeten Datenstrukturen in den Teilkomponenten des P23R sowie ihr Zusammenspiel liefert. Sie soll den Entwicklern eine klare Vorstellung davon geben, welche Funktionalität jede Teilkomponente des P23R bzw. Systemkomponente der P23R-Infrastruktur haben sollte und wie ein mögliches Systemdesign aussehen könnte.

P23R-Sicherheitsarchitektur

P23R-Sicherheitsarchitektur ist ein Dokument, das einen konzeptionellen Überblick über die vollständige Sicherheitsinfrastruktur des Prozess-Daten-Beschleunigers (P23R) und den Systemkomponenten der Sicherheitsarchitektur, die Schnittstellen und die verwendeten Datenstrukturen in den Sicherheits-Teilkomponenten des P23R sowie ihr Zusammenspiel im Kontext der P23R-Rahmenarchitektur liefert. Sie soll den Entwicklern eine klare Vorstellung davon geben, welche Funktionalität jede Teilkomponente der P23R-Sicherheitsarchitektur im Kontext der P23R-Rahmenarchitektur bzw. der zu implementierenden bzw. zu nutzenden Systemkomponenten der P23R-Infrastruktur haben sollte und wie ein mögliches Systemdesign aussehen könnte.

P23R-standalone

Unter einer P23R-standalone-Lösung versteht man eine P23R-Lösung, die einen eigenständigen P23R – sprich nicht in eine vorhandene IT-Lösung integrierte P23R-inside-Lösung – realisiert.

P23R

P23R: Spezifikationen zur Infrastruktur

P23R-TrustedProxy

Standardmäßig kommunizieren der P23R und das Fachverfahren direkt über ein eigenes Protokoll. Der P23R-TrustedProxy als optionale Komponente der P23R-Infrastruktur bietet eine besonders sichere und vertrauenswürdige Kommunikation. Er wird direkt in der internen Infrastruktur bereitgestellt und erlaubt eine vereinfachte Kommunikation im Intranet mit dem P23R.

Der P23R-TrustedProxy ist der Stellvertreter des P23R in der eigenen Infrastruktur und realisiert alle Sicherheitsfunktionen zwischen P23R und Proxy. Ein P23R-TrustedProxy kann durch ein spezielles Deployment eines P23R realisiert werden.

P23R-Unterstützungsstellen

P23R-Unterstützungsstellen ist ein Begriff für die Gesamtheit föderativ verteilter Organisationseinheiten. Diese führen Koordinationsaufgaben durch, die einerseits den Betrieb der P23R-Infrastruktur operativ und andererseits die Erstellung der benötigten Benachrichtigungsregeln unterstützen sowie ggf. deren Konzepte weiterentwickeln. Möglich sind eine Öffentliche Leitstelle, eine fachübergreifende Koordinierungsstelle sowie eine dem Bedarf anpassbare Anzahl von Fachlichen Beratungsstellen.

P23R-Zuständigkeitsverzeichnis

Ein P23R-Zuständigkeitsverzeichnis ist erforderlich, um eine Benachrichtigungsregel im P23R eines Unternehmens zu konkretisieren. Mit seiner Hilfe wird anhand der aktuellen Unternehmenscharakteristik und entsprechend den Vorgaben in der Benachrichtigungsregel ein zuständiger Benachrichtigungsempfänger zugeordnet. Weitere Informationen sind Angaben zur Kommunikation mit dem Benachrichtigungsempfänger und zur erforderlichen Darstellung der Benachrichtigung.

Die Öffentliche Leitstelle ist für die technische Verfügbarkeit der Informationen verantwortlich. Der Betreiber des P23R-Zuständigkeitsverzeichnisses ist nicht notwendigerweise auch der Betreiber der erforderlichen Original-Verzeichnisse. Die erforderlichen Zuständigkeitsinformationen können aus einem zentralen oder aus einem verteilten, föderativen System stammen oder speziell für die Benachrichtigungsregel erstellt werden. Die für die Funktion des P23R erforderlichen Informationen müssen jedoch über das P23R-Zuständigkeitsverzeichnis in einem einheitlichen Format bereitgestellt werden.

Die Anbindung von externen Verzeichnissen an das P23R-Zuständigkeitsverzeichnis ist eine der Unterstützungsaufgaben.

Pivot-Datenmodell

Das Pivot-Datenmodell vermittelt die Semantik zwischen den verschiedenen Semantiken der Quelldatenmodelle (Unternehmensdatenmodelle) zu den verschiedenen Semantiken der Benachrichtigungstypen im P23R (Mapping). Es dient gleichzeitig der Definition des internen Datenmodells. Es ist nicht notwendigerweise ein kanonisches oder normalisiertes Datenmodell.

Siehe auch *Datenmodell*.

