

PUBLIC

SAP Connected Health Platform
Document Version: 1.0 – 2016-08-12

SAP Connected Health Platform Administration Guide



Content

- 1 Getting Started. 5**
- 1.1 About this Document. 6
- 1.2 System Requirements. 6
- 1.3 Technical System Landscape. 7
- 1.4 Administrative Apps. 8

- 2 Data Integration. 10**
- 2.1 Integrating Data Using the Basic Extension for Clinical Data. 10
 - Basic Extension for Clinical Data. 13
- 2.2 Integrating HIX Data. 26
 - HIX Extension. 31
 - Overview of the HIX Format. 33
 - Adapting XML Sources to HIX. 33
 - XML Schema Definition File. 37
 - Example XML for HIX Data. 46
- 2.3 Integrating Genomic Data. 49
 - Integrating Reference Genome Data. 50
 - Integrating Feature Data. 54
 - Integrating Sample Variant Data. 60
- 2.4 Integrating Unstructured Documents. 71
 - Integrating Pathology Reports. 71
 - Integrating Medical Documents. 73
 - Integrating Documents Without Text Analysis. 75
 - Integrating Documents Using the BLOB Adapter and the Basic Extension. 79
 - Extension for Processing Unstructured Documents. 84
- 2.5 Integrating Content from Ontology Content Providers. 86
 - NCIm Import Extension. 88
 - NCIm Transform Extension. 89
 - Value Sets Import Extension. 90
 - Value Sets Transform Extension. 91
 - CMS GEMs Import Extension. 92
 - CMS GEMs Transform Extension. 92
 - RxNorm Import Extension. 93
 - RxNorm Transform Extension. 94
 - Vocabularies Import Extension. 94
 - Vocabularies Transform Extension. 95
- 2.6 Creating Profiles for Data Integration Extensions. 96

2.7	Rolling Back and Cleaning Up Integrated Data.	97
2.8	Connecting to New Data Sources.	98
3	Operations.	99
3.1	Monitoring Profile Runs.	99
3.2	Monitoring the Clinical Data Warehouse.	99
3.3	Viewing and Deleting Patient Data.	102
4	User Management.	104
4.1	User Management Configuration.	104
	Configuring User Accounts.	105
	Configuring Organizational Units.	106
	E-Mail Server Configuration.	107
4.2	Creating Users.	107
	Using CSV File for Multiple Users and Organizational Units.	109
4.3	Creating Organizational Units.	110
4.4	Reassigning Organizational Units.	111
4.5	Deactivation of Users and Organizational Units.	112
4.6	Integrating Users and Organizational Units from LDAP Directory.	112
	LDAP Extension.	116
5	Authorizations.	117
5.1	User Roles.	117
5.2	Technical Users.	119
5.3	SAML Configuration.	120
5.4	Analytic Privileges.	120
6	Application Configuration.	122
6.1	Clinical Data Model Configuration.	123
	Managing Configurations.	124
	Configuring Basic Data for Patients.	125
	Required Attributes.	129
	Configuring Interactions.	130
	Configuring Interaction Attributes.	136
	Advanced Settings for Clinical Data Model Configurations.	142
	Example: Configuring a Primary Tumor Diagnosis Interaction.	144
6.2	Assigning Configurations to Users.	153
6.3	Configuring End-User License Agreements.	154
6.4	Configuring Patient Summary.	156
	Configuring Patient Details.	157
	Activating Patient Summary Tabs.	158
	Configuring Timeline Lanes.	159
	Enabling Users to View Genomic Content.	160

	Activating Patient Summary Configurations.	161
	Adapting to More Recent Versions of the Data Model Configuration.	162
	Duplicating Patient Summary Configurations.	163
	Importing Patient Summary Configurations in JSON Format.	163
6.5	Configuring Search.	164
	Deploying Custom Search Views.	166
	Customizing Search Results Lists.	167
	Changing the Search Background Image.	168
6.6	Customizing Data Consumption for Applications.	169
7	Troubleshooting.	170

1 Getting Started

This guide is intended for system administrators and other users responsible for the daily operation of the SAP Connected Health system.

Daily operations include starting and monitoring data integration runs, managing users, and configuring the clinical data model for use in various applications.

For information about the installation and configuration of the SAP Connected Health system and its components, see the SAP Connected Health Implementation Guide.

Related Information

[About this Document \[page 6\]](#)

[System Requirements \[page 6\]](#)

[Technical System Landscape \[page 7\]](#)

[Administrative Apps \[page 8\]](#)

1.1 About this Document


This guide describes installation, security, and all administrative tasks related to SAP Connected Health that system administrators perform regularly.

1.2 System Requirements

Before installing or using the platform, ensure that the following system requirements are met:

Table 1:

Requirement	Details
<p>SAP HANA platform 1.0 SPS11 (revision 112.03)</p> <p>i Note</p> <p>As the required SAP HANA database maintenance revision may no longer be available in the Software Download Center, please contact your SAP Account team to retrieve the exact revision required.</p> <p>i Note</p> <p>You must deploy SAP HANA platform as a single-container system. SAP Connected Health does not support multitenant database containers.</p> <p>i Note</p> <p>SAP Connected Health does not support scale-out scenarios involving multiple SAP HANA systems.</p>	<ul style="list-style-type: none">• For general release information and information about patches, see SAP Note 2228298.• You can find a complete list of all SAP HANA components and the respective SAP HANA hardware and software requirements in the Product Availability Matrix (PAM) on SAP Support Portal.• For more information about the hardware and software requirements for SAP HANA, see the SAP HANA Server Installation and Update Guide.• For general information about deploying and using an SAP HANA system, see the SAP HANA Master Guide.• Depending on the scope of your implementation, you can use either the SAP HANA Runtime edition for Applications & SAP BW or SAP HANA Enterprise edition. For information about SAP HANA licensing and options, contact your SAP account team.
<p>SAP HANA Enterprise Information Management 1.2.x (SPS11)</p>	<p>For standard data integration scenarios, SAP HANA Enterprise Information Management is required.</p> <p>For custom data integration solutions, other ETL tools may be used.</p>
<p>Architecture</p>	<p>You must use an x86_64 architecture that is supported by SAP HANA, also known as amd64.</p> <p>PowerPC is not supported.</p>

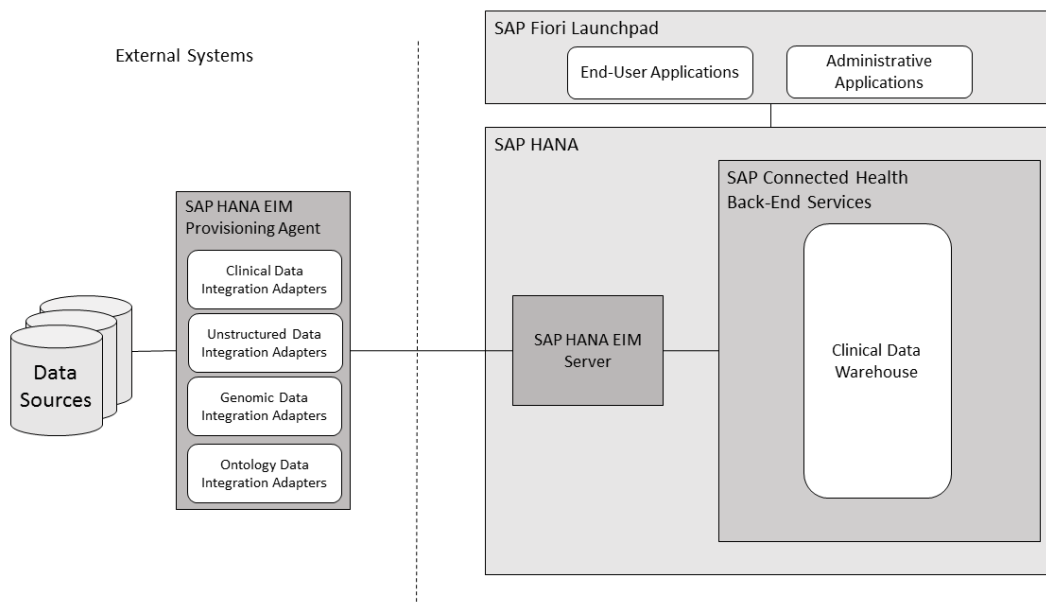
Requirement	Details
Available memory size	Minimum 64 GB RAM For more specific sizing information, see SAP Note 2295041  .
Supported Web browser	<ul style="list-style-type: none"> • Internet Explorer 11.0.9600.18350CO • Mozilla Firefox 47.0.1 • Google Chrome 51.0.2704.106 m • Apple Safari 9.1 on OS X

Related Information

[Product Availability Matrix \(PAM\)](#) 

1.3 Technical System Landscape

The technical system landscape of SAP Connected Health comprises SAP HANA platform, SAP HANA Enterprise Information Management components, SAP Fiori launchpad, and external data source systems.



For integrating clinical, genomic, and ontology data, SAP Connected Health data integration adapters are installed on SAP HANA Enterprise Information Management provisioning agents located on external data source systems. Clinical, genomic, and ontology data is then transmitted to the SAP Connected Health back-

end services through the SAP HANA Enterprise Information Management server and stored in the clinical data warehouse after processing.

Information stored in the clinical data warehouse, as well as underlying configuration information, is then consumed by applications running on the platform. End users and administrators access these applications through the SAP Fiori launchpad.

1.4 Administrative Apps

In addition to the tools available with the standard SAP HANA platform, the following administrative apps are provided specifically for use with SAP Connected Health:

Table 2:

Administrative App	Description
Clinical Data Warehouse Cockpit	For monitoring the data stored in the clinical data warehouse
Data Integration Cockpit	For configuring and using data integration extensions
Plug-In Cockpit	For activating plug-ins, such as data integration plug-ins, that provide additional features
Clinical Data Model Configuration	For configuring the basic patient data, interactions, and attributes used in applications
Patient Summary Configuration	For configuring the features of the Patient view
End-User License Agreements	For creating and editing end-user license agreements
Document Type Configuration	For specifying the text analysis configurations and language-specific names applied to document types
User Management Configuration	For selecting which attributes are available for creating users and organizational units
User Management	For creating and deactivating users and organizational units
Configuration Assignment	For assigning application configurations to users or organizations

Administrative App	Description
Advanced Settings for Clinical Data Model Configurations	<p>For specifying advanced settings, such as placeholder-to-view mappings or query settings, that apply to all clinical data model configurations</p> <div data-bbox="804 488 1396 719" style="background-color: #fff9c4; padding: 10px;"> <p>→ Tip</p> <p>This application is not included in the standard SAP Fiori launchpad catalogs for SAP Connected Health. To use this application, search for it in the tile catalog and add it manually to your SAP Fiori launchpad.</p> </div>
Search Configuration	For integrating and configuring custom search implementations
Original Document Service Configuration	For configuring where original documents are retrieved from in order to be shown on the user interface

2 Data Integration

SAP Connected Health provides standard data integration extensions for integrating both clinical data and genomic data into the clinical data warehouse. For each data integration extension, you create profiles that you use when running data integration. These profiles allow you to define how the data is integrated. You activate and use data integration profiles in the *Data Integration Cockpit* application.

For the data integration extensions to work correctly, you must ensure the implementation steps for the corresponding adapters, virtual tables, and remote sources were completed. For more information, see the SAP Connected Health Implementation Guide.

