P23R

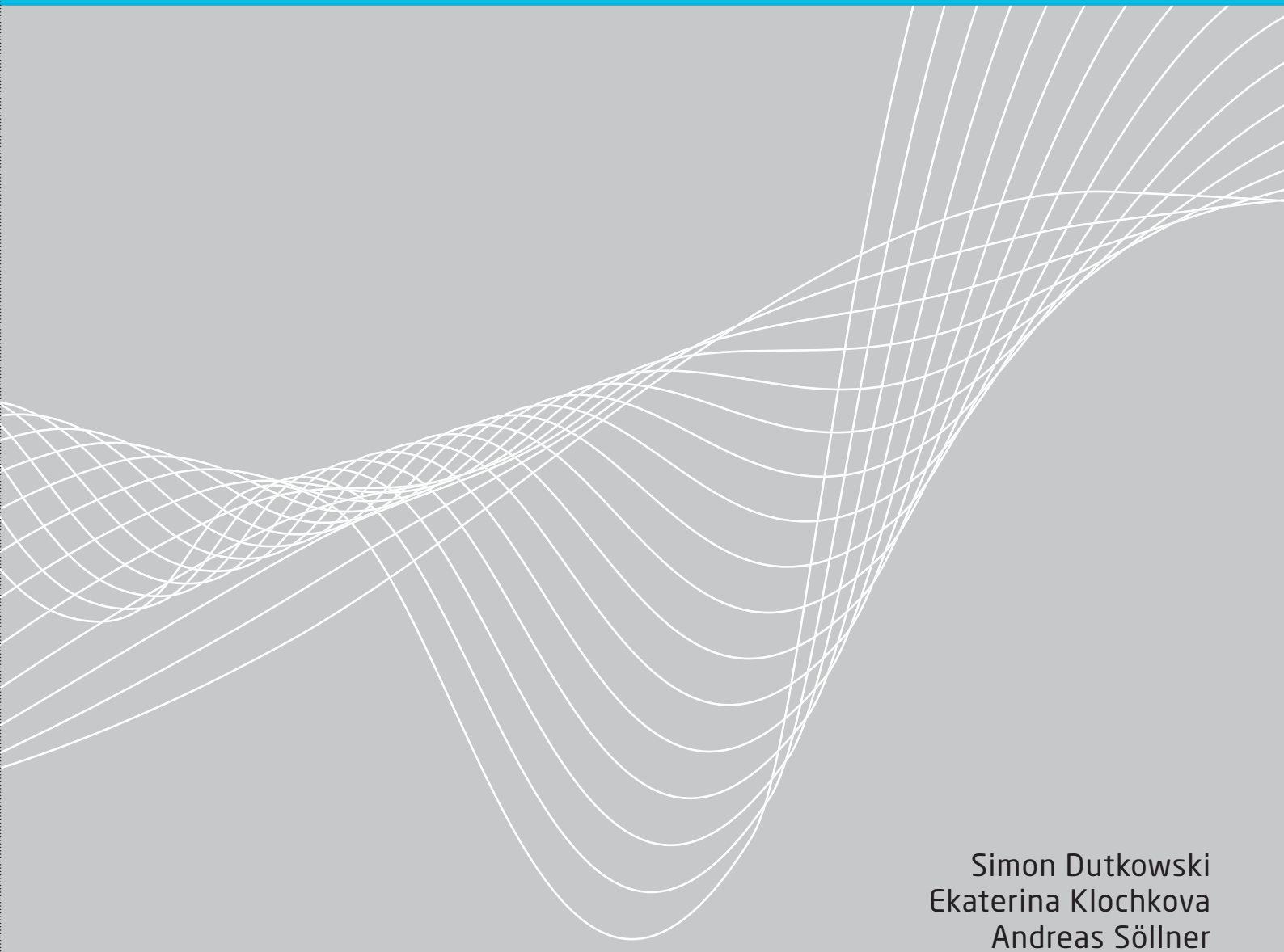
NON-STOP-GOVERNMENT

IT-INVESTITIONS-PROGRAMM

Wir gestalten Zukunft.

P23R: Pflichtenheft 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

Fraunhofer

Projektbeteiligte

Atos

Eberhard Karls Universität Tübingen

ESG Consulting

Fraunhofer FOKUS

Fraunhofer IAO

Fraunhofer IESE

Fraunhofer ISST

Fraunhofer SIT

Metropolregion Rhein-Neckar

provet Projektgruppe verfassungsverträgliche Technikgestaltung

tsm total-sourcing-management

GAD IT für Banken

Projekt

P23R | Prozess-Daten-Beschleuniger

P23R: Pflichtenheft zur Infrastruktur

Ergebnisdokument

Januar 2013

Autoren

Simon Dutkowski, Fraunhofer FOKUS

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

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

Zusammenfassung

Um den im Projekt „Pilotierung und Realisierung eines Prozess-Daten-Beschleunigers“ angestrebten Nachweis der Anwendbarkeit, dessen Vorgaben im Detail im zugehörigen Lastenheft detailliert dargestellt sind, zu erbringen, gilt es, eine exemplarische Musterimplementierung der P23R-Infrastruktur umzusetzen und gemeinsam mit den Pilotpartnern zur Anwendung zu bringen.

Hierzu wird eine weitreichende Umsetzung der durch die P23R-Rahmen- und Sicherheitsarchitektur NORMATIV vorgegebenen Schnittstellen durchgeführt und in einer P23R-Lösung, der P23R-Musterimplementierung, integriert.

Um jedoch auch den INFORMELLEN Teil der Architektur einem Nachweis zu unterziehen, setzt die P23R-Musterimplementierung auch die INFORMELL beschriebenen Komponenten um, anstatt sich auf Basis der NORMATIV vorgegebenen Schnittstellen eine eigene Lösungsarchitektur aufzubauen.

Im Kern beruht die P23R-Musterimplementierung auf einer Umsetzung der Generation-Pipeline zur Generierung und Übermittlung von Benachrichtigungen zwischen Nachrichtensender und Benachrichtigungsempfänger, wie dies in der P23R-Rahmenarchitektur als technische Umsetzung des P23R-Prinzips beschrieben ist.

Alle in diesen Ablauf integrierten Komponenten und Sicherheitskomponenten werden in dem Maße umgesetzt, wie diese durch die zum Nachweis der Anwendbarkeit ausgewählten Informations- und Meldepflichten aus der Domäne Arbeitgebermeldungen gefordert werden.

Dazu zählen neben einer Implementierung des P23R (und seiner Komponenten) selbst auch die exemplarische Umsetzung eines P23R-Clients (als User-Interface-Element) und die Realisierung einer Leitstellenlösung mit den für die Bereitstellung benötigten Depots.

Einhergehend damit werden in einer Laborinstallation der Leitstelle die benötigten Datenmodell- und Regelartefakte (inkl. Zuständigkeitsinformationen) sowie die für die P23R-Sicherheitsarchitektur nötigen Trusted Service Lists bereitgestellt.

Basierend darauf erfolgt dann ein Pilotversuch, für den die P23R-Musterimplementierung um die zur Bereitstellung der Unternehmensdaten erforderlichen Quelldatenkonnektoren zu einer Pilot-Lösung angereichert wird, um eine hinreichende Aussagekraft über die Machbarkeit des P23R-Prinzips zu erlangen.

Das vorliegende Dokument enthält nun Festlegungen über den Grad der Umsetzung der einzelnen Komponenten sowie der benötigen Regeln, Datenmodelle und Zuständigkeitsinformationen im Kontext der Vorgaben der P23R-Rahmen- und Sicherheitsarchitektur und der für die P23R-Musterimplementierung getroffenen Technologieentscheidung.

Executive Summary

To prove the applicability, which was detailed described in “Requirement specification” and which was aimed in the project “management and implementation of the data-process-accelerator”, it was necessary to put together with the project partners a pattern-implementation of the P23R infrastructure into action through its implementation and a following application.

For that a far reaching implementation was conducted by the predefined interfaces in the P23R architectuere and security infrastructure and the pattern implementation was integrated into a P23R-solution.

To prove the informal part of the architecture, the sample implementation includes also the components described informally and doesn’t build its own solution architecture based on the provided interfaces.

The P23R sample implementation is based on the implementation of the Generation-Pipeline for generation and transmission of notifications between message sender and message receiver. This is described as technical implementation in the P23R framework architecture.

All integrated components and safety components will be implemented in a way to demonstrate the applicability of the selected information and reporting requirements from the domain of em-ployers reporting requirements (AGM).

This includes the implementation of P23R (and its components), the sample implementation of a P23R client as a UI element and the realisation of a controll center solution including the sufficient depots which are needed for providing.

A labaratory installation of the control center provides the needed data model and rule artifacts (including responsibility information) as well as the trsuted service lists that are necessary for the security infrastructure.

A pilot test will be done on this basis which is concentrated to a pilot soultion becauso of the P23R sample implementation and the needed source data connectors that are necessary for providing the company data.

The aim is the gratest evidence concerncin the feasibility of the P23R principe.

The document presents the determinations about the level of implementation concerning the individual components as well as the required rules, data models and jurisdiction information in the context of the requirements of the P23R framework and security architecture well as the technology decissions which are made for the P23R sample implementation.

Inhaltsverzeichnis

1 Einleitung 1

1.1 Zweck des Dokuments 1

1.2 Leserkreis 1

1.3 Kontext, Inhalte und Strukturierung 1

2 Umsetzung der P23R-Musterimplementierung 3

2.1 Der Prozess-Daten-Beschleuniger (P23R) 5

2.1.1 Nachrichtenempfang 7

2.1.2 Benachrichtigungsgenerierung 10

2.1.3 Benachrichtigungsversand 12

2.1.4 Benachrichtigungstransport 15

2.1.5 Datenpool 17

2.1.6 Protokollpool 19

2.1.7 Datenmodelle und Benachrichtigungsregeln 21

2.1.8 Termine und Zeitüberschreitungen 25

2.1.9 Bootstrapping 27

2.2 Die P23R-Leitstelle und das Leitstellenportal 29

2.2.1 Das Benachrichtigungsregelpaket-Depot 31

2.2.2 Das Trusted-Service-List-Depot 32

2.2.3 Das P23R-Zuständigkeitsverzeichnis 32

2.3 Der P23R-Client als Onlineanwendung 34

2.3.1 Anwendungsfälle zur Benutzung 36

2.3.2 Verwaltung der Benachrichtigungsregeln 37

2.3.3 Pflege der Unternehmensdaten 38

2.3.4 Steuerung der Benachrichtigungsübermittlung 38

2.3.5 Integration der Sicherheitskomponenten 39

2.4 Die generischen Konnektoren der P23R-Musterimplementierung 43

2.4.1 Der Quelldatenkonnektor 43

2.4.2 Der Kommunikationskonnektor 43

2.5 Die Integration der Sicherheitsarchitektur 44

2.5.1 Integrität, Authentizität und Vertraulichkeit 44

2.5.2 Übersicht der abgebildeten Sicherheitsmechanismen 45

3 Bereitstellung und Bestückung der Laborleitstelle 47

3.1 TSL für die Laborleitstelle 47

3.2 Das Testdatenmodell- und Testbenachrichtigungsregelpaket 47

3.2.1 Datenmodellpaket Testmeldungen 48

3.2.2 Benachrichtigungsregelpaket Testmeldungen 48
3.2.3 Zuständigkeitsinformationen Testmeldungen 49
3.3 Das Pilotdatenmodell- und Pilotbenachrichtigungsregelpaket 49
3.3.1 Datenmodellpaket Arbeitgebermeldungen 49
3.3.2 Benachrichtigungsregelpaket Arbeitgebermeldungen 49
3.3.3 Zuständigkeitsinformationen Arbeitgebermeldungen 50
4 Abwicklung des Pilotversuchs in der Domäne AGM 51
4.1 Austausch der Unternehmensdaten 51
4.1.1 Quelldatenkonnektor „BASFsap“ 52
4.1.2 Quelldatenkonnektor „DATEVraw“ 53
4.2 Realisierung der Kommunikationskonnektoren 54
4.2.1 Kommunikationskonnektor „eSTATISTIKcore“ 55
4.2.2 Kommunikationskonnektor „vDEÜV“ 56
4.2.3 Kommunikationskonnektor „BGviaMail“ 58
4.3 Pilotumgebung bei den Pilotpartnern 59
5 Durchführung von Prüf- und Testverfahren 61
5.1 Einsatz von Testregeln und Testdatensätzen nebst Testergebnissen 61
5.2 Protokollierung relevanter Ereignisse innerhalb des P23R 62
5.3 Durchführen von Anwendungs- und End-To-End-Tests 62
5.3.1 Anwendungs-Tests am Leitstellenportal 63
5.3.2 Anwendungs-Tests am P23R-Client 64
6 Anhang I: Darstellung der Entwicklungs- und Laborumgebung 69
6.1 Entwicklungsumgebung 70
6.2 Integrations- und Testumgebung 70
6.3 Präsentationsumgebung 70
7 Glossar 73
8 Abkürzungsverzeichnis 95
9 Referenzen 97

Verzeichnis der Abbildungen

Abbildung 1: Bausteine der P23R-Infrastruktur (UML) [2] 3

Abbildung 2: Überblick über die NORMATIVEN Schnittstellen im Kontext des P23R (UML) [2] 4

Abbildung 3: Funktionale Blöcke des P23R (UML) [2] 5

Abbildung 4: Überblick über die Generation-Pipeline (UML) [2] 6

Abbildung 5: Überblick über die Support-Packages (UML) [2] 6

Abbildung 6: Komponentenübersicht für den Nachrichtenempfang (UML) [2] 8

Abbildung 7: Komponentenübersicht für die Benachrichtigungsgenerierung (UML) [2] 11

Abbildung 8: Komponentenübersicht für den Benachrichtigungsversand (UML) [2] 13

Abbildung 9: Komponenten für den Benachrichtigungstransport (UML) [2] 15

Abbildung 10: Komponentenübersicht für den Datenpool (UML) [2] 18

Abbildung 11: Komponenten für den Protokollpool (UML) [2] 20

Abbildung 12: Komponentenübersicht für Datenmodelle / Benachrichtigungsregeln (UML) [2] 21

Abbildung 13: Komponentenübersicht für Termine und Zeitüberschreitungen (UML) [2] 26

Abbildung 14: Komponentenübersicht für das P23R-Depot (UML) [2] 30

Abbildung 15: Komponenten für den P23R-Client (UML [2]) 35

Abbildung 16: Zusammenspiel "Unternehmensdaten" zwischen P23R-Client und P23R 38

Abbildung 17: Integration der Sicherheitskomponenten (UML) [3] 40

Abbildung 18: Lokale Authentisierung im Unternehmen (UML) [3] 41

Abbildung 19: Abruf einer oder mehrerer Berechtigungs-Policies (UML) [3] 42

Abbildung 20: Muster für einen Dienst bei eingehenden Anfragen (UML) [2] 44

Abbildung 21: Übersicht Pilotdeployment [1] 51

Abbildung 22: SourceConnector „BASF.sap“ 52

Abbildung 23: SourceConnector „DATEV.raw“ 53

Abbildung 24: Kommunikationskonnektor „eStatistikcore“ (UML) 55

Abbildung 25: Kommunikationskonnektor „vDEÜV“ (UML) 56

Abbildung 26: Ausschnitt DEÜV Datenbausteine für Abgabegrund 50 57

Abbildung 27: Kommunikationskonnektor „BGviaMail“ (UML) 58

Abbildung 28: Stufenweise Entwicklung und Integration 69

Verzeichnis der Tabellen

Tabelle 1: Funktionale Abbildung Nachrichtenempfang 9

Tabelle 2: Funktionale Abbildung Benachrichtigungsgenerierung 11

Tabelle 3: Funktionale Abbildung Benachrichtigungsversand 13

Tabelle 4: Funktionale Abbildung Benachrichtigungstransport 16

Tabelle 5: Funktionale Abbildung Datenpool 19

Tabelle 6: Funktionale Abbildung Protokollpool 20

Tabelle 7: Funktionale Abbildung Datenmodelle und Benachrichtigungsregeln 23

Tabelle 8: Funktionale Abbildung Termine und Zeitüberschreitungen 27

Tabelle 9: Funktionale Abbildung Bootstrapping 28

Tabelle 10: Übersicht der Anwendungsfälle aus dem Lastenheft: Leitstelle 30

Tabelle 11: Wertetabelle „Benutzer im P23R-Leitstellenportal“ 31

Tabelle 12: Funktionale Abbildung Benachrichtigungsregelpaket-Depot 32

Tabelle 13: Funktionale Abbildung Trusted-Service-List-Depot 32

Tabelle 14: Funktionale Abbildung P23R-Zuständigkeitsverzeichnis 33

Tabelle 15: Übersicht der Anwendungsfälle aus dem Lastenheft: P23R-Client 34

Tabelle 16: Funktionale Abbildung P23R-Client: Anwendungsfälle zur Benutzung 36

Tabelle 17: Funktionale Abbildung P23R-Client: Verwaltung der Benachrichtigungsregeln 37

Tabelle 18: Funktionale Abbildung P23R-Client: Steuerung der Übermittlung 39

Tabelle 19: Wertetabelle „Benutzer im P23R-Client“ 41

Tabelle 20: Wertetabelle „Rollen im P23R-Client“ 42

Tabelle 21: Übersicht der Anwendungsfälle aus dem Lastenheft: Quelldatenkonnektor 43

Tabelle 22: Übersicht der Anwendungsfälle aus dem Lastenheft: Kommunikationskonnektor 43

Tabelle 23: Funktionale Abbildung der Sicherheitsmechanismen 45

Tabelle 24: Verknüpfung der Teildatenmodelle des Pivotdatenmodells AGM 49

Tabelle 25: Infrastrukturbedarf Pilotpartner 59

Verzeichnis der Tabellen

Tabelle 26: Kommunikationsbeziehungen Pilotinfrastruktur 60

Tabelle 27: Ereignisprotokollierung je Komponenten 62

Tabelle 28: Tabellenzeilen zur Testfallauswertung 63

Tabelle 29: AT-CC-01 An- und Abmelden eines Benutzers 63

Tabelle 30: AT-CC-02 Verwalten der Benutzerkonten 63

Tabelle 31: AT-CC-03 Verwalten der Regelpakete 64

Tabelle 32: AT-CC-04 Verwaltung des Zuständigkeitsverzeichnisses 64

Tabelle 33: AT-CL-01 An- und Abmelden eines Benutzers 65

Tabelle 34: AT-CL-02 Verwalten der Benutzerkonten 65

Tabelle 35: AT-CL-03 Einsicht in das P23R-Protokoll 65

Tabelle 36: AT-CL-04 Verwaltung der Benachrichtigungsregeln 66

Tabelle 37: AT-CL-05 Auslösen der Abarbeitung einer Benachrichtigungsregel 66

Tabelle 38: AT-CL-06 Pflege der Unternehmensdaten 66

Tabelle 39: AT-CL-07 Bearbeitung und Freigabe einer Benachrichtigung 67

1 EINLEITUNG

1.1 ZWECK DES DOKUMENTS

Das vorliegende Dokument setzt die Vorgaben des Lastenhefts [1] im Kontext der P23R-Rahmenarchitektur [2] sowie der technischen und organisatorischen Rahmenbedingungen und der Anforderungen aus Kommunikation sowie Akzeptanz und Change Management in explizite Vorgaben für die P23R-Musterimplementierung und den Pilotversuch um.

1.2 LESERKREIS

Das Pflichtenheft richtet sich an alle Leser, die sich darüber informieren möchten, wie die im Lastenheft [1] beschriebene P23R-Lösung im Kontext der Rahmenarchitektur für die Musterimplementierung und den Pilotversuch in der Domäne Arbeitgebermeldungen (AGM) umgesetzt wird.

1.3 KONTEXT, INHALTE UND STRUKTURIERUNG

Zunächst beschreibt das Dokument hierzu den Umfang der aus der Rahmenarchitektur umzusetzenden Komponenten und Funktionen für die P23R-Musterimplementierung (siehe Kapitel 2).

Anschließend folgen Vorgaben und Angaben zum Einsatz der P23R-Musterimplementierung für den Pilotversuch in der Domäne AGM (siehe Kapitel 4). Dabei kommt insbesondere auch die Bestückung der Laborleistelle zur Sprache (siehe Kapitel 3).

Zuletzt folgen Vorgaben zur Durchführung der Prüf- und Testverfahren im Rahmen der Entwicklung und des Pilotversuchs (siehe Kapitel 5) und eine Darstellung der zum Einsatz kommenden Entwicklungs- und Laborumgebung (siehe Kapitel 6).

2 UMSETZUNG DER P23R-MUSTERIMPLEMENTIERUNG

Die Pflichten für die P23R-Musterimplementierung, die so auch im Pilotversuch zur Anwendung kommen werden, ergeben sich durch die Abbildung der im Lastenheft [1] beschriebenen Vorgaben auf die P23R-Rahmen- und Sicherheitsarchitektur [2][3].

Die P23R-Musterimplementierung orientiert sich dabei stets so nah wie möglich an der INFORMELLEN Struktur, die die P23R-Rahmenarchitektur als Umsetzungsempfehlung beschreibt, und setzt in jedem Fall deren NORMATIVEN Teile um, um einen durchgängigen Nachweis über die Anwendbarkeit der Architektur zu erhalten.

Der wesentliche Fokus bei der Umsetzung der P23R-Musterimplementierung liegt auf der P23R-Infrastruktur (siehe Abbildung 1), wobei auf die Umsetzung des durch die P23R-Rahmenarchitektur skizzierten „P23RTrustedProxy“ verzichtet wird (durch rotes Kreuz in der Abbildung hervorgehoben).

Auf die Umsetzung des P23RTrustedProxy kann verzichtet werden, da die zugrunde liegende Basistechnologie (im Wesentlichen OSCI 2.0) bereits in zahlreichen Verfahren erfolgreich im Einsatz ist.

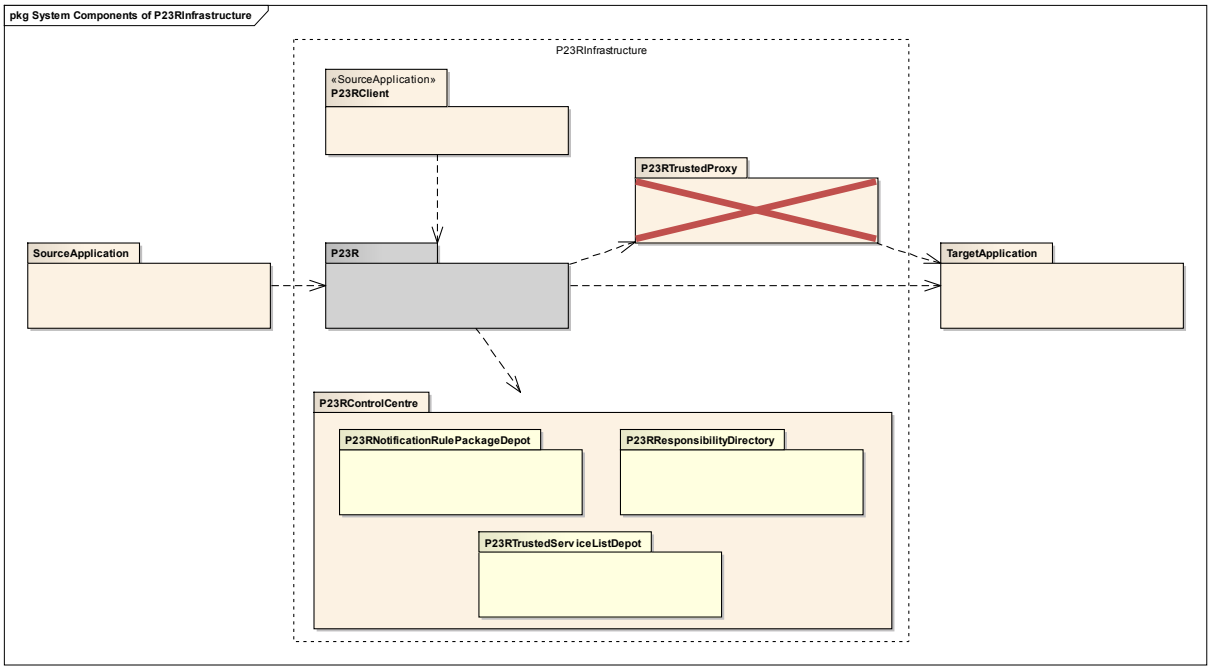


ABBILDUNG 1: BAUSTEINE DER P23R-INFRASTRUKTUR (UML) [2]

Die Umsetzung des P23R selbst wird in Abschnitt 2.1 beschrieben, die der P23R-Leitstelle in Abschnitt 2.2 und die des P23R-Client in Abschnitt 2.3.

Darüber hinaus enthält die P23R-Musterimplementierung jeweils einen exemplarischen Quelldaten- und Kommunikationskonnektor, die für die Durchführung von Testläufen und zur Erprobung gedacht sind. Diese Muster-Konnektoren sind in Abschnitt 2.4 beschrieben.

Schlussendlich wird auch eine exemplarische Umsetzung der P23R-Sicherheitsarchitektur [3] realisiert und mit der Implementierung der P23R-Infrastruktur zur P23R-Musterimplementierung integriert, was in Abschnitt 2.5 beschrieben ist.

Die einzelnen Komponenten der P23R-Musterimplementierung implementieren dabei jeweils die für sie erforderlichen (NORMATIVEN) Schnittstellen (siehe Abbildung 2) in der durch die INFORMELLE P23R-Rahmenarchitektur [2] skizzierten Funktions- und Arbeitsweise.

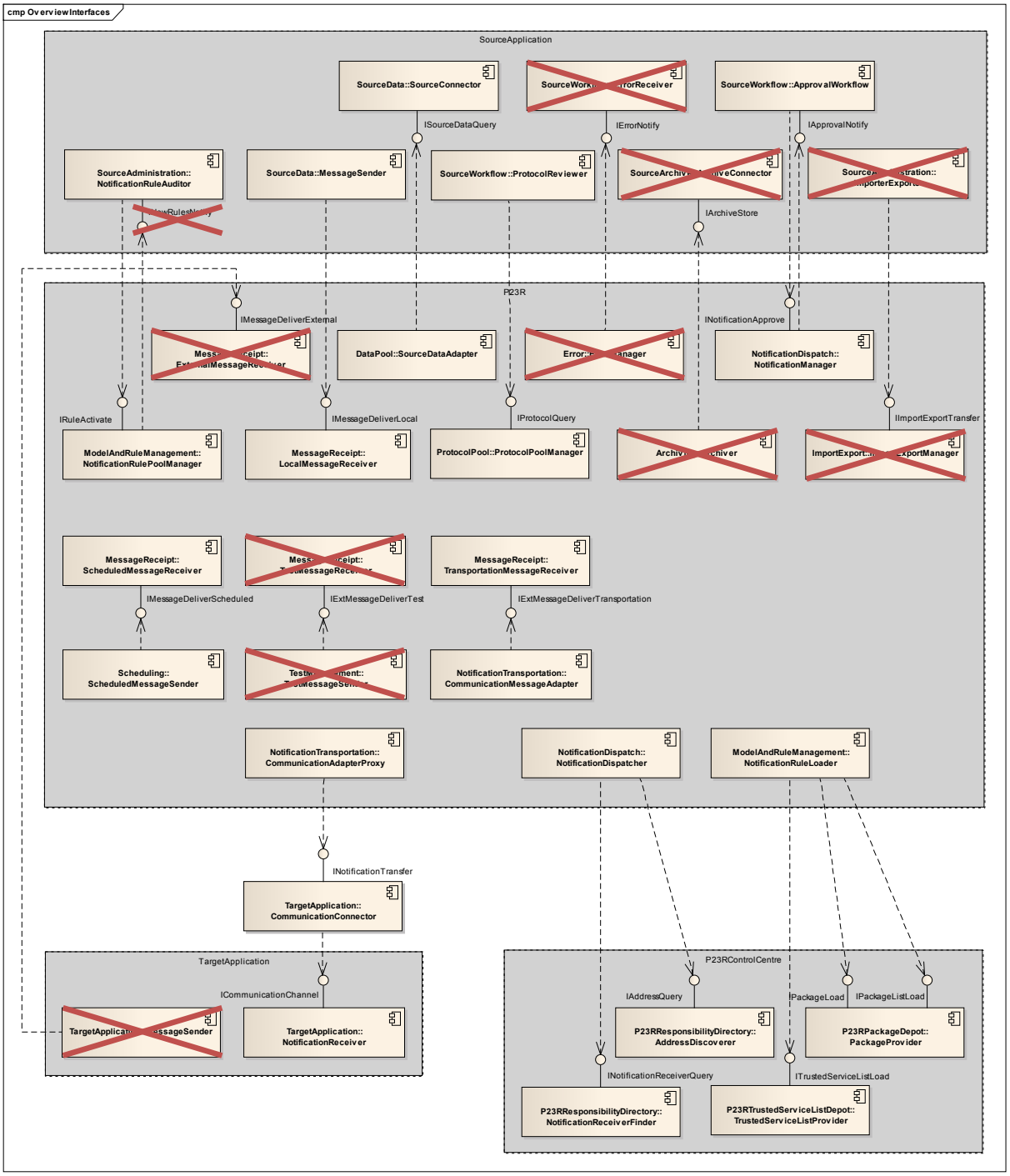


ABBILDUNG 2: ÜBERBLICK ÜBER DIE NORMATIVEN SCHNITTSTELLEN IM KONTEXT DES P23R (UML) [2]

Die Realisierung von User-Interface-Elementen (UI-Elementen), insbesondere des P23R-Clients und des Leitstellenportals, unterliegt darüber hinaus den im Lastenheft [1] beschriebenen Anwendungsfällen und Anforderungen.

Die hierdurch beschriebene P23R-Musterimplementierung wird schlussendlich im Rahmen des Pilotversuchs in der Domäne AGM zum Einsatz gebracht, wie dies in Kapitel 4 beschrieben ist, wobei eine Laborinstanz der P23R-Leitstelle angebunden wird (siehe Kapitel 3).