Policy

Eine Policy ist ein Regelwerk, aus dem sich Entscheidungen herleiten lassen. Im Rahmen der P23R-Sicherheitsarchitektur werden Policies zur Kodierung von Berechtigungen (Berechtigungs-policies) und zur Steuerung des Dienstzugangs (Sicherheitspolicies) verwendet.

Policy Administration Point (PAP)

Über einen Policy Administration Point wird der Lebenszyklus einer Policy gesteuert. Insbesondere erfolgt über den Policy Administration Point als Teil des Berechtigungsmanagements auch die Bereitstellung von Policies zur Nutzung im Rahmen einer Berechtigungsprüfung.

Policy Decision Point (PDP)

Ein Policy Decision Point kapselt die Funktionalität zur Prüfung einer Zugriffsanfrage gegen Berechtigungs-policies.

Policy Enforcement Point (PEP)

Ein Policy Enforcement Point setzt das Designmuster eines Reference Monitors um, der Kontrollflüsse vor dem Zugriff auf geschützte Ressourcen unterbricht, um von einem Policy Decision Point eine Berechtigungsentscheidung abzufragen.

Pool

Ein Pool ist die allgemeine Bezeichnung für Daten- und Informationssammlungen. Ob die Daten dabei in einer Datenbank, in XML-Dateien oder anders abgelegt werden, spielt keine Rolle.

Protokollierung (Logging)

Protokollierung von technischen Ereignissen, z. B. zur Erleichterung einer Fehlerdiagnose oder zur Überwachung der Systemauslastung.

Prozess

Ein Prozess ist eine logische, zielgerichtete Folge von Funktionen, die zur Schaffung eines Produktes oder einer Dienstleistung dienen und in einem direkten Zusammenhang stehen. Prozesse transformieren Inputfaktoren zu einem Outputfaktor.

Prozess-Daten-Beschleuniger

Der Prozess-Daten-Beschleuniger (P23R) ist die zentrale Komponente der P23R-Infrastruktur. Der P23R generiert auf Anforderung automatisch eine Benachrichtigung gemäß den vorliegenden Benachrichtigungsregeln. Er verwendet dazu die vorab vom Unternehmen bereitgestellten Daten. Bevor eine Benachrichtigung an den Benachrichtigungsempfänger versendet wird, muss diese durch das Unternehmen freigegeben werden. Der P23R stellt nur Webservices im Sinne einer SOA bereit.

Prozesskette

Eine Prozesskette kann als eine logische Verknüpfung von Prozessen gesehen werden. Prozessketten stellen damit eine Kette zusammenhängender Prozesse dar, die zur Erstellung einer Dienstleistung oder eines Produkts (Wertschöpfungsorientierung) sowie zu einem gemeinsamen (Geschäfts-)Prozessziel führen sollen.

PRK-Typ I Wertschöpfungsorientierte Prozessketten: Diese Prozessketten beschreiben Wertschöpfungsprozesse, bei denen ein Unternehmen mit mehreren anderen Unternehmen und

P23R

P23R: Spezifikationen zur Infrastruktur

Verwaltungen interagieren muss. Sie zeichnen sich in der Regel durch eine hohe Anzahl an Prozessteilnehmern sowie durch eine komplexe Ablauflogik aus.

PRK-Typ II Datenorientierte Prozessketten: Diese Prozessketten beschreiben Prozesse, deren zentrales Element die daten- und ereignisgetriebene Übermittlung von Daten von den Unternehmen an die Verwaltung ist. Die in einer Prozesskette zwischen den Teilnehmern ausgetauschten Daten und Dokumente fließen oftmals auch in weitere Prozesse, so dass es zu Datenredundanzen kommt. Prozessketten vom Typ II zeichnen sich in der Regel durch eine geringe Anzahl an Prozessteilnehmern und durch eine einfache Ablauflogik aus. Die auszutauschenden Daten und Dokumente müssen im Prozess aufbereitet und an spezifische Formate angepasst werden. Sie weisen i. Allg. einen hohen Grad an Komplexität und Vertraulichkeit auf.

Prozesskettenbündel

Prozesskettenbündel bezeichnen die systematische Verbindung von mehreren Prozessketten zwischen Wirtschaft und Verwaltung mit dem Ziel, Effizienz, Effektivität sowie die Qualität von Informationen für alle Beteiligten zu verbessern. Es gibt unterschiedliche Kriterien, nach denen Prozesskettenbündel gebildet werden können.

In Abhängigkeit der angewendeten Kriterien unterscheidet man in Prozesskettenbündelung vom Typ I und Prozesskettenbündelung vom Typ II. Das Architekturkonzept des Prozess-Daten-Beschleunigers beschreibt technische Komponenten, die zur effizienten IT-Unterstützung von Prozesskettenbündeln eingesetzt werden können.