Related Information

[Integrating Data Using the Basic Extension for Clinical Data \[page 10\]](#)

[Integrating HIX Data \[page 26\]](#)

[Integrating Genomic Data \[page 49\]](#)

[Integrating Unstructured Documents \[page 71\]](#)

[Integrating Content from Ontology Content Providers \[page 86\]](#)

[Creating Profiles for Data Integration Extensions \[page 96\]](#)

[Rolling Back and Cleaning Up Integrated Data \[page 97\]](#)

[Connecting to New Data Sources \[page 98\]](#)

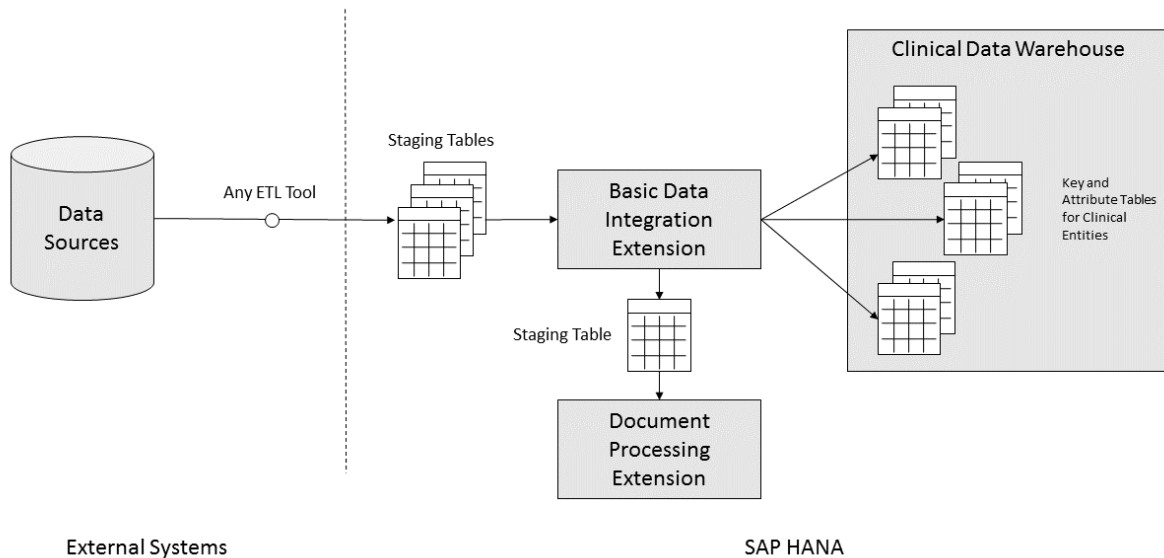
2.1 Integrating Data Using the Basic Extension for Clinical Data

Use any ETL tool and the basic extension for clinical data to integrate structured and embedded unstructured clinical data from sources (tables, virtual tables, views), which act as a staging area, directly into the clinical data warehouse. You configure this extension in the *Data Integration Cockpit*.

Prerequisites

- You have populated the default source tables provided for the `Basic Extension for Clinical Data` with your clinical data, or you have created custom source tables that have identical signatures to the default source tables. For more information, see [Basic Extension for Clinical Data \[page 13\]](#).
- You are assigned the role `HC_HPH_PLUGINS_ADMIN`.

Context



The basic data integration extension for clinical data simplifies data integration by predefining the source tables and dataflows needed for each clinical entity and its attributes. These source tables act as a staging area, from which clinical data is integrated directly into the clinical data warehouse. Unlike other data integration extensions, this extension doesn't require the deployment and configuration of SAP HANA Enterprise Information Management data provisioning adapters.

You can also use the basic data integration extension to integrate unstructured documents. The initial staging process is identical to that for structured clinical data, but instead of writing the data directly to the clinical data warehouse, the extension then passes the documents to the data integration extension for processing documents. This extension then handles all steps required for extracting and writing the document metadata to the clinical data warehouse.

You can populate these source tables from your external data sources by using any ETL tool supported by SAP HANA, as SAP Data Services.

Procedure

1. Using any ETL tool, transform and write your clinical data to the source tables of the SAP Connected Health Platform Basic Data Integration Extension for Clinical Data.

➔ Tip

Check if any of the standard or custom data integration extensions you have are already designed to perform this step.

For more information about the source tables, see [Source Tables for the Basic Extension for Clinical Data \[page 16\]](#).

2. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

3. Open the *Plug-In Cockpit* app.
4. Select and activate the SAP Connected Health Platform Basic Data Integration Extension for Clinical Data.
5. Open the *Data Integration Cockpit* app.
6. Create a profile for data integration.
 - a. Open the *Run and Manage Profiles* view.
 - b. Choose *Create Profile*.
 - c. Choose the relevant data integration extension.

➔ Remember

Data integration extensions are only available if the corresponding plug-in is activated.

- d. Optionally, choose a DW Source.

DW Source is an optional attribute used in the clinical data warehouse to filter data during integration. This means, only data matching the specified DW Source will be integrated during the profile run.

You can create a new DW Source by choosing the **+** icon.

- e. Optionally, choose a schedule for running the profile.

The schedule is based on an existing schedule configuration. You can create a new schedule configuration by choosing the **+** icon.

i Note

You can't create schedule configurations for the following data formats:

- GFF3
- VCF
- BED
- PED

- f. Specify the required parameters.

For more information about the profile parameters, see [Basic Extension for Clinical Data \[page 13\]](#).

7. Activate the profile and choose *Run Profile*.
8. To monitor the status of the profile run, open the *Monitor Profile Runs* view.

Results

After the profile run has completed, you can confirm the data was correctly integrated by either viewing the data in the target tables of the data integration extension or searching for the data in the end-user applications.

Related Information

[Basic Extension for Clinical Data \[page 13\]](#)

2.1.1 Basic Extension for Clinical Data

SAP Connected Health provides a basic data integration extension for integrating structured and embedded unstructured clinical data from sources (tables, virtual tables, views), which act as a staging area, directly into the clinical data warehouse. You configure this extension in the *Data Integration Cockpit*.

i Note

This basic data integration extension supports only the entities and attributes of the default clinical data model for SAP Connected Health. If you have implemented custom attributes, these will not be included during data integration.

Custom Source Tables

If you don't want to use the default tables, you can configure data integration profiles to use your own source tables. However, these tables must have the same signature as the default tables, meaning the number, order, and names of columns must be identical to the default.

For technical reasons, if you want to integrate data from tables where the table signature does not match the required default signature, you must create SQL views to produce the default signature.

If you create SQL views, you must define the SQL views without the read-only option. This is to enable the automatic cleanup of records loaded from the SQL views to the clinical data warehouse.

i Note

The technical user and the `_SYS_REPO` user require DELETE and SELECT privileges on all source tables and SQL views used in the data integration profile.

For more information about the signature of the default tables, see [Source Tables for the Basic Extension for Clinical Data \[page 16\]](#).

Profile Parameters

The following parameters are available when creating profiles for this basic extension:

Table 3:

Parameter	Description	Default
Documents Source	Source table or view for documents	"SAP_HPH"."sap.hc.hph.plugins.basiccdm.db.models::Stage.Document"
Patient Source	Source table or view for the patient entity and its attributes	"SAP_HPH"."sap.hc.hph.plugins.basiccdm.db.models::Stage.Patient"
Practitioner Source	Source table or view for the practitioner entity and its attributes	"SAP_HPH"."sap.hc.hph.plugins.basiccdm.db.models::Stage.Practitioner"
Interactions Source	Source table or view for the interactions entity and its attributes	"SAP_HPH"."sap.hc.hph.plugins.basiccdm.db.models::Stage.Interactions"
Interaction Details Source	Source table or view for the additional attributes describing the interactions entity	"SAP_HPH"."sap.hc.hph.plugins.basiccdm.db.models::Stage.InteractionDetails"
Interaction Measures Source	Source table or view for the measured attributes and their values associated with the interactions entity	"SAP_HPH"."sap.hc.hph.plugins.basiccdm.db.models::Stage.InteractionsMeasures"
Interaction Texts Source	Source table or view for any texts associated with the interactions entity	"SAP_HPH"."sap.hc.hph.plugins.basiccdm.db.models::Stage.InteractionText"
Observations Source	Source table or view for the observations entity and its attributes	"SAP_HPH"."sap.hc.hph.plugins.basiccdm.db.models::Stage.Observations"
Conditions Source	Source table or view for the conditions entity and its attributes	"SAP_HPH"."sap.hc.hph.plugins.basiccdm.db.models::Stage.Conditions"
OrgID	Organizational unit ID If no OrgID is specified for a record relating to an organization, such as patients or interactions, the record is assigned the same OrgID as the profile.	
Cleanup Sources	If you specify Yes, all the data that was loaded during the profile run will be deleted from the source tables.	

Target Tables

When you integrate clinical data using this basic extension, the following SAP Connected Health tables are populated:

Table 4:

Table	Purpose
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities.Patient_Key"	Stores the business key for the patient
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities.Patient_Attr"	Stores attributes taken from the patient details
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities.Interactions_Key"	Stores the business key for the interaction
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities.Interactions_Attr"	Stores recognized elements/attributes, such as type, taken from the interaction
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntitiesEAV.Interaction_Details"	Stores key/value pairs taken from the interaction detail
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntitiesEAV.Interaction_Measures"	
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntitiesEAV.Interaction_Text"	
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities:Observations_Key"	Stores the business key for the observation
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities:Observations_Attr"	Stores attributes taken from the observation details
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities:Condition_Key"	Stores the business key for the condition
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities:Condition_Attr"	Stores attributes taken from the condition details
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWDocuments.Interaction_Documents_Link"	If embedded unstructured data exists, this table stores attributes about the relationship between the interaction and the unstructured data.
"SAP_HPH"."sap.hc.hph.di.documents.staging.db.models::DocumentStage.ProcessingStage"	Temporarily stores embedded unstructured data taken from interaction details. The Extension for Processing Unstructured Documents takes unstructured data from this staging table for further processing.

Table	Purpose
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities.Practitioner_Key"	Stores the business key for the practitioner
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities.Practitioner_Attr"	Stores attributes taken from the practitioner details
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities.Patient_Practitioner_Link"	Stores links between patients and practitioners
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities.Patient_Practitioner_Link_Attr"	This tables stores additional attributes for a link between a patient and practitioner such as a role

Related Information

[Source Tables for the Basic Extension for Clinical Data \[page 16\]](#)

2.1.1.1 Source Tables for the Basic Extension for Clinical Data

The source tables for the basic extension for clinical data are similar to the tables for the target clinical data model entities. Each table contains columns for technical attributes and columns for descriptive attributes that store the actual data. The descriptive attributes are the same as the descriptive attributes of the corresponding target tables of each clinical data model entity.

However, the technical columns of the tables are different. All source tables for the basic extension for clinical data contain technical columns that are prefixed by an underline "_", for example `_PackageID`.

Each source table contains the following technical columns:

Table 5:

Column	SQL Data Type	Description
_PackageID	Integer	<p>This column is mandatory and must not be NULL as it is part of the physical or logical (in case of views) primary key. The column ensures that table comparison works correctly and allows for load distribution.</p> <p>As a rule, the value for the column _PackageID is generated using the following SQL statement:</p> <pre>SELECT IFNULL(MAX("_PackageID"), 0) + 1 FROM "<SOURCE_TABLE>" WHERE "_SourceID" = '<MYSRC>';</pre> <p>i Note</p> <p>A new _PackageID must be generated after each commit into the respective source table. This is because the extension traverses the data in packages and provides a cleanup-mechanism for already loaded records, which is based on the _PackageID.</p>
_LoadTimeStamp	TIMESTAMP	The value can be NULL as this attribute is not currently taken into account.