Die P23R-Musterimplementierung wird auf dem JBoss-Community-Framework umgesetzt. Sowohl P23R als auch der P23R-Client, die P23R-Leitstelle sowie alle Konnektoren werden darauf aufbauend entwickelt und ausgebaut (siehe Kapitel 6).

Die JBoss Community-Projekte bieten ein breites Spektrum von aufeinander abgestimmten Technologien und Frameworks. Zudem stellen sie eine hohe Kompatibilität mit der Forderung einer zukünftigen Veröffentlichung als Open-Source-Software dar.

Den Kern der Community bildet der JBoss Application Server [7]. Praktisch alle Technologien und Frameworks setzen als Laufzeitumgebung den JBoss Application Server voraus. Er bildet den Container für alle Komponenten für die umzusetzende P23R-Musterimplementierung.

Zur Umsetzung von Prinzipien einer serviceorientierten Architektur (SOA) wird an den erforderlichen Stellen der JBoss eigene Enterprise Service Bus (ESB) [8] eingesetzt. Für Web-Applikationen und -Portale kommt Seam zum Einsatz, welches eine JSR 299 kompatibles (CDI) Programmierparadigma umsetzt. Als letzte zentrale Technologie wird für alle Web Service basierten Schnittstellen die JBoss Implementierung des JAX-WS Standards JBossWS verwendet.

Die Komponenten der P23R-Sicherheitsarchitektur (siehe Abschnitt 2.5) werden auf Basis des Metro-Frameworks umgesetzt, wobei die einzelnen Services auf einem GlassFish Application Server bereitgestellt (und von den Services auf dem JBoss Application Server [7] aufgerufen) werden, der eine Kapselung der Sicherheitskomponenten zur besseren Absicherung bewirken soll.

2.1 Der Prozess-Daten-Beschleuniger (P23R)

Gemäß der P23R-Rahmenarchitektur wird die Implementierung in funktionale Komponenten und Schnittstellen nach außen gegliedert.

Im Folgenden wird ausgehend von dem bereits einleitend dargelegten Kontext festgelegt, welche Funktionen und welche Komponenten softwaretechnisch umzusetzen sind, damit alle Anforderungen an die P23R-Musterimplementierung realisiert werden können.

Die funktionalen Anforderungen der P23R-Rahmenarchitektur werden durch verschiedene Pakete gegliedert und beschrieben, wobei diese in folgende funktionale Blöcke gegliedert sind (vgl. Abbildung 3).

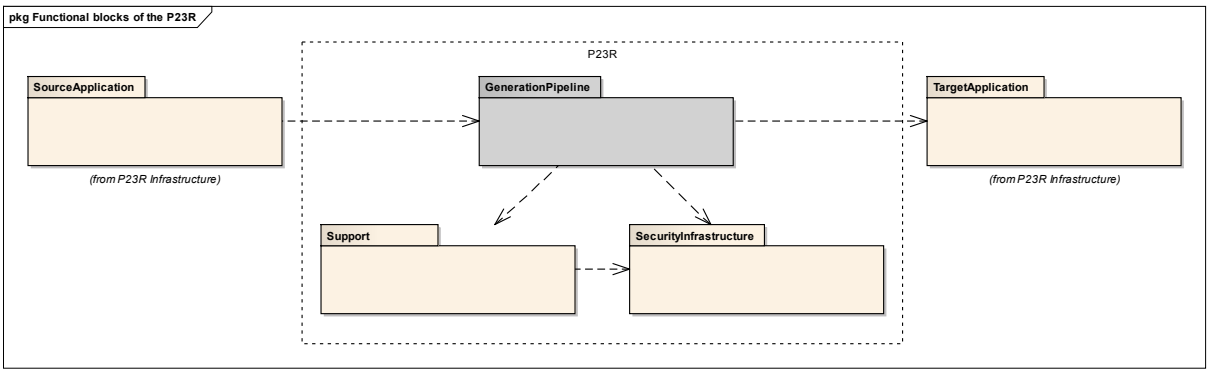


ABBILDUNG 3: FUNKTIONALE BLÖCKE DES P23R (UML) [2]

Den Kern der P23R-Musterimplementierung bildet die Abarbeitung der Generation-Pipeline (vgl. Abbildung 4 für einen Überblick).

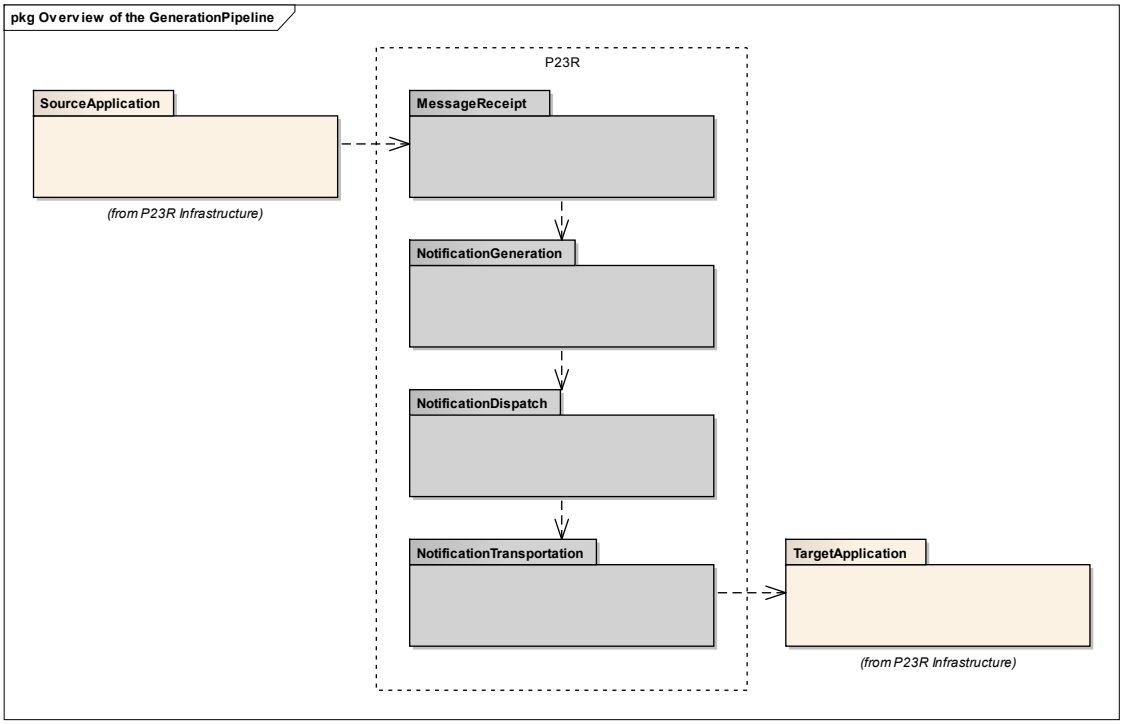


ABBILDUNG 4: ÜBERBLICK ÜBER DIE GENERATION-PIPELINE (UML) [2]

Diese nutzt eine Reihe von „Support-Packages“, die partiell für die P23R-Musterimplementierung umgesetzt werden.

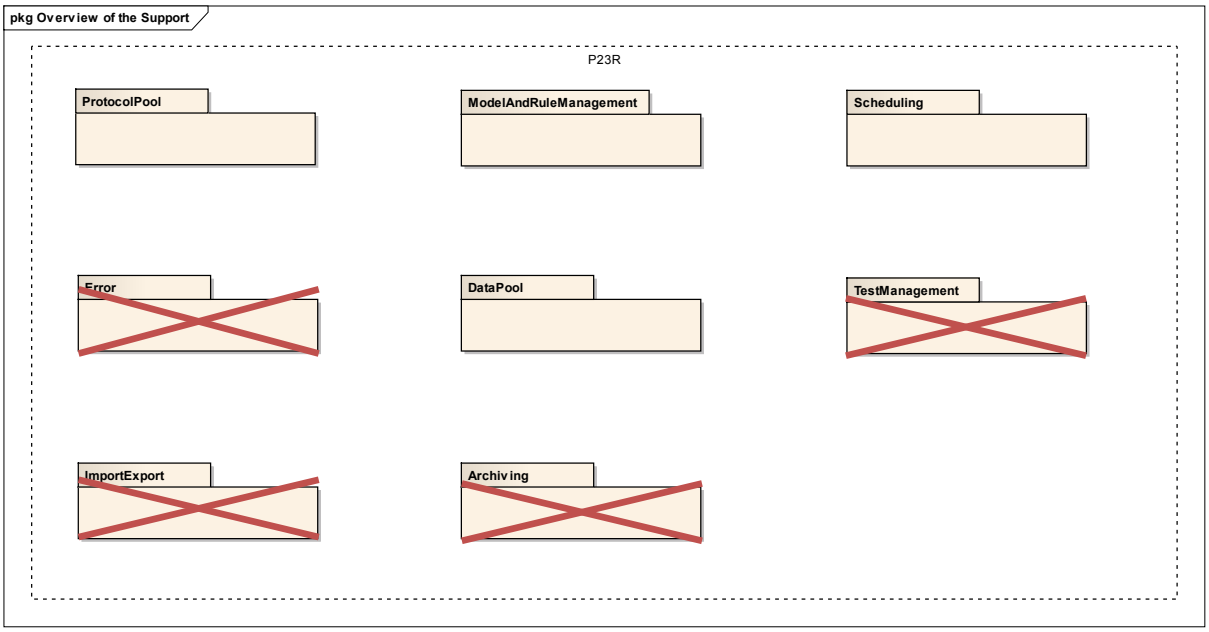


ABBILDUNG 5: ÜBERBLICK ÜBER DIE SUPPORT-PACKAGES (UML) [2]

Alle für die P23R-Musterimplementierung umzusetzenden Komponenten sind entweder Teil der Generation-Pipeline, werden von einem Teil der Generation-Pipeline zwingend benötigt oder gehören

zu einem Support-Package, das zum Nachweis der Funktion unbedingt erforderlich ist (z. B. die Protokollierung über den Protokollpool).

Folgende Komponenten werden umgesetzt und in den nachfolgenden Abschnitten in ihrer jeweiligen Fertigungstiefe weiter ausgeführt:

- Nachrichtenempfang (siehe Abschnitt 2.1.1)
- Benachrichtigungsgenerierung (siehe Abschnitt 2.1.2)
- Benachrichtigungsversand (siehe Abschnitt 2.1.3)
- Benachrichtigungstransport (siehe Abschnitt 2.1.4)

Folgende Support-Packages werden als Komponenten umgesetzt und in den nachfolgenden Kapiteln in ihrer jeweiligen Fertigungstiefe weiter ausgeführt:

- Support-Package Datenpool (siehe Abschnitt 2.1.5)
- Support-Package Protokollpool (siehe Abschnitt 2.1.6)
- Support-Package Datenmodelle und Benachrichtigungsregeln (siehe Abschnitt 2.1.7)
- Support-Package Termine und Zeitüberschreitungen (siehe Abschnitt 2.1.8)

Darüber hinaus werden das Bootstrapping zur Initialisierung des P23R umgesetzt (siehe Abschnitt 2.1.9) und die erforderlichen Komponenten der P23R-Sicherheitsarchitektur [3] realisiert (siehe Abschnitt 2.5) und in die P23R-Musterimplementierung integriert.

Zum besseren Verständnis wird im Folgenden kurz erläutert, welche Pakete nicht durch die P23R-Musterimplementierung umgesetzt werden:

- Support-Package Error (siehe Abschnitt 3.3 in [2])
- Support-Package Archivierung (siehe Kapitel 3.13 in [2])
- Support-Package Testmanagement (siehe Kapitel 3.14 in [2])
- Support-Package Import/Export (siehe Kapitel 3.15 in [2])

Alle diese Komponenten entstammen lediglich verschiedenen Support-Packages und sind für den generellen Nachweis der Machbarkeit nicht erforderlich.

2.1.1 NACHRICHTENEMPFANG

Der Nachrichtenempfang (Komponente „*MessageReceipt*“) ist Bestandteil des P23R innerhalb der P23R-Infrastruktur. Er kapselt alle Funktionalitäten, die mit dem Empfang von Nachrichten zusammenhängen (siehe P23R-Rahmenarchitektur [2] und Kapitel 2 in den Spezifikationen zur P23R-Rahmenarchitektur [4]).

Die Komponente stellt den Beginn der Generation-Pipeline dar (siehe Kapitel 3 in [2]). Sie initialisiert anhand der eingegangenen Nachricht sowie des Eingangskanals die Erzeugung einer Benachrichtigung und dadurch die Anwendung einer Benachrichtigungsregel.

Nachdem der Nachrichtenempfang über einen der Empfangskanäle eine Nachricht empfangen und validiert hat, identifiziert er als erstes die anzuwendende Benachrichtigungsregel. Mit deren Hilfe ist er nun in der Lage, das Benachrichtigungsprofil zu erzeugen und mit ersten Werten zu füllen.

Abschließend übergibt er das so generierte Benachrichtigungsprofil dem nächsten Abschnitt der Generation-Pipeline, dem Benachrichtigungsgenerator (siehe Abschnitt 2.1.2) und beendet damit seine Arbeit.

Die Umsetzung des Nachrichtenempfangs umfasst die Implementierung einer Untermenge der in der P23R-Rahmenarchitektur [2] definierten Schnittstelle und Komponenten für den Empfang von Nachrichten, wie diese in Abbildung 6 dargestellt sind.

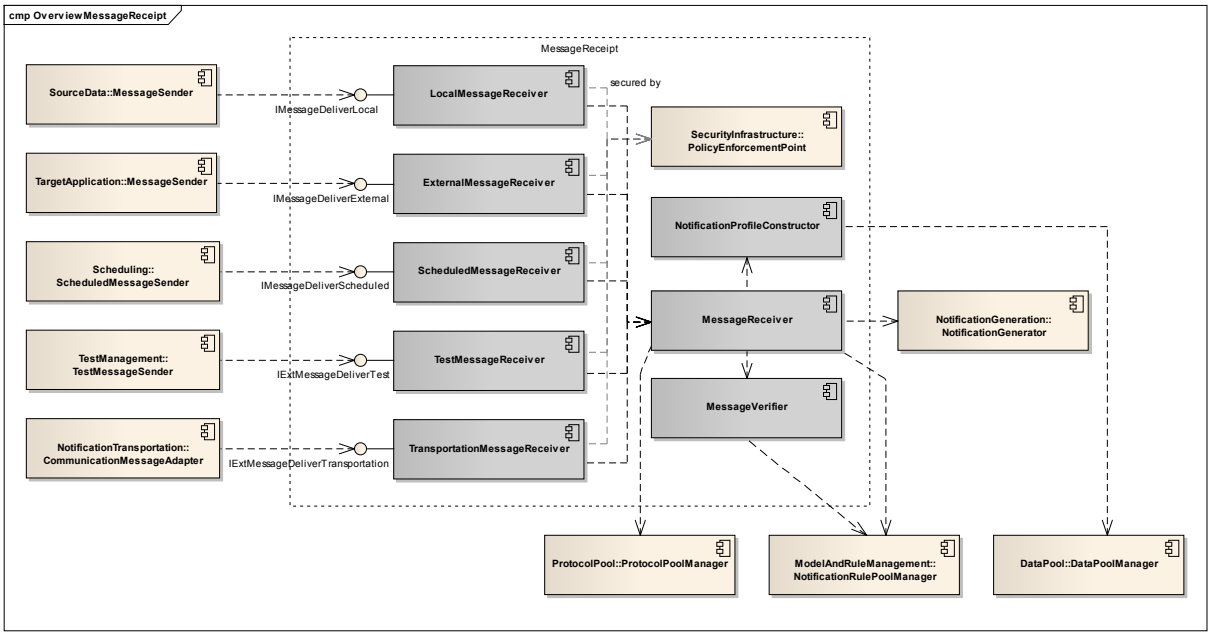


ABBILDUNG 6: KOMPONENTENÜBERSICHT FÜR DEN NACHRICHTENEMPFANG (UML) [2]

Für die P23R-Musterimplementierung werden für den Nachrichtenempfang folgende Schnittstellen und Komponenten aus der P23R-Rahmenarchitektur [2] umgesetzt:

- MessageReceiver
  - LocalMessageReceiver mit IMessageDeliverLocal
  - ScheduledMessageReceiver mit IMessageDeliverScheduled
  - TransportationMessageReceiver mit IMessageDeliverTransportation
- MessageVerifier
- NotificationProfileConstructor

Von den durch die P23R-Rahmenachritektur [2] geforderten Funktionen werden die in Tabelle 1 dargestellten Funktionen realisiert:

TABELLE 1: FUNKTIONALE ABBILDUNG NACHRICHTENEMPFANG

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Verarbeitung lokaler Nachrichten.	Die Benachrichtigungsgenerierung kann auf Grund von lokal (z. B. durch das Unternehmen) eingespielten Nachrichten gestartet werden.	Wird umgesetzt.
Verarbeitung externer Nachrichten.	Die Benachrichtigungsgenerierung kann auf Grund von extern (z. B. durch Behörde) eingespielten Nachrichten gestartet werden.	Wird **nicht** umgesetzt.
Verarbeitung zeitgesteuerter Nachrichten.	Die Benachrichtigungsgenerierung wird auf Grund einer zeitgesteuert (z. B. auf Grund von Informationen aus einer Benachrichtigungsregel) eingespielten Nachricht gestartet.	Wird umgesetzt.
Verarbeitung von Transport-Nachrichten.	Die Benachrichtigungsgenerierung wird auf Grund einer transportbedingt (z. B. für mehrstufige Freigaben) eingespielten Nachricht gestartet.	Wird umgesetzt.
Verarbeitung von Testnachrichten.	Verarbeitung von Testnachrichten in Verbindung mit dem P23R-Testmanagement auf Basis definierter Testdaten und -ergebnisse.	Wird **nicht** umgesetzt.
Initialisierung der Benachrichtigungsprofils oder der Benachrichtigungsprofile.	Initialisierung eines oder mehrerer Benachrichtigungsprofile anhand einer Selektion und eines Transformationsskriptes auf Basis des korrespondierenden Benachrichtigungsregel.	Wird umgesetzt.
Schemaprüfung der Eingangsnachricht.	Prüfung der Nachrichten gegen das zugehörige Schema aus der korrespondierenden Benachrichtigungsregel.	Wird umgesetzt.
Weitergabe in der Generation-Pipeline.	Weitergabe des bzw. der Benachrichtigungsprofile zum nächsten Abschnitt der Generation-Pipeline (der Benachrichtigungsgenerierung).	Wird umgesetzt.

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Absicherung des WebService zum Empfang lokaler Nachrichten.	Der zum Empfang von lokalen Nachrichten bereitgestellte WebService muss über einen PEP-Aufruf abgesichert werden.	Wird umgesetzt.
Protokollierung aller relevanten Ereignisse	Protokollierung aller relevanten Ereignisse auf Basis der Vorgaben der Prüf- und Testverfahren.	Wird umgesetzt.

2.1.2 BENACHRICHTIGUNGSGENERIERUNG

Die Benachrichtigungsgenerierung (Komponente „*NotificationGeneration*“) ist Bestandteil des P23R innerhalb der P23R-Infrastruktur und kapselt alle Funktionalitäten, die mit der Generierung von Benachrichtigungen zusammenhängen (siehe P23R-Rahmenarchitektur [2] und [6]).

Sie setzt die Generation-Pipeline fort und ist der direkte Schritt nach dem Nachrichtenempfang (siehe Abschnitt 2.1.1). Sie beginnt mit der Übernahme des bereitgestellten Benachrichtigungsprofils und endet mit der Weiterleitung des finalen Benachrichtigungsprofils und der generierten Benachrichtigung an den Benachrichtigungsversand (siehe Abschnitt 2.1.3).

Die Benachrichtigungsgenerierung bezieht dabei die vorselektierten Daten vom Datenpool (siehe Abschnitt 2.1.5) und alle Elemente der Benachrichtigungsregeln aus der Komponente Datenmodelle und Benachrichtigungsregeln (siehe Abschnitt 2.1.7).

Während der Transformation werden bei Bedarf zudem Informationen aus dem P23R-Zuständigkeitsverzeichnis (siehe Abschnitt 2.2.3) abgerufen, um den Empfänger zu ermitteln.

Als Ergebnis liefert die Benachrichtigungsgenerierung eine im Sinne der Generierung finale Benachrichtigung sowie das dazugehörige finale Benachrichtigungsprofil. Im Fehlerfall werden die Generierung abgebrochen und ein entsprechender Protokolleintrag gemacht.

Die Umsetzung der Benachrichtigungsgenerierung umfasst die Implementierung einer Untermenge der in der P23R-Rahmenarchitektur [2] definierten Schnittstellen und Komponenten für die Generierung von Benachrichtigungen, wie diese in Abbildung 7 dargestellt sind.

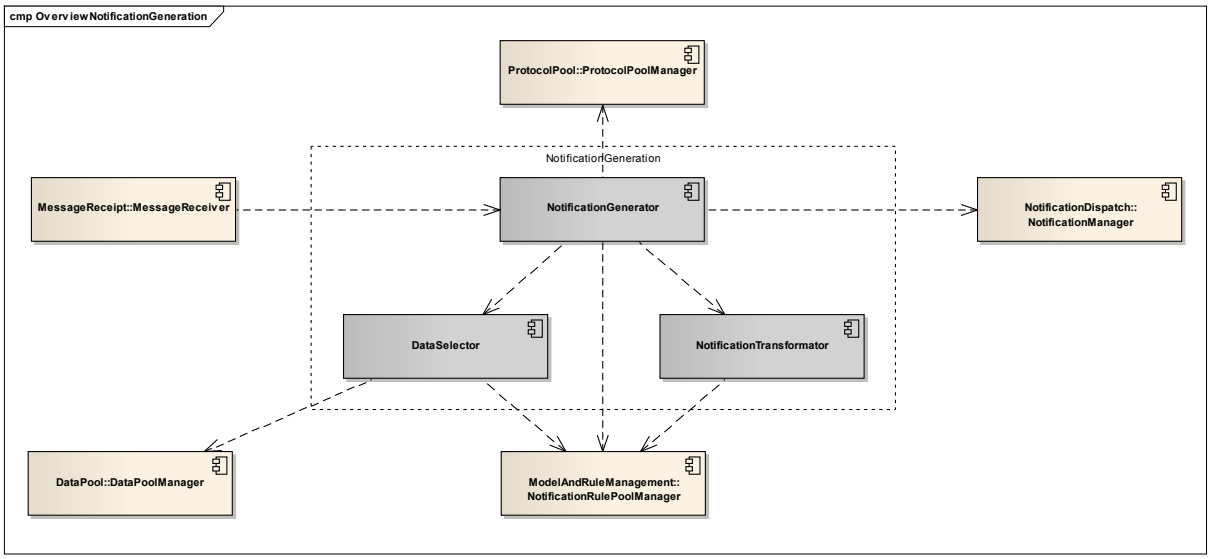


ABBILDUNG 7: KOMPONENTENÜBERSICHT FÜR DIE BENACHRICHTIGUNGSGENERIERUNG (UML) [2]

Für die P23R-Musterimplementierung werden für die Benachrichtigungsgenerierung folgende Schnittstellen und Komponenten aus der P23R-Rahmenarchitktur [2] umgesetzt:

- NotificationGenerator
- DataSelector
- NotificationTransformator

Von den durch die P23R-Rahmenachritektur [2] geforderten Funktionen werden die in Tabelle 2 dargestellten Funktionen realisiert.

TABELLE 2: FUNKTIONALE ABBILDUNG BENACHRICHTIGUNGSGENERIERUNG

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Platzierung in der Generation-Pipeline.	Erhalt des erstellten Benachrichtigungsprofils aus dem vorherigen Abschnitt der Generation-Pipeline (dem Nachrichtenempfang).	Wird umgesetzt.
Selektion der Daten für die Benachrichtigungsgenerierung.	Weitergabe des Selektionsskripts der Benachrichtigungsregel an den Datenpool und Verarbeitung der dadurch selektierten Daten.	Wird umgesetzt.
Generierung und Validierung der Benachrichtigung.	Generierung einer Benachrichtigung auf Basis der in der Benachrichtigungsregel hinterlegten Transformationsskripte mit anschließender Validierung an Hand der in der Benachrichtigungsregel hinterlegten Schemas.	Wird umgesetzt.

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Transformation und Validierung des Benachrichtigungsprofils.	Transformation des Benachrichtigungsprofils auf Basis der in der Benachrichtigungsregel hinterlegten Transformationsskripte mit anschließender Validierung an Hand der in der Benachrichtigungsregel hinterlegten Schemata.	Wird umgesetzt.
Signierung des Hauptbereichs einer Benachrichtigung.	Extraktion des Hauptbereichs aus der generierten Benachrichtigung, Signierung des selbigen und Hinterlegen der Signatur im Benachrichtigungsprofil.	Wird umgesetzt.
Abfrage von Empfängern aus dem Benachrichtigungsempfängerverzeichnis.	Abfrage des Empfängers mit den Kriterien aus dem Pivot-Datenmodell. Sofern nötig Umsetzung von internen Transformationsschritten für die Ermittlung der Werte von Kriterien innerhalb des Zuständigkeitsverzeichnisses.	Wird umgesetzt.
Weitergabe in der Generation-Pipeline.	Weitergabe des Benachrichtigungsprofils und der generierten Benachrichtigung zum nächsten Abschnitt der Generation-Pipeline (dem Benachrichtigungsversand).	Wird umgesetzt.
Protokollierung aller relevanten Ereignisse	Protokollierung aller relevanten Ereignisse auf Basis der Vorgaben der Prüf- und Testverfahren.	Wird umgesetzt.

2.1.3 BENACHRICHTIGUNGSVERSAND

Der Benachrichtigungsversand (Komponente „`NotificationDispatch`“) ist Bestandteil des P23R innerhalb der P23R-Infrastruktur. Sie kapselt alle Funktionalitäten, die mit dem Freigabeprozess im Unternehmen zusammenhängen (siehe P23R-Rahmenarchitektur [2] und Kapitel 3 in den Spezifikationen zur P23R-Rahmenarchitektur [4]).

Er setzt die Generation-Pipeline fort und ist der direkte Schritt nach der Benachrichtigungsgenerierung (siehe Abschnitt 2.1.2). Er beginnt mit der Übernahme des bereitgestellten Benachrichtigungsprofils und der bereitgestellten Benachrichtigung und endet mit der Weiterleitung der selbigen im freigegebenen Zustand an den Benachrichtigungstransport (siehe Kapitel 2.1.4).

Zur Freigabeaufforderung ruft der Benachrichtigungsversand den durch eine Quellanwendung bereitgestellte und im P23R konfigurierte Freigabe-WebService auf (z. B. den des P23R-Client, siehe Abschnitt 2.3).

Nach der Freigabe werden die aktuellen Kommunikationsparameter für den bereits ermittelten Benachrichtigungsempfänger aus dem P23R-Zuständigkeitsverzeichnis (vgl. Abschnitt 2.2.3) ermittelt.

Die Umsetzung des Benachrichtigungsversands umfasst die Implementierung einer Untermenge der in der P23R-Rahmenarchitektur [2] definierten Schnittstellen und Komponenten für den Versand von Benachrichtigungen, wie diese in Abbildung 8 dargestellt sind.

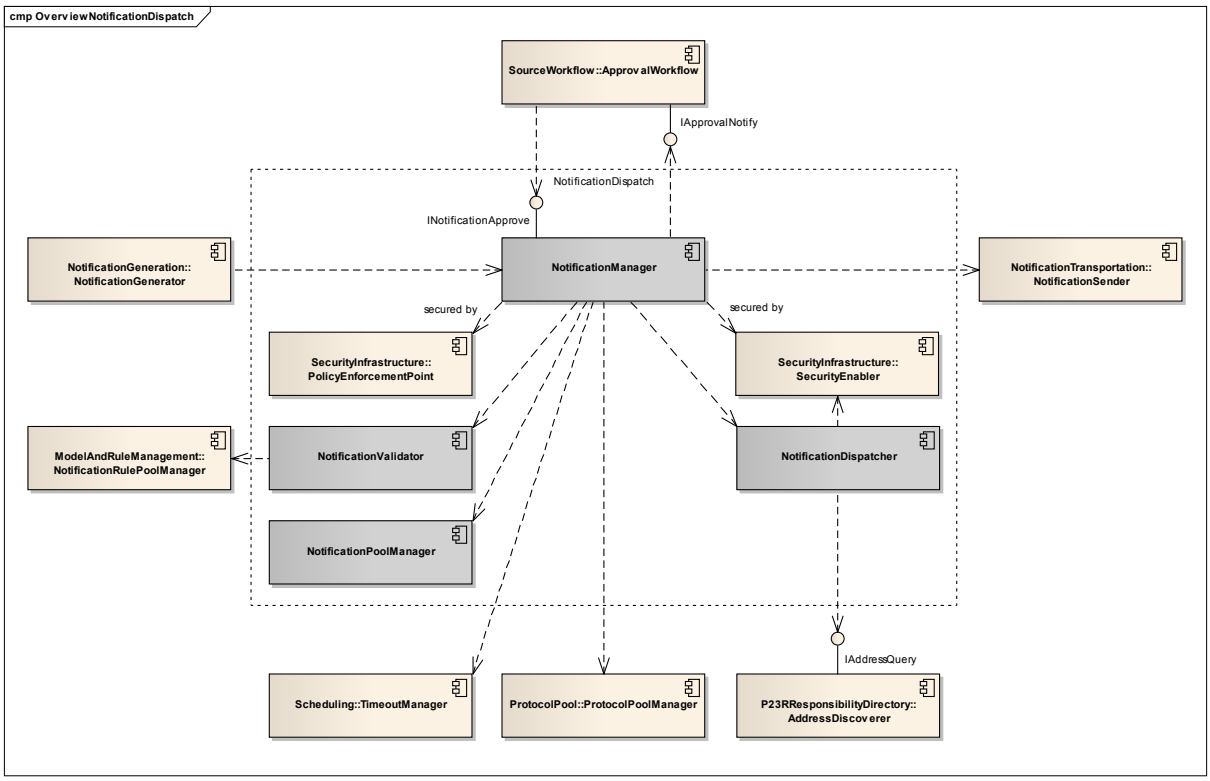


ABBILDUNG 8: KOMPONENTENÜBERSICHT FÜR DEN BENACHRICHTIGUNGSVERSAND (UML) [2]

Für die P23R-Musterimplementierung werden für den Benachrichtigungsversand folgende Schnittstellen und Komponenten aus der P23R-Rahmenarchitektur [2] umgesetzt:

- NotificationManager
- NotificationPoolManager mit NotificationApprove
- NotificationValidator
- NotificationDispatcher

Von den durch die P23R-Rahmenarchitektur [2] geforderten Funktionen werden die in Tabelle 3 dargestellten Funktionen realisiert.

