



IsyFact-Standard

IsyFact – Namenskonventionen

Version 1.8
25.04.2016



„IsyFact – Namenskonventionen“ des Bundesverwaltungsamts ist lizenziert unter einer Creative Commons Namensnennung 4.0 International Lizenz.



„IsyFact – Namenskonventionen“
des Bundesverwaltungsamts ist lizenziert unter einer
Creative Commons Namensnennung 4.0 International Lizenz.

Die Lizenzbestimmungen können unter folgender URL heruntergeladen
werden: <http://creativecommons.org/licenses/by/4.0>

Ansprechpartner:

Referat Z II 2
Bundesverwaltungsamt
E-Mail: isyfact@bva.bund.de
Internet: www.isyfact.de

Dokumentinformationen

Dokumenten-ID:	IsyFact-Namenskonventionen.docx
----------------	---------------------------------

Inhaltsverzeichnis

1. Einleitung	6
1.1. Kurzüberblick	6
2. Namenskonventionen	7
2.1. Allgemein	7
2.1.1 Namen von Anwendungssystemen	7
2.2. Spezifikation	8
2.2.1 Geschäftsprozess	8
2.2.2 Geschäftsvorfall	8
2.2.3 Aktivität	8
2.2.4 Dokument	8
2.2.5 Persistente Daten	8
2.2.6 Organisationseinheit	9
2.2.7 Anwendungssystem	9
2.2.8 Anwendungskomponente	9
2.2.9 Anwendungsfall	9
2.2.10 Anwendungsfunktion	9
2.2.11 Batch	9
2.2.12 Modellkomponente	10
2.2.13 Entitätstyp	10
2.2.14 Dialog	10
2.2.15 Maske	10
2.2.16 Nachbarschnittstelle	10
2.2.17 Schnittstellenentitätstyp	10
2.2.18 Nicht-Funktionale Anforderung	11
2.3. Entwicklung	11
2.3.1 Technische Systemnamen	11
2.3.2 Name der Web Application	13
2.3.3 Interface-Implementierungen	13
2.3.4 Persistenz	13
2.3.5 Business-Objekte	13
2.3.6 Anwendungsfälle	14
2.3.7 Anwendungsfunktionen	14
2.3.8 Batches	14

2.3.9 Schnittstellen.....	15
2.3.10Web-GUI	16
2.4. Deployment/Betrieb von Anwendungen	17
2.4.1 RPM-Pakete.....	17
2.4.2 Installationspfade	18
3. Quellenverzeichnis	20

1. Einleitung

Die umfassenden Konzepte der IsyFact automatisieren und vereinheitlichen den Bau von Anwendungen. Um dies zu erreichen ist es wichtig, dass Namen und Benennungen über die gesamte Anwendungslandschaft hinweg einheitlich sind. Es existiert daher eine Vielzahl von Namenskonventionen in den verschiedenen Konzepten der IsyFact. Dieses Dokument ist Teil der IsyFact-Standards.

1.1. Kurzüberblick

Dieses Dokument soll die Namenskonventionen der IsyFact an einem Ort zentral sammeln. In den jeweiligen Konzepten wird die Vorgabe nochmals erwähnt.

Nach der Einleitung, die einen kurzen Abschnitt über generelle Anforderungen an Namen enthält, befasst sich das Kapitel 2 mit den Namenskonventionen der IsyFact. Es ist dabei unterteilt in allgemeine Konventionen sowie Konventionen in der Entwicklung von Anwendungen.

Jede Namenskonvention ist in Tabellenform dargestellt. Es ist jeweils das Namensschema der Konvention mit Beispielen zu sehen. Ergänzt wird diese Darstellung durch eine Erklärung der genutzten Variablen und ihrer möglichen Ausprägungen.

2. Namenskonventionen

In diesem Abschnitt werden die Namenskonventionen der IsyFact beschrieben. In komplexen Anwendungslandschaften werden Anwendungssysteme in der Regel in funktionale Kategorien eingeordnet und typisiert (bspw. Register und Geschäftsanwendungen). In weniger komplexen Kontexten, in dem nur ein einziges oder wenige Fachanwendungen existieren, wird meist allgemein von „Fachanwendungen“ gesprochen.

In folgenden Vorgaben wird daher zwischen Fachanwendungen, Registern und Geschäftsanwendungen unterschieden. In anderen Kontexten können andere Systemtypen eingeführt werden, für die entsprechende Vorgaben nach dem gleichen Schema zu definieren sind.

