



## **R016 – Beilage 1: Erläuterungen zu den SOA-Policies**

|                  |                               |
|------------------|-------------------------------|
| Klassifizierung: | Nicht klassifiziert           |
| Typ:             | Beilage zu einem IKT-Standard |
| Ausgabedatum:    | 2016-06-13                    |

# Inhaltsverzeichnis

|          |                                               |                                    |
|----------|-----------------------------------------------|------------------------------------|
| <b>1</b> | <b>Einleitung</b>                             | <b>3</b>                           |
| <b>2</b> | <b>Handlungsfelder</b>                        | <b>4</b>                           |
| 2.1      | Strategie                                     | 4                                  |
| 2.2      | Finanzierung                                  | 4                                  |
| 2.3      | Service-Bereitstellung                        | 4                                  |
| 2.4      | Service-Nutzung                               | 5                                  |
| 2.5      | Serviceverzeichnis und -Dokumentation         | 7                                  |
| 2.6      | Referenz-Architekturen                        | 7                                  |
| 2.6.1    | Operational System Layer                      | 9                                  |
| 2.6.2    | Service Component Layer                       | 9                                  |
| 2.6.3    | Service Layer                                 | 9                                  |
| 2.6.4    | Business Process Layer                        | 9                                  |
| 2.6.5    | Consumer Layer                                | 10                                 |
| 2.6.6    | Integration Layer                             | 10                                 |
| 2.6.7    | Quality of Service- Layer                     | 10                                 |
| 2.6.8    | Information Layer                             | 10                                 |
| 2.6.9    | Governance Layer                              | 11                                 |
| 2.7      | Infrastruktur                                 | 11                                 |
| 2.8      | Knowhow                                       | 11                                 |
| 2.9      | Prozesse, Projektabwicklung                   | 11                                 |
| <b>3</b> | <b>Details zu den einzelnen Policies</b>      | <b>13</b>                          |
| 3.1      | Rollenmodell (Policy S1)                      | 13                                 |
| 3.1.1    | Rolle Service-Anbieter                        | 13                                 |
| 3.1.2    | Rolle Service-Betreiber                       | 14                                 |
| 3.1.3    | Rolle Service-Nutzer                          | 14                                 |
| 3.1.4    | Rolle Anwendungsbetreiber                     | 14                                 |
| 3.1.5    | Services-Governance                           | 14                                 |
| 3.2      | Vorgaben (Policy S2)                          | 15                                 |
| 3.3      | Serviceverzeichnis (Policy S3)                | 16                                 |
| <b>4</b> | <b>SOA-Prinzipien</b>                         | <b>18</b>                          |
| 4.1      | Fachliche Unabhängigkeit (SOA-Prinzip 1)      | 18                                 |
| 4.2      | Technologische Unabhängigkeit (SOA-Prinzip 2) | 18                                 |
| 4.3      | Abstraktion (SOA-Prinzip 3)                   | 19                                 |
| 4.4      | Wiederverwendbarkeit (SOA-Prinzip 4)          | 19                                 |
| 4.5      | Statuslosigkeit (SOA-Prinzip 5)               | 19                                 |
|          | <b>Anhänge</b>                                | <b>20</b>                          |
| A.       | Abkürzungen                                   | 20                                 |
| B.       | Glossar der grundlegenden Begriffe            | 21                                 |
| C.       | Glossar technischer Begriffe                  | 22                                 |
| D.       | Referenzen                                    | 23                                 |
| E.       | Bedeutung der Status-Angabe                   | Fehler! Textmarke nicht definiert. |

# 1 Einleitung

Services gemäss SOA (Service-Orientierter Architektur) stellen Information und Funktionalität einem grösseren Kreis von Anwendungen zur Verfügung. So können viele Nutzer auf Services eines Anbieters zugreifen. Hierzu wurden bereits im Jahr 2008 mit dem Dokument „R016 - SOA-Policies“ Leitlinien für die Bereitstellung von SOA-Services erstellt und in Kraft gesetzt worden, die nun aktualisiert werden. Verschiedenen Firmen und Organisationen setzen Services inzwischen erfolgreich ein, dabei zeigt es sich, dass so Synergien genutzt, Redundanzen vermieden und Kosten gespart werden können.

Die hauptsächliche Zielsetzung für den Einsatz von Services bei der Bundesverwaltung ist, dass ein Amt Information und Funktionalität ihrer Domäne der Allgemeinheit – intern und/oder extern – zur Verfügung stellt. Dadurch, dass Information und Funktionalität nur an einem Ort gepflegt werden müssen, entstehen Synergien mit einem grossen Nutzen-Potential sowie Kostenreduktionen. Die folgende Grafik illustriert, wie die Organisationen A und B die Services der Organisation C benutzen können und nicht mehr selbst für den Unterhalt ihrer Applikationen A und B inklusive der Pflege der dazu gehörenden Datenbestände aufkommen müssen.

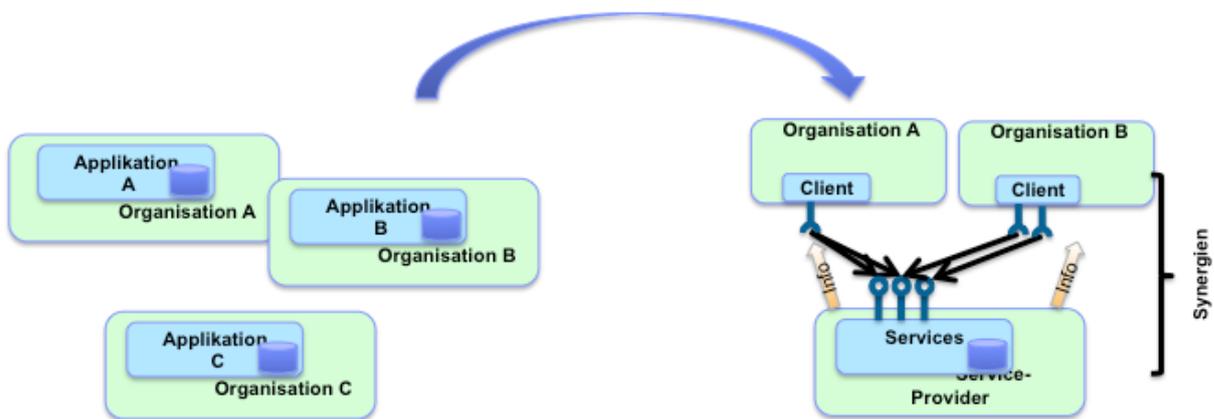


Abbildung 1: Illustration Synergien

Services werden als Bestandteil einer fachspezifischen IKT-**Leistung** dezentral gemäss den üblichen Projekt-Methodiken erstellt und mit üblichen betrieblichen Abläufen erbracht.

**Vorteil Nutzer:** „Fremde“ Services dienen als Baustein einer eigenen Lösung und dies erspart Entwicklungsaufwand sowie Betriebskosten. Da die „fremden“ Services gut unterhalten sind (denn sie sind in der Kernkompetenz des Anbieters) wird die Lösung effizienter und verlässlicher, oft entfällt auch der Aufwand zur Pflege redundanter Datenbestände.

**Vorteil Anbieter:** Wenn andere Nutzer vorhanden sind, entsteht ein besserer ROI für den Entwicklungsaufwand von Services. Das bedingt jedoch eine holistische Betrachtung über die Grenzen einer Verwaltungseinheit hinaus. Services mit modularen Paketen sind besser wartbar und haben einen nachhaltigeren Lebenszyklus.

Gestützt auf Erfahrungen der letzten Jahre ist nun eine Revision der SOA-Policies durchgeführt worden. In diesen „**Erläuterungen**“ werden einerseits mittels Handlungsfelder der Kontext der Richtlinien beschrieben und andererseits weitere Details zu den SOA-Policies selbst erläutert.