TABELLE 3: FUNKTIONALE ABBILDUNG BENACHRICHTIGUNGSVERSAND

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Platzierung in der Generation-Pipeline.	Erhalt des Benachrichtigungsprofils und der generierten Benachrichtigung aus dem vorherigen Abschnitt der Generation-Pipeline (der Benachrichtigungsge-	Wird umgesetzt.

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
	nerierung).	
Freigabeprozess und Anpassung der Benachrichtigung.	Übermittlung der freizugebenden Benachrichtigung an die Quellanwendung und Verarbeitung der rückübermittelten Benachrichtigung.	Wird umgesetzt.
Validierung der freigegebenen Benachrichtigung.	Validierung der von der Quellanwendung rückübermittelten Benachrichtigung auf Basis des in der Benachrichtigungsregel hinterlegten Schemas.	Wird umgesetzt.
Verarbeitung von Freigabesignaturen.	Die durch die Quellanwendung im Rahmen der Freigabe übermittelte Freigabesignatur wird im Benachrichtigungsprofil hinterlegt und weitergereicht.	Wird umgesetzt.
Benachrichtigungspoolmanager	Vorhalten der Benachrichtigungen für die asynchrone Kommunikation, wobei die Benachrichtigung und das Benachrichtigungsprofil dauerhaft gespeichert werden.	Wird umgesetzt
Abfrage von Kommunikationsparametern über das Zuständigkeitsverzeichnis.	Abfrage der Kommunikationsparameter und Übermittlung an den Benachrichtigungstransport.	Wird umgesetzt.
Weitergabe in der Generation-Pipeline.	Weitergabe des Benachrichtigungsprofils und der freigegebenen Benachrichtigung zum nächsten Abschnitt der Generation-Pipeline (dem Benachrichtigungstransport).	Wird umgesetzt.
Absicherung des WebService zum Empfang der freigegebenen Benachrichtigung.	Der zum Empfang der freigegebenen Benachrichtigung bereitgestellte WebService muss über einen PEP-Aufruf abgesichert werden.	Wird umgesetzt.
Protokollierung aller relevanten Ereignisse	Protokollierung aller relevanten Ereignisse auf Basis der Vorgaben der Prüf- und Testverfahren.	Wird umgesetzt.

2.1.4 BENACHRICHTIGUNGSTRANSPORT

Der Benachrichtigungstransport (Komponente „*NotificationTransport*“) ist Bestandteil des P23R innerhalb der P23R-Infrastruktur und kapselt alle Funktionalitäten, die mit dem Transport von Benachrichtigungen zusammenhängen (siehe P23R-Rahmenarchitektur [2] und Kapitel 4 in den Spezifikationen zur P23R-Rahmenarchitektur [4]).

Er schließt die Generation-Pipeline ab und ist der direkte Schritt nach dem Benachrichtigungsversand (siehe Abschnitt 2.1.3). Er beginnt damit mit der Übernahme des bereitgestellten Benachrichtigungsprofils und der bereitgestellten Benachrichtigung und endet mit dem Aufruf des passenden Kommunikationskonnektors.

Als erstes ermittelt er anhand des Kommunikationskanals, welcher dem Benachrichtigungsprofil zu entnehmen ist, den zu verwendenden Kommunikationskonnektor. Daraufhin übergibt er dem Kommunikationskonnektor die Benachrichtigung und alle notwendigen Informationen zur Übertragung an die Zielanwendung, inklusive möglicherweise vorhandener Transformations- und Präsentationsskripte zur Anpassung an das gewünschte Zielformat.

Die Umsetzung des Benachrichtigungstransport umfasst die Implementierung einer Untermenge der in der P23R-Rahmenarchitektur [2] definierten Schnittstellen und Komponenten für den Transport von Benachrichtigungen, wie diese in Abbildung 9 dargestellt sind.

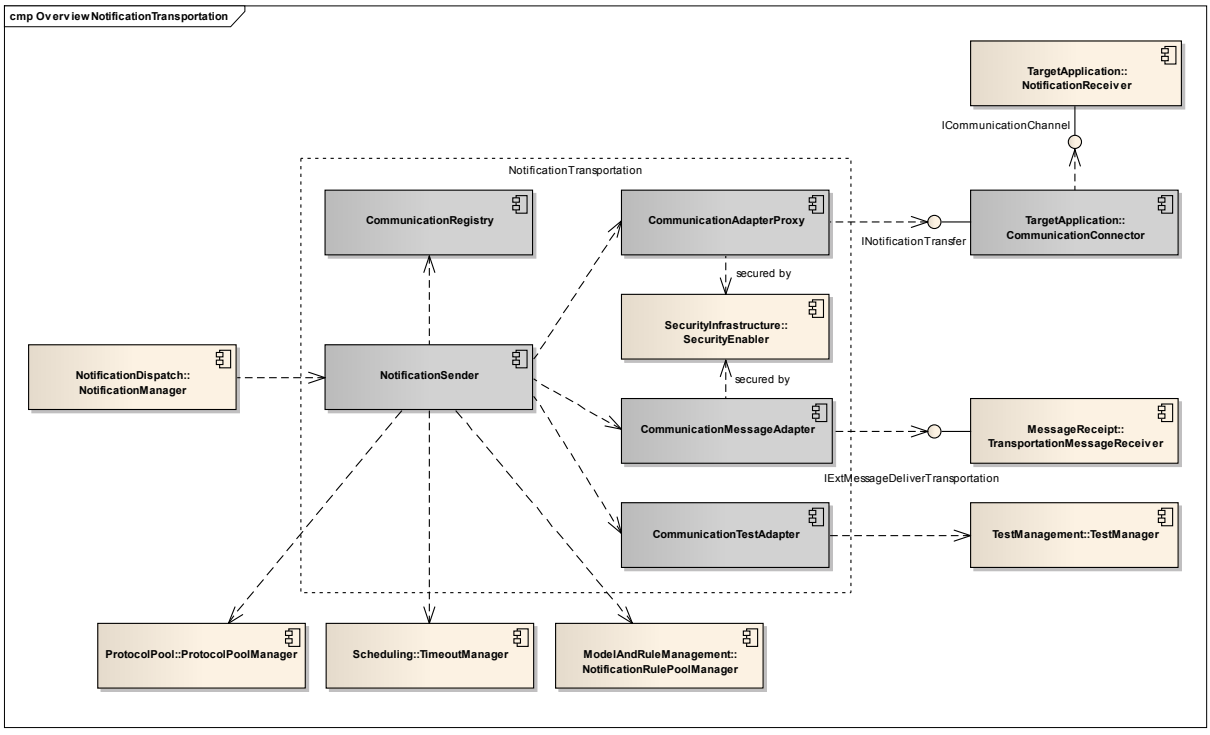


ABBILDUNG 9: KOMPONENTEN FÜR DEN BENACHRICHTIGUNGSTRANSPORT (UML) [2]

Für die P23R-Musterimplementierung werden für den Benachrichtigungstransport folgende Schnittstellen und Komponenten aus der P23R-Rahmenarchitektur [2] umgesetzt:

- NotificationSender
- CommunicationAdapterRegistry
- CommunicationAdapaterProxy

Von den durch die P23R-Rahmenarchitektur [2] geforderten Funktionen werden die in Tabelle 4 dargestellten Funktionen realisiert.

TABELLE 4: FUNKTIONALE ABBILDUNG BENACHRICHTIGUNGSTRANSPORT

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Platzierung in der Generation-Pipeline.	Erhalt des Benachrichtigungsprofils und der freigegebenen Benachrichtigung aus dem vorherigen Abschnitt der Generation-Pipeline (dem Benachrichtigungsversand).	Wird umgesetzt.
Registrierung von Kommunikationskonnektoren	Die zur Verfügung stehenden Kommunikationskonnektoren sind den Übertragungskanälen zugeordnet und werden danach selektiert.	Wird umgesetzt.
Versand über mehrere Kommunikationskanäle	Versand der freigegebenen Benachrichtigung über mehrere Kommunikationskanäle unter Berücksichtigung der Priorität, sowie Nutzung alternativer Kanäle, wenn ein Kanal nicht erreichbar ist.	Wird umgesetzt.
Versand an mehrere Empfänger.	Versand der freigegebenen Benachrichtigung an mehrere Empfänger. In dem Fall ist der Transport erst abgeschlossen, wenn alle Empfänger erreicht wurden.	Wird umgesetzt.
Weiterleitung als neue Transportnachricht.	Die freigegebene Benachrichtigung wird als Transportnachricht an den Nachrichteneingang weitergereicht.	Wird umgesetzt.
Übermittlung von Testbenachrichtigungen.	Die Übermittlung von Testbenachrichtigungen über eine spezielle Ausprägung eines Kommunikationskonnektors, welcher den Zustellauftrag nicht an einen Benachrichtigungsempfänger weiterleitet, sondern im Testfall die übergebenen Zustellinformationen protokolliert.	Wird **nicht** umgesetzt.
Übergabe an den Kommunikationskonnektor.	Abschluss der Generation-Pipeline durch Übergabe der Benachrichtigung und aller relevanten Parameter an den anhand des Kommunikationskanals ermittelten Kommunikationskonnektor.	Wird umgesetzt.

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Protokollierung aller relevanten Ereignisse	Protokollierung aller relevanten Ereignisse auf Basis der Vorgaben der Prüf- und Testverfahren.	Wird umgesetzt.

2.1.5 DATENPOOL

Der Datenpool (Komponente „*DataPool*“) ist Bestandteil der Support-Packages der P23R-Infrastruktur und kapselt alle Funktionalitäten, die zur Bereitstellung der erforderlichen Daten im P23R erforderlich sind (siehe P23R-Rahmenarchitektur [2] und Kapitel 5 in den Spezifikationen zur P23R-Rahmenarchitektur [4]).

Die Daten werden anhand einer Selektion ausgewählt, die durch ein Selektionsskript aus der Benachrichtigungsregel gegeben ist. Die Selektion wird auf Basis eines oder mehrerer Teildatenmodelle des Pivot-Datenmodells beschrieben, welches durch ein Schema im Datenmodellpaket und den dazugehörigen Namespace des Schemas definiert ist.

Die Namespaces dienen gleichzeitig zur Ermittlung der zu verwendenden Quelldatenkonnektoren, die die eigentlichen selektierten Daten in Form eines XML-Dokuments oder -Fragments liefern.

Zusätzlich kann der Datenpool eine vorgesehene Selektion als Konfigurierung abspeichern, so dass spätere Selektionen darauf zurückgreifen können. Die dafür vorgesehenen Selektionsskripte werden analog zur normalen Selektion ausgeführt.

Die Umsetzung des Datenpools umfasst die Implementierung einer Untermenge der in der P23R-Rahmenarchitektur [2] definierten Schnittstellen und Komponenten für die Bereitstellung von Daten, wie diese in Abbildung 10 dargestellt sind.

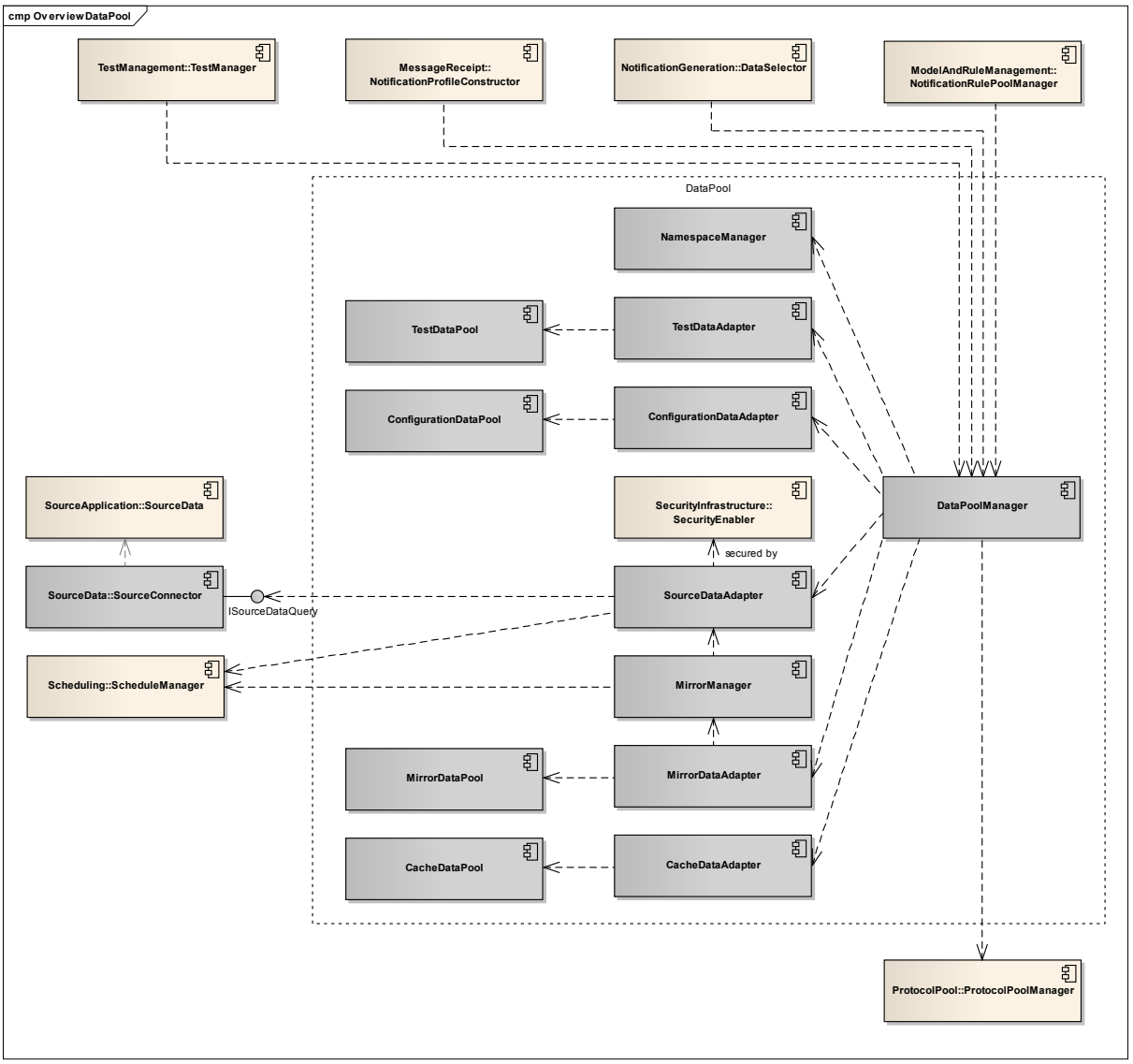


ABBILDUNG 10: KOMPONENTENÜBERSICHT FÜR DEN DATENPOOL (UML) [2]

Für die P23R-Musterimplementierung werden für den Datenpool folgende Schnittstellen und Komponenten aus der P23R-Rahmenarchitektur [2] umgesetzt:

- DataPoolManager
- ConfigurationDataAdapter
  - ConfigurationDataPool
- SourceDataAdapter
- NamespaceManager

Von den durch die P23R-Rahmenarchitektur [2] geforderten Funktionen werden die in Tabelle 5 dargestellten Funktionen realisiert.

TABELLE 5: FUNKTIONALE ABBILDUNG DATENPOOL

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Verarbeiten von Quelldaten.	Selektion von Daten über den korrekten Quelldatenkonnektor in Abhängigkeit des betreffenden Namespaces mit Hilfe des in der Benachrichtigungsregel hinterlegten Selektionsskripts.	Wird umgesetzt.
Konfiguration der Namespacezuordnung.	Konfiguration der Zuordnung von Namespaces zu Quelldatenkonnektor zur Verwaltung der eingesetzten Quelldatenkonnektoren. Diese Konfiguration wird durch den Datenpool verarbeitet.	Wird umgesetzt.
Verarbeiten von Konfigurationsdaten.	Ablegen und Selektieren von Konfigurationsdaten zu einer Benachrichtigungsregel anhand eines Namespaces mit Hilfe der in der Benachrichtigungsregel hinterlegten Selektionsinformationen.	Wird umgesetzt.
Verarbeiten von gespiegelten Quelldaten.	Abruf der kompletten Daten aus den SourceDataAdaptern, Selektion der Daten aus dem Spiegel-Datenpool für Benachrichtigungen. Nach Ablauf eines des Gültigkeitsdatums werden die Daten erneut abgerufen.	Wird **nicht** umgesetzt.
Verarbeiten von zwischengespeicherten Quelldaten.	Ergebnisse aus vorherigen Abfragen werden zwischengespeichert und für neue Datenabfragen vorrangig verwendet.	Wird **nicht** umgesetzt.
Verarbeiten von spezifischen Testdaten.	Abruf von Testdaten, die im Test-Datenpool vorgehalten werden.	Wird **nicht** umgesetzt.
Protokollierung aller relevanten Ereignisse	Protokollierung aller relevanten Ereignisse auf Basis der Vorgaben der Prüf- und Testverfahren.	Wird umgesetzt.

2.1.6 PROTOKOLLPOOL

Der Protokollpool (Komponente „`ProtocolPool`“) ist Bestandteil der Support-Packages der P23R-Infrastruktur und kapselt alle Funktionalitäten, die die zur Protokollierung im P23R erforderlich sind (siehe P23R-Rahmenarchitektur [2] und Kapitel 6 in den Spezifikationen zur P23R-Rahmenarchitektur [4]).

Über den ProtocolPoolManager können diverse Komponenten des P23R zur Nachvollziehbarkeit der Prozesse Protokollinformationen speichern. Die Komponente ProtocolPoolManager stellt außerdem eine Webservice Schnittstelle bereit, über die externe Systeme Protokolleinträge abrufen können.

Die Umsetzung des Protokollpool umfasst die Implementierung einer Untermenge der in der P23R-Rahmenarchitektur [2] definierten Schnittstellen und Komponenten für die Verarbeitung von Protokolldaten, wie diese in Abbildung 11 dargestellt sind.

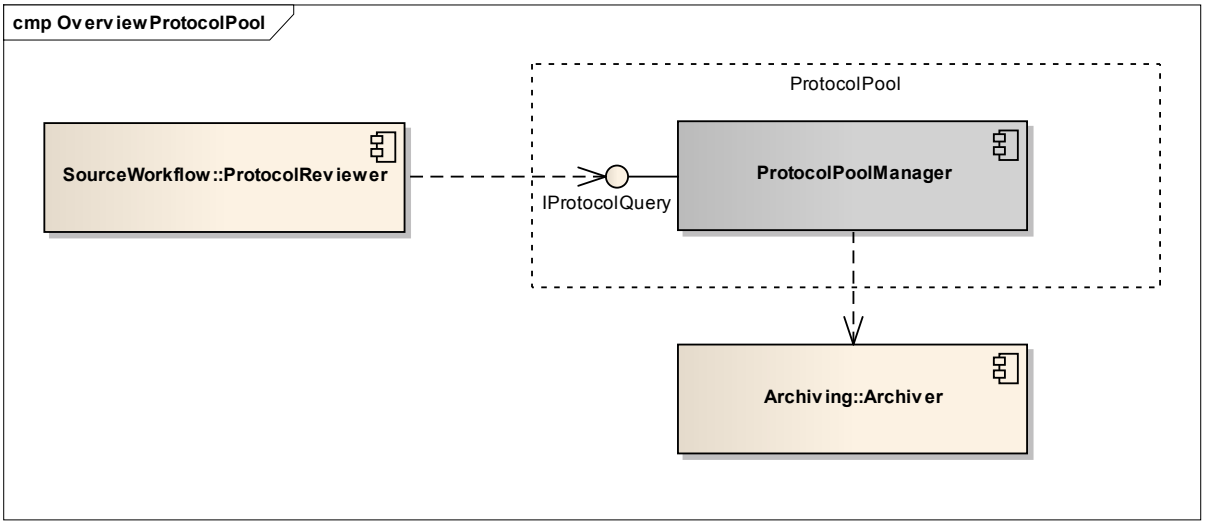


ABBILDUNG 11: KOMPONENTEN FÜR DEN PROTOKOLLPOOL (UML) [2]

Für die P23R-Musterimplementierung werden für den Protokollpool folgende Schnittstellen und Komponenten aus der P23R-Rahmenarchitektur [2] umgesetzt:

- ProtocolPoolManager mit IProtocolQuery

Von den durch die P23R-Rahmenarchitektur [2] geforderten Funktionen werden die in Tabelle 6 dargestellten Funktionen realisiert:

TABELLE 6: FUNKTIONALE ABBILDUNG PROTOKOLLPOOL

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Lesezugriff auf die Protokolldaten.	Ein Lesezugriff über die Schnittstelle IProtocolQuery.	Wird umgesetzt.
Aufbewahrungsfrist bzw. Vorhaltungsfrist	Lebensdauer eines Protokolleintrags, steuert im Zusammenhang mit der Relevanz, wann ein Protokolleintrag dieses Protokollnachrichtentyps archiviert bzw. gelöscht werden kann.	Wird **nicht** umgesetzt
Zusammenhalt, Bündelung	Bündelung einzelner Protokolleinträge zu einem Gesamtvorgang an Hand von TransactionId, NoificationId, oder MessageId.	Wird umgesetzt.

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Archivspeicher	Die Komponente Archivspeicher organisiert das revisionssichere Vorhalten von archivierten Protokollinformationen.	Wird **nicht** umgesetzt

2.1.7 DATENMODELLE UND BENACHRICHTIGUNGSREGELN

Das Management von Datenmodellen und Benachrichtigungsregeln (Komponente „*ModelAndRuleManagement*“) ist Bestandteil der Support-Packages der P23R-Infrastruktur und kapselt alle Funktionalitäten, die zur Verwaltung und Bereitstellung von Benachrichtigungsregeln und Benachrichtigungsregelpaketen im P23R erforderlich sind (siehe P23R-Rahmenarchitektur [2] und Kapitel 7 in den Spezifikationen zur P23R-Rahmenarchitektur [4]).

Es nutzt dazu, die von der P23R-Leitstelle bereitgestellten Schnittstellen des Benachrichtigungsregelpaket-Depots (siehe Abschnitt 2.2.1) um die Liste aktuell verfügbarer Benachrichtigungsregelpakete und Datenmodellpakete bzw. die einzelnen Benachrichtigungsregelpakete und Datenmodellpakete selbst herunterzuladen.

Die Komponente steuert das Herunterladen und die Aktualisierung von Benachrichtigungsregelpaketen Datenmodellpaketen von der P23R-Leitstelle sowie deren Aktivierung innerhalb des P23R. Für die Quellanwendung stellt sie zudem verschiedene Schnittstellen für die Ansicht und Aktualisierung sowie für die Aktivierung und Deaktivierung von Benachrichtigungsregelpaketen bereit.

Die Umsetzung des Managements von Datenmodellen und Benachrichtigungsregeln umfasst die Implementierung einer Untermenge der in der P23R-Rahmenarchitektur [2] definierten Schnittstellen und Komponenten für die Verarbeitung von Datenmodellen und Benachrichtigungsregeln, wie diese in Abbildung 12 dargestellt sind.

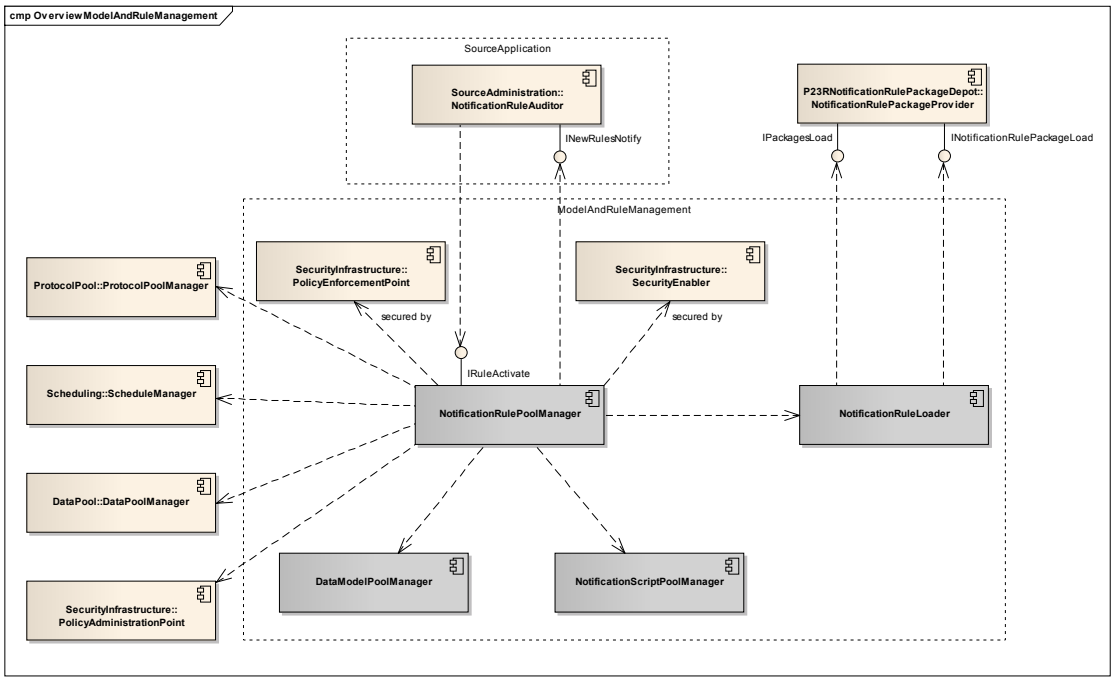


ABBILDUNG 12: KOMPONENTENÜBERSICHT FÜR DATENMODELLE / BENACHRICHTIGUNGSREGELN (UML) [2]

Für die P23R-Musterimplementierung werden für das Management von Datenmodellen und Benachrichtigungsregeln folgende Schnittstellen und Komponenten aus der P23R-Rahmenarchitektur [2] umgesetzt:

- NotificationRulePoolManager

  Der *NotificationRulePoolManager* wird als zentrale Verwaltungsinstanz für Informationen bzw. Funktionalitäten hinsichtlich von Benachrichtigungsregeln (und -Paketen, -Gruppen) innerhalb des P23R implementiert. Er verwaltet alle in diesem Zusammenhang stehenden Daten und Metainformationen und sorgt für deren Aktualität über das Zusammenspiel mit dem *NotificationRuleLoader*, welcher den Abgleich mit der Leitstelle realisiert.
  Der *NotificationRulePoolManager* bietet eine Schnittstelle (siehe unten) für den P23R-Client an, um die Verwaltung und Aktivierung der Benachrichtigungsregeln durch den fachlichen Administrator zu unterstützen (in der Abbildung *IRuleActivate***)** .

  Über den *ScheduleManager* werden gemäß den Definitionen der Benachrichtigungsregeln die Termine eingerichtet, zu denen die Benachrichtigungsprozesse (Generation-Pipeline) gestartet werden müssen. Gültigkeitszeiträume für Benachrichtigungsregeln hingegen werden in der P23R-Musterimplementierung nicht unterstützt.

  Der *NotificationRulePoolManager* sorgt außerdem für die Berechnung von Konfigurationsdaten für aktive Benachrichtigungsregelgruppen bzw. -regeln und deren Weiterverarbeitung bzw. Speicherung durch den *DataPoolManager*.

- NotificationRuleLoader

  Der *NotificationRuleLoader* ist für die Kommunikation mit dem P23R-Depot bei der Leitstelle (*NotificationRulePackageProvider*) zuständig und lädt die Benachrichtigungsregelpaket-Liste bzw. dedizierte Benachrichtigungsregelpakete über die von der Leitstelle definierten Schnittstellen herunter. Diese werden durch den *NotificationRulePoolManager* persistiert.

Die in der Abbildung weiterhin aufgeführten Bestandteile *DataModelPoolManager* und *NotificationScriptManager* werden in der P23R-Musterimplementierung nicht dediziert umgesetzt. Die in der Rahmenarchitektur beschriebenen Funktionalitäten werden – sofern für den Nachweis der Anwendbarkeit erforderlich – durch den *NotificationRulePoolManager* direkt abgebildet.

Über folgende Schnittstelle erfolgt die Kommunikation mit internen Systemen:

- IRuleActivate

  Die Schnittstelle stellt u. a. alle notwendigen Statusinformationen u. a. alle notwendigen Statusinformationen über die aktuellen aktiven sowie neuen, aber noch nicht aktiven Benachrichtigungsregelgruppen und Benachrichtigungsregeln, deren Releases und Änderungsinformationen, für die Aktivierung / Deaktivierung etc. bereit. Darüber hinaus nimmt die Schnittstelle geänderte Statusinformationen entgegen. Der Service führt gemäß dieser Änderungen (Herunterladen Paket, Aktivierung oder Deaktivierung Gruppe etc.) die entsprechend notwe-

nigen Operationen inklusive aller zugehörigen Initialisierungs- und Konfigurationsprozesse aus.

Über folgende Schnittstellen erfolgt eine Kommunikation mit externen Systemen (in Klammern):

- INotificationReceiverQuery (P23R-Leitstelle, P23R-Zuständigkeitsverzeichnis)

  Erfolgt ggf. bei Aktivierung einer Benachrichtigungsregel bei der Konfigurationsberechnung (Transformation) implizit über eine in der T-BRS [5] bereitgestellte Funktion.