Prozesskettenbündelung Typ I

Bei der Prozesskettenbündelung vom Typ I werden Prozessketten zwischen Wirtschaft und Verwaltung mit einander verbunden, die entlang einer Wertschöpfungskette im Unternehmen auftreten. Solche Prozesskettenbündel sind durch eine hohe Anzahl von Akteuren und hohe Frequenz gekennzeichnet, da sie jedes Mal im Zusammenhang mit der Wertschöpfungskette im Unternehmen auftreten. Ziel der Bündelung ist eine möglichst reibungslose, medienbruchfreie Abwicklung der Wertschöpfungskette im Unternehmen sowie die effiziente Erfüllung gesetzlicher Informationspflichten. Ein Beispiel für eine derartige Prozesskettenbündelung vom Typ I ist die Vergabe von privaten Immobilienkrediten (vgl. [45]). Analysekriterien für die Identifikation von Prozessketten, die nach Typ I gebündelt werden können sind: Zugehörigkeit zu einem Wertschöpfungs- bzw. zu einem Prozess-Cluster Die Prozesskette wird ausgelöst durch den Wertschöpfungsprozess im Unternehmen, wie z. B. Meldung, Antrag, Registerauskunft.

Prozesskettenbündelung Typ II

Bei der Prozesskettenbündelung vom Typ II werden Prozessketten zwischen Wirtschaft und Verwaltung mit einander verbunden, die durch eine inhaltliche Überschneidung gekennzeichnet sind. Prozessketten, die gleiche oder ähnliche Inhalte zum Gegenstand haben werden mit einander so verbunden, dass sie nur noch eine gemeinsame Informationsbasis nutzen. Berichts- oder Meldedaten müssen auf diese Weise nicht mehr redundant ermittelt, gepflegt und archiviert werden. Ziel ist es, Berichts- und Meldepflichten an unterschiedliche Adressaten auf Verwaltungsseite möglichst effizient und mit hoher Informationsqualität abwickeln zu können. Analyse Kriterien für die Identifikation von Prozessketten, die nach Typ II gebündelt werden

können, sind: Übereinstimmung von Inhalt, Unternehmenstyp des Informationspflichtigen und Richtung des Informationsflusses (von Wirtschaft zu Verwaltung).

PRTR

Das PRTR (Pollutant Release and Transfer Register) ist ein Register für Schadstoffemissionen in der Luft, in den Böden, in Gewässern, in externen Kläranlagen sowie für entsorgte, gefährliche und nicht-gefährliche Abfälle. Das Register ist öffentlich im Internet zugänglich und informiert über insgesamt 91 Schadstoffe, die von großen Industriebetrieben freigesetzt werden. Das PRTR verfolgt das Ziel, die Öffentlichkeit für Umweltfragen zu sensibilisieren und an der Entscheidungsfindung im Umweltbereich zu beteiligen. Darüber hinaus soll die Umweltleistung von Unternehmen verbessert werden.

Quelle: [46]

Quellsystem (Source Application)

Mit Quellsystem wird das Softwaresystem beim Benachrichtigungssender bezeichnet, das die Daten für die Generierung der Benachrichtigungen in einem P23R bereitstellt.

Das kann bspw. das IT-Fachsystem eines Unternehmens sein.

Qualifizierte elektronische Signatur

Eine qualifizierte elektronische Signatur ist gemäß § 2 Nr. 3 [SigG] eine fortgeschrittene elektronische Signatur, die unter Verwendung einer sicheren Signaturerstellungseinheit erzeugt wurde und zum Zeitpunkt der Signaturerstellung auf einem gültigen qualifizierten Zertifikat beruht. Durch die qualifizierte elektronische Signatur kann die Schriftform ersetzt und somit auf kostenintensive Papierprozesse verzichtet werden. Außerdem hat eine qualifizierte elektronische Signatur gemäß § 371a [ZPO] eine sehr hohe Beweiskraft vor Gericht.

Reference Monitor

Ein Reference Monitor ist ein Designmuster für die Kontrolle und Durchsetzung von Zugriffsberechtigungen. Ein Reference Monitor kapselt eine zu schützende Ressource vollständig (complete mediation) und stellt sicher, dass jeder Zugriffsversuch auf diese Ressource mit definierten Zugriffsregeln abgeglichen wird. In der P23R-Sicherheitsarchitektur wird ein Reference Monitor durch das Access Control Subsystem realisiert. Die Anbindung an die Anwendungsarchitektur erfolgt über Policy Enforcement Points (PEPs), die aus den Abläufen der Anwendung heraus den Übergang zum Access Control Subsystem der Sicherheitsarchitektur bilden.