2.1. Allgemein

In diesem Abschnitt werden allgemeine Namenskonventionen festgehalten.

2.1.1 Namen von Anwendungssystemen

Je nach Typ eines Anwendungssystems ist der Name unterschiedlich zu wählen. Die verschiedenen Varianten sind wie folgt abgebildet.

2.1.1.1 Fachanwendung

Name einer Fachanwendung	
Schema	<verfahren> Anwendung
Beispiele	Xyz Anwendung
Variable	Mögliche Ausprägungen
<verfahren>	Der (Kurz)name des Verfahrens/Anwendungsbereichs.

2.1.1.2 Register

Name eines Registers	
Schema	<verfahren> Register
Beispiele	Xyz Register
Variable	Erklärung / Ausprägungen
<verfahren>	Der (Kurz)name des Verfahrens/Anwendungsbereichs.

2.1.1.3 Geschäftsanwendung

Name einer Geschäftsanwendung	
-------------------------------	--

Schema	<verfahren> Geschäftsanwendung
Beispiele	Xyz Geschäftsanwendung
Variable	Mögliche Ausprägungen
<verfahren>	Der (Kurz)name des Verfahrens/Anwendungsbereichs.

2.2. Spezifikation

In diesem Kapitel werden die Namenskonventionen der Spezifikation festgehalten. Sie sind vor allem der Systemspezifikationsvorlage [IsyFactSystemspezifikation] entnommen.

2.2.1 Geschäftsprozess

Geschäftsprozess	
Schema	GEP_<geschäftsprozessname>
Beispiele	GEP_Akte_Suchen

2.2.2 Geschäftsvorfall

Geschäftsvorfall	
Schema	GVF_<geschäftsvorfallname>
Beispiele	GVF_Akte_Suchen

2.2.3 Aktivität

Aktivität	
Schema	AKT_<aktivitätsname>
Beispiele	AKT_Akte_Suchen

2.2.4 Dokument

Dokument	
Schema	DOK_<dokumentname>
Beispiele	DOK_Spezifikationsvorlage_Test

2.2.5 Persistente Daten

Persistente Daten	
-------------------	--

Schema	DAT_<datenbestandsname>
Beispiele	DAT_Akte_Suche

2.2.6 Organisationseinheit

Organisationseinheit	
Schema	ORG_<organisationsheit>
Beispiele	DAT_AbteilungXyz

2.2.7 Anwendungssystem

Anwendungssystem	
Schema	SYS_<anwendungssystemname>
Beispiele	SYS_RegisterXyz

2.2.8 Anwendungskomponente

Anwendungskomponente	
Schema	ANK_<anwendungskomponentenname>
Beispiele	ANK_Meldung_Beispiel

2.2.9 Anwendungsfall

Anwendungsfall	
Schema	AWF_<anwendungsfallname>
Beispiele	AWF_Meldung_Durchfuehren

2.2.10 Anwendungsfunktion

Anwendungsfunktion	
Schema	AFU_<anwendungsfunktionsname>
Beispiele	AFU_Wert_Berechnen

2.2.11 Batch

Batch	
Schema	BAT_<batchname>

Beispiele	BAT_Bereinigungslauf_Durchfuehren
------------------	-----------------------------------

2.2.12 Modellkomponente

Modellkomponente	
Schema	MKO_<modellkomponentenname>
Beispiele	MKO_Basisdaten

2.2.13 Entitätstyp

Entitätstyp	
Schema	ETY_<entitaetstypname>
Beispiele	ETY_Akte

2.2.14 Dialog

Dialog	
Schema	DIA_<dialogname>
Beispiele	DIA_Akten_Suche

2.2.15 Maske

Maske	
Schema	MAS_<maskenname>
Beispiele	MAS_Treffer_Anzeige

2.2.16 Nachbarschnittstelle

Nachbarschnittstelle	
Schema	NST_<schnittstellename>
Beispiele	NST_Meldung

2.2.17 Schnittstellenentitätstyp

Schnittstellenentitätstyp	
Schema	NSE_<entitaetstypname>
Beispiele	NSE_Personen_Suche

2.2.18 Nicht-Funktionale Anforderung

Nicht-Funktionale Anforderung	
Schema	NFA_<anforderungsname>
Beispiele	NFA_Durchsatz_Pro_Minute

2.3. Entwicklung

Dieser Abschnitt fasst die Namenskonventionen zusammen, die bei der Entwicklung eines Anwendungssystems nach IsyFact relevant sind. Das sind vor allem Klassen- und Dateinamen.