- IPackagesLoad, INotificationRulePackageLoad (P23R Leitstelle)

  (HTTP(S)-Schnittstellen Benachrichtigungsregel-Depot)
  Über diese Schnittstellen lädt die Komponente die Liste verfügbarere Benachrichtigungsregelpakete bzw. die Benachrichtigungsregelpakete selbst.

- ITrustedServiceListLoad (P23R Leitstelle)

  (HTTP(S)-Schnittstellen Trusted-Service-List-Depot)
  Über diese Schnittstellen werden die verfügbaren Dienst-Endpunkte und deren Zertifikate geladen.

Von den durch die P23R-Rahmenarchitektur [2] geforderten Funktionen werden die in Tabelle 7 dargestellten Funktionen realisiert:

TABELLE 7: FUNKTIONALE ABBILDUNG DATENMODELLE UND BENACHRICHTIGUNGSREGELN

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Verarbeitung der Regelpaketliste der P23R-Leitstelle.	Package.lst enthält eine Liste der Regelpakete (oder –gruppen) mit den Abhängigkeiten zum Unternehmensdatenbestand.	Wird umgesetzt.
Regelpaket- und Benachrichtigungsregelverwaltung.	Verwaltung, sprich Aktivierung, Deaktivierung, Löschung und Aktualisierung von Benachrichtigungsregelpaketen bzw. Benachrichtigungsregeln.	Wird umgesetzt: • Aktivierung, Deaktivieren von Paketen, Gruppen und Regeln • Löschung, wenn Regelpaket oder Gruppe nicht mehr vorhanden ist • Aktualisierung, wenn neue Version vorhanden ist
Zugriffsbereitstellung auf die Artefakte einer Benachrichtigungsregel.	Die Komponente stellt Zugriff auf interne Strukturen eines Benachrichtigungsregelpakets (bzw. von Benachrichtigungsregelgruppen, Benachrichtigungsregeln) zur Verfügung.	Wird umgesetzt: • Die von der T-BRS [5] vorgeschriebenen DataSel-Skripte werden intern als XQuery-Skripte verwaltet.
Manifest des Regelpakets	Beschreibung der grundlegenden Paketinformationen.	Wird umgesetzt.
Recommendations des Regelpaketes	Prüfung der Anforderungen für ein Re-	Wird umgesetzt.

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
	gelpaket.	
Manifest der Regelgruppe	Beschreibung der grundlegenden Regelgruppeninformationen.	Wird umgesetzt.
Recommendations der Regelgruppe	Prüfung der Anforderungen für eine Regelgruppe.	Wird umgesetzt.
Ausführung von Konfigurationsskripten.	Konfigurationsermittlung für die Benachrichtigungsregelgruppe und Benachrichtigungsregel.	Wird umgesetzt.
Bestückung des Schedulers.	Ermittlung der geplanten Ausführungszeiten für jede Benachrichtigungsregel und Übergabe an den Scheduler.	Wird umgesetzt.
Abhängigkeiten und Konflikte.	Unterstützung von Abhängigkeiten und Konflikten (z. B. parallele Aktivierung von sich ausschließenden Benachrichtigungsregeln) zwischen Benachrichtigungsregelpaketen und Benachrichtigungsregelgruppen.	Wird **nicht** umgesetzt.
Gültigkeitszeiträume.	Unterstützung von Gültigkeitszeiträumen für Benachrichtigungsregelgruppen, Regeln und Konfigurationen.	Wird **nicht** umgesetzt.
Benachrichtigungsregelvarianten.	Unterstützung für Benachrichtigungsregelvarianten einzelner Benachrichtigungsregeln.	Wird **nicht** umgesetzt.
Signaturprüfung.	Prüfung der Signatur die Seitens der P23R-Leitstelle an ein Benachrichtigungsregelpaket angebracht werden kann.	Wird umgesetzt.
Protokollierung aller relevanten Ereignisse.	Protokollierung aller relevanten Ereignisse auf Basis der Vorgaben der Prüf- und Testverfahren.	Wird umgesetzt.

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Verarbeitung von benutzerdefinierten Funktionen.	Die T-BRS sieht es für eine Reihe von Aktivitäten vor benutzerdefinierte Funktionen aus den Skripten heraus aufzurufen um die Verarbeitung einer Benachrichtigungsregel zu steuern bzw. deren Abarbeitung zu ermöglichen.	Umsetzung von benutzerdefinierten Funktionen für: • Abruf von Quelldaten aus einem Quelldatenkonnektor. • Abruf von Informationen aus dem Zuständigkeitsverzeichnis. • Erstellen eines Protokolleintrags im Protokollpool.
Import bzw. Referenzieren von gemeinsamen Schemata und Funktionen.	Die T-BRS bietet die Möglichkeit in mehreren Regeln auf dieselben Schemata bzw. Funktionsdefinitionen zurückzugreifen und hierzu relative Namespace-Pfade und Import-Statements einzusetzen.	Wird **nicht** umgesetzt.

Hinweis: Eine Umsetzung der Funktionalitäten „Abhängigkeiten und Konflikte“, „Gültigkeitszeiträume“, „Benachrichtigungsregelvarianten“ und „Import von Schemata und Funktionen“ stellt den Nachweis der Anwendbarkeit des P23R-Prinzips hinsichtlich der technischen Machbarkeit nicht in Frage, da diese Fragestellungen bereits hinlänglich im Umfeld des Software-Engineering bearbeitet und beantworten wurden.

So stellt z. B. der Import lediglich eine durch das P23R-System zu unterstützende Funktionalität des XSTL-Frameworks da, welche bereits in eine Reihe von Projekten praktisch erprobt wurde. Ebenso verhält es sich mit Regelvarianten, Gültigkeiten und Versionskonflikten.

2.1.8 TERMINE UND ZEITÜBERSCHREITUNGEN

Das Management von Terminen und Zeitüberschreitungen (Komponente „`Scheduling`“) ist Bestandteil der Support-Packages der P23R-Infrastruktur und kapselt alle Funktionalitäten, die von anderen P23R-Komponenten zur Zeitsteuerung verwendet werden können (siehe P23R-Rahmenarchitektur [2]).

Die Umsetzung des Managements von Terminen und Zeitüberschreitungen umfasst die Implementierung einer Untermenge der in der P23R-Rahmenarchitektur [2] definierten Schnittstellen und Komponenten für die Verarbeitung von Terminen und Zeitüberschreitungen, wie diese in Abbildung 13 dargestellt sind.

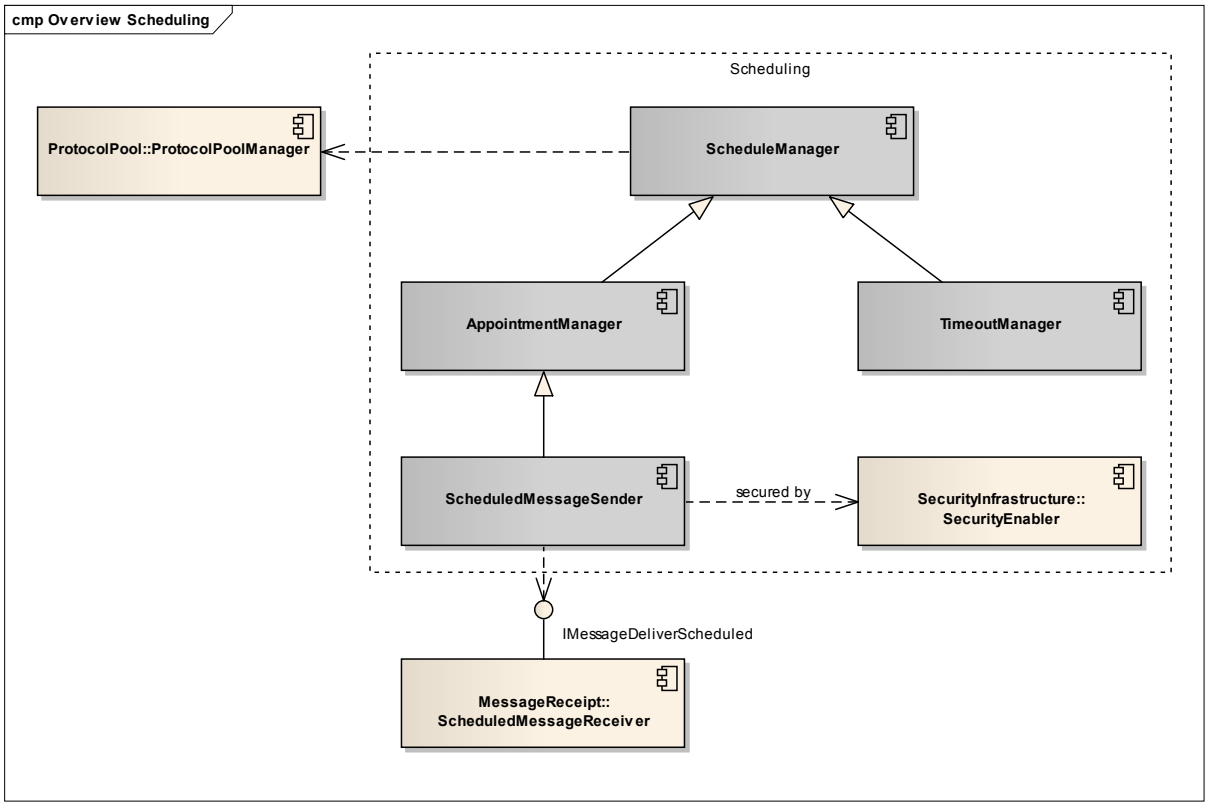


ABBILDUNG 13: KOMPONENTENÜBERSICHT FÜR TERMINE UND ZEITÜBERSCHREITUNGEN (UML) [2]

Für die P23R-Musterimplementierung werden für das Management von Terminen und Zeitüberschreitungen folgende Schnittstellen und Komponenten aus der P23R-Rahmenarchitektur [2] umgesetzt:

- ScheduleManager – Interne Schnittstelle zu den anderen Paketen und Verarbeitung der eingehenden Anfragen.
- TimeoutManager – Verwaltung von Timeouts. Timeouts sind Ereignisse, die für den Initiator als Call-Back oder als Ereignis ausgelöst werden können.
- ScheduledMessageSender – Der ScheduledMessageSender stößt zu gegebenen Zeitpunkten die Generierung einer Benachrichtigung an. Er wird vom `ModelAndRuleManagement` bei der Aktivierung einer Regel initialisiert. Er ist in der Lage eine vorgegebene Nachricht an den `MessageReceipt` über den ScheduledMessageReceiver-Empfangskanal abzusetzen.

Von den durch die P23R-Rahmenarchitektur [2] geforderten Funktionen werden die in Tabelle 8 dargestellten Funktionen realisiert:

TABELLE 8: FUNKTIONALE ABBILDUNG TERMINE UND ZEITÜBERSCHREITUNGEN

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Parametrisierbarkeit und Verwaltbarkeit.	Die Komponente muss mit zeitlichen Vorgaben parametrisiert werden können. Einträge müssen zudem löschbar, bzw. überschreibbar sein.	Wird umgesetzt.
Regelmäßiger Start von Benachrichtigungsläufen.	Gemäß Daten innerhalb der Regelpakete erfolgt ein zeitgesteuerter Start über das Erzeugen einer entsprechenden lokalen Nachricht, die mit der Identität des Benutzers versehen wird, der die entsprechende Benachrichtigungsregel aktiviert hatte.	Wird umgesetzt.
Aktualisierung von Daten im Datenpool.	Aktualisierung von Daten im Datenpool bei Ablauf des Gültigkeitsdatums.	Wird **nicht** umgesetzt.
Zeitüberwachung bei der Freigabe	Aufzeichnen der Terminüberschreitung bei der Freigabe von Benachrichtigungen am Quellsystem.	Wird umgesetzt.
Überwachung der Rechtzeitigkeit.	Überwachung der Rechtzeitigen Übertragung der Benachrichtigung.	Wird **nicht** umgesetzt.
Aktualisierung von Benachrichtigungsregelpaketen	Nutzung von Terminen zur regelmäßigen Aktualisierung	Wird umgesetzt.
Überwachung von Zeitüberschreitungen nach außen.	Zeitüberschreitungen für Kommunikation nach außen über Webservices. Dies wird über das Webservice Protokoll direkt geregelt.	Wird **nicht** umgesetzt.
Protokollierung aller relevanten Ereignisse	Protokollierung aller relevanten Ereignisse auf Basis der Vorgaben der Prüf- und Testverfahren.	Wird umgesetzt.

2.1.9 BOOTSTRAPPING

Für die P23R-Musterimplementierung wird eine vereinfachte Umsetzung der in der P23R-Rahmenarchitektur [2] beschriebenen Bootstrapping-Mechanismen vorgenommen. Das Bootstrapping stellt ein in der P23R-Rahmenarchitektur vorgeschlagenes Vorgehen der Initialisierung eines P23R dar, welches hierdurch erprobt werden soll.

So wird eine Konfigurationsmöglichkeit für den Bereich P23R-Leitstelle geschaffen. Dort wird es möglich sein, für eine zentrale Leitstelle den Dienst-Endpunkt der Trusted-Service-List (in der Rahmenar-

chitektur „controlCentre.tsl“) zu hinterlegen. Dort sind die weiteren Endpunkte der P23R-Leitstelle definiert, z. B. der Dienst-Endpunkt des Benachrichtigungsregelpaket-Depots.

Neben der Konfigurationsmöglichkeit für die Leitstelle wird auch eine Konfigurationsmöglichkeit für die Unternehmens-Infrastruktur geschaffen (in der Rahmenarchitektur sources.tsl). Hier kann ggf. der Dienst-Endpunkt des Unternehmensportals bzw. des P23R-Clients für die Freigabe von Benachrichtigungen konfiguriert werden. Für die Endpunkte der Quelldatenkonnektoren muss eine Lösung etabliert werden, welche die Konfiguration mehrerer Quelldatenkonnektoren (z. B. in Abhängigkeit der benötigten Benachrichtigungsregelpakete) erlaubt.

In einem weiteren Konfigurationsbestandteil, der seine Informationen zum Teil aus dem Build- bzw. Deploymentprozesses erhält, werden Informationen zur P23R Instanz und dessen Systemumgebung hinterlegt (wie beispielweise P23R-Identifier, P23R-Versionsnummer, unterstützte Version der P23R Rahmenarchitektur usw.).

Beim Starten einer konfigurationsabhängigen Deploymenteinheit der P23R-Instanz werden die relevanten Konfigurationsbestandteile geladen. Falls angebracht, wird beim Hochfahren der entsprechenden Deploymenteinheit ein automatischer Selbsttest durchgeführt. Bestandteil dieses Selbsttests sind u. a. Erreichbarkeitstests für alle relevanten Dienst-Endpunkte. Das Ergebnis des Selbsttests, insbesondere der Fehlerfall, wird über die Protokoll-Komponente protokolliert.

In der Deploymenteinheit `ModelAndRuleManagement` erfolgt nach jedem Hochfahren und dem erfolgreichen Selbsttest die Aktualisierung oder Erstbeschaffung der Benachrichtigungsregelpaket-Liste von der Leitstelle mit anschließender Aktualisierung der Datenbasis (`ModelAndRulePool`). Anschließend ist der P23R einsatzbereit.

Dienstadressen werden im Rahmen der P23R-Musterimplementierung statisch vorkonfiguriert. Die zur Abbildung des Vertrauensmodells notwendigen Zertifikate aus den Trusted Service Lists (TSLs) werden im Rahmen des P23R-Bootstrappings registriert. Dazu zählen die Verschlüsselungs- und Signaturzertifikate.

Von den durch die P23R-Rahmenarchitektur [2] geforderten Funktionen werden die in Tabelle 9 dargestellten Funktionen realisiert:

TABELLE 9: FUNKTIONALE ABBILDUNG BOOTSTRAPPING

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Erstmaliger Start der Leitstelle.	Beim Erstmaligen Start der Leitstelle wird aus den Host-Informationen automatisch eine zu dieser Leitstelle zugehörige TSL generiert und im TSL-Depot bereitgestellt.	Wird umgesetzt.
Erstmaliger Start eines P23R.	Laden grundlegender Information in vorgegebener Reihenfolge. Laden der TSL der P23R-Leitstelle (Verarbeitung der TSL i.B. des Dienstendpunkts des Zuständigkeitsverzeichnisses)	Wird umgesetzt.

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Manuelle Konfiguration von Dienstendpunkten.	Manuelle Konfiguration des Dienst-Endpunkts der P23R-Leitstelle und der Quellanwendung.	Wird umgesetzt.
Unterstützung mehrerer P23R-Leitstellen.	Das Anbinden von mehreren, dezentralen Leitstellen (bspw. die öffentliche Leitstelle und die eigene Unternehmensleitstelle).	Wird **nicht** umgesetzt.
Konfiguration der Dienste der Quellanwendung.	Initiale Konfiguration des Datenpools zur Abarbeitung der Recomendations zur Aktivierung der Benachrichtigungsregelpakete und Benachrichtigungsregelgruppen.	Wird umgesetzt.
Dienstendpunkte auf Quellanwendungsseite.	TSL für die Konfiguration der Quellanwendung werden nicht umgesetzt.	Wird **nicht** umgesetzt.
Durchführung eines Selbsttests.	Relevante P23R-Komponenten sollen beim Hochfahren einen Selbsttest durchführen. Bestandteil dieses Selbsttests soll eine Überprüfung der Erreichbarkeit von benötigten Dienst-Endpunkten sein.	Wird **nicht** umgesetzt.

2.2 Die P23R-Leitstelle und das Leitstellenportal

Gemäß der P23R-Rahmenarchitektur [2] zählt die P23R-Leitstelle zur P23R-Infrastruktur (siehe Abbildung 1). Sie untergliedert sich in die logischen Komponenten P23R-Benachrichtigungsregelpaket-Depot, P23R-Trusted-Service-List-Depot (TSL-Depot) und das P23R-Zuständigkeitsverzeichnis.

Die Umsetzung der P23R-Leitstelle umfasst die Implementierung einer Untermenge der in der P23R-Rahmenarchitektur [2] definierten Schnittstellen und Komponenten der P23R-Leitstelle, wie diese in Abbildung 14 dargestellt sind.

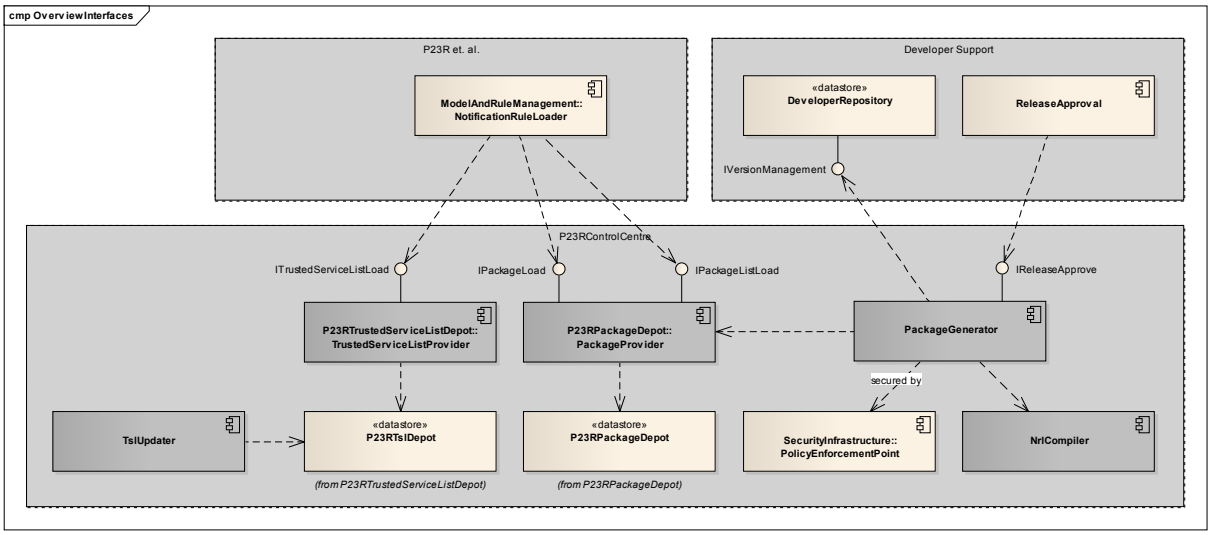


ABBILDUNG 14: KOMPONENTENÜBERSICHT FÜR DAS P23R-DEPOT (UML) [2]

Zur Verarbeitung von Benachrichtigungsregeln in der Generation-Pipeline (Abbildung 4) des P23R werden auf Seiten der P23R-Leitstelle folgende Komponenten umgesetzt:

- P23R-Benachrichtigungsregelpaket-Depot (siehe Abschnitt 2.2.1)
- P23R-Trustet-Service-List-Depot (siehe Abschnitt 2.2.2)
- P23R-Zuständigkeitsverzeichnis (siehe Abschnitt 2.2.3)

Die vorgesehenen Komponenten des „Developer Support“ sowie alle dazugehörigen Schnittstellen werden nicht umgesetzt. Bereits im Lastenheft wurde hierzu ausgeführt, dass dies auf Grund der zahlreichen analog zu verwendenden Erfahrungen aus dem Softwareentwicklungsbereich nicht zum Nachweis der korrekten Funktion des P23R-Prinzips erforderlich ist.

Wie durch das Lastenheft [1] gefordert wird zur Administration eine einfache Onlineanwendung, das Leitstellenportal (Komponente „`P23RControlCenter-Portal`“), entwickelt, die lediglich für den Einsatz im Rahmen der P23R-Musterimplementierung ausgerichtet ist und keine langfristige Lösung für die Verwendung durch die P23R-Unterstützungsstellen darstellt.

Das Leitstellenportal setzt die in Tabelle 10 dargestellten Anwendungsfälle um, die ein am Portal angemeldeter „Leitstellenadministrator“ durchführen können muss.

TABELLE 10: ÜBERSICHT DER ANWENDUNGSFÄLLE AUS DEM LASTENHEFT: LEITSTELLE

Gruppe	Use Case Bezeichner und Beschreibung
Benutzung des Leitstellenportals	UC-CC-1: An- und Abmelden eines Benutzers UC-CC-2: Verwalten der Benutzerkonten
Verwaltung der Datenmodell- und Benachrichtigungsregelpakete	UC-CC-3: Hinzufügen eines Datenmodellpakets UC-CC-4: Hinzufpgen eines Benachrichtigungsregelpakets UC-CC-5: Erzeugen einer aktuellen Paketliste
Verwaltung des Zuständigkeitsverzeichnisses	UC-CC-4: Editieren der Kataloge und Verzeichnisse

In Bezug auf die „Benutzung des Leitstellenportals“ (siehe Tabelle 10) verzichtet das Leitstellenportal dabei auf eine differenzierte Rollenbetrachtung. Der Rolle des „Administrator“ wird vollumfänglicher Zugriff auf alle Bereiche der Leitstelle gewährt.

Für die P23R-Musterimplementierung werden die in Tabelle 11 dargestellten Benutzer (mit vollem Zugriffsumfang) standardmäßig im P23R-Leitstellenportal vorgehalten.

TABELLE 11: WERTETABELLE „BENUTZER IM P23R-LEITSTELLENPORTAL“

Benutzername	Passwort	Rolle (siehe Tabelle 20)
Administrator	P23R_admin	Administrator

Zur „Verwaltung der Regelpakete“ (siehe Tabelle 10) wird im Leitstellenportal ein UI bereitgestellt, über das Regelpakete per Datei-Upload hochgeladen und verwaltete werden können.

Die „Verwaltung des Zuständigkeitsverzeichnisses“ (siehe Tabelle 10) erfolgt direkt über das Administrationsinterface der für die Datenverwaltung im Zuständigkeitsverzeichnis vorgesehenen Datenbanklösung.

Änderungen am TSL-Depot hingegen können lediglich im Backend durch einen entsprechend autorisierten Administrator durchgeführt werden. Im Leitstellenportal ist es lediglich möglich sich den aktuellen Status der im TSL-Depot vorgehaltenen TSLs anzeigen zu lassen.

2.2.1 DAS BENACHRICHTIGUNGSREGELPAKET-DEPOT

Die Verwaltung und Bereitstellung von Benachrichtigungsregelpakete erfolgt über das Benachrichtigungsregelpaket-Depot (Komponente „*P23RNotificationRulePackageDepot*“).

Die Schnittstellen des Benachrichtigungsregelpaket-Depots werden als RESTful WebServices umgesetzt. Im Einzelnen werden folgende Schnittstellen realisiert:

- IPackageListLoad

  Über diese Schnittstelle kann gemäß [5] ein Archiv mit einer Übersicht über alle im Benachrichtigungsregelpaket-Depot enthaltenen Benachrichtigungsregelpakete heruntergeladen werden.

  In diesem Archiv ist u. a. ein URI für jedes Benachrichtigungsregelpaket definiert, über die es vom Depot heruntergeladen werden kann.

- <Identifier Benachrichtigungsregelpaket>/IPackageLoad

  Über diese Schnittstelle kann gemäß [5] das angegebene Benachrichtigungsregelpaket-Archiv heruntergeladen werden.

  Die Schnittstelle steht für jedes Benachrichtigungsregelpaket innerhalb des Benachrichtigungsregelpaket-Depots zur Verfügung.

Von den durch die P23R-Rahmenarchitektur [2] geforderten Funktionen werden die in Tabelle 12 dargestellten Funktionen realisiert:

TABELLE 12: FUNKTIONALE ABBILDUNG BENACHRICHTIGUNGSREGELPAKET-DEPOT

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Bereitstellung alle Benachrichtigungsregelpakete.	Die Liste aller Benachrichtigungsregelpakete berechnet sich automatisch aus allen im Benachrichtigungsregelpakete-Depot befindlichen Benachrichtigungsregelpaketen und steht somit stets konsistent zur Verfügung.	Wird umgesetzt.
Signieren von Benachrichtigungsregelpaketen.	Signieren aller durch die P23R-Leitstelle bereitgestellten Benachrichtigungsregelpakete bei der Einstellung in das Benachrichtigungsregelpakete-Depot.	Wird umgesetzt.

2.2.2 DAS TRUSTED-SERVICE-LIST-DEPOT

Die Bereitstellung der erforderlichen TSLs erfolgt über das Trusted-Service-List-Depot (Komponente „*P23RTrustedServiceListDepot*“). Es stellt alle eingestellten TSLs unter einem konfigurierbaren URL zum Abruf bereit.

Die Schnittstellen des TSL-Depots werden als RESTful WebServices umgesetzt. Im Einzelnen wird folgende Schnittstelle realisiert:

- ITrustedServiceListLoad

  Über diese Schnittstelle können die TSLs aus dem TSL-Depot der P23R-Leitstelle heruntergeladen werden.

Von den durch die P23R-Rahmenarchitektur [2] geforderten Funktionen werden die in Tabelle 13 dargestellten Funktionen realisiert:

TABELLE 13: FUNKTIONALE ABBILDUNG TRUSTED-SERVICE-LIST-DEPOT

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Bereitstellung aller im Trusted-Service-List-Depot enthaltenen TSLs.	Es ist möglich die bereitgestellten TSLs aus dem Trusted-Service-List-Depot abzurufen, um diese zu Verarbeiten.	Wird umgesetzt.

2.2.3 DAS P23R-ZUSTÄNDIGKEITSVERZEICHNIS

Das P23R-Zuständigkeitsverzeichnis (siehe P23R-Rahmenarchitektur [2]) ist der Komplex von Katalogen, Verzeichnissen und Diensten zur Ermittlung von Benachrichtigungsempfängern und deren Kommunikationskanälen (Komponenten „*P23RResponsabilityDirectory*“).

Dabei sind gemäß P23R-Rahmenarchitektur [2] und Kapitel 3 der zugehörigen Spezifikationen [4] zwei Schnittstellen zur Nutzung durch den P23R bereitzustellen:

- INotificationReceiverQuery realisiert durch NotificationReceiverFinder

  Ermöglicht das Auffinden der Benachrichtigungsempfänger während der Benachrichtigungsgenerierung (siehe Abschnitt 2.1.2).

- IAddressQuery realisiert durch AddressDiscoverer

Ermöglicht das Bestimmen des Kommunikationskanals beim Benachrichtigungstransport (siehe Abschnitt 2.1.4).

Für die Kataloge und Verzeichnisse wird eine Trennung der Inhalte vorgenommen um Erweiterungen im Sinne einer nachhaltigen Nutzung zu gewährleisten. Zudem sollen diese möglichst einfach zu handhaben sein. Die Zugriffe werden über eigene Zugriffskomponenten gekapselt.

Für das P23R-Zuständigkeitsverzeichnis werden gemäß P23R-Rahmenarchitektur [2] folgende Kataloge und Verzeichnisse realisiert:

- Leistungskatalog (ServiceCatalogue)
- Kriterienkatalog (CriteriaCatalogue)
- Benachrichtigungsempfängerverzeichnis (NotificationReceiverDirectory)
- Elektronisches Diensteverzeichnis (EserviceDirectory)
- Zuständigkeitsverzeichnis (ResponsibilityDirectory)
  - Verknüpfung für Receiver (MappingReceiver)
  - Verknüpfung für elektronische Dienste (MappingEservice)

Die Schnittstellen werden als WebService realisiert und nutzen zur Abfrage die dahinter liegende Kataloge und Verzeichnisse, welche jeweils in einer eigenen Datenbank realisiert sind.

Von den durch die P23R-Rahmenarchitektur [2] geforderten Funktionen werden die in Tabelle 14 dargestellten Funktionen realisiert, wozu alle zuvor genannten Kataloge und Verzeichnisse umgesetzt werden.