Release

Als Release werden verschiedene Stände der Datenmodelle (Schema-Änderung) bezeichnet. Bei der Anwendung von Regeln zur Erzeugung von Benachrichtigungen muss geprüft werden, ob die Daten dem vorgegebenen Datenmodell entsprechen. Zu einem Release gehören nicht nur ein neues Datenmodell, sondern auch die zugehörigen Änderungen der Transformationskripte. Diese werden i. Allg. von der Fachübergreifenden Koordinierungsstelle herausgegeben.

Rollen

Jeder Mitarbeiter erfüllt innerhalb seines Tätigkeitsprofils bestimmte Aufgaben, die Rollen definieren. Rollen können auch im Zusammenhang mit Anwendungsfällen definiert werden. Über

P23R

P23R: Spezifikationen zur Infrastruktur

diese Rollen werden Berechtigungen definiert, z. B. Zugriffsrechte auf Daten oder Schnittstellen einer Anwendung.

Schutzbedarf

Der Schutzbedarf beschreibt, welcher Schutz für die Geschäftsprozesse, die dabei verarbeiteten Informationen und die eingesetzte Informationstechnik ausreichend und angemessen ist.

Quelle: [41]

Schutzbedarfsfeststellung

Bei der Schutzbedarfsfeststellung wird der Schutzbedarf der Geschäftsprozesse, der verarbeiteten Informationen und der IT-Komponenten bestimmt. Hierzu werden für jede Anwendung und die verarbeiteten Informationen die zu erwartenden Schäden betrachtet, die bei einer Beeinträchtigung der Grundwerte der Informationssicherheit – Vertraulichkeit, Integrität oder Verfügbarkeit – entstehen können. Wichtig ist es dabei auch, die möglichen Folgeschäden realistisch einzuschätzen. Bewährt hat sich eine Einteilung in die drei Schutzbedarfskategorien „normal“, „hoch“ und „sehr hoch“.

Quelle: [41]

Serviceorientierte Architektur

Serviceorientierte Architekturen (SOA) beschreiben fachliche Architekturkonzepte zur Vernetzung und Verwendung verteilter Dienste bzw. Services (meist Webservices). Dabei werden die Anwendungsbausteine (Services) lose miteinander gekoppelt und je nach Bedarf zu umfassenden Diensten und Dienstleistungen verbunden (Service-Orchestrierung).

E-Government-Architekturen basieren zunehmend auf SOA-Konzepten.

Sicherheitsdienst

Ein Sicherheitsdienst trägt innerhalb einer Sicherheitsarchitektur zur Umsetzung von einem oder mehreren Sicherheitszielen (Vertraulichkeit, Integrität, Verfügbarkeit) bei. Beispiele für Sicherheitsdienste sind Nutzerauthentifizierung und Zugriffskontrolle.

Sicherheitskonzept

Ein Sicherheitskonzept dient zur Umsetzung der Sicherheitsstrategie und beschreibt die geplante Vorgehensweise, um die gesetzten Sicherheitsziele einer Institution zu erreichen. Das Sicherheitskonzept ist das zentrale Dokument im Sicherheitsprozess eines Unternehmens bzw. einer Behörde. Jede konkrete Sicherheitsmaßnahme muss sich letztlich darauf zurückführen lassen.

Quelle: [41]

Sicherheitsmaßnahme

Mit Sicherheitsmaßnahme (kurz Maßnahme) werden alle Aktionen bezeichnet, die dazu dienen, um Sicherheitsrisiken zu steuern und um diesen entgegenzuwirken. Dies schließt sowohl organisatorische, als auch personelle, technische oder infrastrukturelle Sicherheitsmaßnahmen ein. Synonym werden auch die Begriffe Sicherheitsvorkehrung oder Schutzmaßnahme benutzt. Als englische Übersetzung wurde „safeguard“, „security measure“ oder „measure“

gewählt. Im englischen Sprachraum wird neben „safeguard“ außerdem häufig der Begriff „control“ verwendet.

Quelle: [41]

Sicherheitsobjekt

Sicherheitsobjekte sind in Bezug auf ihre Integrität, Authentizität und ggf. auch Vertraulichkeit besonders abgesicherte Objekte, die als Ankerpunkte für darauf aufsetzende Sicherheitsmechanismen dienen. Beispiele für Sicherheitsobjekte sind kryptografische Schlüssel und Identifikatoren.

SourceConnector (Quelldatenkonnektor)

Der SourceConnector ist eine externe Systemkomponente, die nicht zum P23R gehört, und typischerweise vom Hersteller der SourceApplication oder dem P23R-Betreiber bereitgestellt wird. Der SourceConnector muss die normative Schnittstelle ISourceDataRead für den P23R bereitstellen, damit dieser auf die Daten der SourceApplication zugreifen kann. Ob der SourceConnector eine separate Systemkomponente oder eine in die SourceApplication integrierte Schnittstelle ist, ist der Implementierung selbst überlassen, solange die Schnittstelle realisiert wird.