Einige der hier genannten Namenskonventionen sind von denen der Spezifikation abhängig beziehungsweise werden davon abgeleitet.

2.3.1 Technische Systemnamen

Die technischen Systemnamen entsprechen der technischen Bezeichnung für ein IsyFact-konformes Anwendungssystem. Sie werden unter anderem für Projektnamen in Eclipse, Dokument Basis / Context Root bei Schnittstellen-URLs und als Namen von Deployment-Einheiten (vgl. Kapitel 2.4.1) verwendet.

2.3.1.1 Fachanwendungen

Name einer Fachanwendung	
Schema	<verfahren>-anwendung
Beispiele	meinverfahren-anwendung
Variable	Erklärung / Ausprägungen
<verfahren>	Der (Kurz)name des Verfahrens/Anwendungsbereichs.

2.3.1.2 Register

Name eines Registers	
Schema	<verfahren>-register
Beispiele	xyz-register
Variable	Erklärung / Ausprägungen
<verfahren>	Der (Kurz)name des Verfahrens/Anwendungsbereichs.

2.3.1.3 Geschäftsanwendung

Name einer Geschäftsanwendung	
-------------------------------	--

Schema	<verfahren>-ga
Beispiele	xyz-ga
Variable	Mögliche Ausprägungen
<verfahren>	Der (Kurz)name des Verfahrens/Anwendungsbereichs.

2.3.1.4 Servicegateway

Name eines Servicegateways	
Schemata	<verfahren>-sgw <verfahren>-<zielverfahren>-sgw
Beispiele	xyz-sgw xyz-dienstabc-sgw
Variable	Mögliche Ausprägungen
<verfahren>	Der (Kurz)name des Verfahrens/Anwendungsbereichs.
<zielverfahren>	Der (Kurz)name des Verfahrens/Anwendungsbereichs, mit dem dieser SGW kommuniziert bzw. für den er eine Schnittstelle bereitstellt.

2.3.1.5 Mailgateway

Name eines Mailgateways	
Schema	<verfahren>-mailgw
Beispiele	xyz-mailgw
Variable	Mögliche Ausprägungen
<verfahren>	Der (Kurz)name des Verfahrens/Anwendungsbereichs.

2.3.1.6 Batch-Anwendungen

Mit vielen Systemen wird eine eigene Batch-Deploymenteinheit ausgeliefert. Diese wird folgendermaßen benannt.

Name einer Batchanwendung	
Schema	<anwendungsname>-batch
Beispiele	xyz-ga-batch

	register-xyz-batch
Variable	Mögliche Ausprägungen
<anwendungsname>	Der Name einer Anwendung, wie in den vorigen Abschnitten beschrieben. Er setzt sich meistens aus einem Systemkürzel und dem Systemtyp zusammen.

2.3.2 Name der Web Application

Der Name einer Web-Applikation (Webapp-Root) ist immer gleich dem technischen Systemnamen (siehe Abschnitt 2.3.1.)

2.3.3 Interface-Implementierungen

Implementierungen von Interfaces werden durch das Suffix „Impl“ gekennzeichnet.

Interface-Implementierungen	
Schema	<Interface>Impl
Beispiele	MeldungImpl NachrichtErzeugungImpl

2.3.4 Persistenz

Data Access Objects: Interfaces + Implementierung	
Schema	<Entitaetsname>Dao <Entitaetsname>DaoImpl
Beispiele	AkteDao AkteDaoImpl

2.3.5 Business-Objekte

Unter Business-Objekten werden nicht-persistente Objekte verstanden, die in Schnittstellen im Anwendungskern verwendet werden.

Business-Objekte	
Schema	<Entitaetsname>Bo
Beispiele	AkteBo

2.3.6 Anwendungsfälle

Anwendungsfallklassen werden durch das Präfix `Awf` gekennzeichnet. Der weitere Teil des Klassennamens leitet sich aus dem Namen des Anwendungsfalls aus der Spezifikation ab, in CamelCase-Schreibweise.