TABELLE 14: FUNKTIONALE ABBILDUNG P23R-ZUSTÄNDIGKEITSVERZEICHNIS

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Bereitstellung der erforderlichen WebServices.	Es ist möglich Informationen aus dem P23R-Zuständigkeitsverzeichnis über die definierten WebServices abzurufen.	Wird umgesetzt.
Einbindung externer Kataloge und Verzeichnisse.	Das P23R-Zuständigkeitsverzeichnis kann sich automatisiert aus Verzeichnissen und Katalogen von externen Informationsdienstleistern und Datenbanken versorgen.	Wird **nicht** umgesetzt.
Sicherung gegen externe Veränderung.	Schutz der Datenbestände und Suchergebnisse gegen externe Veränderung.	Wird **nicht** umgesetzt.

2.3 DER P23R-CLIENT ALS ONLINEANWENDUNG

Gemäß der P23R-Rahmenarchitektur [2] zählt der P23R-Client zur P23R-Infrastruktur (siehe Abbildung 1). Die Kommunikation mit dem P23R und der P23R-Leitstelle erfolgt über eine Reihe von Schnittstellen, die einerseits vom P23R andererseits vom P23R-Client bereitgestellt werden (siehe Abbildung 2).

Der P23R-Client (Komponente „*P23RClient*“) wird in der P23R-Musterimplementierung als einfache Onlineanwendung umgesetzt, das eine mögliche Realisierung des P23R-Clients auf Basis der P23R-Rahmenarchitektur [2] darstellt.

Hierzu setzt der P23R-Client die in Tabelle 15 dargestellten Anwendungsfälle um, die ein am P23R-Client angemeldeter „Benutzer“ durchführen können muss:

TABELLE 15: ÜBERSICHT DER ANWENDUNGSFÄLLE AUS DEM LASTENHEFT: P23R-CLIENT

Gruppe	Use Case Bezeichner und Beschreibung
Benutzung des P23-Clients	UC-CL-1: An- und Abmelden eines Benutzers UC-CL-2: Verwalten der Benutzerkonten UC-CL-3: Einsicht in das P23R-Protokoll
Verwaltung der Benachrichtigungsregeln	UC-CL-4: Verwalten der auszuführenden Benachrichtigungsregeln UC-CL-5: Auslösen der Abarbeitung einer Benachrichtigungsregel
Pflege der Unternehmensdaten	UC-CL-6: Editieren der zu verarbeitenden Unternehmensdaten
Steuerung der Benachrichtigungsübermittlung	UC-CL-7: Bearbeiten und Freigeben einer Benachrichtigung

Die einzelnen Anwendungsfälle werden jeweils über UI-Elemente realisiert, die an die von der P23R-Rahmenarchitektur [2] definierten Schnittstellen und Komponenten „angebunden“ werden, wie diese in Abbildung 15 dargestellt werden.

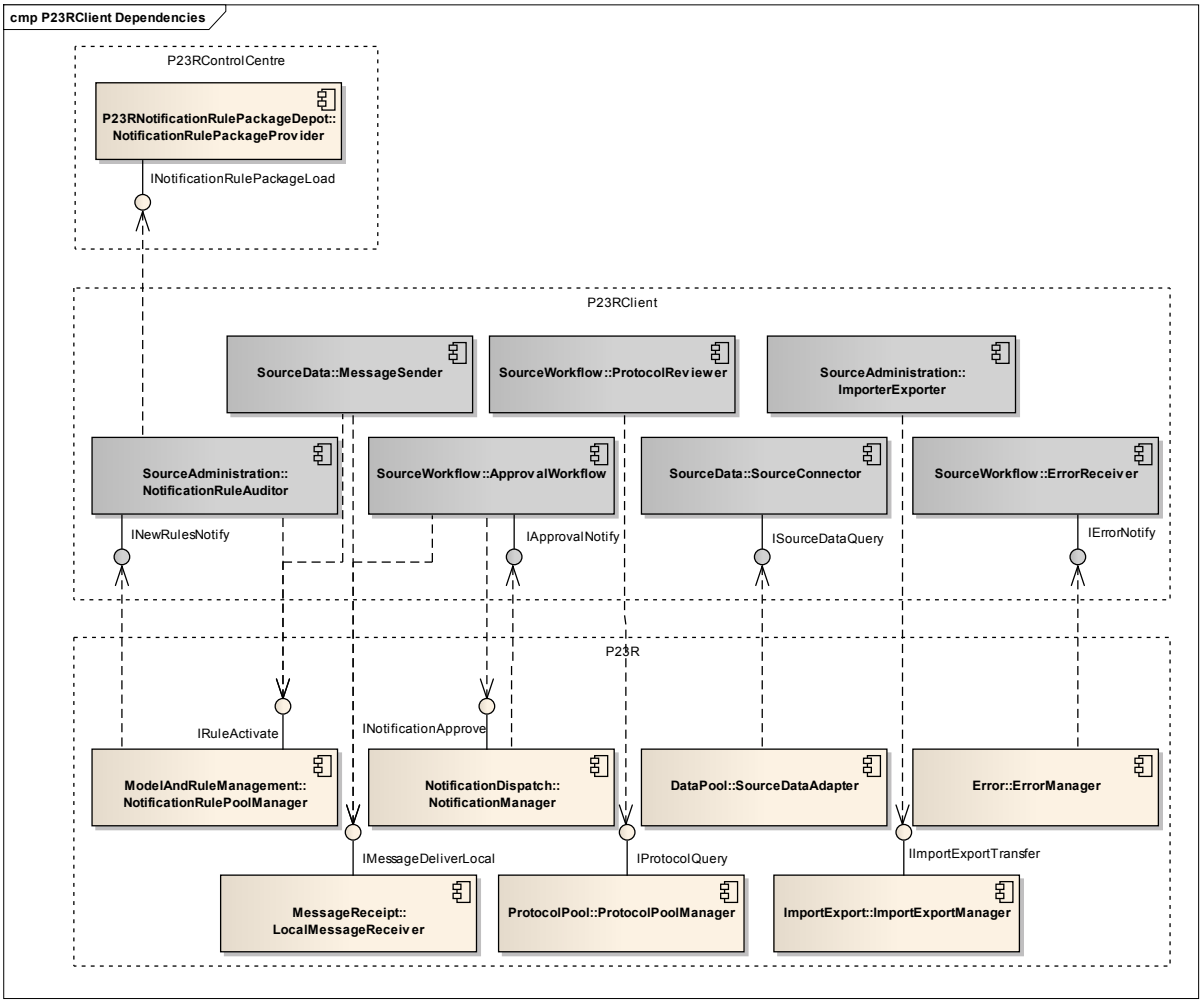


ABBILDUNG 15: KOMPONENTEN FÜR DEN P23R-CLIENT (UML [2])

Folgende Komponenten werden dabei resultierend aus den Anwendungsfällen umgesetzt und in den nachfolgenden Abschnitten in ihrer jeweiligen Fertigungstiefe weiter ausgeführt:

- NotificationRuleAuditor (siehe Abschnitt 2.3.2)

  Bezogen auf den Anwendungsfall UC-CL-4 (siehe Tabelle 15).

- MessageSender (siehe Abschnitt 2.3.2)

  Bezogen auf den Anwendungsall UC-CL-5 (siehe Tabelle 15).

- ApprovalWorkflow (siehe Abschnitt 2.3.4)

  Bezogen auf den Anwendungsall UC-CL-7 (siehe Tabelle 15).

- ProtocolReviewer (siehe Abschnitt 2.3.1)

  Bezogen auf den Anwendungsall UC-CL-3 (siehe Tabelle 15).

- SourceConnector (siehe Abschnitt 2.3.3).

  Bezogen auf den Anwendungsall UC-CL-6 (siehe Tabelle 15).

Zum besseren Verständnis wird im Folgenden kurz erläutert, welche Komponenten nicht durch die P23R-Musterimplementierung umgesetzt werden:

- ImporterExporter
- ErrorReceiver

Darüber hinaus sind eine Reihe von Sicherheitskomponenten in den P23R-Client integriert (siehe Abschnitt 2.3.5), um den P23R und den P23R-Client mit den erforderlichen Sicherheitsmerkmalen zu versorgen.

Dabei spielen auch die Anwendungsfälle UC-CL-1 und UC-CL-2 (siehe Tabelle 15) eine Rolle, und zwar sowohl im P23R-Client als auch in den zugehörigen Sicherheitskomponenten.

2.3.1 ANWENDUNGSFÄLLE ZUR BENUTZUNG

Umgesetzt wird Anwendungsfall UC-CL-1 über das Erfassen der Anmeldeinformationen in einem Benutzername- / Passwortdialog. Das anschließende Abmelden erfolgt durch einen Abmelde-Link innerhalb des P23R-Clients.

Die bei der Anmeldung erfolgten Eingaben werden an den Authentifizierungs-Dienst der Sicherheitsarchitektur weitergeleitet, die den P23R-Client dann über das Authentifizierungsergebnis informiert (siehe Abschnitt 2.3.5.1).

Die Verwaltung der Benutzerkonten, welche aus UC-CL-2 hervorgeht, erfolgt über den direkten Zugriff des P23R-Client auf die der Sicherheitsarchitektur zugrundeliegende Benutzerverwaltung. Der Benutzer ist dabei in der Lage alle Attribute eines anderen Benutzers zu setzen bzw. zu lesen, sowie etwaig benötigtes Schlüsselmaterial für die Signierung im Zuge der Freigabe einzustellen.

Während der Laufzeit verwendet der P23R-Client zudem die durch die Sicherheitsarchitektur bereitgestellten Policies und Rolleninformationen, um den Zugriff des angemeldeten Benutzers auf die einzelnen Elemente des P23R-Client zu steuern (siehe Abschnitt 2.3.5.2) und etwaige nicht zugreifbare UI-Elemente zu sperren bzw. auszublenden.

Zur Realisierung des UC-CL-3 werden im P23R-Client folgende Schnittstellen und Komponenten aus der P23R-Rahmenarchitektur [2] im Bezug umgesetzt:

- ProtocolReviewer i.V.m. IProtocolQuery

Von den durch die P23R-Rahmenachritektur [2] geforderten Funktionen werden die in Tabelle 16 dargestellten Funktionen realisiert:

TABELLE 16: FUNKTIONALE ABBILDUNG P23R-CLIENT: ANWENDUNGSFÄLLE ZUR BENUTZUNG

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Authentifizierung und Autorisierung eines Benutzers.	Ein Benutzer kann sich am P23R-Client an- und abmelden und erhält eine an seinen Benutzerrechten ausgerichtete Darstellung und entsprechende Funktionszugriffe angezeigt.	Wird umgesetzt.
Einsicht in das P23R-Protokoll.	Über den durch den P23R bereitgestellte WebService auf Basis von IProtocol-	Wird umgesetzt.

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
	Query wird das P23R-Protokoll abgerufen und im P23R-Client dargestellt (siehe Kapitel 2.1.6 „Protokollpool“).	

2.3.2 VERWALTUNG DER BENACHRICHTIGUNGSREGELN

Im P23R-Client wird dem Benutzer zur Verwaltung der Benachrichtigungsregeln angezeigt, welche Benachrichtigungsregelgruppen und Benachrichtigungsregeln im P23R eingespielt sind (siehe Abschnitt 2.1.7) und ob ein Update für eine Benachrichtigungsregel vorliegt (sprich, ob es von einer Benachrichtigungsregel mehrere Releases gibt).

Für jede Benachrichtigungsregelgruppe kann der Benutzer im P23R-Client dann die einzelnen Benachrichtigungsregeln (UC-CL-4) aktivieren und auch wieder deaktivieren. Dabei hat er die Möglichkeit das einer Benachrichtigungsregel zugeordnete Benachrichtigungspaket einzusehen (Bezug direkt über die P23R-Leitstelle).

Darüber hinaus ist der P23R-Client in der Lage eine generierte Benachrichtigung automatisch freizugeben, die der Benutzer ebenfalls für jede Benachrichtigungsregel hinterlegen kann, was zur Folge hat, dass die Freigabe einer darauf genierten Benachrichtigung mit seiner Identität (und Signatur) freigegeben wird.

Jede einzelne Benachrichtigungsregel kann darüber hinaus in der Benachrichtigungsregelverwaltung manuell ausgelöst werden (UC-CL-5). Die Benachrichtigungsregel muss dafür zuvor aktiviert worden sein.

Von den durch die P23R-Rahmenarchitektur [2] geforderten Funktionen werden die in Tabelle 17 dargestellten Funktionen realisiert:

TABELLE 17: FUNKTIONALE ABBILDUNG P23R-CLIENT: VERWALTUNG DER BENACHRICHTIGUNGSREGELN

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Bereitstellung Regelinfodienst (INewRulesNotify)	Der P23R-Client wird durch den P23R über neue Regeln informiert, sobald diese bereitstehen.	Wird **nicht** umgesetzt, da die Musterimplementierung des P23R vorsieht, dass das Abfragen von neuen Regeln explizite durch das Aufrufen des P23R-Client erfolgt.
Verwaltung von Benachrichtigungsregeln.	Aktivierung und Deaktivierung von Benachrichtigungsregeln im P23R über IRuleActivate.	Wird umgesetzt.
Möglichkeit zur automatischen Freigabe.	Es ist möglich die durch eine Benachrichtigungsregel erstellte Benachrichtigung automatisch freigeben und signieren zu lassen.	Wird umgesetzt.

2.3.3 PFLEGE DER UNTERNEHMENSDATEN

Die durch den im Datenpool konfigurierten Quelldatenkonnektor bereitgestellten Unternehmensdaten können im P23R-Client sowohl eingesehen als auch bearbeitet werden (UC-CL-6).

Mit Hilfe einer entsprechenden Benutzeroberfläche kann der Benutzer die einzelnen Teildatenmodelle des Pivotdatenmodells aufrufen und den für den jeweiligen Namespace bereitgestellten Datensatz, welcher durch den Quelldatenkonnektor eingelesen werden würde, verarbeiten.

Die Pflege der Daten erfolgt direkt auf der dem P23R-Client zugeordneten XML-Datenbank, die auch als Quelle für den Test-Quelldatenkonnektor dient.

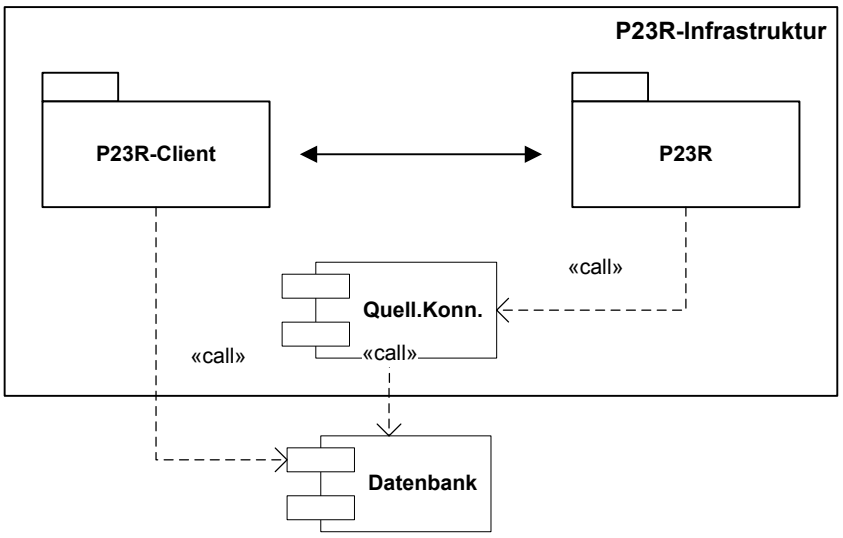


ABBILDUNG 16: ZUSAMMENSPIEL "UNTERNEHMENSDATEN" ZWISCHEN P23R-CLIENT UND P23R

2.3.4 STEUERUNG DER BENACHRICHTIGUNGSÜBERMITTLUNG

Im Zuge der Steuerung der Benachrichtigungsübermittlung wird die vom P23R generierte Benachrichtigung über ein UI auf Basis des in der Benachrichtigungsregel enthaltenen XML-Schemas angezeigt. Mit Hilfe dieser Informationen wird im P23R-Client ein HTML-Formular in barrierefreier Listen-Gliederung dargestellt.

Zur Bearbeitung einer Benachrichtigung via Web-Browser im P23R-Client werden dabei folgende Funktionen zur Verfügung gestellt:

- Einfügen von Datensätzen in die Baumstruktur
- Ändern von Daten innerhalb der Baumstruktur
- Löschen von Datensätzen in der Baumstruktur

Darüber hinaus wird eine Validierung der Eingabe (XSD) vorgenommen und eine Pflichtfeldprüfung durchgeführt. Ergänzend kann die Benachrichtigung auch exportiert und nach Bearbeitung wieder importiert werden.

Zur Freigabe einer Benachrichtigung kann der Benutzer über die Oberfläche des P23R-Client die zuvor optional bearbeite Benachrichtigung über einen einfachen „Klick“ auf die entsprechende Freigabeoption freigeben. Dabei stehen ihm folgende Möglichkeiten zur Verfügung:

- Freigabe der Benachrichtigung zum Versand

- Ablehnen des Versands der Benachrichtigung

Die Übermittlung der Benachrichtigung vom P23R-Client an den P23R erfolgt – gemäß den Vorgaben der P23R-Sicherheitsarchitektur [3] – unter Einsatz einer Verschlüsselung des SOAP-Bodys mittels XML Encryption.

Von den durch die P23R-Rahmenarchitektur [2] geforderten Funktionen werden die in Tabelle 18 dargestellten Funktionen realisiert.

TABELLE 18: FUNKTIONALE ABBILDUNG P23R-CLIENT: STEUERUNG DER ÜBERMITTLUNG

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Freigeben einer Benachrichtigung.	Eine vom P23R generierte Benachrichtigung wird zur Freigabe vorgelegt und das Ergebnis an den P23R zurückübermittelt.	Wird umgesetzt.
Bearbeiten einer freizugebenden Benachrichtigung.	Der Benutzer kann eine freizugebende Benachrichtigung bearbeiten und zu diesem Zweck auch exportieren und wieder reimportieren.	Wird umgesetzt.
Signieren der freizugebenden Benachrichtigung.	Die freizugebende Benachrichtigung wird vor der Rückübermittlung an den P23R auf Basis eines hinterlegten Benutzerzertifikats (Class-1) signiert.	Wird umgesetzt.
Überprüfung auf Benachrichtigungsregeländerungen.	Im Zuge der Freigabe wird vom P23R-Client überprüft, ob sich die zugehörige Benachrichtigungsregel in der Zwischenzeit geändert hat.	Wird **nicht** umgesetzt.
Einsatz des NotificationView.xslt zur Darstellung.	Aus der Benachrichtigungsregel kann ein Skript entnommen werden, dass die zur Freigabe vorgelegte Benachrichtigung in eine HTML-Darstellung transformiert.	Wird **nicht** umgesetzt.

2.3.5 INTEGRATION DER SICHERHEITSKOMPONENTEN

Neben den in der P23R-Musterimplementierung realisierten Komponenten der P23R-Rahmenarchitektur werden eine Reihe von Mechanismen aus der P23R-Sicherheitsarchitektur [3] umgesetzt. Dazu zählen die Konzepte zur Authentifizierung und Autorisierung von Nutzern des P23R bzw. P23R-Client sowie die Zugriffskontrollprüfung im P23R.

Die folgende Abbildung aus der Sicherheitsarchitektur zeigt hierzu das Zusammenspiel von Identity und Access Management Subsystemen zwischen P23R-Client und P23R, die in den folgenden Abschnitten weiter ausgeführt werden:

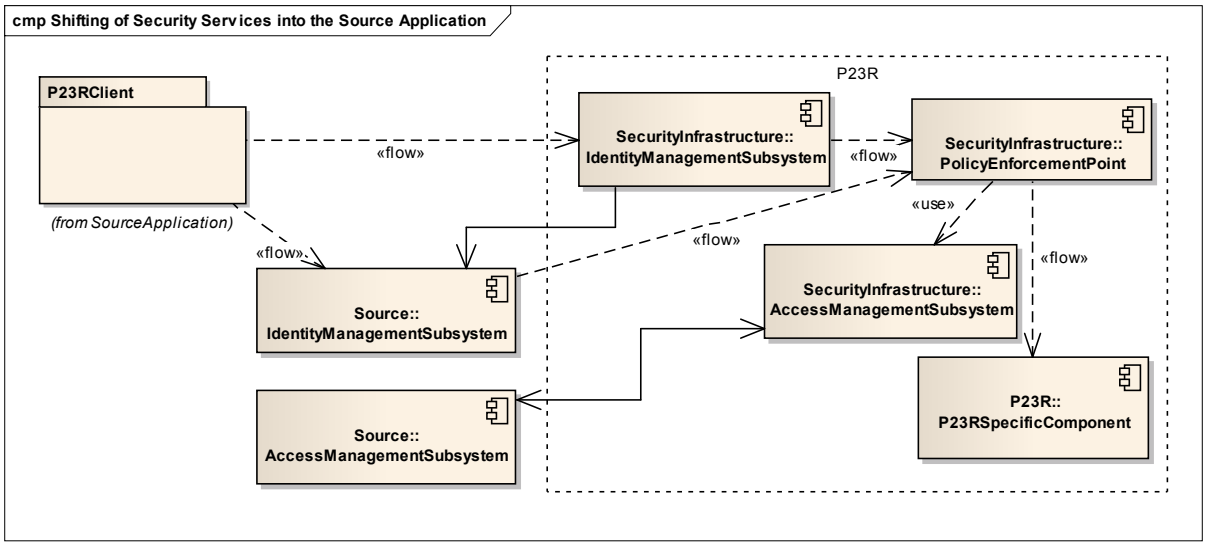


ABBILDUNG 17: INTEGRATION DER SICHERHEITSKOMPONENTEN (UML) [3]

2.3.5.1 AUTHENTIFIZIERUNG (IDENTITY MANAGEMENT)

Die Benutzer des P23R aus den Unternehmen müssen durch den P23R-IdentityProvider (Teil des IdentityManagementSubsystem) direkt oder indirekt authentifiziert werden.

Direkt meint, dass Benutzer dem P23R bekannt sind und diese auch authentifiziert werden können. Indirekt bedeutet hingegen, dass der Authentifizierung durch den P23R-IdentityProvider (Bestandteil des Identity Management Subsystems (siehe Abbildung 17) eine lokale Authentifizierung vorangeht.

In der P23R-Musterimplementierung erfolgt die Authentifizierung über den P23R-Client, wobei der Benutzer lokal über eine Benutzername/Passwort-Kombination authentifiziert wird. Dazu wird ein registrierter Handler („Authenticator“) vom Applikationsserver aufgerufen, welcher die eingegebene Kombination verifizieren soll.

Hierzu wird über einen Guarantor Token Service, der innerhalb des P23R-Client implementiert ist, für den P23R eine Guarantor Assertion erstellt und ausgetauscht.

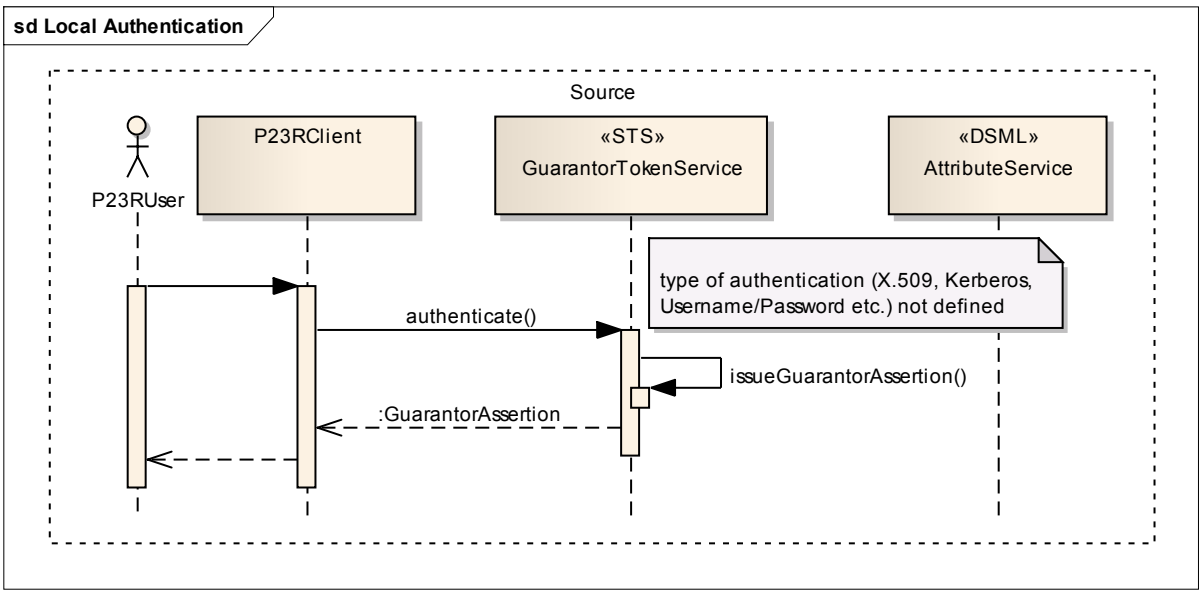


ABBILDUNG 18: LOKALE AUTHENTISIERUNG IM UNTERNEHMEN (UML) [3]

Da Subsysteme nachgenutzt werden sollen, erfolgt ein Aufruf zum Guarantor Token Service mit einem Username Token, welche zudem das Passwort enthält. Erst dieser Dienst ist in der Lage, die Kombination zu überprüfen und im Erfolgsfall einen Authentifizierungsnachweis (SAML 2.0 Assertion bzw. Guarantor Assertion) auszustellen.

Dieser Nachweis wird in der aktuellen Sitzung gespeichert und für die Authentisierung am P23R-IdentityProvider verwendet. Der P23R-IdentityProvider stellt abermals einen Authentifizierungsnachweis aus (SAML 2.0 Assertion bzw. Identity Assertion), der ebenfalls in der Sitzung des Benutzers gespeichert wird.

Sowohl der P23R-IdentityProvider als auch der Guarantor Token Service ist mit Dienst-Policies versehen, welche mittels WS-Policy und WS-SecurityPolicy kodiert sind. Die Aufrufe beider Dienste erfolgt aus dem Web-Service-Framework „Metro“ heraus, welches auf dem Applikationsserver vorinstalliert ist.

Für die P23R-Musterimplementierung werden folgende Benutzer standardmäßig im P23R-Client vorgehalten.

TABELLE 19: WERTETABELLE „BENUTZER IM P23R-CLIENT“

Benutzername	Passwort	Rolle (siehe Tabelle 20)
Administrator (Max Mustermann)	P23R_admin	Administrator und Anwender
Anwender (Erika Mustermann)	P23R_user	Anwender

Für beide Benutzer wird darüber hinaus ein gültiges (Class-1) Dokumentenzertifikat bereitgestellt, um die Signierung der freizugebenden Benachrichtigung (siehe Abschnitt 2.3.4) zu realisieren.

2.3.5.2 ABRUF DER ACCESS-POLICIES

Die Spezifikation der P23R-Sicherheitsarchitektur erlaubt es, interne – im P23R vorgehaltene – Access Policies und / oder externe – aus dem Unternehmen stammende – Access Policies für die Zugriffskontrollprüfung heranzuziehen.

Die P23R-Musterimplementierung beschränkt sich bei der Autorisierung darauf, zwei grobgranulare Access Policies bereitzustellen, die den zuvor beschriebenen Rollen entsprechen (siehe Tabelle 20).

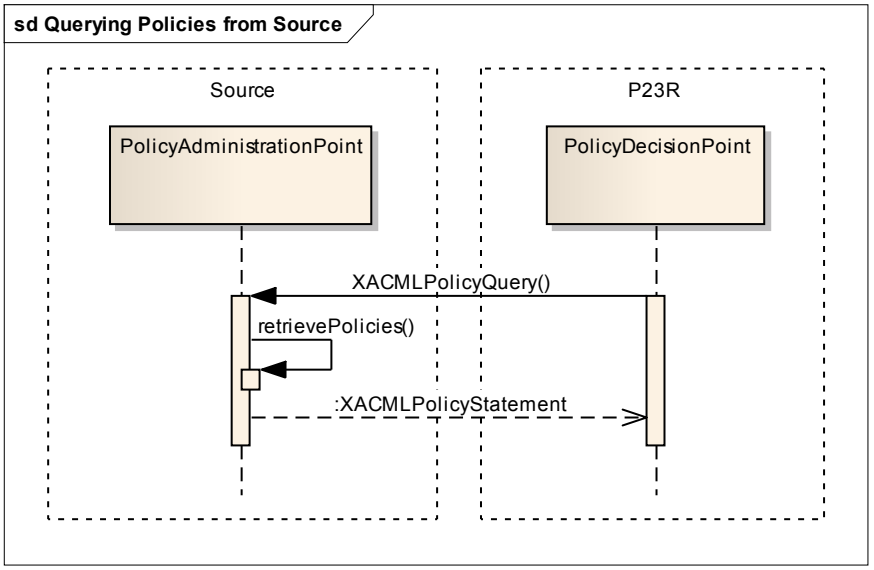


ABBILDUNG 19: ABRUF EINER ODER MEHRERER BERECHTIGUNGS-POLICIES (UML) [3]

Die Access Policies werden dafür über einen simulierten Dienst (einen logischen Policy Administration Point – PAP) verfügbar gemacht, wobei folgender Umfang realisiert werden soll:

- Zwei vordefinierte, grobgranulare Access Policies werden designt und vom P23R aus dem PAP abgerufen, der sich innerhalb des P23R-Client befindet. Die Zugriffskontrollprüfung im P23R basiert auf diese beiden Access Policies.
- Die beiden Policies unterscheiden dabei Administratoren und Anwender, wobei der Administrator Benachrichtigungsregeln aktivieren (sprich in den Scheduler einstellen), der Anwender die Freigabe und Bearbeitung machen darf.

Die Umsetzung einer generischen Schnittstelle für PIP-Anfragen (Policy Information Point) per DSML an den P23R-Client wird im Rahmen der P23R-Musterimplementierung nicht umgesetzt.