Stakeholder

Als Stakeholder wird eine natürliche Person (der Mensch in seiner Rolle als Rechtssubjekt) oder eine juristische Person (z. B. eine Institution) bezeichnet, die ein Interesse am Verlauf oder Ergebnis des P23R-Projekts hat.

Quelle: [47]

Unternehmenscharakteristik

Die Unternehmenscharakteristik ist die Menge aller relevanten Eigenschaften eines Unternehmens, die zur Bestimmung der durch den P23R zu empfehlenden Benachrichtigungsregelgruppen und -regeln erforderlich sind.

Unterstützungsaufgaben

Hier handelt es sich um die Gesamtheit an Aufgaben, die die erfolgreiche Umsetzung des P23R-Prinzips unterstützen. Dies sind bspw. die fachübergreifende Koordinierungsaufgaben, die technische Bereitstellung der Benachrichtigungsregel- und Datenmodellpakete sowie Beratungsaufgaben bei der Erstellung von Benachrichtigungsregeln.

Siehe *Fachübergreifende Koordinierungsaufgaben*.

Verfügbarkeit

Die Verfügbarkeit von Dienstleistungen, Funktionen eines IT-Systems, IT-Anwendungen oder IT-Netzen oder auch von Informationen ist vorhanden, wenn diese von den Anwendern stets wie vorgesehen genutzt werden können.

Quelle: [41]

Verschlüsselung

Verschlüsselung (Chiffrieren) transformiert einen Klartext in Abhängigkeit von einer Zusatzinformation, die "Schlüssel" genannt wird, in einen zugehörigen Geheimtext (Chifftrat), der für

P23R

P23R: Spezifikationen zur Infrastruktur

diejenigen, die den Schlüssel nicht kennen, nicht entzifferbar sein soll. Die Umkehrtransformation - die Zurückgewinnung des Klartextes aus dem Geheimtext - wird Entschlüsselung genannt.

Quelle: [41]

Version

Der Begriff Version wird verwendet, um verschiedene zeitliche Zustände der gleichen Daten zu beschreiben. Jede Änderung von Daten erzeugt eine neue Version (eine neue Instanz) dieser Daten. Beim P23R müssen ältere Versionen archiviert werden, d. h. sie dürfen nicht verloren gehen oder gelöscht werden. Änderungen, z. B. die Beseitigung eines Schreibfehlers in einem Attribut, Datenergänzungen etc., werden vom Unternehmen angestoßen und vom Datenpool erzeugt.

Versionsnummer

Eine Versionsnummer ist die konkrete Bezeichnung für den Stand der Daten (Version).

Vertraulichkeit

Vertraulichkeit ist der Schutz vor unbefugter Preisgabe von Informationen. Vertrauliche Daten und Informationen dürfen ausschließlich Befugten in der zulässigen Weise zugänglich sein.

Quelle: [41]

Vorgang

Menge aller Dokumente, Aktionen und Ereignisse im P23R im Zusammenhang mit einer bestimmten Benachrichtigung vom ersten auslösenden Ereignis bis zur letzten korrigierten abschließenden Benachrichtigung, ihrer Auslieferung und Archivierung.

Vorgangsnummer

Die Vorgangsnummer ist der Identifikator eines Vorgangs.

Vorschriftengeber

Vorschriftengeber ist in der Regel der Gesetzgeber. In einigen Fällen ist die Situation komplexer. Dies gilt dann, wenn der Gesetzgeber nur einen Rechtsrahmen schafft, der von einer anderen Körperschaft auszugestaltet ist (z. B. Rechtsrahmen für die Berufsgenossenschaften oder die Ausgestaltung von Durchführungsverordnungen). In solchen Fällen müssen alle beteiligten Stellen an den Aufgaben des Vorschriftengebers mitwirken. Nur indirekt betroffen sind die Empfänger auf Vollzugsebene.

Webservice

Ein Webservice ist eine interoperable Softwareschnittstelle, die über XML beschrieben ist und die über in XML kodierte Nachrichten angesprochen wird.

Wert

Werte sind alles, was wichtig für eine Institution ist (Vermögen, Wissen, Gegenstände, Gesundheit).

Quelle: [41]

Wertschöpfung

Wenn ein Wirtschaftsakteur eine Wertschöpfung erstellt, transformiert er durch seine wirtschaftliche Leistung vorhandene Güter in Güter mit höherem Geldwert. Wertschöpfung bezeichnet diese in den unterschiedlichen Wirtschaftsbereichen erbrachte wirtschaftliche Leistung. Die Wertschöpfung berechnet sich wirtschaftsmathematisch aus der Differenz zwischen Produktionswert und Vorleistungen.