Anwendungsfälle	
Schema	<code>Awf<AnwendungsfallName></code>
Beispiele	<code>AwfAntragVerarbeiten</code> <code>AwfEntscheidungDurchfuehren</code>

2.3.7 Anwendungsfunktionen

Analog zu den Anwendungsfällen werden Anwendungsfunktionen mit dem Präfix `Afu` gekennzeichnet.

Anwendungsfunktionen	
Schema	<code>Afu<Anwendungsfunktionsname></code>
Beispiele	<code>AfuBerechnungFristdatum</code> <code>AfuErmittlungEntscheidungsrelevanz</code>

2.3.8 Batches

Analog zu den Anwendungsfällen werden Batchklassen mit dem Präfix `Bat` gekennzeichnet.

Batches: Klassen	
Schema	<code>Bat<Batchname></code>
Beispiele	<code>BatLoeschfristPruefen</code> <code>BatSendenAllerInformationen</code>
Batches: Property-Dateien (unter resources/resources/batch)	
Schemata	<code><batchname-des-batches>.properties</code>
Beispiele	<code>loeschfrist-pruefen.properties</code> <code>import-bhknz-liste.properties</code>
Batches: Shellskripte	
Schemata	<code><batchname-des-batches>.sh</code>
Beispiele	<code>loeschfrist-pruefen.sh</code> <code>import-bhknz-liste.sh</code>

2.3.9 Schnittstellen

Schnittstellen werden in der IsyFact in eigenen Schnittstellenprojekten bereitgestellt. Die Konventionen diesbezüglich sind wie in den folgenden Abschnitten dargestellt.

2.3.9.1 Interface-Klassennamen

Interface-Klassennamen	
Schema	<Schnittstellennamen>RemoteBean
Beispiele	MeldungRemoteBean KommunikationXyzRemoteBean

2.3.9.2 URL

Die Service-URL einer HTTP-Invoker-Schnittstelle muss folgendem Schema entsprechen:

Schnittstellen: URL	
Schema	http://<Hostname>/<Anwendungsname>/<Servicename>_v<Version>
Beispiele	http://register-xyz.test.de/xyz-register/AuskunftBean_v4_1 http://qs-xyz.test.de/plis-benutzerverzeichnis/AuskunftBean_v4_1
Variable	Mögliche Ausprägungen
<Hostname>	Der Hostname des Servers
<Anwendungsname>	Der Name der Anwendung. Siehe Kapitel 2.1.1. Wenn es sich um eine Querschnittskomponente handelt, muss dieser mit dem Präfix „isy-“ beginnen.
<Servicename>	Der Name des Services.
<Version>	Die Versionsnummer des Service, die zweielementig und aufsteigend vergeben wird. Hierbei wird sich am Schema Major- und Minor-Level orientiert. Beispiele: <ul style="list-style-type: none"> • 1_0 • 12_7

2.3.9.3 Projektname / Artefakt-ID

Die Maven Artefakt-ID von Schnittstellen ist nach dem folgenden Muster aufgebaut. Die ersten beiden Teile der Versionsnummer sind Teil der Artefakt-ID, um mehrere Versionen einer Schnittstelle gleichzeitig einbinden zu können. Die dritte Stelle wird die Maven-Versionsnummer des Artefakts.

Artefakt-ID von Schnittstellen	
Schema	<Anwendungsname>-<protokoll>-sst-<sstname>-<version>
Beispiele	xyz-ga-httpinvoker-sst-auskunft-v29.1

2.3.9.4 Transportobjekte

Klassennamen Transportobjekte	
Schema	<Entitätsname>To
Beispiele	AkteTo InformationenXyzTo

2.3.10 Web-GUI

2.3.10.1 Modelklassen

Klassennamen Modelklassen	
Schema	<Entitätsname>Model
Beispiele	CdErwerbModel

2.3.10.2 Controllerklassen

Klassennamen Controllerklassen	
Schema	<Maskenname>Controller
Beispiele	CdErwerbController

2.3.10.3 Flows

Flows	
Schema	<Dialogname>Flow.xml
Beispiele	cdErwerbFlow.xml

2.3.10.4 Main-View von Flows

Flows	
Schema	<Maskenname>ViewState.xhtml
Beispiele	cdErwerbViewState.xhtml cdSucheViewState.xhtml

2.3.10.5 Weitere Facelets

Facelets	
Schema	<maskenname><maskenteil>.xhtml
Beispiele	cdErwerbFormular1.xhtml

2.3.10.6 View-State Javascript-Datei

View-State Javascript-Datei	
Schema	<viewstatename>.js <guikomponentenname>.js (bei übergreifenden, wiederverw. Dateien)
Beispiele	cdErwerbFormular.js cdErwerb.js (übergreifend)

2.4. Deployment/Betrieb von Anwendungen

2.4.1 RPM-Pakete

Für die Benennung von RPM-Paketen existiert eine Konvention, welche durch den RPM-Standard vorgeben wird¹. Diese Vorgaben sind ebenfalls im [DeploymentKonzept] zu finden.