Für die P23R-Musterimplementierung werden folgende Rollen / Policies standardmäßig im P23R-Client vorgehalten:

TABELLE 20: WERTETABELLE „ROLLEN IM P23R-CLIENT“

Rollenbezeichnung	Rollenberechtigungen
Administrator	Alle Anwendungsfälle mit Ausnahme von: - UC-CL-7: Bearbeiten und Freigeben einer Benachrichtigung

Rollenbezeichnung	Rollenberechtigungen
Anwender	Alle Anwendungsfälle mit Ausnahme von: - UC-CL-2: Verwalten der Benutzerkonten - UC-CL-4: Verwalten der auszuführenden Benachrichtigungsregeln - UC-CL-5: Auslösen der Abarbeitung einer Benachrichtigungsregel

2.4 Die generischen Konnektoren der P23R-Musterimplementierung

Um die P23R-Musterimplementierung auch ohne konkretes Pilotszenario anwenden und testen zu können, werden sowohl ein minimaler Quelldaten- als auch ein Kommunikationskonnektor umgesetzt und integriert.

2.4.1 Der Quelldatenkonnektor

Der Quelldatenkonnektor ist in der Lage die Daten der Teildatenmodelle des Pivot-Datenmodells in einer XML-Datenbank zu verwalten und für den P23R bereitzustellen. Der P23R-Client ist zudem in der Lage diese Daten ebenfalls über die XML-Datenbank zu verwalten.

Dabei werden folgende im Lastenheft [1] beschriebenen Anwendungsfälle realisiert.

TABELLE 21: ÜBERSICHT DER ANWENDUNGSFÄLLE AUS DEM LASTENHEFT: QUELLDATENKONNEKTOR

Gruppe	Use Case Bezeichner und Beschreibung
Austausch der Unternehmensdaten	UC-SA-1: Bereitstellen der Unternehmensdaten UC-SA-2: Vorhalten des Pivotdatensatzes

Sowohl das Bereitstellen der Unternehmensdaten als auch das Vorhalten des zu den einzelnen Teildatenmodellen des Pivot-Datenmodells konformen Datensatzes erfolgt auf demselben Datensatzschema (dem Schema der einzelnen Teildatenmodelle des Pivot-Datenmodells selbst) und in der Datenbank ohne den Einsatz einer Transferdatenbank.

2.4.2 Der Kommunikationskonnektor

Der Kommunikationskonnektor ist in der Lage eine freigegebene Benachrichtigung hinsichtlich der angebrachten Signatur zu prüfen und das Ergebnis der Generierung nach erfolgreicher Repräsentationstransformation in den Log-Stream zu schreiben, wobei die eigentliche Benachrichtigung im Anschluss verworfen wird.

Der Kommunikationskonnektor implementiert hierzu entsprechend den NORMATIVEN Vorgaben aus Kapitel 4 in [4] – die geforderte Webservice-Schnittstelle.

Er realisiert folgende im Lastenheft [1] beschrieben Anwendungsfälle.

TABELLE 22: ÜBERSICHT DER ANWENDUNGSFÄLLE AUS DEM LASTENHEFT: KOMMUNIKATIONSKONNEKTOR

Gruppe	Use Case Bezeichner und Beschreibung
Benachrichtigungsübermittlung an die Verwaltung	UC-TA-1: Übermitteln einer Benachrichtigung UC-TA-2: Erstellen einer Übermittlungsbestätigung

2.5 DIE INTEGRATION DER SICHERHEITSARCHITEKTUR

Zur Realisierung der Zugriffskontrollprüfung ist neben der eigentlichen Umsetzung des Security-Frameworks zur Authentisierung eine Einsprungsfunktion aus dem P23R („Hook“) zu realisieren, die in der Verarbeitung der Web Service-Aufrufe integriert ist, wie in Abbildung 20 dargestellt:

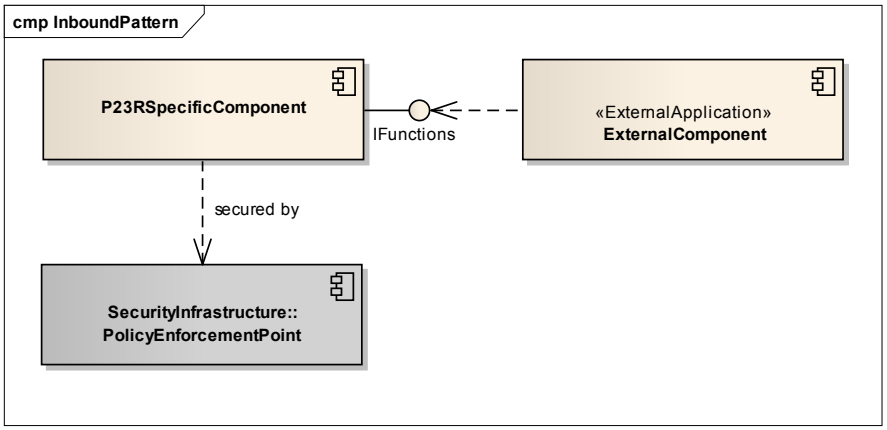


ABBILDUNG 20: MUSTER FÜR EINEN DIENST BEI EINGEHENDEN ANFRAGEN (UML) [2]

Die technische Spezifikation der P23R-Sicherheitsarchitektur beschreibt exemplarisch, wie dies passieren kann. Dies wird in der P23R-Musterimplementierung umgesetzt:

- Ein SOAP-Aufruf wird über einen SOAP-Handler geprüft (Entwurfsmuster eines Interceptor).
- Der Handler analysiert die SOAP-Nachricht und generiert eine Autorisierungsanfrage an einen Policy Decision Point (PDP), wobei der Handler als Policy Enforcement Point (PEP) fungiert.
- Der PDP ruft die Access Policies vom P23R-Client (Repräsentant eines PAP) ab. Dabei wird unternehmensseitig die passende Access Policy identifiziert und nur diejenige zurückgesandt, welche zur Autorisierungsanfrage passt (Policy Activation).

2.5.1 INTEGRITÄT, AUTHENTIZITÄT UND VERTRAULICHKEIT

Integrität, Authentizität und Vertraulichkeit von Benachrichtigungen sind wesentliche Merkmale der P23R-Infrastruktur, über die auch ein Nachweis erbracht werden soll. Hierzu gilt es die normativen Vorgaben zu Signaturen hinsichtlich der von einem P23R an eine Verwaltung übermittelten Benachrichtigungen aus der Sicherheitsarchitektur aufzugreifen.

Die Kommunikationskonnektoren beschränken sich dabei im Bedarfsfall jedoch auf den Einsatz von fortgeschrittenen Signaturen. Damit assoziiertes Schlüsselmaterial, Zertifikate und CAs werden für die P23R-Musterimplementierung selbst generiert und haben ausschließlich Testcharakter. Alle Zertifikate werden dabei über TSLs verfügbar gemacht.

Hierzu werden folgende Annahmen getroffen bzw. Umsetzungsvorgaben festgehalten:

- (Test-)Zertifikate und Schlüsselmaterial sind vorhanden und können über Bootstrapping publiziert werden.
- Alle Dienste des P23R haben assoziierte Keystores, welche das Schlüsselmaterial sowie Zertifikate aufnehmen.

2.5.2 ÜBERSICHT DER ABGEBILDETEN SICHERHEITSMECHANISMEN

Tabelle 23 zeigt eine Übersicht der in der P23R-Musterimplementierung abgebildeten Sicherheitsmechanismen.

TABELLE 23: FUNKTIONALE ABBILDUNG DER SICHERHEITSMECHANISMEN

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Guarantor Assertion Verfahren.	Eine Authentisierung erfolgt lokal im Online-Portal und wird um eine P23R-konforme Authentisierung ergänzt.	Umsetzung: Diese in der Sicherheitsarchitektur spezifizierte Schnittstelle wird vollständig umgesetzt.
Authentifizierung durch P23R-Identity-Provider.	Ein Benutzer wird mittels Guarantor Assertion authentifiziert.	Umsetzung: Diese in der Sicherheitsarchitektur spezifizierte Funktionalität wird vollständig umgesetzt.
Abruf von Access Policies.	Access Policies werden über einen PAP aus dem P23R-Client abgerufen.	Umsetzung: Diese in der Sicherheitsarchitektur spezifizierte Schnittstelle wird vollständig umgesetzt.
Abruf von Attributen aus Verzeichnisdiensten.	Zur Zugriffskontrollprüfung können weitere Attribute notwendig sein. Diese werden über einen Policy Information Point abgerufen.	Keine Umsetzung. Allerdings wird die DSML-Schnittstelle diesbzgl. implementiert. Es erfolgt keine Integration etwaiger Aufrufe in den Zugriffskontrollprozess (Hintergrund: Das begrenzte Szenario mit zwei Benutzern und Access Policies erfordert keine zusätzlichen Attribute. Werden feingranulare Access Policies im Unternehmen erstellt bzw. liegen vor, kann dies erforderlich werden).
Autorisierung.	Der Prozess zur Berechtigungserteilung zum Zugriff auf eine Ressource erfolgt im Rahmen der Autorisierung.	Keine Umsetzung: Wie Benutzer autorisiert werden, wird nicht vorgegeben. Es werden vorhandene Access Policies vorausgesetzt, welche die exemplarische Autorisierung zweier Benutzer widerspiegeln.
Zugriffskontrollprüfung.	Prüfung und Umsetzung einer erteilten Berechtigung.	Umsetzung: Diese in der Sicherheitsarchitektur spezifizierte Funktionalität wird vollständig umgesetzt.
Einsatz von TSLs	Einsatz von TSLs zur Sicherung von Dienstendpunkten. Die TSLs im TSL-Depot werden darüber hinaus signiert.	Umgesetzt: Es wird die CTRLCTR-EXTERNAL-TSL unterstützt. Alle weiteren TSL-Varianten werden ausgeklammert. Dies betrifft insbesondere die P23R-Callback-TSL und den Einsatz von GOV-External- und P23R-External-TSLs.

Funktionalität / Architekturvorgabe	Detail	P23R-Musterimplementierung
Nachrichtensignierung zwischen P23R und P23R-Client	Die SOAP-Nachrichten, welche zwischen P23R und P23R-Client ausgetauscht werden, werden mit Hilfe je eines Zertifikats für P23R und P23R-Client signiert.	Wird umgesetzt.

3 Bereitstellung und Bestückung der Laborleitstelle

Als Teil der Präsentationsumgebung (siehe Kapitel 6) wird eine Instanz einer P23R-Leitstelle als Laborleitstelle bereitgestellt, welche über folgenden Zugriffspunkt erreichbar ist:

- https://leitstelle.p23r.de/

Das zugehörige Leitstellenportal ist über folgenden Zugriffspunkt erreichbar:

- https://leitstelle.p23r.de/portal

  Zugriff auf das Leitstellenportal, welches sich auf p23rdemocc (siehe Kapitel 6) befindet.

Die Schnittstellen zum Abruf der Benachrichtigungsregelpakete aus dem Depot (siehe Abschnitt 2.2.1) sind über folgende URL-Schemata erreichbar:

- https://leitstelle.p23r.de/notificationrulepackagedepot/notificationrulepackagelist/
- https://leitstelle.p23r.de/notificationrulepackagedepot/<Identifier NotificationRulePackage>

Die Paketliste der Benachrichtigungsregelpakete wird unter folgendem URL bereitgestellt:

- https://leitstelle.p23r.de/packages.lst

Darüber hinaus wird im Kontext der Laborleitstelle der SFTP-Server für den Empfang von Benachrichtigungen des vDEÜV-Kommunikationskonnektors (siehe Abschnitt 4.2.2) betrieben. Dieser wird unter folgendem URL bereitgestellt:

- ftp://vdeuev.leitstelle.p23r.de/

3.1 TSL für die Laborleitstelle

Im TSL-Depot ist das erforderliche (signierte) TSL abgelegt, dieses ist unter folgender URL erreichbar:

- https://leitstelle.p23r.de/tsl/

In das Depot wird die Wurzel-TSL für die Kommunikation mit der Öffentlichen Leitstelle eingestellt:

- https://leitstelle.p23r.de/p23r.tsl

Der Aufbau der TSLs erfolgt gemäß der Vorgaben der P23R-Sicherheitsarchitektur [3] und beinhaltet das Paketdepot der P23R-Leitstelle und das Zuständigkeitsverzeichnis als Dienst-Endpunkt. Darüber hinaus ist die TSL, wie in Tabelle 25 der Spezifikationen zur P23R-Sicherheitsarchitektur [9] beschrieben, zu signieren.

3.2 Das Testdatenmodell- und Testbenachrichtigungsregelpaket

Auf Basis einer fiktiven jährlichen Statistikmeldung wird ein Testpaket erstellt, mit dem die Kernfunktionalitäten des P23R (Empfehlung, Konfiguration, Auslösen, Selektion und Transformation, Repräsentation, Signierung / Freigabe und Versand) getestet werden können.

Hierzu wird ein Datenmodellpaket mit grundlegenden Unternehmens- und Mitarbeiterinformationen eingesetzt um darauf aufbauend die Anzahl der im Unternehmen tätigen Mitarbeiter an einen fiktiven Verwaltungsempfänger zu übermitteln.

Die Daten werden dabei aus dem generischen Quelldatenkonnektor abgerufen und an den generischen Kommunikationskonnektor übermittelt, um einen vollständigen Durchlauf des Generierungsprozesses aufzuzeigen.

Zudem wird das Testbenachrichtigungsregelpaket in zwei Versionen bereitgestellt, um die Funktionsweise der Verwaltung in mehreren Versionen innerhalb des P23R demonstrieren zu können.

3.2.1 DATENMODELLPAKET TESTMELDUNGEN

Für das Datenmodellpaket „TEST“ ergeben sich daraus folgende Strukturen und Festlegungen. Die Darstellung der technischen Umsetzung der einzelnen Bestandteile in die T-BRS [5] gemäß P23R-Rahmenarchitektur [2] erfolgt in der Feinspezifikation [6]:

- Datenmodellpaket „TEST“ (Testmeldungen)
  - Teildatenmodell „TEST_UNTERNEHMEN” (Namensraum Unternehmen)
  - Teildatenmodell „TEST_MITARBEITER“ (Namensraum Mitarbeiter)

Die Verknüpfung von Mitarbeiter- und Unternehmensinformationen erfolgt dabei analog zum Vorgehen im Datenmodellpaket AGM (siehe Abschnitt 3.3.1).

3.2.2 BENACHRICHTIGUNGSREGELPAKET TESTMELDUNGEN

Resultierend daraus ergeben sich für das Benachrichtigungsregelpaket „TEST“ folgende Strukturen und Festlegungen. Die Darstellung der technischen Umsetzung der einzelnen Bestandteile in die T-BRS [5] gemäß P23R-Rahmenarchitektur [2] erfolgt in der Feinspezifikation [6]:

- Benachrichtigungsregelpaket „TEST“ (Version 1.0)
  - Benachrichtigungsregelgruppe „TEST_STAT“ (Statistikmeldungen)

    Benachrichtigungsregel „TEST_STAT_JAHRESMELDUNG“
- Benachrichtigungsregelpaket „TEST“ (Version 2.0)
  - Benachrichtigungsregelgruppe „TEST_STAT“ (Statistikmeldungen)

    Benachrichtigungsregel „TEST_STAT_JAHRESMELDUNG“

Die Benachrichtigungsregel „TEST_STAT_JAHRESMELDUNG“ soll folgende Prinzipien der T-BRS aufzeigen und deren Anwendbarkeit nachweisen:

- Auslösen der Generierung durch die Terminüberwachung oder optional auf Wunsch durch den Anwender (lokale Nachricht)
- Generierung der Benachrichtigung über mehr als einen Transformationsschritt zum Nachweis der mehrstufigen Generierung auf Basis selektierter Daten
- Verarbeitung von Informationen aus dem Zuständigkeitsverzeichnis und Anwendung eines Repräsentationsskripts für den zurückgelieferten Empfänger
- Durchführen von Konfigurationserstellung und Auswertung von Recommendations auf Basis der Daten im Datenpool des P23R
- Nachweis der Anwendbarkeit der DataSel der T-BRS (vgl. Kapitel 7 in [5]) unter Verwendung eines entsprechenden XQuery-Compilers in der Musterimplementierung

3.2.3 ZUSTÄNDIGKEITSINFORMATIONEN TESTMELDUNGEN

Ebenso sind für das Zuständigkeitsverzeichnis fiktive Zuständigkeitsdaten zu erstellen, um auch diesen Aspekt in die Nachweisführung mit aufzunehmen. Hierzu sollen die mittels eines statischen Parameters zu ermittelnden Empfängerinformationen zur Auswahl des korrekten Repräsentationsskripts und Bestimmung des generischen Konnektors herangezogen werden.

3.3 DAS PILOTDATENMODELL- UND PILOTBENACHRICHTIGUNGSREGELPAKET

Basierend auf den Vorgaben des Lastenhefts [1] erfolgt die Umsetzung der Pilotdatenmodell- und Pilotbenachrichtigungsregelpakete (AGM).

3.3.1 DATENMODELLPAKET ARBEITGEBERMELDUNGEN

Das für den Pilotversuch erforderliche Datenmodellpaket wird auf Basis der im Lastenheft [1] skizzierten Teildatenmodelle für das Pivot-Datenmodell umgesetzt.

Für das Datenmodellpaket „AGM“ ergeben sich daraus folgende Strukturen und Festlegungen. Die Darstellung der technischen Umsetzung der einzelnen Bestandteile in die T-BRS gemäß Rahmenarchitektur [5] erfolgt in der Feinspezifikation [6]:

- Datenmodellpaket „AGM“ (Arbeitgebermeldungen)
  - Teildatenmodell „AGM_UNTERNEHMEN” (Namensraum Unternehmen)
  - Teildatenmodell „AGM_MITARBEITER“ (Namensraum Mitarbeiter)

Dabei ist das Teildatenmodell „Mitarbeiter“ über die Zugehörigkeit eines jeden Mitarbeiters zu einer bestimmten Arbeitsstätte mit dem Datenmodell „Unternehmen“ verknüpft.

TABELLE 24: VERKNÜPFUNG DER TEILDATENMODELLE DES PIVOTDATENMODELLS AGM

Namespace „Unternehmen“	Namespace „Mitarbeiter“
<unternehmen name=“###“ ...> <betriebsstaette bezeichnung=“123“> <arbeitsstaette bezeichnung=“ABC“ /> <arbeitsstaette bezeich-nung=“XYZ“ /> ... </betriebsstaette> ... </unternehmen>	<mitarbeiterdaten> <mitarbeiter> <arbeitsstaette bezeichnung=“ABC“ von=“...“ bis=“...“ /> ... </mitarbeiter> <mitarbeiter> <arbeitsstaette bezeichnung=“ABC“ von=“...“ bis=“...“ /> ... <mitarbeiter> ... </mitarbeiterdaten>

3.3.2 BENACHRICHTIGUNGSREGELPAKET ARBEITGEBERMELDUNGEN

Das für den Pilotversuch erforderliche Benachrichtigungsregelpaket wird auf Basis der im Lastenheft [1] analysierten Meldepflichten in Verbindung mit dem zuvor dargestellten Datenmodellpaket umgesetzt.

Daraus ergeben sich für das Benachrichtigungsregelpaket „AGM“ folgende Strukturen und Festlegungen. Die Darstellung der technischen Umsetzung der einzelnen Bestandteile in die T-BRS [5] gemäß P23R-Rahmenarchitektur [2] erfolgt in der Feinspezifikation [6]:

- Benachrichtigungsregelpaket „AGM“ (Arbeitgebermeldungen)
  - Benachrichtigungsregelgruppe „AGM_STAT“ (Statistikmeldungen)

    Benachrichtigungsregel „AGM_STAT_VERDIENSTERHEBUNG“
  - Benachrichtigungsregelgruppe „AGM_SOZ“ (Sozialversicherung)

    Benachrichtigungsregel „AGM_SOZ_DEUEV-JAHRESMELDUNG“

    Benachrichtigungsregel „AGM_SOZ_BG-JAHRESMELDUNG“

Nicht definiert werden für die einzelnen Benachrichtigungsregeln folgende Bestandteile:

- NotificationView.xslt zur Transformation der Benachrichtigung in eine HTML-Darstellung zur Übergabe an den P23R-Client.
- Die Selektionsskripte werden nicht in DataSel (vgl. Kapitel 7 in [5]), sondern direkt in der Verarbeitungssprache der Musterimplementierung, XQuery, formuliert. Ein Kompiliervorgang – wie bei den Testregeln – wird nicht realisiert.

3.3.3 Zuständigkeitsinformationen Arbeitgebermeldungen

Zur Ermittlung der Zuständigkeitsinformationen im Regelpaket „AGM“ sind im Zuständigkeitsverzeichnis folgende Zuständigkeiten zu hinterlegen:

- Für die Benachrichtigungsregel „AGM_STAT_VERDIENSTERHEBUNG”

  Die mittels eines statischen Parameters zu ermittelnde Dienstadresse des Annahmeverfahrens für den eSTATISTIK.core-Konnektor.
- Für die Benachrichtigungsregel „AGM_SOZ_DEUEV-JAHRESMELDUNG“

  Die mittels der Bezeichnung der Krankenkasse zu ermittelnde Verzeichnisinformation zur Übermittlung der Benachrichtigung mit dem vDEUEV-Konnektor.
- Für die Benachrichtigungsregel „AGM_SOZ_BG-JAHRESMELDUNG“

  Die mittels der Betriebsnummer der Berufsgenossenschaft zu ermittelnde E-Mail-Adresse für den Versand durch den BGviaMail-Konnektor.

4 ABWICKLUNG DES PILOTVERSUCHS IN DER DOMÄNE AGM

Für den Pilotversuch bei der BASF SE und der DATEV e.G. kommt jeweils die im vorausgehenden Kapitel beschriebene P23R-Musterimplementierung (siehe Kapitel 2) zum Einsatz.

Diese wird dazu in die jeweilige Betriebsumgebung der Pilotpartner eingespielt, welche dem Projektteam bereitgestellt wird (siehe Abschnitt 4.3), und an die Laborleitstelle (siehe Kapitel 3) angebunden.

Darüber hinaus werden in die P23R-Musterimplementierung jeweils die erforderlichen Kommunikationskonnektoren integriert, die in Abschnitt 4.2 beschrieben werden.

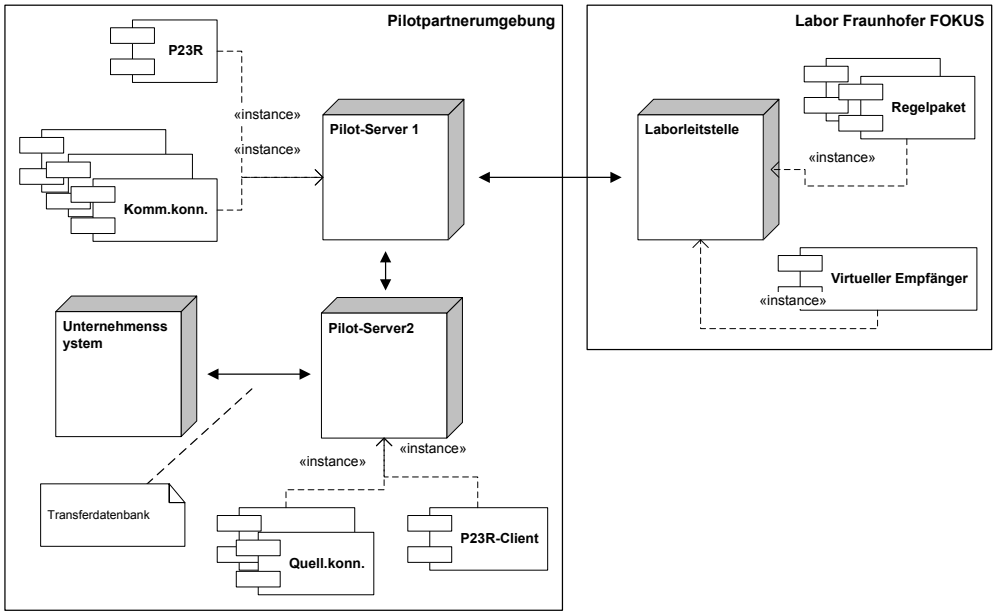


ABBILDUNG 21: ÜBERSICHT PILOTDEPLOYMENT [1]

Über die Laborleitstelle wird dabei das für den Pilotversuch benötigte Datenmodell- und Benachrichtigungsregelpaket und die zugehörigen Zuständigkeitsinformationen (siehe Kapitel 3) bereitgestellt.

In den beiden Versuchsumgebungen müssen jeweils die Unternehmensdaten über einen entsprechenden Quelldatenkonnektor ausgetauscht werden (siehe Abschnitt 4.1).

Im DATEV-Szenario werden zudem alle Benachrichtigungsregeln nach der Aktivierung zur automatischen Freigabe vermerkt, da bei diesem Versuch keine manuelle Freigabe erfolgen soll.

Die eigentliche Durchführung des Pilotversuchs besteht abschließend in der testweisen Durchführung der in Kapitel 5 beschriebenen Testverfahren in der jeweiligen Pilotumgebung.

4.1 AUSTAUSCH DER UNTERNEHMENSDATEN

Die für die Teildatenmodelle des Pivot-Datenmodells erforderlichen Daten müssen über einen entsprechenden Quelldatenkonnektor (SourceConnector) mit der Unternehmensinfrastruktur des jeweiligen Pilotpartners ausgetauscht werden.

Dieser stellt hierzu die erforderlichen Datensatz für den Datenpool zur Abholung bereit (siehe Abschnitt 2.1.5), indem dieser die Schnittstelle ISourceDataRead implementiert.

Zum Einsatz kommt hierbei jeweils eine Transferdatenbank, über die die Daten mit den Unternehmenssystemen ausgetauscht werden.

Der Konnektor bildet hierzu die folgenden Anwendungsfälle ab:

- UC-SA-1: Bereitstellen der Unternehmensdaten

  Die erforderlichen Unternehmensdaten werden hierzu als „Export“ aus dem jeweiligen Unternehmenssystem zur Verarbeitung durch den SourceConnector in einem Transferspeicher bereitgestellt (Transferdatenbank).

- UC-SA-2: Vorhalten des Pivot-Datensatzes

  Die in der Transferdatenbank bereitgestellten Exportdaten werden in Pivot transformiert und für den P23R über den SourceConnector bereitgehalten.

4.1.1 QUELLDATENKONNEKTOR „BASFSAP“

Für den Pilotversuch stellt die BASF eine SAP-HR-Sandbox mit repräsentativen anonymisierten Testdaten für den P23R bereit, deren Abruf im Folgenden näher beschrieben wird (siehe Abbildung 22).

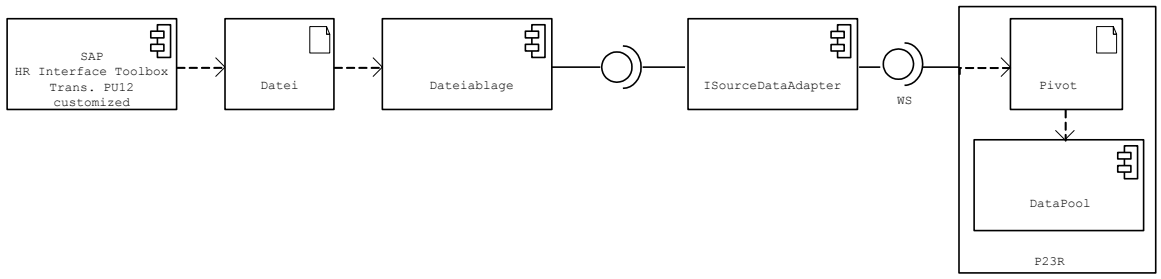


ABBILDUNG 22: SOURCECONNECTOR „BASF.SAP“

Für den Export der Namensräume „Unternehmen“ und „Mitarbeiter“ wird im SAP-HR-System die SAP-HR-Interface-Toolbox (PU12) den entsprechenden Teildatenmodellen des Pivot-Datenmodells angepasst und als Datei in eine Dateiablage übertragen.

Dieser Prozess erfolgt einmal täglich automatisiert über einen SAP-Hintergrund-Job, um sicherzustellen, dass die Daten dem aktuellen Unternehmensstand entsprechen. Zusätzlich ist auch ein manuelles Anstoßen des Exportprozesses möglich.

4.1.1.1 UMSETZUNG DER ANWENDUNGSFÄLLE

Die erforderlichen Anwendungsfälle werden für den Quelldatenkonnektor „BASF.sap“ wie folgt umgesetzt:

- UC-SA-1: Bereitstellen der Unternehmensdaten

  Die Exportdaten der Namensräume „Unternehmen“ und „Mitarbeiter“ werden vom SAP-HR-System über die SAP-HR-Interface-Toolbox exportiert und als SAP-IDoc-Dateien in einem für SAP und den Quelldatenkonnektor beidseitig zugänglichen Dateiordner abgelegt.

  Der Dateiaustausch erfolgt dabei über FTP und wird unterhalb des Home-Verzeichnisses des entsprechenden FTP-Benutzers abgelegt.

Dieser Dateiordner stellt die „Transferdatenbank“ dar, wobei die SAP-IDoc-Daten erst noch vom SourceConnector in die entsprechenden Teildatenmodelle des Pivot-Datenmodells transformiert werden (siehe UC-DA-2).

- UC-SA-2: Vorhalten des Pivot-Datensatzes

  Der SourceConnector transformiert das in der Transferdatenbank abgelegten Datei-Artefakt in einen zu den entsprechenden Teildatenmodellen des Pivot-Datenmodells konformen Datensatz und stellt diese über den spezifizierten WebService dem P23R zur Verfügung.

  Da das IDoc-Format XML-ähnlich aufgebaut ist, wird dieser Transformationsprozess durch eine Vorverarbeitung (Erstellung einer XML-validen Datenstruktur) und eine XSL-Transformation erfolgen.

  Der Datensatz wird nach erfolgter Transformierung im SourceConnector-Speicher für den Abruf durch den P23R bereitgehalten.