Zertifizierung

Als Zertifizierung wird die Zertifizierung nach ISO 27001 auf der Basis von IT-Grundschutz bezeichnet.

Zielgruppe

Als Zielgruppen wird eine bestimmte Menge von Stakeholdern bezeichnet, die auf kommunikationspolitische Maßnahmen homogener reagieren als die Gesamtmenge aller Stakeholder.

Quelle: [43]

Zugang

Mit Zugang wird die Nutzung von IT-Systemen, System-Komponenten und Netzen bezeichnet. Zugangsberechtigungen erlauben somit einer Person, bestimmte Ressourcen, wie IT-Systeme bzw. System-Komponenten und Netze, zu nutzen.

Quelle: [41]

Zugriff

Mit Zugriff wird die Nutzung von Informationen bzw. Daten bezeichnet. Über Zugriffsberechtigungen wird geregelt, welche Personen im Rahmen ihrer Funktionen oder welche IT-Anwendungen bevollmächtigt sind, Informationen, Daten oder auch IT-Anwendungen, zu nutzen oder Transaktionen auszuführen.

Quelle: [41]

P23R

P23R: Spezifikationen zur Infrastruktur

12 ABKÜRZUNGSVERZEICHNIS

AGM	Arbeitgebermeldepflichten
API	Application Programming Interface
AT	Anwendungstest
BG	Berufsgenossenschaft
BR	Benachrichtigungsregel
BRG	Benachrichtigungsregelgruppe
BRS	Benachrichtigungsregelsprache
BRP	Benachrichtigungsregelpaket
BRV	Benachrichtigungsregelvariante
BRW	Benachrichtigungsregelwerk
BSI	Bundesamt für Sicherheit und Informationstechnik
CC	Control-Center
CL	P23R-Client
CR	Candidate Release
DB	DataBase
DEÜV	Datenerfassungs- und Übermittlungsverordnung
DSL	Domain Specific Language
DTP	Data Pool
e.G.	eingetragene Genossenschaft
ESB	Enterprise-Service-Bus
ESI	Electronic Signatures and Infrastructures
F-BRS	Fachliche Benachrichtigungsregelsprache
FOP	Formatting Objects Processor
FTP	File Transfer Protocol
GA	General Availability
GmbH	Gesellschaft mit Beschränkter Haftung
HTTP(S)	Hyper Text Transfer Protocol (Secure)
IP	Internet Protocol
IT	Information Technology
JAXB	Java Architecture for XML Binding
JPA	Java Persistence API
JSF	JavaServer Faces
LDAP	Lightweight Directory Access Protocol
MARM	Model And Rule Management
MIME	Multipurpose Internet Mail Extensions
P23R	Prozess-Daten-Beschleuniger
PAP	Policy Administration Point
PEP	Policy Enforcement Point
PDF	Portable Document Format
PDP	Policy Decision Point
PEP	Policy Enforcement Point

P23R

P23R: Spezifikationen zur Infrastruktur

PIP	Policy Information Point
PRK	Prozesskette
PRTR	Pollutant Release and Transfer Register
REST	Representational State Transfer
SAML	Security Assertion Markup Language
SE	Societas Europaea, dt.: Europäische Gesellschaft
SFTP	Secure File Transfer Protocol
SMTP	Simple Mail Transfer Protocol
SOA	Service Orientierte Architektur
SOAP	Simple Object Access Protocol
SP	Service Pack
SSH	Secure-Shell
SVN	Apache Subversion
TA	Target Application
T-BRS	Technischen Benachrichtigungsregelsprache
TK	Techniker Krankenkasse
TSL	Trusted Service List
UC	Use Case
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
UWM	Umweltmeldepflichten
VM	Virtual Machine
WS	Webservice
WSDL	Web-Service Description Language
WSIT	Web Services Interoperability Technology
XACML	eXtensible Access Control Markup Language
XSL	Extensible Stylesheet Language
XSLT	Extensible Stylesheet Language Transformations
XML	Extensible Markup Language

13 REFERENZEN

Alle in diesem Kapitel aufgeführten Ergebnisdokumente des P23R-Projekts werden unter www.p23r.de bereitgestellt werden.