Column	SQL Data Type	Description
_SourceID	NVARCHAR(5)	<p>This column is used for data-driven source-mapping, which means it determines which source the record originates from and provides the DWSource for calculating the DWID of each record.</p> <p>Note the following interdependencies:</p> <ul style="list-style-type: none"> • The DWSource assigned to a data integration profile affects the _SourceID value and acts as a filter. • If a DWSource is set, only rows with DWSource = _SourceID are loaded • If no DWSource is set, all rows in the source table are loaded. <div style="background-color: #fff9c4; padding: 5px; border: 1px solid #ccc;"> <p>i Note</p> <p>Multiple profiles must not load the same _SourceID at the same time.</p> </div>
_ <i>Entity</i> >SourceID	NVARCHAR(5)	<p>This column exists for all foreign keys in the source table. For example, PatientID appears as _PatientSourceID. This enables records to link to records from other sources.</p>
_ProcessingMode	NVARCHAR(1)	<p>This column specifies the processing mode of a record. You can specify the following values:</p> <ul style="list-style-type: none"> • NULL = Normal (Insert / Update record target table) • D = Delete (Invalidate record in the target table)

Stage.Conditions Table

Table 6:

Column	SQL Data Type	Primary Key	Nullable
_PackageID	INTEGER	X	
_LoadTimestamp	TIMESTAMP		X

Column	SQL Data Type	Primary Key	Nullable
_SourceID	NVARCHAR(5)	X	
_ProcessingMode	NVARCHAR(1)		X
ConditionID	NVARCHAR(100)	X	
ConditionType	NVARCHAR(100)		X
Description	NVARCHAR(5000)		X

Stage.Document Table

Table 7:

Column	SQL Data Type	Primary Key	Nullable
_PackageID	INTEGER	X	
_LoadTimestamp	TIMESTAMP		X
_SourceID	NVARCHAR(5)	X	
_ProcessingMode	NVARCHAR(1)		X
DocumentID	NVARCHAR(1024)	X	
_ParentEntitySourceID	NVARCHAR(5)		X
ParentEntityID	NVARCHAR(1024)		X
ParentEntityType	NVARCHAR(256)		X
FileName	NVARCHAR(256)		X
MIMETYPE	NVARCHAR(256)		X
Content	BLOB		X
Title	NVARCHAR(1024)		X
Author	NVARCHAR(1024)		X
Type	NVARCHAR(128)		X
LanguageCode	NVARCHAR(2)		X
CreatedAt	TIMESTAMP		X
CreatedBy	NVARCHAR(256)		X
ChangedAt	TIMESTAMP		X
ChangedBy	NVARCHAR(256)		X

Stage.InteractionDetails Table

Table 8:

Column	SQL Data Type	Primary Key	Nullable
_PackageID	INTEGER	X	
_LoadTimestamp	TIMESTAMP		X
_SourceID	NVARCHAR(5)	X	
_ProcessingMode	NVARCHAR(1)		X
InteractionID	NVARCHAR(100)	X	
Attribute.OriginalValue	NVARCHAR(100)	X	
Attribute.Code	NVARCHAR(100)	X	
Attribute.CodeSystem	NVARCHAR(100)	X	
Attribute.CodeSystemVersion	NVARCHAR(100)	X	
Value.OriginalValue	NVARCHAR(5000)		X
Value.Code	NVARCHAR(100)		X
Value.CodeSystem	NVARCHAR(100)		X

Stage.InteractionMeasures Table

Table 9:

Column	SQL Data Type	Primary Key	Nullable
_PackageID	INTEGER	X	
_LoadTimestamp	TIMESTAMP		X
_SourceID	NVARCHAR(5)	X	
_ProcessingMode	NVARCHAR(1)		X
InteractionID	NVARCHAR(100)	X	
Attribute.OriginalValue	NVARCHAR(100)	X	
Attribute.Code	NVARCHAR(100)	X	
Attribute.CodeSystem	NVARCHAR(100)	X	
Attribute.CodeSystemVersion	NVARCHAR(100)	X	
Unit	NVARCHAR(100)		X
Value	DECIMAL(34,10)		X

Stage.InteractionText Table

i Note

If you want to include unstructured text for an interaction, this table will load the text to the table `InteractionText_EAV` and link the text to the interaction.

Table 10:

Column	SQL Data Type	Primary Key	Nullable
_PackageID	INTEGER	X	
_LoadTimestamp	TIMESTAMP		X
_SourceID	NVARCHAR(5)	X	
_ProcessingMode	NVARCHAR(1)		X
InteractionID	NVARCHAR(100)	X	
Attribute	NVARCHAR(100)	X	
Value	NVARCHAR(5000)		X
Lang	NVARCHAR(50)	X	

Stage.Interactions Table

Table 11:

Column	SQL Data Type	Primary Key	Nullable
_PackageID	INTEGER	X	
_LoadTimestamp	TIMESTAMP		X
_SourceID	NVARCHAR(5)	X	
_ProcessingMode	NVARCHAR(1)		X
InteractionID	NVARCHAR(100)	X	
_PatientSourceID	NVARCHAR(5)		X
PatientID	NVARCHAR(100)		
_ParentInteractionSourceID	NVARCHAR(5)		X
ParentInteractionID	NVARCHAR(100)		X
_ConditionSourceID	NVARCHAR(5)		X
ConditionID	NVARCHAR(100)		X
InteractionType.Original-Value	NVARCHAR(100)		X
InteractionType.Code	NVARCHAR(100)		X

Column	SQL Data Type	Primary Key	Nullable
InteractionType.CodeSystem	NVARCHAR(100)		X
InteractionType.CodeSystemVersion	NVARCHAR(100)		X
InteractionStatus	NVARCHAR(100)		X
PeriodStart	TIMESTAMP		X
PeriodEnd	TIMESTAMP		X
OrgID	NVARCHAR(100)		X

Stage.Observations Table

Table 12:

Column	SQL Data Type	Primary Key	Nullable
_PackageID	INTEGER	X	
_LoadTimestamp	TIMESTAMP		X
_SourceID	NVARCHAR(5)	X	
_ProcessingMode	NVARCHAR(1)		X
ObservationID	NVARCHAR(100)	X	
_PatientSourceID	NVARCHAR(5)		X
PatientID	NVARCHAR(100)		
ObsType	NVARCHAR(100)		X
ObsCharValue	NVARCHAR(255)		X
ObsNumValue	DECIMAL(34,10)		X
ObsUnit	NVARCHAR(100)		X
ObsTime	TIMESTAMP		X
OrgID	NVARCHAR(100)		X

Stage.Patient Table

Table 13:

Column	SQL Data Type	Primary Key	Nullable
_PackageID	INTEGER	X	
_LoadTimestamp	TIMESTAMP		X
_SourceID	NVARCHAR(5)	X	

Column	SQL Data Type	Primary Key	Nullable
_ProcessingMode	NVARCHAR(1)		X
PatientID	NVARCHAR(100)	X	
ValidFrom	DATE	X	
ValidTo	DATE		X
FamilyName	NVARCHAR(100)		X
GivenName	NVARCHAR(100)		X
Title.OriginalValue	NVARCHAR(100)		X
Title.Code	NVARCHAR(100)		X
Title.CodeSystem	NVARCHAR(100)		X
Title.CodeSystemVersion	NVARCHAR(100)		X
Gender.OriginalValue	NVARCHAR(100)		X
Gender.Code	NVARCHAR(100)		X
Gender.CodeSystem	NVARCHAR(100)		X
Gender.CodeSystemVersion	NVARCHAR(100)		X
BirthDate	SECONDDATE		X
MultipleBirthOrder	TINYINT		X
DeceasedDate	SECONDDATE		X
MaritalStatus.OriginalValue	NVARCHAR(100)		X
MaritalStatus.Code	NVARCHAR(100)		X
MaritalStatus.CodeSystem	NVARCHAR(100)		X
MaritalStatus.CodeSystem- Version	NVARCHAR(100)		X
Nationality.OriginalValue	NVARCHAR(100)		X
Nationality.Code	NVARCHAR(100)		X
Nationality.CodeSystem	NVARCHAR(100)		X
Nationality.CodeSystemVer- sion	NVARCHAR(100)		X
Address.StreetName	NVARCHAR(200)		X
Address.StreetNumber	NVARCHAR(60)		X
Address.PostOfficeBox	NVARCHAR(60)		X
Address.City	NVARCHAR(100)		X
Address.PostalCode	NVARCHAR(60)		X
Address.State	NVARCHAR(100)		X
Address.Region	NVARCHAR(100)		X
Address.Country.Original- Value	NVARCHAR(100)		X

Column	SQL Data Type	Primary Key	Nullable
Address.Country.Code	NVARCHAR(100)		X
Address.Country.CodeSystem	NVARCHAR(100)		X
Address.Country.CodeSystemVersion	NVARCHAR(100)		X
Telecom.Phone	NVARCHAR(100)		X
Telecom.Mobile	NVARCHAR(100)		X
Telecom.Fax	NVARCHAR(100)		X
Telecom.Email	NVARCHAR(100)		X
OrgID	NVARCHAR(100)		X

Stage.Patient_Practitioner Table

Table 14:

Column	SQL Data Type	Primary Key	Nullable
_PackageID	INTEGER	X	
_LoadTimestamp	TIMESTAMP		X
_SourceID	NVARCHAR(5)	X	
_ProcessingMode	NVARCHAR(1)		X
_PatientSourceID	NVARCHAR(5)		X
PatientID	NVARCHAR(100)	X	
_PractitionerSourceID	NVARCHAR(5)		X
PractitionerID	NVARCHAR(100)	X	
Role.OriginalValue	NVARCHAR(100)		X
Role.Code	NVARCHAR(100)		X
Role.CodeSystem	NVARCHAR(100)		X
Role.CodeSystemVersion	NVARCHAR(100)		X

Stage.Practitioner Table

Table 15:

Column	SQL Data Type	Primary Key	Nullable
_PackageID	INTEGER	X	
_LoadTimestamp	TIMESTAMP		X

Column	SQL Data Type	Primary Key	Nullable
_SourceID	NVARCHAR(5)	X	
_ProcessingMode	NVARCHAR(1)		X
PractitionerID	NVARCHAR(100)	X	
ValidFrom	DATE	X	
ValidTo	DATE		X
FamilyName	NVARCHAR(100)		X
GivenName	NVARCHAR(100)		X
Title.OriginalValue	NVARCHAR(100)		X
Title.Code	NVARCHAR(100)		X
Title.CodeSystem	NVARCHAR(100)		X
Title.CodeSystemVersion	NVARCHAR(100)		X
Gender.OriginalValue	NVARCHAR(100)		X
Gender.Code	NVARCHAR(100)		X
Gender.CodeSystem	NVARCHAR(100)		X
Gender.CodeSystemVersion	NVARCHAR(100)		X
BirthDate	SECONDDATE		X
MultipleBirthOrder	TINYINT		X
DeceasedDate	SECONDDATE		X
MaritalStatus.OriginalValue	NVARCHAR(100)		X
MaritalStatus.Code	NVARCHAR(100)		X
MaritalStatus.CodeSystem	NVARCHAR(100)		X
MaritalStatus.CodeSystem-Version	NVARCHAR(100)		X
Nationality.OriginalValue	NVARCHAR(100)		X
Nationality.Code	NVARCHAR(100)		X
Nationality.CodeSystem	NVARCHAR(100)		X
Nationality.CodeSystemVersion	NVARCHAR(100)		X
Address.StreetName	NVARCHAR(200)		X
Address.StreetNumber	NVARCHAR(60)		X
Address.PostOfficeBox	NVARCHAR(60)		X
Address.City	NVARCHAR(100)		X
Address.PostalCode	NVARCHAR(60)		X
Address.State	NVARCHAR(100)		X
Address.Region	NVARCHAR(100)		X