## 2 Handlungsfelder

Mit Repräsentanten verschiedener Verwaltungseinheiten sind Erfahrungen bei der Bereitstellung und Nutzung von Services in mehreren Workshops besprochen und folgende Handlungsfelder identifiziert und diskutiert worden:

- Anreize und Finanzierung, Investment & ROI
- Kultur der „Unternehmung Bund“
- Ressourcen: Infrastruktur & Know-how
- Serviceverzeichnis und -Dokumentation
- Terminologie und Prinzipien
- SOA-Rollenmodell
- Prozesse, Projektabwicklung
- Strategische Vorgaben – in Abstimmung mit IKT-Strategie Bund
- Referenz-Architekturen
- Technologische Vorgaben
- Nutzung von technisch vorhandenen Services (Governance)

Wesentliche Aspekte dieser Handlungsfelder sind in der Folge in leicht umstrukturierter Form ausgeführt und beschrieben.

### 2.1 Strategie

Gemäss der **IKT-Strategie des Bundes 2016–2019** [IKTS] ist eines der Ziele der strategischen Stossrichtungen „S01 – Geschäftsausrichtung der IKT“ die organisationsübergreifende Zusammenarbeit. Es sollen organisationsübergreifend Synergien genutzt werden. [IKTS, S. 13]. Eine Form der Synergienutzung besteht darin, dass Verwaltungseinheiten Funktionalität und Information ihres Kompetenzbereichs anderen Organisationen und gegebenenfalls auch der Allgemeinheit in geeigneter Form zur Verfügung stellen. Services im Sinne der service-orientierten Architektur sind ein bewährtes Instrument hierfür.

### 2.2 Finanzierung

Es macht einen Unterschied, ob Services für einen grossen externen Anwenderkreis bereitgestellt oder nur lokal verwendet werden. Die Bereitstellung von breit genutzten Services ist aufwändiger. Die Anforderungen an die Verfügbarkeit sind in der Regel höher. Die Mechanismen, wie die dadurch verursachten Mehrkosten zu finanzieren sind, sind nicht Gegenstand der vorliegenden Policies.

### 2.3 Service-Bereitstellung

Services eines Amtes können unterschiedlich charakterisiert werden, als **öffentliche Services**, die frei nutzbare Daten und Funktionen verfügbar machen sowie als **geschützte Services**, die nur in Absprache mit dem Service-Anbieter und unter Einhaltung von Security-Mechanismen benutzt werden können. Vom ISB bereit gestellte Standard-Services können Services enthalten. Dies kann wie folgt detaillierter umschrieben werden:

- **Geschützte Services** dienen im Wesentlichen der Integration von Information und Funktionalität für andere Nutzer. In diesem Fall kennt der Service-Anbieter die Service-Nutzer und trifft mit ihnen entsprechende Vereinbarungen, insbesondere im Bereich Security.  
*Beispiel: EJPD Services, die von der Polizei (auch der Kantone) genutzt werden.*

- **Öffentliche Services** können von allen genutzt werden – möglicherweise mit der Einschränkung, dass Nutzer bekannt oder registriert sind, um z.B. eine Verrechnung der Leistung zu ermöglichen. Öffentliche Services müssen mit den nötigen Qualitäten betrieben werden. Der Service-Anbieter kennt den Service-Nutzer unter Umständen nicht; die Service-Bereitstellung und Service-Nutzung sind völlig entkoppelt.  
*Beispiel: Services für den Bezug von Daten aus Verzeichnissen z.B. über Firmen, Postleitzahl-Services, AHV-Services, Stammdaten-Services (z.B. für Länder-Abkürzungen oder Währungen)*
- Öffentliche und geschützte Services sollen von den Verwaltungseinheiten angeboten werden. Auch vom ISB geführte **Standarddienste** können Services im Sinne der Service-Orientiertheit enthalten. Es gibt jedoch auch Standarddienste, die keine Services in diesem Sinne enthalten.  
*Beispiel: Services von IAM sind Standarddienste, DAKO ist ein Standarddienst ohne Services.*
- Die Bereitstellung von Services wird somit zu einer **Leistung**, die ein Amt im Interesse eines grösseren erbringt. Auch kann eine Service-Gruppe Bestandteil eines **Produkts** sein.  
*Beispiel: Das Produkt UID-Register stellt auch einige Services bereit.*

Angaben sowohl für öffentliche als auch geschützte Services müssen in einem Serviceverzeichnis verwaltet werden. Während bei geschützten Services die Nutzer über Veränderungen und Weiterentwicklungen direkt informiert werden können, sind für öffentliche Services andere Mechanismen vorzusehen, da sie ohne Kenntnis des Anbieters genutzt werden können. Ein weiterer, von der Charakterisierung unabhängiger Aspekt ist die Bezahlung der Leistungserbringer. Sowohl öffentliche als auch geschützte Services können mit oder ohne Kostenfolge genutzt werden.

Die Bereitstellung von Services verbleibt weitgehend in der Kompetenz der betreffenden Ämter. Es wird als sinnvoll erachtet, dass die Ämter ihre „Assets“ selbst bewirtschaften und als Services anbieten. Deshalb ist es wichtig, die Ämter entsprechend zu unterstützen: methodisch bei der Identifikation der Services, finanziell für die Entwicklung und Erstellung der Services sowie für den Betriebs-Infrastruktur (eigene Rechner oder Clouds). Das bedeutet, dass den Ämtern im Rahmen einer IKT-Gesamtplanung Ressourcen und Mittel zur Verfügung gestellt werden müssen, die einerseits Projekte zum Etablieren und Bereitstellen entsprechender Service-Gruppen und andererseits den Betrieb stark nachgefragter Services finanzieren.

Die Begleitung durch das ISB ermöglicht aus einer übergeordneten Perspektive Service-Gruppen zu identifizieren und spezifizieren, sodass das Portfolio der Services gesamthaft über die Bundesverwaltung gesehen Synergien fördert, Redundanzen vermeidet und Kosten reduziert.

## 2.4 Service-Nutzung

Die Nutzung von Services kann sehr vielfältig sein und hängt natürlich auch davon ab, ob es öffentliche oder geschützte Services sind. Während einige Services spezifisch in eine Anwendung des Nutzers eingebunden werden (und u.U. in Zusammenarbeit mit dem Service-Betreiber auch in allen Facetten getestet werden – wie dies z.B. bei EJPD-Services der Fall ist) sind andere Services frei verfügbar und der Anbieter kennt seine Nutzer nicht.

Auf folgende spezifische Nutzungen soll noch hingewiesen werden, sie werden in jüngster Zeit immer wichtiger:

**Mobile Apps:** Auf heutigen Smartphones werden viele verschiedene Apps betrieben, die in der Regel viel autonom verarbeiten können, jedoch hin und wieder auf Funktionalität oder Information im Hintergrund zugreifen. Diese Zugriffe erfolgen in der Regel über Services und deshalb sollte vermehrt bei der Service-Identifikation darauf geachtet werden, dass die Services in Mobilen Apps benutzt werden können. Dies beeinflusst meist auch den Technologie-Entscheid, denn insbesondere RESTful Services und API Management haben sich hier als hilfreich erweisen. Die folgende Grafik illustriert wie ein Service von Mobilen Apps mehrerer Device-Typen genutzt wird:

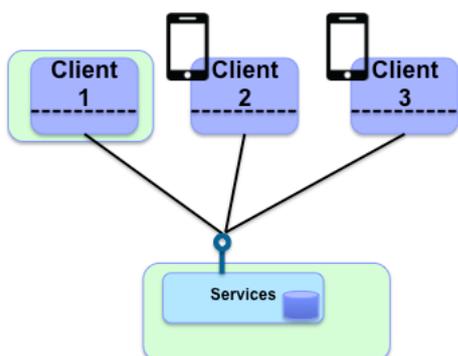


Abbildung 2: Illustration Mobile Applikation

**BPM (Business Process Management):** Mit BPM werden Prozesse modelliert und auch zur Ausführung gebracht, d.h. das Prozess-Modell steuert und kontrolliert den Ablauf laufender einzelner Prozesse. Die einzelnen Verarbeitungsschritte eines Prozesses sind „Tasks“, die gut durch Services ausgeführt werden können. So wird es möglich, in einem Prozess Daten aus einer anderen Anwendung – auch eines anderen Amtes – zu beziehen und zu nutzen. In diesen Fällen haben sich Webservices bewährt. Die folgende Grafik illustriert wie ein Prozess mehrere Services nutzt:

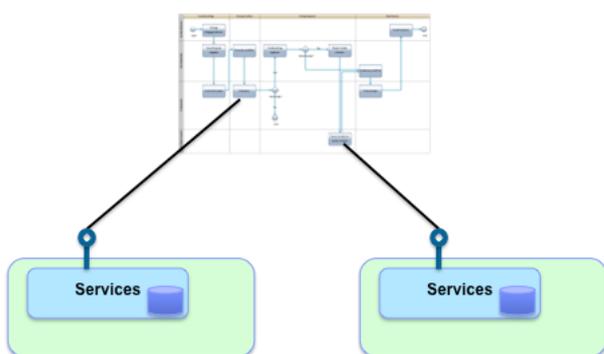


Abbildung 3: Illustration BPM

Diese beiden Beispiele zeigen, dass Services in mehrfacher Hinsicht genutzt werden können und demzufolge einen wertvollen Beitrag für die IT der Bundesverwaltung darstellen können. Die nun revidierten SOA-Policies sollen dazu beitragen, dass dies leicht und zielgerichtet zum Nutzen aller geschehen kann.

## 2.5 Serviceverzeichnis und -Dokumentation

Im Hauptdokument wird festgehalten, dass „public“ Services in einem Serviceverzeichnis eingetragen und dokumentiert werden müssen. Damit macht das Serviceverzeichnis die verfügbaren Services bekannt. Heute kann das Serviceverzeichnis mit dem folgenden Link aufgerufen werden:

<http://www.e-services.admin.ch>

Ein zentralisiertes Serviceverzeichnis dient als verbindliche Informationsquelle über alle Services. Es hat grundsätzlich zwei Zielgruppen, die allerdings nicht klar voneinander abgegrenzt sind:

- Fachpersonen, die sich für die Möglichkeiten von Services interessieren oder in ihren Ämtern auch selbst Services anbieten können und
- IT-Experten / Entwickler.

Daher ist es notwendig, dass das Serviceverzeichnis sowohl für Entwickler als auch für Nicht-Techniker eine verständliche Übersicht gibt. Das Serviceverzeichnis beschränkt sich auf die wesentlichsten Angaben; es wird auch mit einem einfachen CMS (Content Management System) gepflegt. Diese wesentlichsten Angaben können Links enthalten, die z.B. auf Web-Seiten mit für die Nutzung relevanter Angaben verweisen. Insbesondere über die breit verfügbaren öffentlichen Services müssen diese wesentlichsten Angaben mit entsprechender Sorgfalt publiziert und gepflegt werden. Beilage 2 ist das zu verwendende Template für die Erfassung der Angaben für das Serviceverzeichnis.

## 2.6 Referenz-Architekturen

Service-Gruppen und Services sind technologie-unabhängig zu definieren. Jedoch werden Services technologie-abhängig implementiert und diese Abhängigkeit besteht auch für technische Schnittstelle. Dem Service-Anbieter wird frei gestellt, welche dieser Technologien angewendet werden. Dem Service-Nutzer wird dies durch die technische Beschreibung bekannt gemacht, damit er entsprechende Aufruf-Mechanismen einsetzen kann. Eine entsprechende Referenz-Architektur bildet die Grundlage für die Entscheidungen des Service-Anbieters über die Technologien zur Implementierung und den Betrieb der Services.

Die Bundesverwaltung orientiert sich neu an der „**SOA Reference Architecture**“ der Open Group, welche als Standard seit 2011 publiziert und als Open Group Dokument C119 [SOARA] verfügbar ist. Inzwischen hat ISO eine Standardisierung der Service-Orientierten-Architektur (SOA) an die Hand genommen, wofür die Standards der Open Group die Grundlage bilden. Die folgende Grafik gibt einen groben Überblick über die SOA-Referenz-Architektur der Open Group:

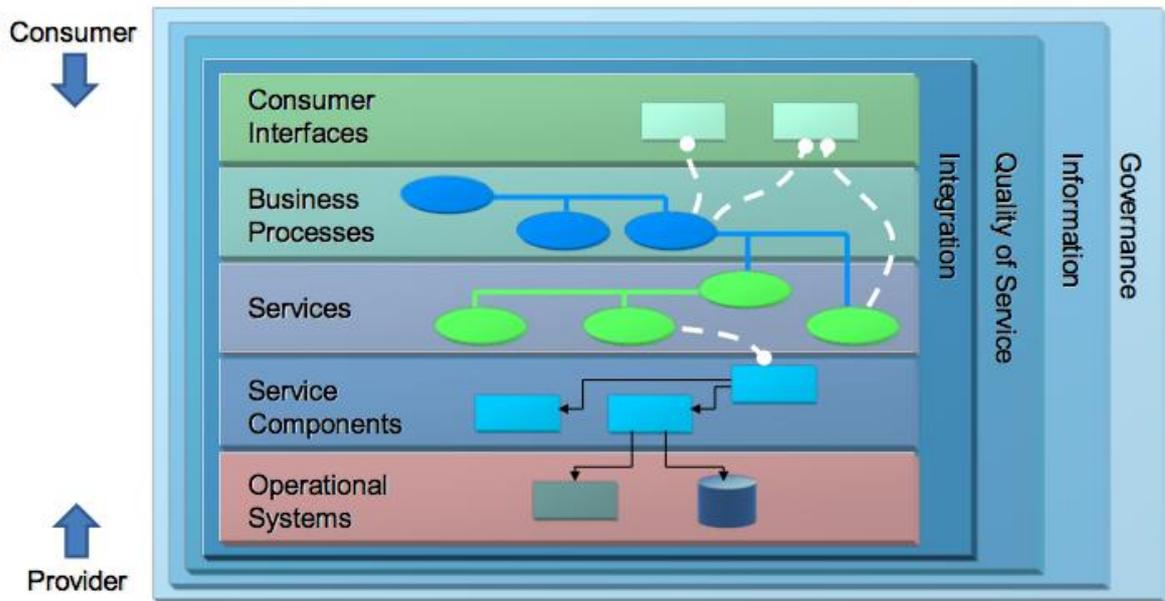


Abbildung 4: Logical Solution View der SOA Referenz-Architektur der Open Group (© Open Group)

Drei Layer befassen sich mit der Bereitstellung von Services (Operational Systems, Service Components, Services), drei Layer mit der Nutzung (Business Processes, Consumer Interfaces, Integration) und vier Layer bezeichnen übergeordnete Aspekte (Integration, Quality of Services, Information, Governance). Damit wird auch verdeutlicht, wie Service-Provider und Service-Consumer voneinander entkoppelt sind.

In [SOARA] sind die Layer, die Beziehungen zwischen den Layer, die zugehörigen Fähigkeiten (Capabilities) und Komponenten (Architecture Building Blocks – ABB) im Detail beschrieben. Die Layer lassen sich wie folgt charakterisieren:

- Der **Operational System Layer** beinhaltet alle Runtime-Komponenten für den produktiven Betrieb.
- Der **Service Component Layer** beinhaltet alle Komponenten für die Realisierung und Bereitstellung von Services.
- Der **Service Layer** definiert und spezifiziert die Funktionalität der Services aus Geschäftssicht.
- Der **Business Process Layer** unterstützt die Ausführung von Business-Prozessen und damit eine Verarbeitungssequenz von Services.
- Der **Consumer Layer** beinhaltet Komponenten des Benutzerinterfaces.
- Der **Integration Layer** stellt die für die Integration benötigte Funktionalität bereit.
- Der QoS (Quality of Service) Layer gewährleistet, dass die nichtfunktionalen Anforderungen erfüllt werden.
- Der **Information Layer** fasst alle Aspekte der Daten und Information zu den Services zusammen.
- Der **Governance Layer** befasst sich mit der Definition, Implementation und operativen Umsetzung der Steuerungsmechanismen für eine effiziente und effektive Nutzung von Services in der Organisation

Die Referenz-Architektur spezifiziert für jeden Layer, die Fähigkeiten (Capabilities), die er umfasst und welche ABBs dafür benötigt werden. Detaillierte Ausführungen über die Layer, ihre Beziehungen untereinander und ihre Architecture Building Blocks (ABB) finden sich im Original-Dokument [SOARA]; in den folgenden Abschnitten werden die Layer kurz charakterisiert.