- [1] J. Gottschick et al. (2012), *P23R: Rahmenarchitektur*. (Ergebnisdokument des P23R-Projekts)
- [2] S. Dutkowski, E. Klochkova und A. Söllner (2013), *P23R: Pflichtenheft zur Infrastruktur*. (Ergebnisdokument des P23R-Projekts)
- [3] *The Java Persistence API - A Simpler Programming Model for Entity Persistence*.
Verfügbar unter: <http://www.oracle.com/technetwork/articles/javaee/jpa-137156.html>
(zuletzt abgerufen am 20.10.2011).
- [4] *W3C XML Schema (XSD)*. Verfügbar unter: <http://www.w3.org/XML/Schema>
(zuletzt abgerufen am 21.10.2011).
- [5] J. Gottschick (2012), *P23R: Spezifikation der Technischen Benachrichtigungsregelsprache*. (Ergebnisdokument des P23R-Projekts)
- [6] *JBoss WS - Web Services Framework for JBoss AS*.
Verfügbar unter: <http://jboss.org/jbossws> (zuletzt abgerufen am 20.10.2011).
- [7] *JBoss ESB, Reliable SOA infrastructure*.
Verfügbar unter: <http://jboss.org/jbossesb> (zuletzt abgerufen am 20.10.2011).
- [8] *XText-Framework*. Verfügbar unter: <http://www.eclipse.org/Xtext/>
(zuletzt abgerufen am 31.10.2011).
- [9] R. Rosenmüller, F. Clauder und H. Hartenstein (2012), *P23R: Spezifikationen zur Rahmenarchitektur*. (Ergebnisdokument des P23R-Projekts)
- [10] *Quartz - Enterprise Job Scheduler*.
Verfügbar unter: <http://www.quartz-scheduler.org/> (zuletzt abgerufen am 20.10.2011).
- [11] *Web Services Interoperability Technologies (WSIT)*.
Verfügbar unter: <http://wsit.java.net/> (zuletzt abgerufen am 20.10.2011).
- [12] J. Caumanns et al. (2012), *P23R: Sicherheitsarchitektur*. (Ergebnisdokument des P23R-Projekts)
- [13] *OpenSAML*. Verfügbar unter:
<https://wiki.shibboleth.net/confluence/display/OpenSAML/Home>
(zuletzt abgerufen am 21.10.2011).
- [14] *The Seam Framework - Next generation enterprise Java development*.
Verfügbar unter: <http://seamframework.org/> (zuletzt abgerufen am 20.10.2011).
- [15] *JBoss Application Server*. Verfügbar unter: <http://jboss.org/jbossas>
(zuletzt abgerufen am 20.10.2011).
- [16] *RichFaces - Rich AJAX enabled components for your JSF applications*.
Verfügbar unter: <http://www.jboss.org/richfaces> (zuletzt abgerufen am 20.10.2011).

P23R

P23R: Spezifikationen zur Infrastruktur

- [17] *eXist-db - Open Source Native XML Database*.
Verfügbar unter: <http://exist.sourceforge.net/> (zuletzt abgerufen am 20.10.2011).
- [18] S. Dutkowski, E. Klochkova und A. Söllner (2013), *P23R: Lastenheft zu Szenarien und Datenmodellen*. (Ergebnisdokument des P23R-Projekts)
- [19] *Apache FOP - Formatting Objects Processor*.
Verfügbar unter: <http://xmlgraphics.apache.org/fop/> (zuletzt abgerufen am 20.10.2011).
- [20] *Avoid basic style errors – Collected Java Practices*. Verfügbar unter:
<http://www.javapractices.com/topic/TopicAction.do;jsessionid=C971FA95A4818DACFF44168A505E16D?Id=227> (zuletzt abgerufen am 20.10.2011).
- [21] *Collected Java Practices*. Verfügbar unter:
<http://www.javapractices.com/home/HomeAction.do> (zuletzt abgerufen am 20.10.2011).
- [22] *Secure Coding Guidelines for the Java Programming Language*. Verfügbar unter:
<http://www.oracle.com/technetwork/java/seccodeguide-139067.html#references>
(zuletzt abgerufen am 20.10.2011).
- [23] *Eclipse Integrated Development Environment*.
Verfügbar unter: <http://www.eclipse.org/> (zuletzt abgerufen am 20.10.2011).
- [24] *Javadoc Tool*. Verfügbar unter:
<http://www.oracle.com/technetwork/java/javase/documentation/index-jsp-135444.html>
(zuletzt abgerufen am 20.10.2011).
- [25] *How to Write Doc Comments for the Javadoc Tool*. Verfügbar unter:
<http://www.oracle.com/technetwork/java/javase/documentation/index-137868.html>
(zuletzt abgerufen am 20.10.2011).
- [26] *Maven – Software Project Management Tool*.
Verfügbar unter: <http://maven.apache.org/> (zuletzt abgerufen am 20.10.2011).
- [27] *Version number rules – Versions Maven Plugin*.
Verfügbar unter: <http://mojo.codehaus.org/versions-maven-plugin/version-rules.html>
(zuletzt abgerufen am 20.10.2011).
- [28] *Semantic Versioning - Semantic Versioning Specification (SemVer)*.
Verfügbar unter: <http://semver.org/> (zuletzt abgerufen am 20.10.2011).
- [29] *Improve default support for version schemes in Maven*. Verfügbar unter:
<http://docs.codehaus.org/display/MAVEN/Versioning> (zuletzt abgerufen am 20.10.2011).
- [30] *Simple Logging Facade for Java (SLF4J)*.
Verfügbar unter: <http://www.slf4j.org/> (zuletzt abgerufen am 20.10.2011).
- [31] *Pulse – Managing Eclipse based Tool Suites*.
Verfügbar unter: <http://www.poweredbypulse.com/> (zuletzt abgerufen am 20.10.2011).
- [32] *Subversion*. Verfügbar unter: <http://subversion.apache.org/>
(zuletzt abgerufen am 20.10.2011).
- [33] *Version Control with Subversion*.
Verfügbar unter: <http://svnbook.red-bean.com/> (zuletzt abgerufen am 20.10.2011).