Column	SQL Data Type	Primary Key	Nullable
Address.Country.OriginalValue	NVARCHAR(100)		X
Address.Country.Code	NVARCHAR(100)		X
Address.Country.CodeSystem	NVARCHAR(100)		X
Address.Country.CodeSystemVersion	NVARCHAR(100)		X
Telecom.Phone	NVARCHAR(100)		X
Telecom.Mobile	NVARCHAR(100)		X
Telecom.Fax	NVARCHAR(100)		X
Telecom.Email	NVARCHAR(100)		X
Role.OriginalValue	NVARCHAR(100)		X
Role.Code	NVARCHAR(100)		X
Role.CodeSystem	NVARCHAR(100)		X
Role.CodeSystemVersion	NVARCHAR(100)		X
Speciality.OriginalValue	NVARCHAR(100)		X
Speciality.Code	NVARCHAR(100)		X
Speciality.CodeSystem	NVARCHAR(100)		X
Speciality.CodeSystemVersion	NVARCHAR(100)		X
PreferredLanguage	NVARCHAR(100)		X
OrgID	NVARCHAR(100)		X

2.2 Integrating HIX Data

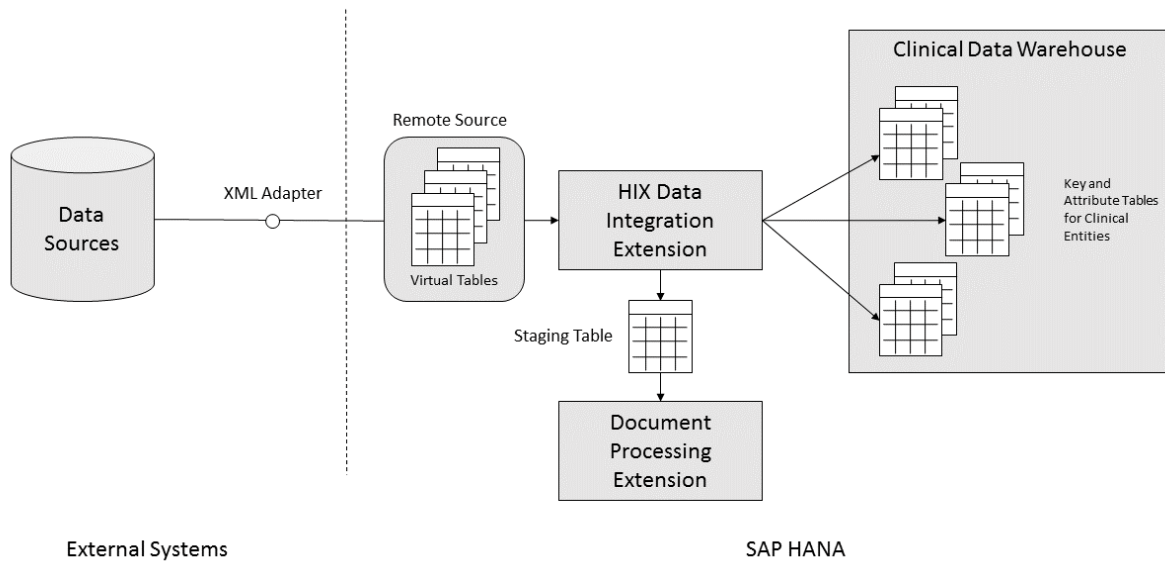
Integrate structured and unstructured clinical data conforming to the HIX format by using the SAP XML adapter and the HIX data integration extension. You configure this extension in the [Data Integration Cockpit](#).

Prerequisites

- An SAP HANA Enterprise Information Management data provisioning agent is installed and configured and has access to the file share with the HIX documents
- The XML adapter is deployed and configured and you have created the required virtual tables and remote sources.
For more information, see the SAP Connected Health Platform Implementation Guide.
- You have activated the relevant plug-in for HIX data in the [Plug-In Cockpit](#).

- You have configured the relevant data integration extensions in the *Data Integration Cockpit*. For more information, see [Creating Profiles for Data Integration Extensions \[page 96\]](#).
- You are assigned the role `HC_HPH_PLUGINS_ADMIN`.

Context



You can also use the HIX data integration extension to integrate unstructured documents that are embedded in the HIX files. The initial staging process is identical to that for structured clinical data, but instead of writing the data directly to the clinical data warehouse, the extension then passes the documents to the data integration extension for processing documents. This extension then handles all steps required for extracting and writing the document metadata to the clinical data warehouse.

This data integration extension uses the deployed XML adapter to retrieve the HIX files from the source system and store them in the corresponding SAP HANA virtual tables, which were created during system implementation. The structured clinical data is then transformed and stored in the tables for clinical entities in the clinical data warehouse.

Procedure

1. If necessary, adapt your sources to the HIX format.

For more information about the HIX format, see the following topics:

 - [Adapting XML Sources to HIX \[page 33\]](#)
 - [XML Schema Definition File \[page 37\]](#)
 - [Example XML for HIX Data \[page 46\]](#)
2. If you haven't already, set up the SAP HPH XML adapter and create virtual tables for it.
 - a. Register the SAP HPH XML adapter with SAP HANA.

To do so, follow the steps described in .

➔ **Tip**

As an easy alternative, execute the following SQL statement:

```
CREATE AGENT '<agent name>' PROTOCOL 'TCP' HOST '<host where agent was installed>' PORT '<port specified during agent installation>'
CREATE ADAPTER 'SAP HPH XML Adapter' AT LOCATION AGENT '<agent name>'
```

- b. In your SAP HANA system, create and configure a remote source for the SAP HPH XML adapter.

To do so, follow the steps described in .

When you configure the remote source, you define the following parameters:

Table 16:

Parameter	Mandatory	Description
Source Directory	X	Directory of HIX files in Data Provisioning Agent instance
Schema Definition	X	Schema definition (.xsd) file to validate the HIX files
Archive Directory		Directory for storing archived documents
Schema Validation	X	A flag to enable and disable the schema validation for HIX files [Default value: Yes]
Archive Mode	X	A flag to enable archive operation for processed documents Allowed values: <ul style="list-style-type: none"> ○ <code>Keep</code>: Keeps the file in the source directory location [Default] ○ <code>Delete</code>: Deletes the files from the source directory ○ <code>Move</code>: Moves the file from source directory to archive directory
Batch Size	X	Defines how many files to be processed in one batch. Default value is 100.

- c. Create the required virtual tables corresponding to the remote source you've created and assign them to your schema.

By default, the system suggests the following virtual tables:

- SAP_HPH_VT_FILES
- SAP_HPH_VT_LOG_DETAIL
- SAP_HPH_VT_XML_STAGE

i Note

Virtual tables for the XML adapter must be created in the SAP_HPH schema.

➔ Tip

You can rename these virtual tables to anything you want.

- d. To avoid early timeouts during complex or extensive data integration runs, increase the time allowed before your SAP HANA Enterprise Information Management data provisioning server session times out.

To do so, you can use the following example SQL statements, which increase the timeout to 86,400 seconds:

```
ALTER SYSTEM ALTER CONFIGURATION ('dpserver.ini', 'SYSTEM') SET ('framework', 'usePrefetch') = 'true' WITH RECONFIGURE;
ALTER SYSTEM ALTER CONFIGURATION ('dpserver.ini', 'SYSTEM') SET ('framework', 'prefetchLimit') = '1' WITH RECONFIGURE;
ALTER SYSTEM ALTER CONFIGURATION ('dpserver.ini', 'SYSTEM') SET ('framework', 'prefetchTimeout') = '86400' WITH RECONFIGURE;
```

For more information, see the Administration Guide for SAP HANA Enterprise Information Management.

3. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: <http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#>

4. Open the *Plug-In Cockpit* app.
5. Select and activate the SAP Connected Health Platform HIX Plug-In.
6. Open the *Data Integration Cockpit* app.
7. Create a profile for data integration.
 - a. Open the *Run and Manage Profiles* view.
 - b. Choose *Create Profile*.
 - c. Choose the relevant data integration extension.

➔ Remember

Data integration extensions are only available if the corresponding plug-in is activated.

- d. Optionally, choose a DW Source.

DW Source is an optional attribute used in the clinical data warehouse to filter data during integration. This means, only data matching the specified DW Source will be integrated during the profile run.

For HIX files, DW Sources are derived from the OrgID, which is based on the <sender_location_code> element of the HIX file, and the Source, which is based <system_type> and <product_name> elements of the HIX file.

The assignment is done automatically during load and a 5-digit number is chosen for each DW Source. These derived DW Sources are available for selection when creating profiles.

- e. Optionally, choose a schedule for running the profile.

The schedule is based on an existing schedule configuration. You can create a new schedule configuration by choosing the **+** icon.

- f. Specify the required parameters.

For more information about the profile parameters, see [HIX Extension \[page 31\]](#).

8. Activate the profile and choose *Run Profile*.
9. To monitor the status of the profile run, open the *Monitor Profile Runs* view.

Results

After the profile run has completed, you can confirm the data was correctly integrated by either viewing the data in the target tables of the data integration extension or searching for the data in the end-user applications.

Related Information

[HIX Extension \[page 31\]](#)

[Overview of the HIX Format \[page 33\]](#)

[Adapting XML Sources to HIX \[page 33\]](#)

[XML Schema Definition File \[page 37\]](#)

[Example XML for HIX Data \[page 46\]](#)

2.2.1 HIX Extension

SAP Connected Health provides a data integration extension for integrating structured and unstructured clinical data in HIX format. You configure this extension in the [Data Integration Cockpit](#).

Profile Parameters

The following parameters are available when creating profiles for the HIX extension:

Table 17:

Parameter	Description	Default
File Filter	<p>Absolute path to a specific HIX file. Wildcards and underscores (% and _) are supported.</p> <div style="background-color: #fff9c4; padding: 5px;"> <p>i Note</p> <p>If you want to find an exact string that contains the wildcard characters "%", "_", or "\", you can escape their function by inserting a backslash (\).</p> <p>For example, <code>folder_de\%file</code> would find a file named <code>folder_de %file</code>.</p> </div>	<null>
File List Table	<p>Name of the FILES virtual table including schema</p> <p>This virtual table needs to be created after installation, with the technical name <code>SAP_HPH_VT_FILES</code>.</p>	"SAP_HPH"."HIX_FILELIST"
Log Detail Table	<p>Name of the LOG_DETAIL virtual table including schema</p> <p>This virtual table needs to be created after installation, with the technical name <code>SAP_HPH_VT_LOG_DETAIL</code>.</p>	"SAP_HPH"."HIX_LOG_DETAIL"
Last Update	Modified date of the HIX file must be more recent than this timestamp	1900-01-01 00:00:00

Parameter	Description	Default
Staging Table	Name of the XML_STAGE virtual table including schema This virtual table needs to be created after installation, with the technical name SAP_HPH_VT_XML_STAGE.	"SAP_HPH"."HIX_STAGE"

Target Tables

When you integrate HIX data using the HIX extension, the following SAP Connected Health tables are populated:

Table 18:

Table	Purpose
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities.Patient_Key"	Stores the business key for the patient
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities.Patient_Attr"	Stores attributes taken from the patient details.
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities.Interaction_Key"	Stores the business key for the interaction
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities.Interaction_Attr"	Stores recognized elements/attributes, such as type, taken from the interaction
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntitiesEAV.Interaction_Details"	Stores key/value pairs taken from the interaction detail.
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities.Practitioner_Key"	Stores the business key for the practitioner
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities.Practitioner_Attr"	
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities.Patient_Practitioner_Link"	
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities.Patient_Practitioner_Link_Attr"	
"SAP_HPH"."sap.hc.hph.di.documents.staging.db.models::DocumentStage.ProcessingStage"	Temporarily stores embedded unstructured data taken from interaction details. The Extension for Processing Unstructured Documents takes unstructured data from this staging table for further processing.