## 2.6.1 Operational System Layer

Der Operational System Layer beinhaltet die Runtime-Komponenten für den produktiven Betrieb aller Komponenten und bildet so die Service-Plattform mit ihrer Software und Hardware ab. Dazu gehören insbesondere Netzwerk, Speicher, Rechner und Datenbanken, aber auch nicht service-orientiert realisierten Legacy-Anwendungen (sog. „Silo-Anwendungen“). Derartige Anwendungen werden in der Referenz-Architektur als Ganzes ohne eine Unterscheidung von Business-Logik und Datenhaltung betrachtet.

## 2.6.2 Service Component Layer

Der Service Component Layer beinhaltet Software-Komponenten für die Realisierung und Bereitstellung von Services. Er repräsentiert die Sicht des Service Providers. Services können unter Nutzung unterschiedlicher Technologien und Techniken angeboten werden, z.B.:

- als SOAP-basierte Web-Services gemäss dem ausgebauten Set der WS\*-Standards, die u.a. auch Sicherheitsaspekte abdecken,<sup>1</sup>
- als RESTful Services,<sup>2</sup>
- über ein API-Management.<sup>3</sup>

Eine Service-Komponente kann eine oder sogar mehrere Applikationen oder Packages einpackeln und ihre Funktionen über standardisierte Schnittstellen zugänglich machen. Funktionalität von Legacy-Systemen können „eingepackt“ und über Adaptoren als Services verfügbar gemacht werden. Es kann auch eine Kombination der Technologien zum Einsatz kommen.

## 2.6.3 Service Layer

Im Service Layer treffen sich die Interessen des Service-Nutzers und des Service-Anbieters aus organisatorischer Sicht, bzw. des Service Consumers und des Service Providers.

Der Service Layer definiert und spezifiziert die Funktionalität der Services aus Geschäftssicht. Er stellt dem Service Consumer die nötigen technischen Informationen der Services bereit (beispielsweise über eine Service Registry) und stellt sicher, dass ein Service gemäss Policy eingesetzt wird. Dazu gehört u.a. die Gewährleistung des geforderten Zugriffsschutzes.

## 2.6.4 Business Process Layer

Der Business Process Layer unterstützt die Ausführung von Business-Prozessen und die Steuerung der Verarbeitungssequenz von Services. Er umfasst alle Komponenten, die notwendig sind, um Prozesse zu definieren, auszuführen und zu überwachen. Zur Prozessausführung muss jede Instanz eines Geschäftsprozesses ihren Status mitführen, damit sie über die nötige Informationen verfügt, um den weiteren Prozessablauf zu steuern.

Falls Service-Betreiber und Anwendungs-Betreiber nicht identisch sind (z.B. bei E-Government-Lösungen), stellt der Anwendungs-Betreiber die nötige Funktionalität dafür bereit.

---

<sup>1</sup> Vgl. <https://de.wikipedia.org/wiki/WS->\*

<sup>2</sup> Vgl. [https://de.wikipedia.org/wiki/Representational\\_State\\_Transfer](https://de.wikipedia.org/wiki/Representational_State_Transfer)

<sup>3</sup> Vgl. <http://searchcloudapplications.techtarget.com/definition/API-management>

## 2.6.5 Consumer Layer

Der Consumer Layer enthält alle Komponenten des Benutzerinterfaces. Er bereitet die Informationen kanalgerecht auf (z.B. für Web-Browser oder Handys) und bietet dem Benutzer Zugang zu den angebotenen Funktionen. Der Consumer Layer entkoppelt das Benutzerinterface von der Implementierung der Funktionalität. Müssen die Inhalte gegen unerwünschten Zugriff geschützt werden, hat der Consumer Layer geeignete Funktionen bereitzustellen, über die sich die Benutzer authentisieren.

Grundsätzlich können Services auf verschiedenste Weisen genutzt werden, z.B.:

- in einem Browser,
- durch einen Rich Client, der noch einige Verarbeitungen anfügt,
- durch Apps auf Smartphones (Android oder iOS),
- durch Anwendungen auf der Nutzer-Seite,
- durch Business-Prozesse beim Aufruf einer automatisierten Aufgabe.

Wenn Service-Betreiber und Anwendungs-Betreiber sich in unterschiedlichen Organisationen befinden (z.B. bei E-Government-Lösungen), stellt der Anwendungs-Betreiber die nötige Funktionalität für diesen Layer bereit.

## 2.6.6 Integration Layer

Der Integration Layer stellt die für die Integration benötigte Funktionalität bereit, insbesondere für Routing, Daten-Transformationen, Protokoll-Konvertierungen, Mediation etc.. Zwischen dem Service-Provider und dem Service-Consumer werden ausserdem Kommunikation-Mechanismen benötigt, die zusätzliche Funktionen zur Gewährleistung der Sicherheit und des System-Managements wahrnehmen.

Zur Abdeckung der Mehrzahl der benötigten Funktionalität dieses Layers wird in der Regel ein Enterprise Service Bus (ESB) eingesetzt.

## 2.6.7 Quality of Service- Layer

Der Quality of Service Layer gewährleistet, dass die Services den an sie gestellten nicht-funktionalen Anforderungen gerecht werden.

Er enthält beispielsweise Komponenten zur Systemüberwachung, dem Policy Monitoring, dem Konfigurations- und Change-Management etc. Diese werden auch für Legacy-Anwendungen eingesetzt. Sie müssen allerdings so ausgestaltet sein, dass sie ein Layer-übergreifendes Qualitätsmanagement unterstützen.

## 2.6.8 Information Layer

Der Information Layer fasst alle Aspekte der Definition und Dokumentation der Daten bzw. Information zusammen, insbesondere jene, welche die Service-Interfaces betreffen. Wenn die Leistung einer Service-Gruppe nicht trivial ist, ist die Dokumentation des „kanonischen Datenmodells“ notwendig, auf dem die publizierten Schnittstellen beruhen. Ein über die Bundesverwaltung einheitliches Datenmodell ist jedoch nicht anzustreben.

Die Einstufung der Daten hat im Rahmen der Schutzbedarfsanalyse bzw. des Informations- und Datenschutzkonzepts zu erfolgen.

Zu diesem Layer gehören alle Komponenten zur Erstellung, Pflege und Publikation der entsprechenden Dokumentation.

## 2.6.9 Governance Layer

Der Governance Layer befasst sich mit der Definition und Implementation der Steuerungsmechanismen für eine effiziente und effektive Nutzung von Services in der Organisation, sowie der Definition und Überwachung der entsprechenden Policies.

Zu diesem Layer gehören u.a. die für die Governance nötigen Repositories; in der Bundesverwaltung ist dies u.a. das zentrale Serviceverzeichnis Bund.

## 2.7 Infrastruktur

Die Infrastruktur für Services ist eine **Services-Plattform**, welche durch eine angepasste Services-Referenz-Architektur beschrieben wird. Mit einer Professionalisierung der Infrastruktur und ihres Unterhalts kann die Bereitstellung und Nutzung von Services nachhaltig gefördert werden.

**Qualitätskriterien:** Die Services müssen so konzipiert und realisiert werden, dass sie die geforderten Qualitätskriterien erfüllen. Diese müssen in einem SLA (Service Level Agreement) festgelegt werden, das gegebenenfalls zwischen Nutzer und Anbieter abgeschlossen wird.