-
- [34] *Nexus | the repository manager.*
Verfügbar unter: <http://nexus.sonatype.org/> (zuletzt abgerufen am 20.10.2011).
- [35] *The Trac Project.*
Verfügbar unter: <http://trac.edgewall.org/> (zuletzt abgerufen am 20.10.2011).
- [36] *Hudson - Extensible continuous integration server.*
Verfügbar unter: <http://hudson-ci.org/> (zuletzt abgerufen am 20.10.2011).
- [37] *Sonar – Code Quality Management Platform.*
Verfügbar unter: <http://www.sonarsource.org/> (zuletzt abgerufen am 20.10.2011).
- [38] Bundesamt für Sicherheit in der Informationstechnik (BSI) (2006), *Das E-Government-Glossar.*
Verfügbar unter:
https://www.bsi.bund.de/cae/servlet/contentblob/476872/publicationFile/31173/6_EGloss_pdf (zuletzt abgerufen am 26.10.2011).
- [39] P. Kramer und M. Meints, „Datenschutz“, in: *Handbuch Multimedia-Recht, T. Hoeren und U. Sieber (Hrsg.)*, 23. Auflage. München: Beck, 19. Einzellieferung vom 19. März 2008., Teil 16.5, Rn. 3 ff.
- [40] *Statistische Ämter des Bundes und der Länder.*
Verfügbar unter: www.statspez.de/core/ (zuletzt abgerufen am 11.05.2011).
- [41] Bundesamt für Sicherheit in der Informationstechnik (BSI), *IT-Grundschutz-Glossar.*
Verfügbar unter:
https://www.bsi.bund.de/DE/Themen/weitereThemen/ITGrundschutzKataloge/Inhalt/Glossar/glossar_node.html (zuletzt abgerufen am 31.10.2012).
- [42] M. Bruhn (2007), *Kommunikationspolitik*, 4. überarbeitete Auflage, Verlag Franz Vahlen GmbH, München.
- [43] Gabler Verlag (Hrsg.), *Gabler Wirtschaftslexikon*, Verfügbar unter:
<http://wirtschaftslexikon.gabler.de/> (zuletzt abgerufen am 07.11.2012).
- [44] Bundesamt für Sicherheit in der Informationstechnik (BSI), *BSI Standard 100-4: Notfallmanagement.* Verfügbar unter:
https://www.bsi.bund.de/cae/servlet/contentblob/471456/publicationFile/30746/standard_1004.pdf (zuletzt abgerufen am 29.10.2012).
- [45] N. Frösche et al. (2009), *Machbarkeitsstudie Entwicklung von Prozessketten zwischen Wirtschaft und Verwaltung: Finanzdienstleistungen.*
Verfügbar unter: <http://www.p23r.de/publikationen/> (zuletzt abgerufen am 17.11.2011).
- [46] Umweltbundesamt, *Leitfaden für die Durchführung der PRTR-Berichtspflicht.*
Verfügbar unter:
http://www.bmwfj.gv.at/Unternehmen/gewerbetechnik/Documents/Nationaler%20E_PRTR_Leitfaden.pdf (zuletzt abgerufen am 10.10.2011).
- [47] R. Olbrich (2009), *Marketing – Eine Einführung in die marktorientierte Unternehmensführung*, 2. Auflage, Springer-Verlag GmbH, Heidelberg.

Herausgeber

Fraunhofer-Institut für Offene Kommunikationssysteme FOKUS
Kaiserin-Augusta-Allee 31
10589 Berlin

Kontakt

info@p23r.de
www.p23r.de

Redaktion

Johannes Einhaus, Fraunhofer FOKUS
Dominique Leikauf, :::tsm total-sourcing-management
Petra Steffens, Fraunhofer FOKUS

Layout und Satz

Marie Luise Birkholz, Fraunhofer FOKUS
Simone Geppert, Fraunhofer FOKUS

Nachdruck und Weitergabe sind nur unter der Bedingung gestattet,
dass das Dokument unverändert bleibt.

www.p23r.de