RPM-Pakete	
Schema	<paketname>-<versionsnummer>-<build-version>.<architektur>.rpm
Beispiele	isy-cd-anwendung-batch-1.2.0-01.noarch.rpm
Variable	Mögliche Ausprägungen
<paketname>	Name der Deployment-Einheit. Setzt sich zusammen aus dem Präfix „isy-“ und dem Anwendungsnamen (siehe Kapitel 2.1.1) Beispiel: <ul style="list-style-type: none"> isy-xyz-ga-batch

¹ Siehe <http://docs.fedoraproject.org/drafts/rpm-guide-en/ch01s03.html>.

	<p>Aus historischen Gründen existieren auch noch Deployment-Einheiten mit dem Präfix „plis-“. Diese Namenskonvention ist veraltet und sollte für neue Anwendungen nicht mehr verwendet werden.</p> <p>Beispiel:</p> <ul style="list-style-type: none"> plis-xyz-ga-batch
<versionsnummer>	<p>Versionsnummer der Deployment-Einheit, z. B. „1.2.0“. Die zweite Stelle wird mit jedem Release² hochgezählt. Bei umfangreicheren Anpassungen kann – nach Abstimmung mit den entsprechenden Kontrollgremien – auch die erste Stelle erhöht werden.</p> <p>Die dritte Stelle ist üblicherweise 0. Sie kann bei Notfallauslieferungen oder Patches erhöht werden.</p>
<build-version>	<p>Hier wird die Build-Nummer eingesetzt. Sie wird bei dem Bau jeder Auslieferungsversion (insbesondere auch bei Nachlieferungen) erhöht.</p> <p>Während der Entwicklung (Nightly-Build) wird hier die SVN-Revisionsnummer eingesetzt.</p>
<architektur>	<p>Gibt die Systemarchitektur an, für welche das Paket erstellt wurde. Da IsyFact-konforme Anwendungen in Java erstellt werden, wird hier immer „noarch“ eingesetzt. Sollten Anwendungen Architektur-spezifische Bestandteile enthalten, wird hier die vom RPM-Standard vorgegebene Architektur-Bezeichnung eingesetzt.</p>

2.4.2 Installationspfade

Die Installationspfade sind ausführlich im Konzept [DeploymentKonzept] beschrieben.

Installationspfade – Anwendung	
Schema	<p>/opt/<rpm-paketname> (Anwendungsbasis)</p> <p>/opt/<rpm-paketname>/bin (Skript-Verzeichnis)</p> <p>/opt/<rpm-paketname>/tomcat (Tomcat)</p> <p>/opt/<rpm-paketname>/tomcat/webapps/<name> (Basis der Webapp)</p>
Beispiele	<p>/opt/isy-xyz-anwendung</p> <p>/opt/isy-xyz-anwendung /bin</p> <p>/opt/isy-xyz-anwendung /tomcat</p> <p>/opt/isy-xyz-anwendung /tomcat/webapp/xyz-anwendung</p>
Installationspfade – Logdateien	
Schema	<p>/var/log/<rpm-paketname></p>

² Unter „Release“ wird üblicherweise die Entwicklung und Auslieferung im Rahmen eines Projektes verstanden.

Beispiele	/var/log/isy-xyz-anwendung
Installationspfade – Betriebliche Konfiguration	
Schema	/etc/<rpm-paketname>
Beispiele	/etc/isy-xyz-anwendung

3. Quellenverzeichnis

[DeploymentKonzept]

Konzept Deployment für IsyFact-Anwendungen
30_Plattform\Konzept_Deployment.pdf.

[IsyFactSystemspezifikation]

Vorlage für Systemspezifikationen
40_Methodik\20_Systementwurf\IsyFact-Vorlage_Systemspezifikation.dot.