Oft enthält eine SOA-Plattform einen **ESB** (Enterprise Service Bus), der eine Entkoppelung von Service-Consumer und Service-Provider anbietet und auch ermöglicht, dass Datenstrukturen transformiert werden (wenn der Consumer eine andere benötigt, als der Provider anbietet) oder ein Protokoll transformiert wird (z.B. von HTTP zu JMS).

Für eine Services-Plattform der Bundesverwaltung wird eine ESB-Infrastruktur sinnvollerweise ämterübergreifend implementiert und unterhalten. Das bedeutet, dass mehrere Service-Anbieter eine gemeinsame Plattform unterhalten. Ein weiteres Element einer solchen Plattform kann eine BPM-Umgebung darstellen, womit Business-Prozesse modelliert werden und der Ablauf einzelner Prozesse gesteuert und kontrolliert werden kann.

## 2.8 Knowhow

Die Beurteilung der heutigen Situation bezüglich der Bereitstellung von Services zeigt auf, dass noch zu wenig internes Knowhow vorhanden ist und dieses auch sehr ungleichmässig verteilt ist.

Für die Professionalisierung der Bereitstellung von Services ist es notwendig, dass in diesem Bereich die Sensibilisierung und Ausbildung verbessert und ausserdem das Wissensmanagement und der Wissenstransfer gefördert werden.

## 2.9 Prozesse, Projektabwicklung

Die Ausgestaltung der Prozesse, insbesondere jene zur Entwicklung von Services hängt stark von den Umständen und den Randbedingungen ab. Die Vorgaben für die Projektdurchführung (wie [HERMES]) und den IKT-Betrieb (ITIL) sind zu respektieren. Die unterschiedlichen Ausprägungen der Service-Gruppen beeinflussen die Prozesse zur Bereitstellung und Nutzung von Services:

- Öffentliche Services, welche ohne Einschränkung der Allgemeinheit zu Gute kommen (wie z.B. der Service für das Bestimmen einer Postleitzahl) werden vom Service-Anbieter bereitgestellt, ohne dass Nutzer bekannt sind. Hierbei ist wesentlich, dass die Services derartiger Service-Gruppe robust designt und implementiert sind und mit hoher Verfügbarkeit und Performance betrieben werden.

- Geschützte Services verlangen, dass der Nutzer sich autorisiert und ausweist. In diesem Falle müssen entsprechende Authentisierungs- und Autorisierungsmechanismen berücksichtigt werden und es benötigt einen Prozess, um die Berechtigung abzuklären.
- Bei besonders sensitiven Anwendungen müssen Vereinbarungen zwischen dem Anbieter und dem Nutzer getroffen werden und u.U. die Kommunikation zwischen den Anwendungen geprüft werden; ein solches Verfahren sieht EJPD für Services vor (siehe hierzu [EJPD1] und [EJPD2]).

## 3 Details zu den einzelnen Policies

### 3.1 Rollenmodell (Policy S1)

Das Rollenmodell der Bundesverwaltung für die Nutzung von Services bezieht sich auf die Stammorganisation und beinhaltet **Service-Anbieter** und **Service-Betreiber** sowie **Service-Nutzer** und **Anwendungsbetreiber**.

**Beispiel:**

- Das BFS ist Service-Anbieter der Service-Gruppe „Unternehmensidentifikatoren (UID)“.
- Das BIT ist Service-Betreiber der UID-Service-Gruppe.
- Das BWL ist Service-Nutzer der UID-Services, denn seine Anwendung AGATE benutzt diese.
- Das ISCeco ist Anwendungsbetreiber der Anwendung AGATE, die auf UID-Services des BFS beim BIT zugreift.

**Aussage:** In der **Stammorganisation** – gemäss [HERMES] – sind insbesondere die Träger der Rollen Service-Anbieter und Service-Betreiber zu definieren.

**Begründung:** Im einfachen Fall stellt ein „Provider“ Services zur Verfügung und ein „Consumer“ nutzt diese. In der Bundesverwaltung beinhaltet der „Provider“ verschiedene Rollen, welche in der Regel nicht von derselben Organisation verantwortet werden; das Gleiche gilt auch für „Consumer“. Deshalb ist das Rollenmodell der Bundesverwaltung differenzierter und unterscheidet die organisatorische Verantwortung von Anbieter und Nutzer von der betrieblichen Verantwortung der Betreiber.

**Auswirkungen:** Organisatorisch wird hiermit ein gewisser Zusatzaufwand nötig, ohne den jedoch in einem föderalen System die Zuständigkeiten nicht klar zugeordnet werden können. Ausserdem:

- Service-Anbieter und Service-Betreiber können von verschiedenen Organisationen wahrgenommen werden (z.B. der Betreiber von der IT, der Anbieter von der Fachorganisation).
- Die Verantwortlichkeiten bezüglich Organisation und Betrieb können separat zugeordnet werden; das Gleiche gilt für entsprechende Support-Organisationen.
- Es gibt Services, welche auf einer technischen Basis von den Anbietern der eingesetzten Software angeboten werden (u.a. von SAP), für deren Nutzung jedoch in der Bundesverwaltung noch nicht die organisatorischen Voraussetzungen geschaffen sind, insbesondere die entsprechenden Rollen gemäss dem Rollenmodell noch nicht definiert sind. In einem solchen Fall muss die entsprechende Governance noch etabliert werden.
- Die Aufgaben der Rollen müssen in den entsprechenden Organisationseinheiten wahrgenommen werden und schliesslich verantwortlichen Personen zugewiesen werden.
- Wer mit wem SLA abschliesst, muss im Einzelfall geregelt werden. Es hängt insb. davon ab, nach welchen Regeln die beteiligten Verwaltungseinheiten und Leistungserbringer arbeiten.

#### 3.1.1 Rolle Service-Anbieter

Innerhalb der institutionellen Rollen der SOA-Policies kommt dem Service-Anbieter eine Schlüsselrolle zu. Er muss aus der Sicht der Bereitstellung die Gesamtverantwortung für einen Service (bzw. Service-Gruppe) wahrnehmen (im Sinne einer „Generalunternehmung“) und über entsprechenden Vereinbarungen mit seinen Anbietern bzw. Betreibern muss er den Betrieb gewährleisten. Der **Service-Anbieter**:

- kommuniziert mit potentiellen Nutzern und nimmt Anforderungen auf,
- identifiziert und spezifiziert die Services einschliesslich einer Roadmap mit einem Releaseplan für verschiedene Versionen,
- synchronisiert die Release-Fenster insbesondere mit Technologie-Releases (wie diejenigen von Java),
- ist für die Definition der nichtfunktionalen Anforderungen verantwortlich und sorgt in der Regel mit dem Abschluss entsprechender SLAs (Service Level Agreement) für deren Einhaltung,
- publiziert die Services im Serviceverzeichnis einschliesslich deren Roadmap,
- prüft während des Betriebs die Einhaltung,
- ist Ansprechperson für Design-Probleme und bearbeitet diese im Sinne eines GU.

### 3.1.2 Rolle Service-Betreiber

Innerhalb der institutionellen Rollen der SOA-Policies sorgt der Service-Betreiber für den produktiven Betrieb. Er muss die Gesamtverantwortung für einen Service (bzw. Service-Gruppe) in der Laufzeit-Umgebung wahrnehmen (im Sinne einer „GU“) und wenn nötig über entsprechenden SLAs mit seinen Anbietern bzw. Betreibern muss er den Betrieb über eine ganze Kaskadenkette gewährleisten. Der **Service-Betreiber**:

- sorgt in der Regel mit dem Abschluss entsprechender SLAs (Service Level Agreement) mit seinen Anbietern für die Einhaltung der geforderten Qualität,
- organisiert eine Support-Organisation für die Behebung von Problemen im laufenden Betrieb,
- ist Ansprechperson für Runtime-Probleme und bearbeitet dies im Sinne eines GU.