4.1.2 QUELLDATENKONNEKTOR „DATEVRAW“

Für den Pilotversuch stellt die DATEV einen Export der relevanten Unternehmensdaten aus dem DATEV-RZ-Datensatz bereit, dessen Abruf im Folgenden näher beschrieben wird (siehe Abbildung 23).

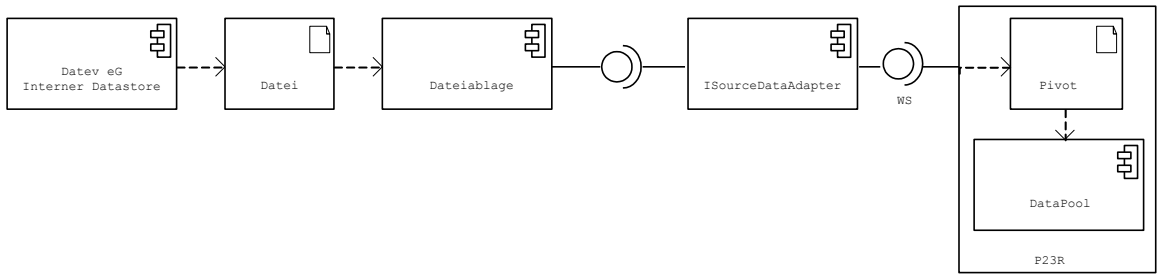


ABBILDUNG 23: SOURCECONNECTOR „DATEV.RAW“

Die Systemlösung der DATEV legt hierzu bei Änderung der zugrundeliegenden Daten ein Update des Datensatzes in einem Transferordner für den P23R ab, den der Quelldatenkonnektor dann weiterverarbeiten kann.

4.1.2.1 UMSETZUNG DER ANWENDUNGSFÄLLE

Die erforderlichen Anwendungsfälle werden für den SourceConnector „DATEV.raw“ wie folgt umgesetzt:

- UC-SA-1: Bereitstellen der Unternehmensdaten

  Die Unternehmensdaten werden von der DATEV-eigenen Systemlösung vorgehalten und regelmäßig in einen für den SourceConnector zugänglichen Dateiordner übertragen.

  Der Dateiaustausch erfolgt dabei über FTP und wird unterhalb des Home-Verzeichnisses des entsprechenden FTP-Benutzers abgelegt.

  Der SourceConnector selbst prüft in kontinuierlichen Abständen das Erzeugungs- bzw. Änderungsdatum der Datei, um Informationen über Änderungen in den Datensätzen zu erhalten.

  Der Dateiablageordner stellt hierbei die „Transferdatenbank“ dar.

- UC-SA-2: Vorhalten des Pivot-Datensatzes

  Der SourceConnector transformiert das in der Transferdatenbank abgelegte Dateiartefakt in einen zu den Teildatenmodellen des Pivot-Datenmodells konformen Datensatz und stellt diese über den spezifizierten WebService dem P23R zur Verfügung.

  Bei den DATEV-Datensätzen handelt es sich um ein XML-strukturiertes internes Format, welches speziell auf die einzelnen Teildatenmodelle des Pivot-Datenmodells gemappt und transformiert wird.

  Der Datensatz wird nach erfolgter Transformierung im SourceConnector-Speicher für den Abruf durch den P23R bereitgehalten.

4.2 Realisierung der Kommunikationskonnektoren

Die Kommunikationskonnektoren (zu denen die Anforderungen in Abschnitt 4.4 des Lastenhefts [1] beschrieben sind) realisieren jeweils das Interface INotificationTransfer der P23R-Rahmenarchitektur [2] und bilden dieses auf das jeweils spezifische Fachverfahren ab.

Für jeden dieser Kommunikationskonnektoren müssen folgende Anwendungsfälle umgesetzt – sprich auf das jeweilige Fachverfahren des Benachrichtigungsempfänger abgebildet – werden:

- UC-TA-1: Übermitteln einer Benachrichtigung

  Hierbei wird die generierte Benachrichtigung durch die Komponenten CommunicationTransport an Hand der ebenfalls übermittelten Profilinformationen über den Kommunikationskonnektor an den Benachrichtigungsempfänger übermittelt.

- UC-TA-2: Erstellen einer Übermittlungsbestätigung

  Immer dann, wenn das Fachverfahren des Benachrichtigungsempfängers dies unterstützt, wird eine Übermittlungsbestätigung an die Komponente CommunicationTransport zurückgemeldet, um diese im Protokoll des P23R abzulegen. Sollte das Fachverfahren dies nicht unterstützen, wird bei positiver Statusrückmeldung des Übertragungsverfahrens ebenfalls eine solche Rückmeldung gegeben.

Folgende Kommunikationskonnektoren müssen für die P23R-Musterimplementierung umgesetzt und in die daraus resultierende P23R-Lösung integriert werden:

- Kommunikationskonnektoren „eSTATISTIKcore“

  Anbindung des Fachverfahrens des statistischen Bundesamtes über die Kommunikationsschnittstelle CORE.connect. Über diesen Konnektor wird die Meldung „vierteljährliche Verdiensterhebung“ übermittelt.

- Kommunikationskonnektoren „vDEÜV“

  Erzeugung eines DEÜV-konformen Datensatzes zur Übermittlung an eine zu konfigurierende virtuelle Kopfstelle („v“ für virtuell). Über diesen Konnektor wird die Meldung „DEÜV-Jahresmeldung zur Sozialversicherung“ übermittelt.

- Kommunikationskonnektoren „BGviaMail“

  Generierung eines PDF-Formulars mit der entsprechenden Jahresmeldung, welches als E-Mail-Anhang an die jeweils zuständige Berufsgenossenschaft übermittelt wird. Über diesen Konnektor wird die Meldung „Jährlicher Lohnnachweis an die Berufsgenossenschaft“ übermittelt.

In den folgenden Abschnitten werden die einzelnen Kommunikationskonnektoren nun basierend auf den vorhergehenden Ausführungen näher beschrieben:

4.2.1 KOMMUNIKATIONSKONNEKTOR „ESTATISTIKCORE“

Der Kommunikationskonnektor „eSTATISTIKcore“ implementiert die Kommunikationsschnittstelle CORE.connect aus dem HTTP-WebService-basierten eSTATISTIK.core-Verfahren des Statistischen Bundesamtes (siehe Abbildung 24) [10].

Über diese Schnittstelle wird der für die Meldung „vierteljährliche Verdiensterhebung“ definierte Datensatz mit der Datensatzkennung 1000107300199 übermittelt. Dieser kann auf der Erhebungsdatenbank des Statistischen Bundesamtes abgerufen werden [11].

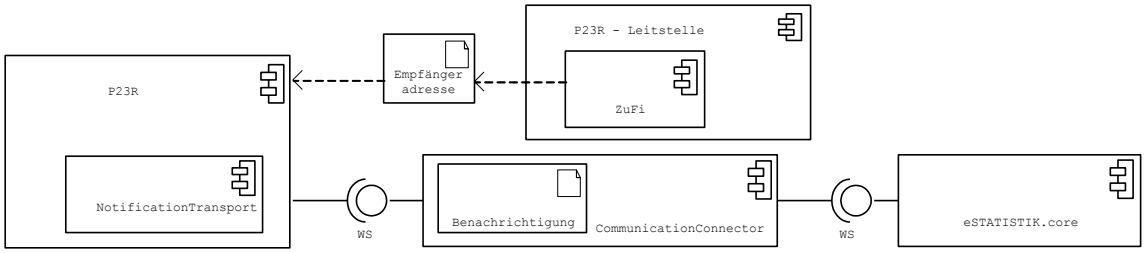


ABBILDUNG 24: KOMMUNIKATIONSKONNEKTOR „ESTATISTIKCORE“ (UML)

Bei der Umsetzung wird darüber hinaus die im eSTATISTIK.core-Verfahren hinterlegte Testkennung eingesetzt, um zu vermeiden, dass durch die Testmeldungen unerwünschtes Handeln auf Seiten des Meldeempfängers ausgelöst wird.

Darüber hinaus identifiziert sich der Konnektor gegenüber dem Verfahren über die vom Statistischen Bundesamt für das Projekt P23R bereitgestellte Testkennung:

- Benutzer: 0000555006
- Passwort: nrMuy5rH

Hierzu wird in der Sektion „*optionen*“ des Meldedatensatzes die Testkennung 100 übergeben, welche die Daten als „nach Eingangs- und Vorprüfung beim Empfänger zu verwerfen“ markiert.

```
<optionen>
<test kennung="100"/>
</optionen>
```

Die genaue Handhabung hierzu ist in den „Verarbeitungsoptionen“ der Liefervereinbarung zur „vierteljährlichen Verdiensterhebung“ beschrieben (Kapitel 3 in [12]).

Darüber hinaus finden sich ergänzende Angaben in Abschnitt 2.4.9.1 der Dokumentation der im eSTATISTIK.core-Verfahren eingesetzten Beschreibungssprache „Data Markup Language (DatML)“ [13].

4.2.1.1 UMSETZUNG DER ANWENDUNGSFÄLLE

Die erforderlichen Anwendungsfälle werden für den Kommunikationskonnektor „eSTATISTIKcore“ wie folgt umgesetzt:

- UC-TA-1: Übermitteln einer Benachrichtigung

  Die Übermittlung erfolgt über das Transportprotokoll HTTPS 1.1 (HTTP 1.1 über SSL) mit der Methode POST und dem Inhaltstyp „*multipart/form-data*“ auf der folgenden Übermittlungs-URL:

  https://www-idev.destatis.de.

  Hierbei wird die Meldung in das HTTP-Post Feld „*daten*“ als binärer Datentyp eingebettet. Dies entspricht der Kommunikationsschnittstellenspezifikation für Dateneingänge des Verfahrens eSTATISTIK.core [14].

- UC-TA-2: Erstellen einer Übermittlungsbestätigung

  Als Übermittlungsbestätigung wird der HTTP-Statuscode im HTTP-Response (bei Fehlern auf Protokollebene erfolgt Statuscode „*200*“) sowie dem empfangenen HTTP-Header „*X-Status*“, welcher den Empfangsstatus kennzeichnet (bei korrekter Übertragung erfolgt Statuscode „*0 = OK*“), eingesetzt.

  Im Erfolgsfall enthält der HTTP-Response zudem einen eindeutigen Empfangsstempel, welcher in Abschnitt 4.2 der eSTATISTIK.core-Verfahrensdokumentation [14] beschrieben ist.

  Die Implementierung des eSTATISTIK.core-Prüfprotokolls wird für den Kommunikationskonnektor nicht umgesetzt, da es sich im Sinne des P23R-Prinzips um den Eingang einer „externen Nachricht“ auf Basis einer gesonderten Benachrichtigungsregel handelt.

  In der Praxis würde dieser Vorgang zudem asynchron mit ca. einem Tag Verzögerung erfolgen, was für den Nachweis der Anwendbarkeit nicht praktikabel ist.

4.2.2 KOMMUNIKATIONSKONNEKTOR „VDEÜV“

Der Kommunikationskonnektor „vDEÜV“ generiert einen DEÜV-konformen Datensatz und übermittelt diesen an eine virtuelle Kopfstelle.

Die virtuelle Kopfstelle wird in der Art simuliert, dass ein FTP-Server für jede reelle Kopfstelle ein entsprechend zu adressierendes Unterverzeichnis bereitstellt, in das die jeweils zugehörigen Meldedaten abzulegen sind (siehe Abbildung 25).

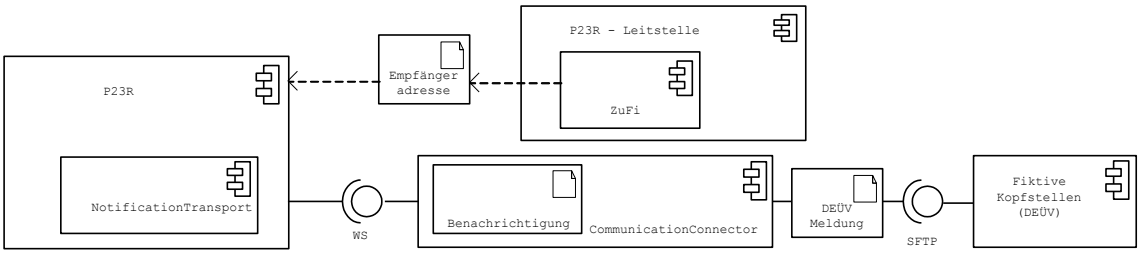


ABBILDUNG 25: KOMMUNIKATIONSKONNEKTOR „VDEÜV“ (UML)

Die Umsetzung der virtuellen Kopfstelle erfolgt hierzu als SFTP-Server, der im Kontext der Laborleitstelle (siehe Kapitel 3) betrieben wird.

Bei der Verwendung des SFTP-Servers handelt es sich um ein spezifiziertes Verfahren der Informationstechnische Servicestelle der gesetzlichen Krankenversicherung (ITSG). Der Kommunikationskonnektor selbst transformiert die Benachrichtigung in das DEÜV-Datenformat und überträgt diese in das korrekte Verzeichnis der virtuellen Kopfstelle.

Die Benachrichtigung wird gemäß der DEÜV-Beschreibung in die Datenbausteine „VOSZ“ (Vorlaufsatz) und dem Datenbaustein „NCSZ“ (Nachlaufsatz) transformiert. Es wird sich dabei auf die Datenbausteine des Abgabegrunds „50“ („Jahresmeldung“) beschränkt, der die „DEÜV-Jahresmeldung zur Sozialversicherung“ abbildet.

Die Abbildung 26 zeigt hierzu die benötigten Datenbausteine auf. Die zugehörige Datensatzbeschreibung kann aus der Dokumentation des DEÜV-Verfahrens entnommen werden [15].

Für die Implementierung des Kommunikationskonnektors kommt Version 1.6.2011 zum Einsatz.

Abgabegrund	DS ME	Datenbausteine DB ME	DB NA	DB GB	DB AN	DB EU	DB UV	DB KS	DB SV	DB VR	DB RG	DB SO
40 Gleichzeitige An- und Abmeldung wegen Ende der Beschäftigung (VSNR liegt vor) bei einem knappschaftlichen Betrieb (Stelle 1-3 im Feld BBNRVU im DSME = 980 oder 098)	J	J	J	J	J	N	m	m	N	N	N	N
40 Gleichzeitige An- und Abmeldung wegen Ende der Beschäftigung (VSNR liegt nicht vor)	J	J	J	J	J	K	m	m	N	N	N	N
49 Abmeldung wegen Tod	J	J	k	N	k	N	m	m	N	N	N	N
50 Jahresmeldung	J	J	k	N	k	N	m	m	N	N	N	N
51 Unterbrechungsmeldung wegen Anspruch auf Entgeltersatzleistungen	J	J	k	N	k	N	m	m	N	N	N	N

Zeichendarstellung:

J = Datenbaustein muss vorhanden sein

N = Datenbaustein darf nicht vorhanden sein

K = Datenbaustein muss vorhanden sein, sofern Daten bekannt sind

k = Datenbaustein kann vorhanden sein (z. B. mehrere Meldegründe)

m = Datenbaustein DBKS muss bei Meldesachverhalten der Personengruppen 140 bis 143 oder 149 oder bei Meldungen knappschaftlicher Betriebe (Stelle 1-3 im Feld BBNRVU im DSME = 980 oder 098) für Personengruppen ungleich 109 und 110 vorhanden sein.

Datenbaustein DBUV muss bei Meldesachverhalten der Personengruppen ungleich 108, 143, 203 bis 205, 207 bis 210 oder 301 bis 305 vorhanden sein.

ABBILDUNG 26: AUSSCHNITT DEÜV DATENBAUSTEINE FÜR ABGABEGRUND 50

Die virtuelle Annahmestelle wird über einen SSH-Tunnel angesteuert, der unter der IP-Adresse „10.42.44.5“ SSH über „193.174.152.221“ bereitgestellt ist. Für den Zugriff auf den eigentlichen FTP-Server wird dann folgende Kombination aus Benutzername und Passwort verwendet:

- Benutzer: deuev
- Passwort: wwzmf2011

Unterhalb des User-Verzeichnisses werden dann die einzelnen Datensätze in jeweils folgendem Verzeichnis abgelegt: ~/vdeuev/[zufi_verzeichniss].

4.2.2.1 UMSETZUNG DER ANWENDUNGSFÄLLE

Die erforderlichen Anwendungsfälle werden für den Kommunikationskonnektor „vDEÜV“ wie folgt umgesetzt:

- UC-TA-1: Übermitteln einer Benachrichtigung

  Die Benachrichtigung wird nach erfolgreicher Transformation als Textdatei über das SFTP-Protokoll (FTP über SSH) zur virtuellen Kopfstelle übertragen.

  Hierbei wird zur Nachbildung der Kopfstellen je ein Meldedatensatz in ein Kopfstellenverzeichnis geschrieben. Die hierfür notwendige Kopfstellenkennung (Zieladresse) wird als Parameter übergeben, der im Zuge der Transportvorbereitung aus dem Zuständigkeitsverzeichnis der Leitstelle ermittelt wurde.

Die Umsetzung der Schnittstellen für die Übermittlung von Dateien mittels FTP ist wie folgt spezifiziert [16]:

- UC-TA-2: Erstellen einer Übermittlungsbestätigung

  Als Übermittlungsbestätigung wird der Zustand des SFTP-Dateitransfers verwendet (bei erfolgreicher Übertragung, sprich ohne Fehler auf Protokollebene, ist der Zustandswert „*OK*“).

  Eine Implementierung des durch das DEÜV-SFTP-Verfahren beschriebenen Bestätigungsverfahrens, welches auf der Erzeugung eines ACK-Files (QUIT-Files) beruht, wird für den Kommunikationskonnektor nicht umgesetzt, da es sich im Sinne des P23R-Prinzips um den Eingang einer „externen Nachricht“ auf Basis einer gesonderten Benachrichtigungsregel handelt [16].

4.2.3 KOMMUNIKATIONSKONNEKTOR „BGVIAMAIL“

Der Kommunikationskonnektor „BGviaMail“ generiert ein PDF-Formular auf Basis der Benachrichtigung und übermittelt dieses als E-Mail-Anhang an den jeweils hinterlegten Empfänger (siehe Abbildung 27).

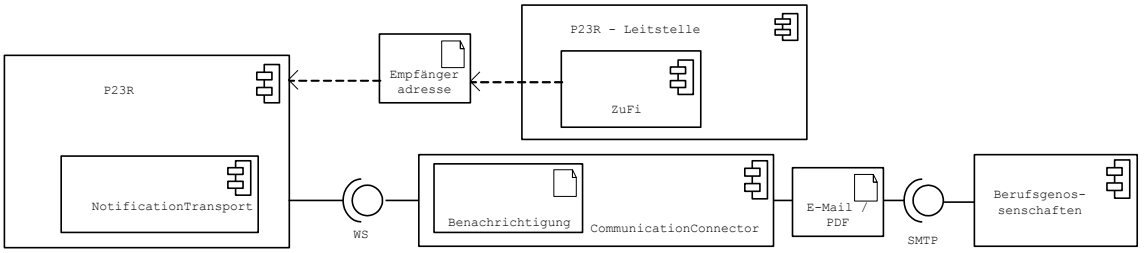


ABBILDUNG 27: KOMMUNIKATIONSKONNEKTOR „BGVIAMAIL“ (UML)

Die hierfür notwendigen Zieladressen werden als Parameter übergeben, die im Zuge der Transportvorbereitung aus dem Zuständigkeitsverzeichnis der P23R-Leitstelle ermittelt wurden.

Die Übermittlung erfolgt für den Pilotversuch an das Postfach „all@leitstelle.p23r.de“, das im Kontext der Laborleitstelle eingerichtet ist und über folgenden Zugang abgerufen werden kann:

- https://w3-isp.com/mail/readmail.html?folder=inbox&id=7b10441d88451bc633bb232eadb7ec2d&get=1
- Username: all@leitstelle.p23r.de

- Password: wwzmf2011

4.2.3.1 UMSETZUNG DER ANWENDUNGSFÄLLE

Die erforderlichen Anwendungsfälle werden für den Kommunikationskonnektor „BGviaMail“ wie folgt umgesetzt:

- UC-TA-1: Übermitteln einer Benachrichtigung

  Die Benachrichtigung wird mittels XSL-FO in ein PDF-Dokument transformiert und als E-Mail-Anhang an die jeweilige Berufsgenossenschaft versendet. Als Transportprotokoll kommt SMTP zum Einsatz, womit der Versand an die als Parameter übergebene Empfangsadresse, welche zuvor aus dem Zuständigkeitsverzeichnis ermittelt wurde, erfolgt.

- UC-TA-2: Erstellen einer Übermittlungsbestätigung

  Aufgrund der Eigenschaften des SMTP-Verfahrens kann nur der Zustand der erfolgreichen Absendung der Nachricht auf Protokollebene ermittelt werden. Der erfolgreiche Empfang beim Empfänger kann aufgrund der Asynchronität (In der Regel mit Hilfe einer E-Mail Bounce Message) nicht ermittelt werden.

4.3 PILOTUMGEBUNG BEI DEN PILOTPARTNERN

Für die Durchführung des Pilotversuchs stellen die Pilotpartner jeweils die in Tabelle 25 beschriebene Infrastruktur innerhalb ihres Rechenzentrums bereit.

TABELLE 25: INFRASTRUKTURBEDARF PILOTPARTNER

Kategorie	Bedarf
Server	Zwei virtuelle oder dedizierte Server mit mindestens je 6 GB Hauptspeicher (besser 8 GB) und je einer Festplattengröße von 50 GB.
Betriebssystem	Nach Möglichkeit: Ubuntu 10.04 oder alternativ aktuelle Debian-Distribution bzw. Red Hat oder openSUSE-System.
SW-Pakete	Oracle Java JDK Version 1.6
Sonstiges	Portöffnung für ein- und ausgehenden Traffic HTTP und HTTPS und VPN oder SSH-Verbindung zur Administration mit entsprechendem Benutzerkonto.

Auf einem der beiden Server wird dabei der P23R selbst, auf dem anderen der P23R-Client und der jeweilige Quelldatenkonnektor installiert. Alle Wartungs- und Installationsarbeiten werden dabei vom Projektteam für den Pilotpartner übernommen.

Darüber hinaus versorgen sie den Transferspeicher der Quelldatenkonnektoren (siehe Abschnitt 4.1) mit den erforderlichen Rohdaten aus dem jeweiligen Unternehmenssystem (z. B. SAP-System).

Das Deployment des P23R selbst erfolgt dabei direkt aus der in Abschnitt 6.2 beschriebenen Integrationsumgebung.

Aus den in Abbildung 21 dargestellten Kommunikationsbeziehungen ergeben sich zudem folgende (auch auf der Firewall) freizugebende Protokolle und Dienste.

TABELLE 26: KOMMUNIKATIONSBEZIEHUNGEN PILOTINFRASTRUKTUR

Quellsystem	Zielsystem	Protokoll / Dienst
Unternehmenssystem	Pilot-Server 2	(S)FTP
Pilot-Server 2	Pilot-Server 1	HTTP(S)
Pilot-Server 1	Pilot-Server 2	HTTP(S)
Pilot-Server 1	Laborleitstelle	HTTP(S), (S)FTP, SMTP
Pilot-Server 1	Internet	HTTP(S), (S)FTP, SMTP
P23R-Team	Pilot-Server 1 / Pilot-Server 2	VPN (IPSec o. ä.)
P23R-Anwender	Pilot-Server 2	HTTP(S)

5 Durchführung von Prüf- und Testverfahren

Die im Lastenheft [1] geforderten Tests werden sowohl auf technischer als auch fachlicher Seite durchgeführt. Auf technischer Seite geschieht dies auf drei Ebenen. Sie sollen das korrekte Arbeiten von Software überprüfbar machen und sicherstellen, dass sie allen gesetzten Anforderungen genügt:

- Unit Tests (Entwickler)

  Unit Tests werden während der Implementierung durch den Entwickler definiert. Sie werden soweit möglich durch entsprechende Tools (z. B. das Continuous Integration Tool Hudson) automatisiert durchgeführt werden. Für Unit Tests muss das Framework TestNG eingesetzt werden. Tests sind vom Entwickler sinnvoll zu definieren und als Bestandteil des Maven Build-Prozesses zu konfigurieren.

- Integrationstests (Entwickler)

  Ein wesentlicher Aspekt des Gesamtsystems ist die Kommunikation zwischen den einzelnen Komponenten. Diese definiert sich durch die Schnittstellen der Komponenten und den möglichen Dialogabläufen. Neben dem manuellen Testen der Integration soll durch eine umfassende Protokollierung sichergestellt werden, dass das reibungslose Zusammenspiel der einzelnen Komponenten nachgewiesen werden kann.

- End-to-End Tests (Team, siehe Abschnitt 5.3)

  Jedes geplante Release muss einer Reihe von End-To-End Tests unterworfen werden, um zu gewährleisten, dass alle Anforderungen an das Release erfüllt wurden. Die erforderlichen End-To-End-Tests sind im Lastenheft [1] vorgeben und werden jeweils durchgeführt.

Auf fachlicher Seite geschieht die Realisierung der Prüf- und Testverfahren mit Hilfe einer Umsetzung des geforderten Testdatensatzes nebst zugehöriger Testergebnisse.

- Fachliche Prüfung der Benachrichtigungsregeln

  Die einzelnen Benachrichtigungsregeln sind mit Hilfe des zur Entwicklung zum Einsatz kommenden XML-Toolsets gegen die vorgegebenen Testergebnisse aus Basis des Testdatensatzes zu prüfen. Dies erfolgt kontinuierlich während der Regelerstellung und vor Einspielen in die P23R-Laborleitstelle für den Pilotversuch.

5.1 Einsatz von Testregeln und Testdatensätzen nebst Testergebnissen

Um die Unit- und Integrationstests untermauen zu können, kommt ein Testregel- und Testdatenmodellpaket auf Basis der im Lastenheft [1] beschriebenen Benachrichtigungsregeln und Teildatensätze des Pivot-Datenmodells zum Einsatz.

Mit Hilfe dieser Pakete können die Komponente zur Verwaltung von Datenmodellen und Benachrichtigungsregeln (siehe Abschnitt 2.1.7) und die Generation-Pipeline umfassend getestet und geprüft werden.

5.2 PROTOKOLLIERUNG RELEVANTER EREIGNISSE INNERHALB DES P23R

Um die Anforderungen zum Nachweis der korrekten Abarbeitung der Benachrichtigungsregeln zu erfüllen, muss durch die P23R-Musterimplementierung eine umfassende Protokollierung der Ereignisse und Vorgänge innerhalb der P23R-Infrastruktur erfolgen.

Dabei werden die in Tabelle 27 dargestellten Ereignisse in den jeweils zugehörigen Komponenten der P23R-Musterimplementierung protokolliert.

TABELLE 27: EREIGNISPROTOKOLLIERUNG JE KOMPONENTEN

Komponenten	Verpflichtender Protokollumfang
Nachrichtenempfang	Annahme einer Nachricht und anschließende Profilerzeugung sowie Übermittlung an die Benachrichtigungsgenerierung.
Benachrichtigungsgenerierung	Transformation der Rohdaten über die einzelnen Transformationsschritte nebst Abruf der relevanten Informationen aus dem Zuständigkeitsverzeichnis.
Benachrichtigungsversand	Abruf einer freizugebenden Benachrichtigung durch den P23R-Client und Status der anschließenden Rückmeldung (Freigabe, Verweigerung).
Benachrichtigungstransport	Übermittlung an den korrekten Kommunikationskonnektor inkl. der zugehörigen Parameter aus dem Zuständigkeitsverzeichnis.
Datenpool	Abruf der geforderten Daten aus dem korrekten Quelldatenkonnektor unter Zuhilfenahme des zugehörigen Selektionsskripts.
Protokollpool	Keine
Datenmodelle und Benachrichtigungsregeln	Abruf eines Datenmodell- oder Benachrichtigungsregelpakets sowie Veränderung des Regelstatus innerhalb des MARM (Aktivieren / Deaktivieren).
Termine und Zeitüberschreitungen	Versand einer Nachricht an den Nachrichteneingang (Auslösen der Benachrichtigungsgenerierung).
Bootstrapping	Keine.
Authentifizierung	Genehmigung oder Verwehrung des Zugriffs.
Access Policies	Abruf der angeforderten Policy aus dem Policy-Store.

5.3 DURCHFÜHREN VON ANWENDUNGS- UND END-TO-END-TESTS

Die im Folgenden aufgelisteten Testfälle müssen auf Basis P23R-Musterimplementierung (siehe Kapitel 2) unter Einsatz der Testdatensätze und Testregeln (siehe Abschnitt 5.1) durchgeführt werden.

Die einzelnen Testfälle orientieren sich an den im Lastenheft [1] beschriebenen Anwendungsfällen und werden durch einen im Umgang mit der P23R-Musterimplementierung geschulten Mitarbeiter durchgeführt.

An die jeweils aufgeführten Testtabellen müssen zur Dokumentation folgende Zeilen angehängt werden (siehe auch Tabellenvorlage im Lastenheft [1]), so dass ein vollständig dokumentierter Testlauf entsteht.

TABELLE 28: TABELLENZEILEN ZUR TESTFALLAUSWERTUNG

Tatsächliches Ergebnis	
Bemerkungen	
PASS / FAIL	
Datum	

5.3.1 ANWENDUNGS-TESTS AM LEITSTELLENPORTAL

Die in den folgenden Tabellen dargestellten Anwendungstests werden am Leitstellenportal der Laborleitstelle durchgeführt.

TABELLE 29: AT-CC-01 AN- UND ABMELDEN EINES BENUTZERS

AT-CC-01	**An- und Abmelden eines Benutzers**
Gegenstand	Das Leitstellenportal der Laborleitstelle.
Vorbedingungen	• Leitstellenportal ist erfolgreich installiert, konfiguriert und für den Anwender erreichbar, sprich über einen Webbrowser abrufbar.
Testprozedur	• Mit Hilfe von Benutzername und Passwort wird der Benutzer am Leitstellenportal angemeldet. • Jede Seite des Leitstellenportals wird einmalig abgerufen um die Autorisierungseinstellungen zu prüfen. • Der Benutzer wird über den entsprechenden Menüpunkt wieder abgemeldet.
Erwartetes Ergebnis	• Der Benutzer kann sich anmelden und alle für ihn sichtbaren Seiten des Leitstellenportals ansteuern.