2.2.2 Overview of the HIX Format

The health data integration XML (HIX) format is an XML format specifically designed for importing clinical data into SAP Connected Health.

HIX files can contain the following types of information:

- Structured information from a patient's electronic medical records
- Unstructured information from database sources
This information can be included as either character large object data (CLOB) or binary large object data (BLOB).
- Unstructured information from documents
This information can be included as either large text documents or large binary documents, specifically PDF or Word documents.

The HIX format is similar to the clinical data model in that it is built around a design focused on key-value pairs that are stored together with metadata. Applications running on SAP Connected Health convert the keys into table column headings while the values become the values of a given row under each matching column heading.

Attachments can be combined with the structured data key-value pairs into a single file. Alternatively, attachments can be devoted to a separate file with sufficient complex elements provided to identify the location of the attachment with the appropriate interaction and interaction detail record.

Further Advantages

The HIX format has the following additional advantages over other common formats:

- Flexibility of data identification through the use of dynamic key values that allow for the dynamic definition of column headings based upon values in the received data.
- Flexibility to receive updates of the data on a business level entity, such as patient. Through the HIX format, you can send only new data, such as lab results or pathology report, for a business level entity without having to resend all of the data on the entity.

2.2.3 Adapting XML Sources to HIX

In most implementation scenarios, to integrate clinical data into SAP Connected Health using the HIX data integration extension, you'll have to adapt your XML source files to the HIX format.

To support you in adapting your content, this guide provides the XML schema definition file for the HIX format, as well as an example HIX file.

As an overview, a HIX file contains the following elements:

Table 19:

Element	Target Tables in Clinical Data Warehouse	Notes
patient	DWEntities.Patient_Key	The HIX data integration extension recognizes the <code>pid</code> element/attribute.
patient_details patient_detail	DWEntities.Patient_Attr	<p>The HIX data integration extension recognizes this element only if the key is in:</p> <ul style="list-style-type: none"> • City of residence • Date of birth • Date of death • Gender (M or F) • Patient address • Patient country • Patient email • Patient fax number • Patient first name • Patient home telephone number • Patient marital status • Patient mobile number • Patient multiple birth order (use a single digit, such as 1, 2, or 3, or null) • Patient nationality • Patient post office box • Patient postal code • Patient street number • Patient surname • Patient title • Region • State • Valid from • Valid to <p>The attributes <code>code_system</code>, <code>code_system_version</code>, and <code>code_value</code> are recognized and mapped only for the keys <code>Patient title</code>, <code>Patient nationality</code>, <code>Gender</code>, <code>Patient country</code>, and <code>Patient marital status</code>.</p>

Element	Target Tables in Clinical Data Warehouse	Notes
interaction	<ul style="list-style-type: none"> • DWEntities.Interaction_Key • DWEntities.Interaction_Attr 	<p>The HIX data integration extension recognizes the following elements/attributes:</p> <ul style="list-style-type: none"> • src_pkid • type • code_system • code_system_version • code_value • isvoided • timestamp
interaction_detail	<ul style="list-style-type: none"> • DWEntitiesEAV.Interaction_Detail • DWDocuments.Interaction_Documents_Link • Document_Stage.Processing_Stage 	<p>The HIX data integration extension recognizes the following elements/attributes:</p> <p>All key/value pairs including attributes code_system, code_system_version, and code_value are recognized and mapped for all keys and values.</p> <p>For embedded documents, please make sure you base64-encode your document and provide the following value attributes:</p> <ul style="list-style-type: none"> • type="BLOB" • encoding="BASE64" • mime_type="<your_mime_type>"
provider	DWEntities.Practitioner_Key	<p>The HIX data integration extension recognizes the pid element/attribute.</p>

Element	Target Tables in Clinical Data Warehouse	Notes
provider_details	DWEntities.Practitioner_Attr	<p>The HIX data integration extension recognizes this element only if the key is in:</p> <ul style="list-style-type: none"> • City of residence • Date of birth • Date of death • Gender (M or F) • Provider address • Provider country • Provider email • Provider fax number • Provider first name • Provider home telephone number • Provider marital status • Provider mobile number • Provider nationality • Provider post office box • Provider postal code • Provider street number • Provider surname • Provider title • Region • State • Valid from • Valid to <p>The attributes <code>code_system</code>, <code>code_system_version</code>, and <code>code_value</code> are recognized and mapped only for the keys <code>Provider title</code>, <code>Provider nationality</code>, <code>Gender</code>, <code>Provider country</code>, and <code>Provider marital status</code>.</p>

General Notes

When adapting clinical data to the HIX format, take note of the following:

- A HIX file also contains the elements `message_header` and `provider`; however, these are not recognized by the provided HIX data integration extension and, therefore, have no impact.
- For all attributes using dates, you must use the date format YYYY-MM-DD. Timestamp fields must use the format YYYY-MM-DD HH24:MI:SS. All dates and times must be UTC.

- If the key/value pair of an attribute doesn't occur in the HIX file or if its value element is empty, the attribute is set to <empty string> if it is of the type String or it is set to NULL in case of numeric/date types.
- If there is no section <patient_details> in the HIX file, no attributes (including "isvoided") are updated. In other words, to update "isvoided", the <patient_details> section must exist, even if empty.

2.2.4 XML Schema Definition File

Code Syntax

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
  <xs:element name="hix_clinical_document">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="message_header"/>
        <xs:element ref="patient"/>
      </xs:sequence>
      <xs:attribute name="hix_version" use="required"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="message_header">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="creation_timestamp"/>
        <xs:element ref="test_data_indicator"/>
        <xs:element ref="phi_data_indicator"/>
        <xs:element ref="sender"/>
        <xs:element ref="creator_agent"/>
        <xs:element ref="receiver"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="creation_timestamp" type="xs:string"/>
  <xs:element name="test_data_indicator" type="xs:boolean"/>
  <xs:element name="phi_data_indicator" type="xs:boolean"/>
  <xs:element name="sender">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="sender_ref_id"/>
        <xs:element ref="sender_code"/>
        <xs:element ref="sender_location_code"/>
        <xs:element ref="system"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="sender_ref_id" type="xs:string"/>
  <xs:element name="sender_code" type="xs:string"/>
  <xs:element name="sender_location_code" type="xs:string"/>
  <xs:element name="creator_agent">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="creator_org_name"/>
        <xs:element ref="software"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="creator_org_name" type="xs:string"/>
  <xs:element name="software">
```



```

<xs:complexType>
  <xs:sequence>
    <xs:element ref="vendor"/>
    <xs:element ref="product_name"/>
    <xs:element ref="product_version"/>
    <xs:element ref="configuration_version"/>
    <xs:element ref="canonical_data_model_version"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="product_version" type="xs:string"/>
<xs:element name="configuration_version" type="xs:string"/>
<xs:element name="canonical_data_model_version" type="xs:string"/>
<xs:element name="receiver">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="receiver_code"/>
      <xs:element ref="receiver_location_code"/>
      <xs:element ref="system"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="receiver_code" type="xs:string"/>
<xs:element name="receiver_location_code">
  <xs:complexType/>
</xs:element>
<xs:element name="patient">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" ref="patient_details"/>
      <xs:element minOccurs="0" maxOccurs="unbounded" ref="interaction"/>
    </xs:sequence>
    <xs:attribute name="isvoided" use="required"/>
    <xs:attribute name="pid" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="patient_details">
  <xs:complexType>
    <xs:sequence>
      <xs:element maxOccurs="unbounded" ref="patient_detail"/>
    </xs:sequence>
    <xs:attribute name="code_system" use="required"/>
    <xs:attribute name="code_system_version" use="required"/>
    <xs:attribute name="code_value" use="required"/>
    <xs:attribute name="isvoided" use="required"/>
    <xs:attribute name="src_pkid" use="required"/>
    <xs:attribute name="timestamp" use="required"/>
    <xs:attribute name="type" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="patient_detail">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="key"/>
      <xs:element ref="value"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="interaction">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" ref="providers"/>
      <xs:element maxOccurs="unbounded" ref="interaction_detail"/>
    </xs:sequence>
    <xs:attribute name="code_system" use="required"/>
    <xs:attribute name="code_system_version" use="required"/>
    <xs:attribute name="code_value" use="required"/>
    <xs:attribute name="isvoided" use="required"/>
  </xs:complexType>
</xs:element>

```

```

    <xs:attribute name="src_pkid" use="required"/>
    <xs:attribute name="timestamp" use="required"/>
    <xs:attribute name="type" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="providers">
  <xs:complexType>
    <xs:sequence>
      <xs:element maxOccurs="unbounded" ref="provider"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="provider">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" ref="provider_detail"/>
    </xs:sequence>
    <xs:attribute name="isvoided"/>
    <xs:attribute name="provider_ID" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="provider_detail">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="key"/>
      <xs:element ref="value"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="interaction_detail">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="key"/>
      <xs:element ref="value"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="system">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="system_type"/>
      <xs:element ref="vendor"/>
      <xs:element ref="product_name"/>
      <xs:element ref="version"/>
      <xs:element ref="system_instance"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="system_type" type="xs:string"/>
<xs:element name="version" type="xs:string"/>
<xs:element name="system_instance" type="xs:string"/>
<xs:element name="vendor" type="xs:string"/>
<xs:element name="product_name" type="xs:string"/>
<xs:element name="key">
  <xs:complexType mixed="true">
    <xs:attribute name="code_system" use="required"/>
    <xs:attribute name="code_system_version" use="required"/>
    <xs:attribute name="code_value" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="value">
  <xs:complexType mixed="true">
    <xs:attribute name="code_system"/>
    <xs:attribute name="code_system_version"/>
    <xs:attribute name="code_value"/>
    <xs:attribute name="encoding"/>
    <xs:attribute name="mime_type"/>
    <xs:attribute name="type" use="required"/>
  </xs:complexType>
</xs:element>

```