### 3.1.3 Rolle Service-Nutzer

Der Service-Nutzer verwendet einen Service indirekt, indem er eine Anwendung benutzt, die ihrerseits auf einen Service zugreift. Dies ist insofern relevant, als er damit einen Dienst eines anderen nutzt und dadurch mit ihm eine geschäftliche Verbindung eingeht. Dies kann rechtliche, finanzielle und technische Konsequenzen haben, denen er Rechnung zu tragen hat.

### 3.1.4 Rolle Anwendungsbetreiber

Die Rolle des Anwendungsbetreibers wurde in das Rollenmodell aufgenommen, um klar zu machen, dass Anwendungsbetreiber und Service-Betreiber nicht identisch zu sein brauchen. Wenn sie nicht identisch sind, muss der Anwendungsbetreiber sicherstellen, dass der Zugriff auf den Service technisch funktioniert. Ausserdem müssen Anwendungs- und Service-Betreiber die Zusammenarbeit im Falle von Problemen regeln.

### 3.1.5 Services-Governance

Während das Rollenmodell im Wesentlichen die Policies für die Bereitstellung und den Betrieb der Services darstellt, befasst sich Services-Governance mit verschiedensten Aspekten wie Design-Time, Run-Time und Change-Time Aspekte aus einer übergeordneten Sicht, indem sie u.a.:

- die Führung, Planung und Kontrolle aus einer Unternehmenssicht wahrnimmt und mit einer kohärenten Sicht die Umsetzungen so unterstützt, dass eine gewisse Einheitlichkeit erzielt wird;

- die Kooperation zwischen den einzelnen Ämtern unterstützt und dabei aus der Sicht einer Unternehmensarchitektur Bereiche für Services bestimmt (Diese orientieren sich sinnvollerweise an den Aufgabengebieten des Bundes.);
- ein Gesamtportfolios von Services aus der Sicht der Bundesverwaltung betreut;
- eine Services-Referenz-Architektur pflegt und Entscheide für die konkrete Umsetzung mit ausgewählten Technologien, Standards sowie schliesslich Produkten unterstützt;
- ordnungsgemässe Abläufe unter den beteiligten Rollen (Anbieter, Betreiber, Nutzer) sicherstellt.

## 3.2 Vorgaben (Policy S2)

Die Vorgaben der Bundesverwaltung für die Entwicklung und Bereitstellung von Services können in zwei Hauptkategorien eingeteilt werden: es gelten Prinzipien, die von den „Best Practices“ der Industrie übernommen werden und es gibt spezifische Vorgaben für die Bundesverwaltung. Die Vorgaben der SOA-Policies sind spezifisch für die Bundesverwaltung.

**Aussage:** Bei der Bereitstellung von Services sind gewisse Vorgaben einzuhalten bezüglich:

- Namenskonventionen,
- Spezifikationsvorgaben
- Qualitätsvorgaben,
- Standardisierte Servicebeschreibungen,
- Service Life Cycle Management.

**Begründung:** Diese Vorgaben sind spezifisch für die Bundesverwaltung und dienen dazu, dass das gesamte Service-Portfolio eine gewisse Einheitlichkeit aufweist. Ausserdem wird so besser den Erwartungen der Nutzer entsprochen, insbesondere bezüglich der Qualität von Services.

**Auswirkungen:** In der ersten Phase einer Service-Bereitstellung müssen entsprechende Entscheide gefällt werden. Insbesondere bei Auslagerung von Entwicklungsprojekten ist dann auf die Einhaltung der Vorgaben zu achten.

### **Namenskonventionen**

Namenskonventionen dienen der Vereinheitlichung des Erscheinungsbildes und geben Leitlinien für eine sinnvolle Benennung vor. Ausserdem sollen unterschiedliche Versionen eines Services erkennbar und unterscheidbar sein. Die Serviceversion kann durch das Anhängen von \_Versionsnummer an den Servicennamen dokumentiert werden.

**Beispiel:** *finance\_payment\_v1*

Um Namenskonflikte mit anderen Services zu vermeiden kann der Bezeichnung des betroffenen Aufgabengebietes in die Namen aufgenommen werden.

**Beispiel:** *agriculture\_grants\_calculation\_v2*

### **Sicherheit**

Öffentliche Services erfordern im Wesentlichen eine Sicherheit, dass die Daten so geschützt sind, dass sie bei der Übertragung nicht geändert werden können, d.h. dass die Services zuverlässig immer die richtigen Resultate liefern. Geschützte Services sind so zu schützen, dass nur autorisierte Zugriffe möglich sind und alle anderen zufälligen oder beabsichtigten Zugriffe unmöglich sind. Es muss – allerdings nur im Rahmen der Spezifikation und nicht öffentlich – aufgezeigt werden mit welchen Sicherheitsmassnahmen dem Schutzbedarf Rechnung getragen wird.

### **Verfügbarkeit**

Die Verfügbarkeit der Services hängt von den Möglichkeiten des Anbieters ab und ist durch die ihm zur Verfügung stehenden Mittel und Ressourcen limitiert. Deshalb kann ein Nutzer nicht weiter gehende Forderungen stellen und muss allenfalls bei ihm nicht genügender Verfügbarkeit auf die Nutzung der Services verzichten.

Eine sehr hohe Verfügbarkeit (z.B. 99.9% - nur 9 Stunden Ausfall in einem ganzen Jahr) erfordert entsprechende Massnahmen für den Betrieb, insbesondere eine redundante Auslegung von Hardware und Software. Diese müssen (im Interesse aller) finanziert werden. Dabei muss berücksichtigt werden, dass Komponenten eine höhere Verfügbarkeit aufweisen müssen als die angestrebte Gesamtverfügbarkeit: Angenommen eine Anwendung benutzt einen Webserver, einen RESTful Service und einen Datenbank-Server. Die Verfügbarkeit aller drei beträgt je 99.9%. Dies ergibt eine kumulierte Verfügbarkeit von nur noch 99.7%, d.h. 26 Stunden Ausfall pro Jahr.

### **Performance**

In der Regel sollten Services eine Performance aufweisen, die eine kaum spürbare Verzögerung in der Antwortzeit aufweist. Wesentlich dabei ist, dass berücksichtigt wird, dass sich die Antwortzeit eines Services abhängig ist von der zu verarbeitenden Datenmenge und sich zusammensetzt aus den Transferzeiten im Netz, der Verarbeitungszeit einer Software-Komponente und meist der Antwortzeit einer Datenbank.

### **Service Life Cycle**

Roadmaps bilden die Grundlage für die Kommunikation des Service Life Cycle und ermöglichen den Nutzern die entsprechende Planung. Das vorgegebene Raster mit den Bezeichnungen gilt nicht nur für Services, sondern wird auch anderweitig in der Bundesverwaltung so genutzt. Dies erlaubt, Roadmaps zu vergleichen und es erhöht die Lesbarkeit.

Bei Service-Gruppen „In Abklärung“ sind Aktivitäten zur Identifikation und Spezifikation, sowie Entscheide zur Implementierung einzubeziehen.

## **3.3 Serviceverzeichnis (Policy S3)**

Das Serviceverzeichnis der Bundesverwaltung umfasst sowohl organisatorische Aspekte als auch die Beschreibung der Services selbst.

**Aussage:** Das ISB stellt den Departementen das „**Serviceverzeichnis Bund**“ zur Verfügung um sämtliche verfügbare Service-Gruppen öffentlich zu machen. Services, welche „Public“ sind, also ämterübergreifend oder öffentlich genutzt werden können, müssen im Serviceverzeichnis Bund geführt werden.

**Begründung:** Ein zentrales „**Serviceverzeichnis Bund**“ erlaubt eine umfassende Unterstützung der Nutzung und Wiederverwendung von Services und fördert damit die Kooperation zwischen den Ämtern sowie zwischen der Bundesverwaltung und der Allgemeinheit. Dies reduziert Redundanz und spart Kosten. Es ist ein zentrales Mittel, um Services bekannt zu machen und über die gesamte Bundesverwaltung hinweg eine holistische Sicht einnehmen zu können.