TABELLE 30: AT-CC-02 VERWALTEN DER BENUTZERKONTEN

AT-CC-02	**Verwalten der Benutzerkonten**
Gegenstand	Das Leitstellenportal der Laborleitstelle.
Vorbedingungen	• Ein Benutzer mit den erforderlichen Benutzerrechten wurde am Leitstellenportal angemeldet.
Testprozedur	• Für die folgenden Testfälle erfolgt jeweils einzeln die Durchführung der Aktion und ein anschließendes „An- und Abmelden“ des jeweiligen Benutzers: ○ Aktivieren und Deaktivieren eines bestehenden Benutzers

AT-CC-02	Verwalten der Benutzerkonten
	○ Zuordnung jeder möglichen Rollenkombination ○ Anlegen eines neuen Benutzers ○ Löschen eine bestehenden Benutzers ○ Ändern der Profilinformationen
Erwartetes Ergebnis	• Mit dem jeweils manipulierten Benutzer lassen sich die durch die Manipulation hervorgerufenen Verhaltensmuster (z. B. Einloggen nicht mehr möglich) aufzeigen.

TABELLE 31: AT-CC-03 VERWALTEN DER REGELPAKETE

AT-CC-03	Verwalten der Regelpakete
Gegenstand	Das Leitstellenportal der Laborleitstelle.
Vorbedingungen	• Ein Benutzer mit den erforderlichen Benutzerrechten wurde am Leitstellenportal angemeldet.
Testprozedur	• Ansteuern der Seite Paketverwaltung im Leitstellenportal. • Hochladen eines Datenmodellpakets und eines Benachrichtigungsregelpakets aus den Testpaketen. • Erzeugen einer aktuellen Version der Paketliste.
Erwartetes Ergebnis	• Eine den eingestellten Paketen entsprechende Paketliste wird seitens der Leitstelle erzeugt und bereitgestellt.

TABELLE 32: AT-CC-04 VERWALTUNG DES ZUSTÄNDIGKEITSVERZEICHNISSES

AT-CC-04	Verwaltung des Zuständigkeitsverzeichnisses
Gegenstand	Das Leitstellenportal der Laborleitstelle.
Vorbedingungen	• Ein Benutzer mit den erforderlichen Benutzerrechten wurde am Leitstellenportal angemeldet.
Testprozedur	• Ansteuern der Seite Zuständigkeitsverzeichnisverwaltung im Leitstellenportal. • Anpassung der Zuständigkeitsinformationen an Hand der Testpakete.
Erwartetes Ergebnis	• Eine den eingestellten Kataloginformationen entsprechende Konfiguration des Zuständigkeitsverzeichnisses wird seitens der Leitstelle bereitgestellt.

5.3.2 ANWENDUNGS-TESTS AM P23R-CLIENT

Die in den folgenden Tabellen dargestellten Anwendungstests werden am P23R-Client durchgeführt und entsprechen zum Teil den Anwendungs-Tests im Rahmen des Pilotversuchs (jeweils in Klammern in den Testfällen vermerkt).

TABELLE 33: AT-CL-01 AN- UND ABMELDEN EINES BENUTZERS

AT-CL-01	An- und Abmelden eines Benutzers (+Pilotversuch)
Gegenstand	Der P23R-Client.
Vorbedingungen	• P23R-Client ist erfolgreich installiert, konfiguriert und für den Anwender erreichbar, sprich über einen Webbrowser abrufbar.
Testprozedur	• Mit Hilfe von Benutzername und Passwort wird der Benutzer am P23R-Client angemeldet. • Jede Seite des P23R-Client wird einmalig abgerufen, um die Autorisierungseinstellungen zu prüfen. • Der Benutzer wird über den entsprechenden Menüpunkt wieder abgemeldet.
Erwartetes Ergebnis	• Der Benutzer kann sich anmelden und alle für ihn sichtbaren Seiten des P23R-Client ansteuern.

TABELLE 34: AT-CL-02 VERWALTEN DER BENUTZERKONTEN

AT-CL-02	Verwalten der Benutzerkonten
Gegenstand	Der P23R-Client.
Vorbedingungen	• Ein Benutzer mit den erforderlichen Benutzerrechten wurde am P23R-Client angemeldet.
Testprozedur	• Für die folgenden Testfälle erfolgt jeweils einzeln die Durchführung der Aktion und ein anschließendes „An- und Abmelden“ des jeweiligen Benutzers: ○ Aktivieren und Deaktivieren eines bestehenden Benutzers ○ Zuordnung jeder möglichen Rollenkombination ○ Anlegen eines neuen Benutzers ○ Löschen eine bestehenden Benutzers ○ Ändern der Profilinformationen
Erwartetes Ergebnis	• Mit dem jeweils manipulierten Benutzer lassen sich die durch die Manipulation hervorgerufenen Verhaltensmuster (z. B. Einloggen nicht mehr möglich) aufzeigen.

TABELLE 35: AT-CL-03 EINSICHT IN DAS P23R-PROTOKOLL

AT-CL-03	Einsicht in das P23R-Protokoll
Gegenstand	Der P23R-Client.
Vorbedingungen	• Ein Benutzer mit den erforderlichen Benutzerrechten wurde am P23R-Client angemeldet.
Testprozedur	• Ansteuern der Protokollanzeige im P23R-Client. • Abgleich der Protokollanzeige mit dem P23R-Protokoll.

AT-CL-03	Einsicht in das P23R-Protokoll
Erwartetes Ergebnis	• Die Protokollanzeige wird angezeigt und entspricht dem Protokollstand aus dem P23R.

TABELLE 36: AT-CL-04 VERWALTUNG DER BENACHRICHTIGUNGSREGELN

AT-CL-04	Verwaltung der Benachrichtigungsregeln
Gegenstand	Der P23R-Client.
Vorbedingungen	• Ein Benutzer mit den erforderlichen Benutzerrechten wurde am P23R-Client angemeldet.
Testprozedur	• Für die folgenden Testfälle erfolgt jeweils einzeln die Durchführung der Aktion und eine anschließende Überprüfung in der Regelverwaltung: ○ Selektieren und Deselektieren eines Benachrichtigungsregelpakets ○ Aktivieren und Deaktivieren einer Benachrichtigungsregel ○ Aktivieren und Deaktivieren einer Regelgruppe
Erwartetes Ergebnis	• Der Status im MARM des P23R entspricht dem jeweils im P23R-Client konfigurierten.

TABELLE 37: AT-CL-05 AUSLÖSEN DER ABARBEITUNG EINER BENACHRICHTIGUNGSREGEL

AT-CL-05	Auslösen der Abarbeitung einer Benachrichtigungsregel (+Pilotversuch)
Gegenstand	Der P23R-Client.
Vorbedingungen	• Ein Benutzer mit den erforderlichen Benutzerrechten wurde am P23R-Client angemeldet.
Testprozedur	• Ansteuern der Seite Regelverwaltung im P23R-Client. • Auslösen einer beliebigen Regel unter folgenden beiden Optionen: ○ Ohne Aktivierung der automatischen Freigabe ○ Mit Aktivierung der automatischen Freigabe
Erwartetes Ergebnis	• Bei aktivierter automatischer Freigabe muss die korrespondierende Benachrichtigungsregel korrekt ausgeführt werden. • Ohne automatische Freigabe muss die korrespondierende Benachrichtigung zur Freigabe vorgelegt werden.

TABELLE 38: AT-CL-06 PFLEGE DER UNTERNEHMENSDATEN

AT-CL-06	Pflege der Unternehmensdaten
Gegenstand	Der P23R-Client.
Vorbedingungen	• Ein Benutzer mit den erforderlichen Benutzerrechten wurde am P23R-Client angemeldet.

AT-CL-06	Pflege der Unternehmensdaten
Testprozedur	• Ansteuern der Seite Nutzerdatenverwaltung im P23R-Client. • Veränderung der Unternehmensdaten: ○ Hinzufügen und Löschen eines Mitarbeiters ○ Umadressierung einer Unternehmens ○ Manipulation der Mitarbeiterdaten
Erwartetes Ergebnis	• Beim Abrufen der Daten durch den Datenpool werden die veränderten Daten korrekt abgerufen und durch die Verarbeitung im P23R verarbeitet.

TABELLE 39: AT-CL-07 BEARBEITUNG UND FREIGABE EINER BENACHRICHTIGUNG

AT-CL-07	Bearbeitung und Freigabe einer Benachrichtigung (+Pilotversuch)
Gegenstand	Der P23R-Client.
Vorbedingungen	• Ein Benutzer mit den erforderlichen Benutzerrechten wurde am P23R-Client angemeldet. • Eine Benachrichtigungsgenerierung wurde durch eine vorgelagerte Auslösung der Benachrichtigungsgenerierung ohne automatische Freigabe ausgelöst.
Testprozedur	• Ansteuern der Seite Benachrichtigungsfreigabe im P23R-Client. • Freigabe / Signierung der freizugebenden Benachrichtigung in folgenden Varianten: ○ Zustimmung ○ Ablehnung • Wiederholung des Vorgangs unter Veränderung / Bearbeitung der generierten Benachrichtigung mit erneut denselben Varianten. • Verlassen der Seite Benachrichtigungsfreigabe im P23R-Client.
Erwartetes Ergebnis	• Korrekte Generierung und Übermittlung der korrespondierenden Benachrichtigungsregel durch den P23R an den „zuständigen“ Kommunikationskonnektor.

6 Anhang I: Darstellung der Entwicklungs- und Laborumgebung

Die Entwicklungsumgebung (als Teil der Laborumgebung) bietet drei unabhängige Infrastrukturen, die die Softwareentwicklung in einem bestimmten Entwicklungsabschnitt unterstützt. Die Software wird dabei stufenweise (siehe Abbildung 28), gemäß ihrem Reifegrad, von einer Infrastruktur zur nächsten migriert.

So steht die erste Umgebung für die tägliche Entwicklung zur Verfügung und enthält im Allgemeinen reine Snapshots, die weder fehlerfrei noch aufeinander abgestimmt sein müssen.

In der nächsten Stufe werden Komponenten installiert, wenn diese nach Plan fertig implementiert wurden und nun im Zusammenspiel mit anderen bereits entwickelten Komponenten getestet werden müssen. Voraussetzung ist, dass alle Komponenten in ihrem Zustand synchronisiert zur Verfügung stehen.

Erst wenn alle Tests in der Test- und Integrationsumgebung erfolgreich sind, wird die Software als Demonstrator-Release freigegeben und in der Demo-Umgebung von außen zugreifbar installiert. Dort kann sie dann zu Kommunikations- und Präsentationszwecken vorgeführt werden.

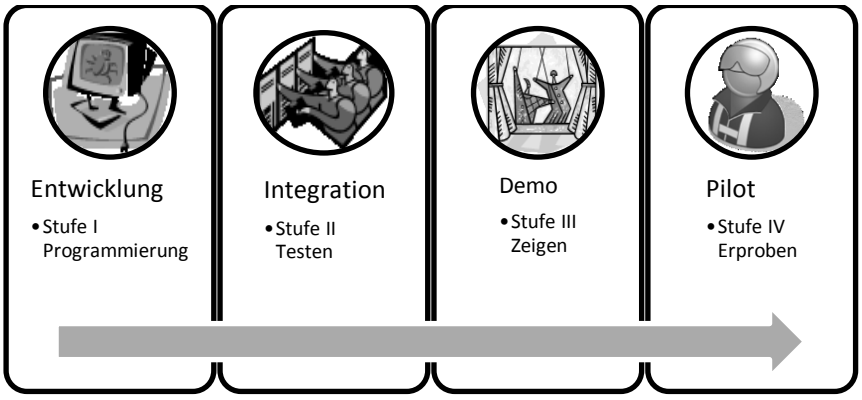


ABBILDUNG 28: STUFENWEISE ENTWICKLUNG UND INTEGRATION

Darüber hinaus werden für den Pilotversuch bei den einzelnen Pilotpartnern externe Installationen durchgeführt. Die Infrastruktur des Labors stellt für diese Installationen wiederum die Leitstelle zur Verfügung.

Grundsätzlich sind die Entwicklungsumgebung sowie die Test- und Integrationsumgebung nicht von außen erreichbar und können nicht ohne weiteres miteinander verbunden werden. Dadurch kann sichergestellt werden, dass durch unachtsame Verknüpfung von Komponenten aus verschiedenen Entwicklungsstufen Debug- oder Testergebnisse verfälscht werden.

Jeder Entwickler hat durch einen eigenen Account direkten Zugang über das SSH-Protokoll [17]. Es ist zudem gewährleistet, dass alle Dienste und Anwendungen vollen Zugriff auf das Internet sowie auf den SVN-Server und das Maven-Repository des Projekts haben (siehe Feinspezifikation [6]).

Alle Server laufen virtuell auf dafür ausgelegter Hardware. Als Betriebssysteme kommt die Ubuntu 10.4 LTE (Linux) Distribution [18] zum Einsatz. Auf Anforderung können aber auch andere Systems auf der VM-Hardware installiert werden.

6.1 ENTWICKLUNGSUMGEBUNG

Im Labor des Fraunhofer FOKUS werden virtuelle Server zur Verfügung gestellt, die für die Entwickler eine komplette Infrastruktur mit Koordinierungsstelle, Unternehmen und der eigentlichen P23R-Lösung darstellen. Die Rolle „Verwaltung“ wird in der Infrastruktur nicht abgebildet.

Die Entwicklungsumgebung stellt im Wesentlichen drei Server zur Verfügung.

- p23rdevmain – Stellt die Laufzeitumgebung für den eigentlichen P23R zur Verfügung. Auf ihm läuft ein JBoss AS [7] mit installiertem JBoss ESB [8].
- p23rdevcc – Stellt die Laufzeitumgebung der Laborleitstelle zur Verfügung. So lange nicht mehr erforderlich ist, läuft auf ihm ein JBoss AS [7].
- p23rdevent - Stellt die Laufzeitumgebung des Unternehmens zur Verfügung. So lange nicht mehr erforderlich ist, läuft auf ihm ein JBoss AS [7].

Der Server p23rdevent ist der einzige Server, der von außen erreichbar ist. Er dient als Gateway, um auf die anderen Server zu gelangen. Jeder Entwickler besitzt einen Account für den Server. Der Zugang erfolgt ausschließlich über SSH [17]. Alle weiteren Verbindungen müssen mit Hilfe von Port-Tunneling über die SSH-Verbindung hergestellt werden.

6.2 INTEGRATIONS- UND TESTUMGEBUNG

Die Integrations- und Testumgebung ist vom Aufbau her identisch mit der Entwicklungsumgebung. Das Kürzel „dev“ im Servernamen wird hier durch das Wort „test“ ersetzt, so dass die drei Server entsprechende Namen haben:

- p23rtestmain – siehe Entwicklungsumgebung
- p23rtestcc – siehe Entwicklungsumgebung
- p23rtestent – siehe Entwicklungsumgebung

Wie in der Entwicklungsumgebung ist nur der „ent“ Server von außen erreichbar und bietet einen Zugang per SSH [17] an.

In gewissen Abständen, die sich an festgelegten Entwicklungszyklen und Releases orientieren, werden in der Test- und Integrationsumgebung die erforderlichen Integrations- und End-to-End Tests durchgeführt. Erst wenn alle Tests bestanden wurden, kann eine Version als Demonstrator in der Demoumgebung zur Verfügung gestellt werden.

6.3 PRÄSENTATIONSUMGEBUNG

In der Demoumgebung (Stage) werden nur Versionen installiert, die vollständig erfolgreich getestet und für stabil befunden wurden. Das heißt, die Software muss vorher in der Test- und Integrationsumgebung alle Tests bestehen.

Die Präsentationsumgebung ist vom Aufbau her identisch mit der Integrations- und Testumgebung. Das Kürzel „test“ im Servernamen wird hier durch das Wort „demo“ ersetzt, so dass die drei Server entsprechende Namen haben:

- p23rdemomain – siehe Entwicklungsumgebung
- p23rdemocc – siehe Entwicklungsumgebung
- p23rdemoent – siehe Entwicklungsumgebung

Wie in der Entwicklungsumgebung ist nur der „ent“ Server von außen erreichbar und bietet einen Zugang per SSH an. Zusätzlich ist gewährleistet, dass alle nötigen Zugänge freigeschaltet sind, so dass eine Demonstration des Systems von außen her jederzeit möglich ist.

7 GLOSSAR

Access Policy

Eine Access Policy ist ein Regelwerk, aus dem sich Entscheidungen herleiten lassen. Im Rahmen der P23R-Sicherheitsarchitektur werden Access Policies zur Kodierung von Berechtigungen (Berechtigungspolicies) und zur Steuerung des Dienstzugangs (Sicherheitspolicies) verwendet.

Adapter

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

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.

Authentisierung

Unter einer Authentisierung versteht man die Vorlage eines Nachweises eines Kommunikationspartners, in dem bestätigt wird, dass er tatsächlich derjenige ist, der er vorgibt zu sein.

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.

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.

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: [20]

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.

Benachrichtigungspool (Notification Pool)

Speicher und Archiv für die Benachrichtigungen, in dem alle generierten Benachrichtigungen nachweisbar abgelegt werden. Die freigegebenen Benachrichtigungen werden aus dem Pool an den ermittelten Benachrichtigungsempfänger versendet.

Benachrichtigungsprofil (Notification Profile)

Das Benachrichtigungsprofil enthält alle Metadaten, die intern vom P23R und den Benachrichtigungsregeln zur Steuerung bei der Erzeugung und dem Versand einer Benachrichtigung benötigt werden.

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.

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 vorgegeben 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.

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: [21]

Domäne (Fachdomäne)

Die Abgrenzung eines Themenbereiches für die Regelerstellung wird im Kontext des P23R-Prinzips als „Fachdomäne“ (kurz: „Domäne“) bezeichnet. Die Abgrenzungskriterien sind unter-

schiedlicher 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.

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.

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.

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

Siehe auch *Fachübergreifende Koordinierungsaufgaben*.

Genehmigung

Eine Genehmigung ist das Ziel eines Antragsprozesses (siehe Antrag).

Identität

Eine Identität im Sinne der IT-Sicherheit ist die Summe der die Eigentümlichkeit einer Entität erfassenden Merkmale (Attribute und Werte). Die Identität eines Nutzers kann so z. B. über eine eindeutige Nummer innerhalb eines definierten Wertesystems (z. B. Sozialversicherungsnummer) oder über demografische Attribute (Name, Geburtstag, Geburtsort etc.) erfasst werden.

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: [22]

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 Informationsverbünden 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: [22]

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: [22]

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 kommunikationstreibenden Unternehmen bewusst zur Erreichung kommunikativer Zielsetzungen eingesetzt werden.

Quelle: [23]

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: [24]

Komponente

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

Quelle: [22]

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 Benachrichtigungsregelpaketen, 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.

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: [22]

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.

Nachrichtentyp (Message Type)

Der P23R empfängt unterschiedliche Nachrichten in einem XML-Format von externen Quellen, die die Generierung spezifischer Benachrichtigungen im P23R auslösen. Jedem dieser XML-Formate kann ein Nachrichtentyp zugeordnet werden. Ein Nachrichtentyp wird durch einen eindeutigen Namen in Form des standardmäßigen XML-Namensraums identifiziert, der in dem XML-Dokument verwendet wird.

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

Quelle: [25]

Notfallkonzept

Das Notfallkonzept umfasst das Notfallvorsorgekonzept und das Notfallhandbuch.

Quelle: [25]

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-Identity-Provider

Siehe Identity Provider.

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-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-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 Unterneh-

menscharakteristik 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 (Berechtigungspolicies) 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 Berechtigungspolicies.

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.

Policy Information Point (PIP)

Ein Policy Information Point erlaubt einen on-demand Abruf von Attributwerten, die zur Auswertung einer Policy erforderlich sind. Ein Policy Information Point fungiert dabei als Zugang zu bestehenden Informationssystemen im Unternehmen.

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 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. [26]). 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. Analysekriterien 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: [27]

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.

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 Transformationsskripte. 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 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: [22]

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: [22]

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: [22]

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: [22]

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: [28]

Trusted Service List (TSL)

Zur Bekanntgabe von Zertifikaten und Schnittstellen vertrauenswürdiger Dienste werden entsprechende Dokumente als Trusted Service List publiziert. Syntax und Semantik dieser Dokumente werden durch den Standard ETSI TS 102 231 [29] definiert.

Unified Modeling Language

UML (Unified Modeling Language) ist ein Standard der OMG (http://www.omg.org/uml) und definiert Notation und Semantik zur Visualisierung, Konstruktion und Dokumentation von Modellen für die Geschäftsprozessmodellierung und für die objektorientierte Softwareentwicklung. Im Sinne einer Sprache definiert UML dabei Bezeichner für die meisten der bei einer Modellierung wichtigen Begriffe und legt mögliche Beziehungen zwischen diesen Begriffen fest. UML definiert weiter grafische Notationen für diese Begriffe und für Modelle statischer Strukturen und dynamischer Abläufe, die man mit diesen Begriffen formulieren kann.

Quelle: [30]

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: [22]

Verschlüsselung

Verschlüsselung (Chiffrieren) transformiert einen Klartext in Abhängigkeit von einer Zusatzinformation, die "Schlüssel" genannt wird, in einen zugehörigen Geheimtext (Chiffrat), der für diejenigen, die den Schlüssel nicht kennen, nicht entzifferbar sein soll. Die Umkehrtransformation - die Zurückgewinnung des Klartextes aus dem Geheimtext - wird Entschlüsselung genannt.

Quelle: [22]

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: [22]

Verzeichnisdienst

Ein Verzeichnisdienst ist ein Infrastrukturdienst, der Informationen (Attributwerte) zu hierarchisch strukturierten Entitäten eines Typs zur Verfügung stellt.

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.

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 auszugestalten 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: [22]

Zertifikat

Ein (digitales) Zertifikat bindet mit Hilfe einer digitalen Signatur einen öffentlichen Schlüssel an eine Identität. Eine digitale Signatur, die gegen diesen öffentlichen Schlüssel geprüft werden kann, ist damit der an diesen Schlüssel gebundenen Identität zuzuordnen. Zertifikate bilden Hierarchien, an deren Spitze ein von einer vertrauenswürdigen Stelle selbst zertifiziertes Zertifikat steht.

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: [24]

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: [22]

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.

8 ABKÜRZUNGSVERZEICHNIS

AGM	Arbeitgebermeldepflichten
AT	Anwendungstest
BMI	Bundesministerium des Innern
BG	Berufsgenossenschaft
BR	Benachrichtigungsregel
BRG	Benachrichtigungsregelgruppe
BRS	Benachrichtigungsregelsprache
BRP	Benachrichtigungsregelpaket
BSI	Bundesamt für Sicherheit in der Informationstechnik
CC	Control-Center
CDI	Context and Dependency Injection
CL	P23R-Client
DEÜV	Datenerfassungs- und Übermittlungsverordnung
DSML	Directory Service Markup Language
e.G.	eingetragene Genossenschaft
ESB	Enterprise Service Bus
ESI	Electronic Signatures and Infrastructures
ETSI	European Telecommunications Standards Institute
F-BRS	Fachliche Benachrichtigungsregelsprache
FTP	File Transfer Protocol
HTML	Hypertext Markup Language
HTTP(S)	Hyper Text Transfer Protocol (Secure)
IP	Internet Protocol
IT	Information Technology
ITSG	Informationstechnische Servicestelle der gesetzlichen Krankenversicherung
JAX-WS	Java API for XML Web Services
JDK	Java Development Kit
MARM	Model And Rule Management
MLF	Methodenleitfaden (Online)
NCSZ	Nachlaufsatz
OMG	Object Management Group
P23R	Prozess-Daten-Beschleuniger
PAP	Policy Administration Point
PDF	Portable Data Format
PDP	Policy Decision Point
PEP	Policy Enforcement Point
PIP	Policy Information Point
PRK	Prozesskette
PRTR	Pollutant Release and Transfer Register
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
SSH	Secure-Shell
SVN	Apache Subversion
TA	Target Application
T-BRS	Technische Benachrichtigungsregelsprache
TSL	Trusted Service List
UC	Use Case
UI	User Interface
UML	Unified Modeling Language
URI	Unified Resource Identifier
URL	Unified Resource Locator
V	Version
VM	Virtual Machine
VOSZ	Vorlaufsatz
WS	Web Service
XML	Extensible Markup Language
XSD	XML Schema Definition
XSL-FO	Extensible Stylesheet Language – Formatting Objects

9 REFERENZEN

Alle in diesem Kapitel aufgeführten Ergebnisdokumente des P23R-Projekts werden unter www.p23r.de bereitgestellt werden.

[1] S. Dutkowski, E. Klochkova und A. Söllner (2013), *P23R: Lastenheft zu Szenarien und Datenmodellen.* (Ergebnisdokument des P23R-Projekts)

[2] J. Gottschick et al. (2012), *P23R: Rahmenarchitektur*. (Ergebnisdokument des P23R-Projekts)

[3] J. Caumanns et al. (2012), *P23R: Sicherheitsarchitektur*. (Ergebnisdokument des P23R-Projekts)

[4] R. Rosenmüller et al. (2012), *P23R: Spezifikationen zur Rahmenarchitektur*. (Ergebnisdokument des P23R-Projekts)

[5] J. Gottschick (2012), *P23R: Spezifikation der Technischen Benachrichtigungsregelsprache.* (Ergebnisdokument des P23R-Projekts)

[6] S. Dutkowski, E. Klochkova und A. Söllner (2013), *P23R: Spezifikation zur Infrastruktur.* (Ergebnisdokument des P23R-Projekts)

[7] *JBoss Application Server*. Verfügbar unter: http://jboss.org/jbossas (zuletzt abgerufen am 18.10.2011).

[8] *JBoss ESB – Reliable SOA infrastructure*. Verfügbar unter: http://jboss.org/jbossesb (zuletzt abgerufen am 18.10.2011).

[9] J. Caumanns et al. (2012), *P23R: Spezifikationen zur Sicherheitsarchitektur*. (Ergebnisdokument des P23R-Projekts)

[10] *Entwicklerportal des Statistischen Bundesamts*. Verfügbar unter: http://www.statspez.de/core/entwickler.html (zuletzt abgerufen am 21.10.2011).

[11] Statistisches Bundesamt, *Erhebungsdatenbank des Statistischen Bundesamts*. Verfügbar unter: https://erhebungsdatenbank.destatis.de/eid/index.html (zuletzt abgerufen am 18.10.2011).

[12] Statistisches Bundesamt, *Liefervereinbarung zur vierteljährlichen statistischen Verdiensterhebung.* Verfügbar unter: https://erhebungsdatenbank.destatis.de/eid/download.html?download=100010730019998000002 (zuletzt abgerufen am 28.07.2011).

[13] Statistisches Bundesamt, *Beschreibung der Data Markup Language (DatML).* Verfügbar unter: http://www.statspez.de/core/Downloads/DatML/raw/v2_0/datml-raw-2_0-spezifikation.pdf (zuletzt abgerufen am 28.07.2011).

[14] Statistisches Bundesamt, Die *Kommunikationsschnittstelle des gemeinsamen Dateneingangs von eSTATISTIK.core.* Verfügbar unter: http://www.statspez.de/core/Downloads/Connect/Beschreibung_CORE_Schnittstelle.pdf (zuletzt abgerufen am 28.07.2011).

[15] GKV Spitzenverband et al., *Dokumentation des DEÜV-Verfahrens i.B. Datensatzbeschreibung.* Verfügbar unter http://www.gkv-datenaustausch.de/upload/2011-06-01_GG28b.zip (zuletzt abgerufen am 28.072.2011).

[16] GKV Spitzenverband, *Spezifikation der Schnittstellen zur Kommunikation mittels File Transfer Protokoll.* Verfügbar unter http://www.gkv-ag.de/upload/TA_FTP_V_1.2_9431_3581.pdf (zuletzt abgerufen am 29.04.2011).

[17] *SSH – Secure Shell*. Verfügbar unter: http://de.wikipedia.org/wiki/Secure_Shell (zuletzt abgerufen am 18.10.2011).

[18] *Ubuntu*. Verfügbar unter: http://www.ubuntu.com/ (zuletzt abgerufen am 18.10.2011).

[19] *Statistische Ämter des Bundes und der Länder*. Verfügbar unter: www.statspez.de/core/ (zuletzt abgerufen am 11.05.2011).

[20] 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.pdf (zuletzt abgerufen am 18.10.2011).

[21] 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.

[22] 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).

[23] M. Bruhn (2007), *Kommunikationspolitik*, 4. überarbeitete Auflage, Verlag Franz Vahlen GmbH, München.

[24] Gabler Verlag (Hrsg.), *Gabler Wirtschaftslexikon*. Verfügbar unter: http://wirtschaftslexikon.gabler.de/ (zuletzt abgerufen am 07.11.2012).

[25] 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).

[26] N. Fröschle 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).

[27] 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).

[28] R. Olbrich (2009), *Marketing – Eine Einführung in die marktorientierte Unternehmensführung*, 2. Auflage, Springer-Verlag GmbH, Heidelberg.

[29] ETSI Technical Committee Electronic Signatures and Infrastructures (ESI) (2009), *ETSI TS 102 231 Version 3.1.2 - Electronic Signatures and Infrastructures (ESI): Provision of harmonized Trust-service status Information*, ETSI. Verfügbar unter: http://www.etsi.org/deliver/etsi_ts/102200_102299/102231/03.01.02_60/ts_102231v030102p.pdf (abgerufen am 01.11.2011).

[30] *UML Version 2.3 Spezifikationen*. Verfügbar unter: http://www.omg.org/spec/UML/2.3/Infrastructure/PDF und http://www.omg.org/spec/UML/2.3/Superstructure/PDF (zuletzt abgerufen am 23.09.2011).

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

www.p23r.de