```

</xs:complexType>
</xs:element>
</xs:schema>

```

Related Information

[XML Content Reference - Message Header \[page 40\]](#)

[XML Content Reference - Patient \[page 42\]](#)

[XML Content Reference - Interaction \[page 44\]](#)

2.2.4.1 XML Content Reference - Message Header

Element: message_header

Table 20:

Attribute	Example Value	Description
creation_timestamp	2009-04-23 15:35:03	When was the file created
test_data_indicator	False	Data is for test purposes only
phi_data_indicator	True	Internal use

Element: sender

Table 21:

Attribute	Description
sender_code	Code of the sending organization
sender_location_code	Code for location of sender, such as the data center or branch
sender_ref_id	Sender's tag for this dataset

Element: system (parent: sender)

Table 22:

Attribute	Example Value	Description
system_type	ERP	Type of system
vendor	SAP	Vendor name
product_name	IS-H	
version	6.03	Version number of system
system_instance	1	Instance number

Element: creator_agent

Table 23:

Attribute	Example Value	Description
vendor	Health Data Inc.	Organization that created the data

Element: software (parent: creator_agent)

Table 24:

Attribute	Example Value	Description
vendor		Company providing the service for <creator_org_name>
product_name		Product used for extraction
version	4.5	Product version
config_version	1.2b	Configuration or profile version used
canonical_data_model_version	1.0	Which data model version was it created for

Element: receiver

Table 25:

Attribute	Description
receiver_code	Code of the receiving organization
receiver_location_code	Location of the receiving organization

Element: system (parent: receiver)

Table 26:

Attribute	Example Value	Description
system_type	HEC	Type of software system
vendor	SAP	Name of the vendor of the receiving system
product_name	SAP Platform for Health	Product name of the receiving system
version	1.0 SP2	Version of the receiving system
system_instance	01	Instance number

2.2.4.2 XML Content Reference - Patient

Element: patient

Table 27:

Attribute	Example Value	Description
pid	ID47110815	ID of the patient in the source system
isvoided	0 or 1	Indicator if patient should be deleted

Element: patient_details

Table 28:

Attribute	Example Value	Description
src_pkid		Same as pid
timestamp	2015-04-12 16:43:30	Last modified date of the patient details
type	Patient Demographics	Possibly codified type of patient_details
code_value	123456	patient_details code
code_system	SNOMEDCT_US	patient_details code system
code_system_version	2015	patient_details code system version
isvoided	0 or 1	indicator if patient_details should be deleted

Element: patient_detail (parent: patient_details)

Table 29:

Child Element	Attributes	Example Value	Description
key	<ul style="list-style-type: none"> code_value code_system code_system_version 	<ul style="list-style-type: none"> 123456 SNOMEDCT_US 2015 	<ul style="list-style-type: none"> Key code for patient_detail Key code system for patient_detail Version of key code system for patient_detail
value	<ul style="list-style-type: none"> code_value code_system code_system_version 	<ul style="list-style-type: none"> 123456 SNOMEDCT_US 2015 	<ul style="list-style-type: none"> Value code for patient_detail Value code system for patient_detail Version of value code system for patient_detail

2.2.4.3 XML Content Reference - Interaction

Element: interaction (parent: patient)

Table 30:

Attribute	Example Value	Description
src_pkid	ID47110815	Same as pid
timestamp	2015-04-12 16:43:30	Last modified date of the interaction
type	Diagnosis	Possibly codified type of interaction
code_value	123456	Interaction's value code
code_system	SNOMEDCT_US	Interaction's value code system
code_system_version	2015	Interaction's value code system version
isvoided	0 or 1	Indicator if interaction should be deleted

Element: provider (parent: interaction)

Table 31:

Attribute	Example Value	Description
provider_id	ID237464	ID of the provider of this interaction (physician, nurse)
isvoided	0 or 1	Indicator if provider should be deleted

Element: provider_detail (parent: provider)

Table 32:

Child Element	Attributes	Example Value	Description
key	<ul style="list-style-type: none">code_valuecode_systemcode_system_version	<ul style="list-style-type: none">123456SNOMEDCT_US2015	<ul style="list-style-type: none">Key code for provider_detailkey code system for provider_detailversion of key code system for provider_detail
value	<ul style="list-style-type: none">code_valuecode_systemcode_system_version	<ul style="list-style-type: none">123456SNOMEDCT_US2015	<ul style="list-style-type: none">Value code for provider_detailValue code system for provider_detailVersion of value code system for provider_detail

Element: interaction_detail (parent: interaction)

Table 33:

Child Element	Attributes	Example Value	Description
key	<ul style="list-style-type: none">code_valuecode_systemcode_system_version	<ul style="list-style-type: none">123456SNOMEDCT_US2015	<ul style="list-style-type: none">Key code for interaction_detailkey code system for interaction_detailVersion of key code system for interaction_detail

Child Element	Attributes	Example Value	Description
value	<ul style="list-style-type: none"> type code_value code_system code_system_version mime_type encoding 	<ul style="list-style-type: none"> BLOB 123456 SNOMEDCT_US 2015 application / pdf "BASE64" 	<ul style="list-style-type: none"> Content type of the value Value code for provider_detail Value code system for provider_detail Version of value code system for provider_detail mime_type type of the value in case of type="BLOB" encoding of the value in case if type "BLOB"

2.2.5 Example XML for HIX Data

Sample Code

```

<hix_clinical_document hix_version="V 0.93d">
  <message_header>
    <creation_timestamp>2009-01-29 15:34</creation_timestamp>
    <test_data_indicator>>false</test_data_indicator>
    <phi_data_indicator>>false</phi_data_indicator>
    <sender>
      <sender_ref_id>file 4711</sender_ref_id>
      <sender_code>XY Practice</sender_code>
      <sender_location_code>XY WSH</sender_location_code>
      <system>
        <system_type>XY BI</system_type>
        <vendor>XY EPIC</vendor>
        <product_name>XY Clarity</product_name>
        <version>V1.6</version>
        <system_instance>Q 33</system_instance>
      </system>
    </sender>
    <creator_agent>
      <creator_org_name>XY Provider</creator_org_name>
      <software>
        <vendor>XY Provider</vendor>
        <product_name>XY Harmony</product_name>
        <product_version>Enterprise Version 7</product_version>
        <configuration_version>V 1.0</configuration_version>
        <canonical_data_model_version>V 1.0</
canonical_data_model_version>
      </software>
    </creator_agent>
    <receiver>
      <receiver_code>XY ABC</receiver_code>
      <receiver_location_code></receiver_location_code>
      <system>
        <system_type>XY ABC</system_type>

```

```

        <vendor>XY ABC</vendor>
        <product_name>XY ABC</product_name>
        <version>V 1.6</version>
        <system_instance></system_instance>
    </system>
</receiver>
</message_header>
<patient_pid="XY A86743" isvoided = "XY N">
    <patient_details src_pkid="XY 1234567890" timestamp="2009-01-29
15:34" type="XY Demographics" code_value="XY 21321" code_system="SNO MEDCT_US"
code_system_version="V 2012" isvoided = "XY N">
    <patient_detail>
        <key code_value="XY 130470349875" code_system="XY
SNOMEDCT_US" code_system_version="V 2012">DOB</key>
        <value type="XY date">1952-12-04</value>
    </patient_detail>
    <patient_detail>
        <key code_value="XY 2108740" code_system="XY SNOMEDCT_US"
code_system_version="V 2012">Fname</key>
        <value type="string">Jack</value>
    </patient_detail>
</patient_details>
    <interaction src_pkid="XY 1234567890" timestamp="2009-01-29 15:34"
type="XY Diagnosis" code_value="XY 21321" code_system="SNO MEDCT_US"
code_system_version="V
2012" isvoided = "XY N">
    <providers>
        <provider provider_ID="XY 4711" isvoided = "XY N">
            <provider_detail>
                <key code_value="XY 2198347676" code_system="XY
SNOMEDCT_US" code_system_version="V 2012">npi</key>
                <value type="XY string" code_value="XY 153.6"
code_system="XY ICD_CM" code_system_version="V 9">1114977923</value>
            </provider_detail>
            <provider_detail>
                <key code_value="XY 2198347676" code_system="XY
SNOMEDCT_US" code_system_version="V 2012">last_name</key>
                <value type="XY string" code_value="XY 153.6"
code_system="XY ICD_CM" code_system_version="V 9">Miller</value>
            </provider_detail>
            <provider_detail>
                <key code_value="XY 2198347676" code_system="XY
SNOMEDCT_US" code_system_version="V 2012">title</key>
                <value type="XY string" code_value="XY 153.6"
code_system="XY ICD_CM" code_system_version="V 9">MD</value>
            </provider_detail>
            <provider_detail>
                <key code_value="XY 2198347676" code_system="XY
SNOMEDCT_US" code_system_version="V 2012">role</key>
                <value type="XY string" code_value="XY 153.6"
code_system="XY ICD_CM" code_system_version="V 9">treating physician</value>
            </provider_detail>
        </provider>
    </providers>
    <provider provider_ID="XY 4713">
    </provider>
</interaction_detail>
    <interaction_detail>
        <key code_value="XY 2198 347676" code_system="XY
SNOMEDCT_US" code_system_version="V 2012">pdx_cd</key>
        <value type="XY string" code_value="XY 153.6"
code_system="XY ICD_CM" code_system_version="V 9">153.6</value>
    </interaction_detail>
    <interaction_detail>
        <key code_value="XY 2198347676" code_system="XY
SNOMEDCT_US" code_system_version="V 2012">2354 325</key>
        <value type="XY string" code_value="XY 153.6"
code_system="XY ICD_CM" code_system version="V 9">Malignant ... colon</value>
    </interaction_detail>

```

```

        <interaction_detail>
          <key code_value="XY 2198347677" code_system="XY
SNOMEDCT_US" code_system_version="V 2012">sdxc_d</key>
          <value type="XY string" code_value="XY 249.0"
code_system="XY ICD_CM" code_system_version="V 9">249.0</value>
        </interaction_detail>
        <interaction_detail>
          <key code_value="XY 2198347678" code_system="XY
SNOMEDCT_US" code_system_version="V 2012">sdxc_d_desc</key>
          <value type="XY string" code_value="XY 249.0"
code_system="XY ICD_CM" code_system_version="V 9">Secondary ... or
unspecified</value>
        </interaction_detail>
      </interaction>