**Auswirkungen:** Eine zentrale Stelle führt das Portfolio der Services der Bundesverwaltung und muss hierzu auch die entsprechenden Mittel und Ressourcen aufwenden. Das Serviceverzeichnis der Bundesverwaltung dient nicht dazu, Entwicklungs- und Betriebsgrundlagen zu dokumentieren.

Spezifisch für die Bundesverwaltung ist das Rollenmodell, das organisatorische Aspekte beinhaltet und somit Grundlage für die Angaben im Serviceverzeichnis ist. Um potentielle Nutzer anzusprechen müssen die Angaben über die fachliche Funktionen der Services soweit beschrieben sein, dass bereit gestellte Funktionalität, Information und Nutzen beim Einsatz klar ersichtlich sind. Grundsätzlich sind Angaben redundanzfrei zu halten, deshalb sollen Angaben gegebenenfalls über Links verfügbar gemacht werden.

Für den Zugriff auf die Services ist relevant, ob diese generell verfügbar oder nur nach Vereinbarung genutzt werden. Für öffentliche Services müssen sämtliche Angaben für einen Aufruf vorhanden sein inklusive Protokoll, URL, Message-Formate eventuell als XSD-Spezifikation und gegebenenfalls die WSDL-Beschreibung (z.T. nicht im Serviceverzeichnis selbst, sondern über einen im Serviceverzeichnis angegebenen Link auf eine Web-Page des Service-Anbieters). Für geschützte Services reicht u.U. eine Referenz der Ansprechperson aus.

Damit unterscheidet sich dieses Serviceverzeichnis von einem üblichen Service-Repository. Auch ist dieses Serviceverzeichnis keine Registry, eine solche Funktion wäre bereits Teil der technischen Implementation.

Services, die nicht „public“ sind, müssen nicht im Service-Verzeichnis publiziert werden. In diesen Fällen müssen die Service-Anbieter ihren Service-Nutzern die benötigten Informationen über andere Wege zugänglich machen (z.B. durch Versand der Informationen auf Anfrage, über geschützte Web-Seiten, gemeinsam zugänglich Collaborationsplattformen...).

## 4 SOA-Prinzipien

In den letzten 20 Jahren haben sich für den Einsatz von Services Regeln und Prinzipien etabliert und sogenannte „Best Practices“ für Service-Orientiertheit herausgebildet. Diese gelten auch für die Bereitstellung von Services beim Bund. Die folgenden SOA-Prinzipien sind für die Bereitstellung von Services beim Bund zu berücksichtigen.

### 4.1 Fachliche Unabhängigkeit (SOA-Prinzip 1)

**Aussage:** Services – und auch die Operationen eines Services – müssen fachlich unabhängig und autonom sein. Services sollen eine maximale Kontrolle über ihre Geschäftslogik haben. Die Grenzen der Geschäftslogik, die von einem Service kontrolliert wird, sollen genau definiert sein. „Seiteneffekte“ sollen vermieden werden, sodass nur das ausgeführt wird, was für den Service gemäss Beschreibung spezifiziert ist.

**Begründung:** „Separation of Concerns“ ist ein SOA-Grundsatz um Abhängigkeiten möglichst gering zu halten und Bereiche klar voneinander abzugrenzen.

Hierzu sollten auch Read (bzw. GET), Create (bzw. ADD) idempotent sein, d.h. dass das mehrfache Ausführen derselben Operation erzeugt dasselbe Resultat (ein ADD mit denselben Daten also nur genau einen Datensatz anlegt). Dann können diese Operationen mehrfach aufgerufen werden, was die Kommunikation vereinfacht, weil ein Client nicht auf die Rückmeldung achten muss.

**Auswirkungen:** Einkapselte Geschäftslogik kann als Baustein verwendet werden und man kann sich darauf verlassen, dass die entsprechende Funktionalität ausgeführt wird (deshalb sind auch die Seiteneffekte zu vermeiden).

**Beispiel:** Ein ADD des Services **Operation** legt nur einen Datensatz Operation an, nicht noch andere Datensätze.

### 4.2 Technologische Unabhängigkeit (SOA-Prinzip 2)

**Aussage:** Ein Service-Provider darf nicht voraussetzen, dass der Consumer bestimmte Technologien verwendet. Dieses Prinzip betrifft die Technologie, mit der der Service implementiert ist (z.B. die verwendete Programmiersprache oder Datenbanktechnologie). Es betrifft NICHT das Protokoll, mit dem die Schnittstelle implementiert ist (z.B. WSDL).

**Begründung:** Ein wesentliches Merkmal des Erfolgs der Service-Orientiertheit ist die technologische Unabhängigkeit von Provider und Consumer (was durch entsprechende Standards unterstützt wird). Damit ist es möglich Services ohne Rücksicht auf die möglichen Nutzer bereit zu stellen und die Basis-Software sowie betriebliche Plattform selbst zu wählen. So ist auch die Nutzung von Services über das Internet möglich geworden. Deshalb darf ein Service-Provider nicht technologische Abhängigkeiten dem Consumer übertragen.

**Beispiel:** Webservices benutzen HTTP und WSDL, RESTful Services HTTP und JSON (obwohl JSON Java im Namen führt, ist dies ein sprachenunabhängiges Datenformat).

**Auswirkungen:** Technologische Unabhängigkeit ist eine Grundvoraussetzung für das unabhängige Implementieren von Services. Erst damit wird eine Bereitstellung und Nutzung von Services im Sinne der Service-Orientiertheit möglich. Dies impliziert, dass grundsätzlich ein Nutzer dem Anbieter keine Forderungen, insbesondere über die Gestaltung der Schnittstelle und der Messages, stellen darf. Gegebenenfalls kann der Einsatz eines ESB notwendige Daten-Transformationen und Protokoll-Konversionen unterstützen.

### 4.3 Abstraktion (SOA-Prinzip 3)

**Aussage:** Service-Interfaces müssen so definiert werden, dass sie nicht angepasst werden müssen, wenn die Implementation, aber nicht die Funktion des Services sich ändert.

**Begründung:** Services stellen einen Kontrakt über eine zu erbringende Leistung dar, wobei das Zustandekommen der Leistung dem Leistungserbringer überlassen wird.

Ein Service abstrahiert die Details und Art und Weise der zugrunde liegenden Umsetzung und Implementation. Die Service-Interfaces müssen deshalb implementierte Logik so abstrahieren und verbergen, dass sie für den Service-Consumer und die Service-Kommunikation irrelevant sind. Die Schnittstellen der Services müssen ermöglichen, dass eine Implementation eines Services ohne Beeinträchtigung der Schnittstelle des Services ausgewechselt werden kann; d.h. eine interne Veränderung der Anwendung oder der Infrastruktur des Services hat für einen Consumer keine Auswirkung – dies wird ebenfalls als „Separation of Concerns“ bezeichnet.

Die Beschreibung der Services sollte soweit abstrakt sein, dass sie unabhängig von den unterschiedlichen Implementierungsvarianten ist.

**Auswirkungen:** Bei einer konsequenten Umsetzung dieses Grundsatzes können Services mit unterschiedlichen Technologien implementiert werden und damit dieselbe Leistung mit einer aktualisierten Implementierung erbracht werden. Dies erhöht die Flexibilität des Service-Anbieters sehr.

### 4.4 Wiederverwendbarkeit (SOA-Prinzip 4)

**Aussage:** Services sollen so gestaltet werden, dass sie möglichst breit eingesetzt und wiederverwendet werden können. Service-Gruppen sollen so designt werden, dass sie robust gegen Veränderungen des Gebrauchs und des Umfelds sind.

**Begründung:** Sobald ein Service mehrfach genutzt wird, lohnt sich der für seine Erstellung zusätzliche Aufwand. Jede Wiederverwendung reduziert die Gesamtkosten und erzielt Synergien.