      <interaction src_pkid="XY 1234567890" timestamp="2009-01-29 15:34"
type="Medication" code_value="XY 95" code_system="XY ASCO"
code_system_version="V 2015" isvoided =
"XY N">
        <interaction_detail>
          <key code_value="XY 233443" code_system="XY SNOMEDCT_US"
code_system_version="V 2012">agent</key>
          <value type="XY string" code_value="XY 1-102166"
code_system="XY RXNORM" code_system_version="V 11">Tylenol</value>
        </interaction_detail>
        <interaction_detail>
          <key code_value="XY 35032148" code_system="XY SNOMEDCT_US"
code_system_version="V 2012">dose</key>
          <value type="XY int">100</value>
        </interaction_detail>
        <interaction_detail>
          <key code_value="XY 2442244" code_system="XY SNOMEDCT_US"
code_system_version="V 2012">unit_of_measure</key>
          <value type="XY string">ml</value>
        </interaction_detail>
        <interaction_detail>
          <key code_value="2198347676" code_system="SNOMEDCT_US"
code_system_version="2012">note</key>
          <value type="BLOB" code_value="" code_system=""
code_system_version="" mime_type="application/pdf" encoding="BASE64" >
ivBOw0KGgoAAAANSUheEUgAAABAAAAAQ
CAYAAAAf8/9hAAAABGdBTUEAAK/INwWK
6QAAAB10RVh0U29mdHdhcmUAQWRvYmUg
SWlhZ2VSZWZkeXhJZTAAAJOSURBVdjL
pZl9T1RBFIAf3buAoBgJ8rl6QVBJVNDc
ShMLOhBj6T+wNUADjY0WmpBIgYpAjL/A
ShJ+gVYYRPIony5IETkQxZ2770zc2fG
YpflQy2MJzk5J5M5z/vO5ESstfxPx4e
rL4Zuh4pLnoaiUZdq7XAGKzRJVbIBZ3J
PLJaD9c/eCj/CFgzfN15qK5q8EhTXdxx
LKgQjAFr0NK0ppOpt9n51D2gd2cmsvOE
lVcvOoprKvuPtriNzsY8rH+H0ECoQEg4
WklYlczP8akZby51p6G3b6QAWB14311S
VTlUfuZE3NmYh9Vl0HkHSuVq4ENFNWfd
C+uJ5JI/9/V2Y//rkShA1HF6yk/VxJ0f
07CcgkCB7+fSC8Dzcy7mp419/khlUzwe
caI9hT+wRrsOISylcsphCFL11RXIvBmp
YDZJrKYRjHELACNEgC/KCQQofWBQ5nuV
64UAP8AEfrDrQEiLlJD18+p7BguwfAoB
UmKEsLsAGZSiFWxtgWPP4gGAkuB5YDRW
ylKAKIDJZBa1H8Kx47C1Cdls7qLnQTzf
fQ+201B7EiU1ent7sQBQ6+vdq2PJ5dC9
ABWlsJnOQbL5Qc/HpNOYehf/41W+jY4v
h2tr3fsWafRwzRt1DW5f9aVzjUVj72Fm
CqzBypBQCKzbyLp8jZUPo70ZyYm7bYkv
w/sAAFMD7V3lp5sGqs+fjRcZhVYKY0xu
pwysfpogk0jcb5ucffbbKu9Esv1K11N2
+Ekk5rg2DIXRmog1Jdr3F/Tm5mO0edc6

```

```

MSP/CvjX+AV0DoH1Z+D54gAAAABJRU5E
rkJggg==
    </value>
      </interaction_detail>
    </interaction>
  </patient>
</hix_clinical_document>

```

2.3 Integrating Genomic Data

In SAP Connected Health, users can view genomic information in the [Variant Browser](#) or use genomic information to filter and analyze patients within other applications running on the platform. To enable this, you must integrate your available genomic data by using the provided data integration extensions.

SAP provides the following data integration extensions for genomic data:

Table 34:

Data Integration Extension	Purpose	Further Details
SAP Connected Health Platform VCF SDA Extension	Integrate sequenced genome information taken from patient samples or studies	VCF SDA Extension [page 66]
SAP Connected Health Platform VCF Table Extension	Alternative to VCF SDA Extension	VCF Table Extension [page 69]
SAP Connected Health Platform FASTA File Processing Extension	Integrate FASTA files to process sequence data of a reference genome	FASTA Extension [page 52]
SAP Connected Health Platform GFF3 File Processing Extension	Integrate feature data for reference genome in GFF3 format	GFF3 Extension [page 58]
SAP Connected Health Platform PED File Processing Extension	Integrate pedigree data taken from samples and studies in PED format	
SAP Connected Health Platform BED File Processing Extension	Integrate region-specific gene information within an annotation track in BED format	
SAP Connected Health Platform VCF Annotation Extension	Integrate variant annotation for sample variant data	
SAP Connected Health Platform GFF3 Annotation Extension	Integrate feature annotation for GFF3	

Related Information

[Integrating Reference Genome Data \[page 50\]](#)

[Integrating Feature Data \[page 54\]](#)

[Integrating Sample Variant Data \[page 60\]](#)

2.3.1 Integrating Reference Genome Data

Prerequisites

- An SAP HANA Enterprise Information Management data provisioning agent is installed and configured on the source system.
- The FASTA adapter is deployed and configured.
For more information, see the SAP Connected Health Platform Implementation Guide.
- You have activated the SAP Connected Health Platform FASTA File Processing plug-in in the *Plug-In Cockpit*.
- You have configured the SAP Connected Health Platform FASTA File Processing extension in the *Data Integration Cockpit*.
- You are assigned the role `HC_HPH_PLUGINS_ADMIN`.

Context

In SAP Connected Health, genomic information taken from samples or studies related to specific patients or cohorts is compared to a reference genome to, for example, identify variants. For this reference genomic information, SAP Connected Health supports the FASTA format for reference genome data.

SAP provides the following data integration extensions for reference genome data:

Table 35:

Data Integration Extension	Purpose	Further Details
SAP Connected Health Platform FASTA File Processing Extension	Integrate reference genome information in FASTA format	FASTA Extension [page 52]

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Open the *Data Integration Cockpit* app.
3. Open the *Run and Manage Profiles* view.
4. Create or select a profile to use when integrating the reference genome data.

If you haven't already created a profile, see, [Creating Profiles for Data Integration Extensions \[page 96\]](#).

Profiles for integrating reference genome data specify where the data is taken from. For each profile, you must define the following parameters:

Table 36:

Parameter	Description
Reference ID	Names/versions of a reference genome
Schema name of the virtual table used for accessing FASTA-related virtual tables	Name of the schema where virtual tables are added
Chromosome Table	Name of the Chromosome virtual table You create and name this virtual table during installation. The technical name of the table on the remote source is <code>SAP_HPH_VT_HEADER_CHROMOSOMES</code> .
Sequence Table	Name of the Sequence virtual table You create and name this virtual table during installation. The technical name of the table on the remote source is <code>SAP_HPH_VT_REF_SEQUENCE</code> .
Log Detail Table	Name of the Log Detail virtual table You create and name this virtual table during installation. The technical name of the table on the remote source is <code>SAP_HPH_VT_LOG_DETAIL</code> .
Log Trace Table	Name of the Log Trace virtual table You create and name this virtual table during installation. The technical name of the table on the remote source is <code>SAP_HPH_VT_LOG_TRACE</code> .
Trim Chromosome Prefix	Trim the specified chromosome prefix character. You can specify the character to be trimmed from the chromosome names. The characters are case sensitive.

i Note

You must ensure the profile is active.

- Choose *Run Profile*.
- To monitor the status of the data integration run, choose the *Monitor Profile Runs* view to see the start time, duration, and status for each applied profile.

i Note

To cancel the data integration run, choose *Cancel* from the *Actions* column.

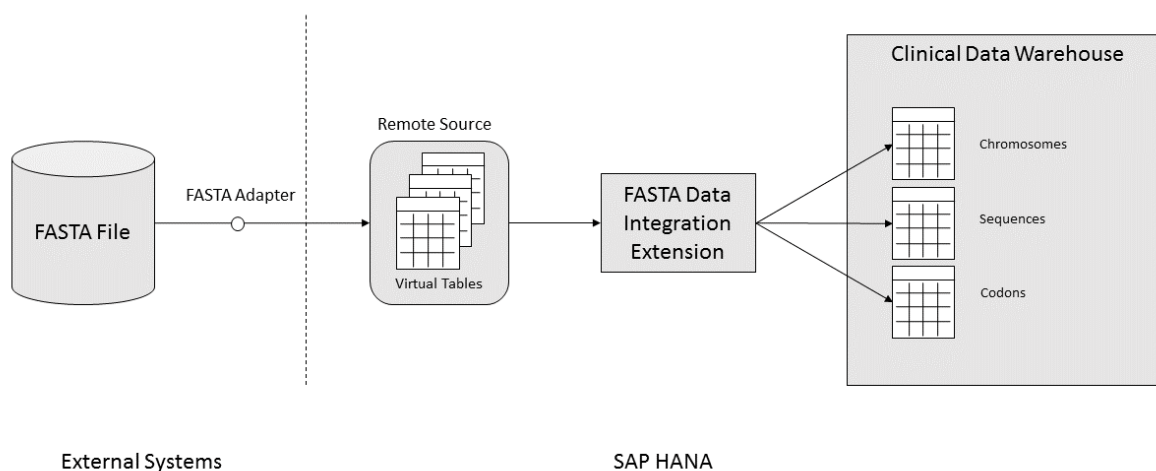
- To monitor any impacts on the related tables in the clinical data warehouse, use the *Clinical Data Warehouse Cockpit* app.

Related Information

[FASTA Extension \[page 52\]](#)

2.3.1.1 FASTA Extension

SAP Connected Health provides a data integration extension for integrating reference genome sequences in FASTA format. You configure this extension in the *Data Integration Cockpit*.



This data integration extension uses the deployed FASTA adapter to retrieve the FASTA file from the source system and store it in the corresponding SAP HANA virtual tables, which were created during system implementation. The data integration extension then transforms the data and stores it in the clinical data warehouse.

FASTA Format

FASTA format is a text-based format for representing nucleotide sequences, in which nucleotides are represented using single-letter codes. A sequence in FASTA format begins with a single-line description, followed by lines of sequence data. This is used in the variant analysis and annotation.

FASTA is a very simple format consisting of one or multiple sections with multiple lines each. A new section is indicated by a header starting with '>' followed by the section name (without spaces) and optionally a description separated by a space and ended by a newline. The following lines contain the sequence data encoded using ASCII characters of the bases A, C, G and T. The line separations are only for readability and have no meaning.

Sample Code