**Auswirkungen:** Wiederverwendung erfordert einige Vorkehrungen beim Design und für den Betrieb. Der Design sollte möglichst robust erfolgen und eventuell zukünftige Nutzungen so einbeziehen, dass Änderungen (z.B. für zusätzliche Parameter in der Message) nicht nötig werden. Beim Betrieb ist zu berücksichtigen, dass mehrfache Nutzung auch eine höhere Last zur Folge hat und deshalb mehr Kapazität zur Verfügung gestellt werden sollte. Diese zusätzlichen Aufwendungen kosten und deshalb sind entsprechende zusätzliche Mittel bereit zu stellen. Umgekehrt erspart sich ein Nutzer von Services Aufwand und Kosten durch das Erzielen von Synergien bei der Wiederverwendung bestehender Services.

### 4.5 Statuslosigkeit (SOA-Prinzip 5)

**Aussage:** Service-Aufrufe müssen voneinander unabhängig sein, d.h. es darf zwischen Client und Server keine „Session“ aufgebaut werden.

**Begründung:** Die Statuslosigkeit im Sinne, dass „Sessions“ ausgeschlossen werden, bildet eine Grundlage erfolgreicher Services-Nutzung. Damit ist jedes Paar eines Service Requests und einer Service Response von den anderen Calls unabhängig; auch hat die Reihenfolge der Ausführung keinen Einfluss. Das bedeutet jedoch nicht, dass ein Client für sich den Status der Bearbeitung führt (mittels Cookies oder in einer BPM-Engine) und auch der Service verändert seinen Status (u.a. in der Datenbank). Dabei muss von anderen Verwendungen des Begriffs „statuslos“ unterschieden werden.

**Auswirkungen:** Der Design der Services muss entsprechend gestaltet sein und die Unabhängigkeit der Paare von Service Request und einer Service Response garantieren.

**Beispiel:** Von BPM-Engines (Business Process Management) werden Services in einer definierten Sequenz aufgerufen; entsprechende BPM Tool-Umgebungen gewährleisten, dass die einzelnen Prozess-Instanzen ihren Status mitführen.

## Anhänge

### A. Abkürzungen

| <b>Kürzel</b> | <b>Bedeutung</b>                     |
|---------------|--------------------------------------|
| ABB           | Architecture Building Block          |
| ESB           | Enterprise Service Bus               |
| HTTP          | Hypertext Transfer Protocol          |
| ISB           | Informatiksteuerungsorgan des Bundes |
| JMS           | Java Message Service                 |
| JSON          | JavaScript Object Notation           |
| REST          | Representational State Transfer      |
| SLA           | Service Level Agreement              |
| SOA           | Service-Orientierte Architektur      |
| WSDL          | Web Services Description Language    |
| XML           | Extensible Markup Language           |
| XSD           | XML Schema Definition                |

## B. Glossar der grundlegenden Begriffe

In diesem Glossar werden nur die im Dokument R016 und den Erläuterungen verwendete Begriffe aufgeführt, jedoch nicht die bereits definierten Begriffe (wie z.B. einzelne Rollen).

| Begriff                | Definition                                                                                                                               |
|------------------------|------------------------------------------------------------------------------------------------------------------------------------------|
| Institutionelle Rolle  | Eine institutionelle Rolle beschreibt die Aufgaben, Funktionen und Verantwortung einer Organisation in einem bestimmten Kontext.         |
| Personenbezogene Rolle | Eine personenbezogene Rolle beschreibt die Aufgaben, Funktionen, Verantwortung und Kompetenzen einer Person in einem bestimmten Kontext. |
| Rolle                  | Im Kontext von SOA werden zwei Arten von Rollen verwendet:<br><br>-> Institutionelle Rolle<br><br>-> Personenbezogene Rolle              |

## C. Glossar technischer Begriffe

| Begriff                  | Definition                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|--------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| WSDL                     | <p>Web Service Description Language</p> <p>Ein Standard der W3C (<a href="http://www.w3.org/TR/wsdl">http://www.w3.org/TR/wsdl</a>) für die Beschreibung der Schnittstellen von Webservices.</p>                                                                                                                                                                                                                                                                                                                         |
| REST                     | <p>Representational State Transfer</p> <p>Eine in manchen Fällen valide Alternative zu Webservices bei der Realisierung von Services. Nicht standardisiert, basiert auf einer Dissertation (<a href="http://www.ics.uci.edu/~fiel-ding/pubs/dissertation/top.htm">http://www.ics.uci.edu/~fiel-ding/pubs/dissertation/top.htm</a>).</p> <p>Vgl. auch die Zusammenfassung in: <a href="http://en.wikipedia.org/wiki/Representational_State_Transfer">http://en.wikipedia.org/wiki/Representational_State_Transfer</a></p> |
| JMS                      | <p>Java Message Service</p> <p>Eine genormte Schnittstelle für den Zugriff auf Messaging Middleware (<a href="http://www.jcp.org/en/jsr/detail?id=914">http://www.jcp.org/en/jsr/detail?id=914</a>)</p>                                                                                                                                                                                                                                                                                                                  |
| Stateful / Stateless     | <p>Stateful = Statusbehaftet</p> <p>Stateless = ohne (interne) Status</p> <p>Ein statusloses System verhält sich bei gleichem Input immer exakt gleich.</p> <p>Im Zusammenhang von SOA wird gefordert, dass Serviceaufrufe statuslos sein sollen; sich also immer unabhängig von zuvor erfolgten Aufrufen verhalten.</p>                                                                                                                                                                                                 |
| SOAP                     | <p>Ein XML-basiertes Protokoll zum Austausch von Nachrichten zwischen Systemen, das u.a. Webservices zu Grunde liegt. SOAP ist durch die W3C standardisiert: <a href="http://www.w3.org/TR/soap/">http://www.w3.org/TR/soap/</a></p>                                                                                                                                                                                                                                                                                     |
| BPMN                     | <p>Business Process Modeling Notation</p> <p>Ein OMG-Standard für die Notation und den Ablauf von Business-Prozessen - BPMN 2.0.1 auf <a href="http://www.omg.org/docs/formal/2013-09-02">http://www.omg.org/docs/formal/2013-09-02</a></p>                                                                                                                                                                                                                                                                              |
| Anwendung<br>Application | <p>Eine allgemein anerkannte Definition gibt es nicht, jedoch versteht man meist unter einer Anwendung die Zusammenfassung von Funktionalitäten einer Software mit einer gemeinsamen Benutzeroberfläche. Organisatorisch ist eine Anwendung oft dadurch definiert, dass es einen Anwendungsverantwortlichen gibt.</p> <p>„Anwendung“ ist in vielen Fällen gleichbedeutend mit IKT Lösung.</p>                                                                                                                            |
| Webservice               | <p>Eine mögliche Realisierungsvariante eines Service auf Basis der Webservice-Technologie. Das Interface eines Webservice wird durch eine WSDL Dokument beschrieben. Ein Webservice kommuniziert auf Basis von SOAP-Nachrichten.</p>                                                                                                                                                                                                                                                                                     |

## D. Referenzen

- [IKTS] IKT-Strategie des Bundes 2016–2019 vom Dezember 2015.
- [BinfV] Verordnung über die Informatik und Telekommunikation in der Bundesverwaltung vom 09. Dezember 2011 (Stand am 01. Januar 2012); SR 172.010.58
- [HERMES] Standard P007 – Richtlinien für Projektführung in Informatikprojekten
- [RVOG] Regierungs- und Verwaltungsgesetz vom 21. März 1997 (Stand am 01. Januar 2015) SR 172.010
- [EJPD1] Richtlinie für Zugriffe auf EJPD Informatikanwendungen via Webservice Schnittstellen vom 8. April 2014 – V1.3 (EJPD Webservice Richtlinie)
- [EJPD2] HP NonStop Ablösung Prozessbeschreibung EJPD WSBV, 8.3.2013, In Arbeit
- [SOARA] SOA Reference Architecture, Open Group Technical Standard 2011, ISBN-Nr: 1-937218-01-0, Document Number C119