```
>chr1 Chromosome 1 of unknown species
CGACTACCTATTTCGAAAAAGCTTTCCAGGTAAAACCGTACAT
GTACCAATGAAGATCTCTCACGTTTACCATTGTGTAACCTTC
TCCCATAGGGTCAT
```

Profile Parameters

When you integrate FASTA data using the FASTA extension, you can create profiles to define how data is integrated. The following parameters are available when creating profiles for the FASTA extension:

Table 37:

Parameter	Description
Reference ID	Names/versions of a reference genome
Schema name of the virtual table used for accessing FASTA-related virtual tables	Name of the schema where virtual tables are added
Chromosome Table	Name of the Chromosome virtual table You create and name this virtual table during installation. The technical name of the table on the remote source is <code>SAP_HPH_VT_HEADER_CHROMOSOMES</code> .
Sequence Table	Name of the Sequence virtual table You create and name this virtual table during installation. The technical name of the table on the remote source is <code>SAP_HPH_VT_REF_SEQUENCE</code> .

Parameter	Description
Log Detail Table	Name of the Log Detail virtual table You create and name this virtual table during installation. The technical name of the table on the remote source is <code>SAP_HPH_VT_LOG_DETAIL</code> .
Log Trace Table	Name of the Log Trace virtual table You create and name this virtual table during installation. The technical name of the table on the remote source is <code>SAP_HPH_VT_LOG_TRACE</code> .
Trim Chromosome Prefix	Trim the specified chromosome prefix character. You can specify the character to be trimmed from the chromosome names. The characters are case sensitive.

Target Tables

When you integrate FASTA data using the FASTA extension, the following SAP Connected Health tables are populated:

Table 38:

Table	Purpose
"SAP_HPH"."sap.hc.hph.genomics.db.models::Reference.Chromosomes"	Stores the Chromosome header information like number, name, or description
"SAP_HPH"."sap.hc.hph.genomics.db.models::Reference.Sequences"	Stores the Sequence information
"SAP_HPH"."sap.hc.hph.genomics.db.models::Reference.Codons"	Stores the Amino Acid type information

2.3.2 Integrating Feature Data

Prerequisites

- An SAP HANA Enterprise Information Management data provisioning agent is installed and configured on the source system.

- To integrate feature data (GFF3), you must ensure that you've already integrated reference genome data (FASTA).
- The GFF3 adapter is deployed and configured.
For more information, see the SAP Connected Health Platform Implementation Guide.
 - For more information about deploying the GFF3 adapter, see .
- You have activated SAP Connected Health Platform GFF3 plug-in for genomic data in the *Plug-In Cockpit*.
- You have configured SAP Connected Health Platform GFF3 File Processing Extension in the *Data Integration Cockpit*.
- You are assigned the role `HC_HPH_PLUGINS_ADMIN`.

i Note

Additionally, you can configure SAP Connected Health Platform GFF3 Extension to run feature annotation, if required.

Context

In SAP Connected Health, genomic information taken from samples or studies related to specific patients or cohorts is compared to a reference genome to identify variants. For this reference genomic information, SAP Connected Health supports the GFF3 format for annotations for the reference genome.

SAP provides the following data integration extensions for reference genome data:

Table 39:

Data Integration Extension	Purpose	Further Details
SAP Connected Health Platform GFF3 File Processing Extension	Integrate feature information for reference genome in GFF3 format	GFF3 Extension [page 58]
SAP Connected Health Platform GFF3 Annotation Extension	Integrate feature annotation for GFF3	For more information, see .

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Open the *Data Integration Cockpit* app.
3. Open the *Run and Manage Profiles* view.
4. Create or select a profile to use when integrating the feature data.

If you haven't already created a profile, see [Creating Profiles for Data Integration Extensions \[page 96\]](#).

Profiles for importing genomic features specify where the data is taken from. For each profile, you must define the following parameters:

Table 40:

Parameter	Description
Call Annotation	Call the internal feature annotation procedures after the GFF3 file has been imported.
Reference ID	Names/versions of human genome references (Same as provided in the FASTA extension)
Schema Name	Name of the schema where virtual tables are added
Feature Table	Name of the Feature virtual table You create and name this virtual table during installation. The technical name of the table on the remote source is <code>SAP_HPH_VT_GFF3_FEATURE</code> .
Chromosome Table	Name of the Chromosome virtual table You create and name this virtual table during installation. The technical name of the table on the remote source is <code>SAP_HPH_VT_GFF3_CHROMOSOME</code> .
Log Detail Table	Name of the Log Detail virtual table You create and name this virtual table during installation. The technical name of the table on the remote source is <code>SAP_HPH_VT_LOG_TRACE</code> .
Log Trace Table	Name of the Log Trace virtual table You create and name this virtual table during installation. The technical name of the table on the remote source is <code>SAP_HPH_VT_LOG_TRACE</code> .

i Note

You must ensure the profile is active.

For more information about profiles, see [Creating Profiles for Data Integration Extensions \[page 96\]](#).

5. Choose *Run Profile*.
6. From the *Chromosome* view, map the feature chromosomes to the chromosomes of the reference genome you want to use.
 - a. Select a chromosome from a chromosome file.

➔ Tip

The *Size* column indicates how many base pairs a chromosome contains.

- b. Drag and drop the feature chromosome onto the reference chromosome.

i Note

For each feature file, you can map a feature chromosome to only one reference chromosome.

➔ Tip

You can also use the *Auto Map* feature to automatically map all chromosomes in the feature file to the corresponding chromosomes in the reference files. The system identifies chromosomes as matching if one of the following criteria is the same for both the feature and the reference chromosomes:

- Chromosome size
- Chromosome name

i Note

Any feature chromosomes that you don't map will not be imported.

7. To monitor the status of the data integration run, choose the *Monitor* view to see the start time, duration, and status for each applied profile.

i Note

To cancel the data integration run, choose *Cancel* from the *Actions* column.

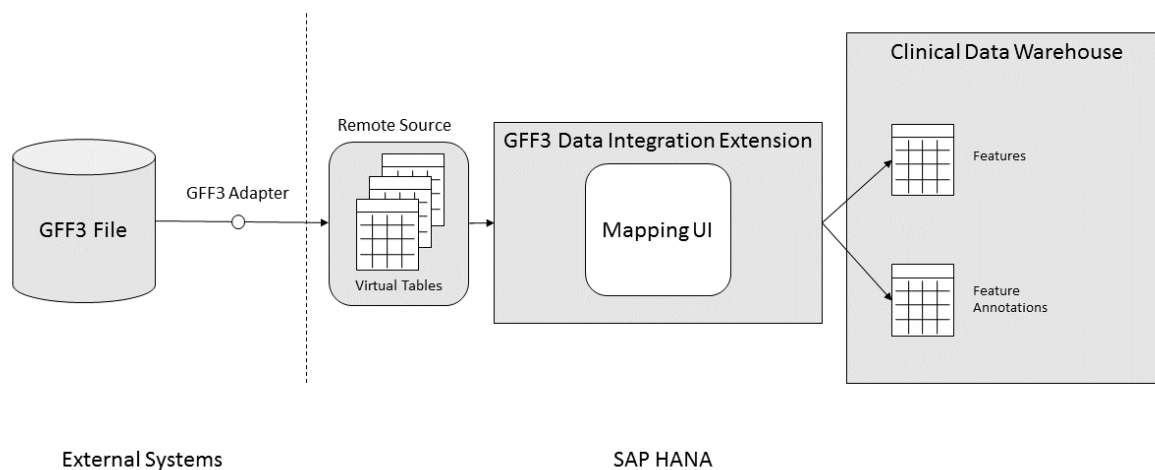
8. To monitor any impacts on the related tables in the clinical data warehouse, use the *Clinical Data Warehouse Cockpit* app.

Related Information

[GFF3 Extension \[page 58\]](#)

2.3.2.1 GFF3 Extension

SAP Connected Health provides a data integration extension for integrating feature data and annotations for the reference genome in GFF3 format. You configure this extension in the [Data Integration Cockpit](#).



This data integration extension uses the deployed GFF3 adapter to retrieve the GFF3 file from the source system and store it in the corresponding SAP HANA virtual tables, which were created during system implementation. Before the GFF3 data can be integrated into the clinical data warehouse, an administrative user must manually map the GFF3 data by using the provided [GFF3 Data Integration Mapping](#) user interface.

GFF3 Format

GFF3 is a standard file format for storing genomic features in a text file. GFF stands for Generic Feature Format. GFF3 files are plain text, 9 column, tab-delimited text files used to represent genomic sequence features.

A GFF3 file consists of one or more records, each of which represents a simple start to stop feature. Records are separated by newlines, one record per row. Each record has 9 fields, the last of which is optional. A record in GFF3 may represent many things: an exon, a transcript, a gene, etc. Arbitrary levels of hierarchy may be defined by using the 'ID' and 'Parent' attributes.

Sample Code

```
human15.1 . gene          214301  215772 . + . ID=HsG8283
human15.1 . mRNA         214360  215771 . + . Comments=fixed+one+splice
+junction;Parent=HsG8283;Evidence=700000069743825;Transcript_type=Novel_Transc
ript;Name=Novel+Transcript%2C+variant+%28partial%29;ID=HsT20206
human15.1 . CDS          214360  214441 . + . Parent=HsT20206
human15.1 . CDS          215299  215444 . + . Parent=HsT20206
human15.1 . CDS          215641  215766 . + . Parent=HsT20206
human15.1 . three_prime_UT 215767  215771 . + . Parent=HsT20206
```

Profile Parameters

The following parameters are available when creating profiles for the GFF3 extension:

Table 41:

Parameter	Description
Call Annotation	Call the internal feature annotation procedures after the GFF3 file has been imported.
Reference ID	Names/versions of human genome references (Same as provided in the FASTA extension)
Schema Name	Name of the schema where virtual tables are added
Feature Table	Name of the Feature virtual table You create and name this virtual table during installation. The technical name of the table on the remote source is <code>SAP_HPH_VT_GFF3_FEATURE</code> .
Chromosome Table	Name of the Chromosome virtual table You create and name this virtual table during installation. The technical name of the table on the remote source is <code>SAP_HPH_VT_GFF3_CHROMOSOME</code> .
Log Detail Table	Name of the Log Detail virtual table You create and name this virtual table during installation. The technical name of the table on the remote source is <code>SAP_HPH_VT_LOG_TRACE</code> .
Log Trace Table	Name of the Log Trace virtual table You create and name this virtual table during installation. The technical name of the table on the remote source is <code>SAP_HPH_VT_LOG_TRACE</code> .

Target Tables

When you integrate GFF3 data using the GFF3 extension, the following SAP Connected Health tables are populated:

Table 42:

Table	Purpose
<code>"SAP_HPH"."sap.hc.hph.genomics.db.models::Reference.Features"</code>	Stores the genome features

Table	Purpose
"SAP_HPH". " sap.hc.hph.genomics.db.models::Reference.FeaturesAnnotation"	Stores the feature annotations

2.3.3 Integrating Sample Variant Data

Prerequisites

- You have activated the relevant plug-ins for genomic data in the *Plug-In Cockpit*.
- You have configured the relevant data integration extensions in the *Data Integration Cockpit*. For more information, see [Creating Profiles for Data Integration Extensions \[page 96\]](#).
- You have deployed the BLOB adapter to a remote source and specified the location of your VCF files. For more information, see the SAP Connected Health Platform Implementation Guide.

i Note

The SAP HPH BLOB Adapter cannot integrate any files larger than 1 GB.

- You have integrated reference genome information using the FASTA and GFF3 extensions.
- You are assigned the role `HC_HPH_PLUGINS_ADMIN`.
- If you use the VCF Table extension, you have created tables in SAP HANA to act as a staging area from which the extension can retrieve VCF data.

Context

In SAP Connected Health, users can view genomic information taken from samples or studies relating to specific patients or cohorts. For this type of genomic information, SAP Connected Health supports the Variant Call Format (VCF).

SAP provides the following data integration extensions for VCF:

Table 43:

Data Integration Extension	Purpose	Further Details
SAP Connected Health Platform VCF SDA Extension	Primary extension for integrating VCF data	VCF SDA Extension [page 66]

Data Integration Extension	Purpose	Further Details
SAP Connected Health Platform VCF Table Extension	Secondary extension for integrating VCF data if VCF SDA extension fails or is not supported	VCF Table Extension [page 69]

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Open the *Data Integration Cockpit* app.
3. Open the *Run and Manage Profiles* view.
4. Create or select a profile to use when integrating the genomic data.

If you haven't already created a profile, see [Creating Profiles for Data Integration Extensions \[page 96\]](#).

Profiles for genomic data specify where the data is taken from.

For the VCF SDA extension, you must define the following profile parameters:

Table 44:

Parameter	Description
Remote Source	External source containing VCF files You define the name of the remote source during deployment of the related data integration adapter.
OrgID	The organizational unit ID used when searching for patients during sample mapping.

For the VCF Table extension, you must define the following profile parameters:

Table 45:

Parameter	Description
Schema name	Schema containing the input virtual table
Table Name	Name of the input virtual table
OrgID	The organizational unit ID used when searching for patients during sample mapping

i Note

You must ensure the profile is active.

5. Choose *Run Profile*.
6. From the *Mapping* view, select the specific set of VCF files to import and click *Next*.
7. From the *Chromosome* view, map the sample chromosomes to the chromosomes of the reference genome you want to use.
 - a. Select the reference genome you want to use.
 - b. Select a chromosome from a sample file.

→ Tip

The *Size* column indicates how many base pairs a chromosome contains.

- c. Drag and drop the sample chromosome onto the reference chromosome.

i Note

For each sample file, you can map a sample chromosome to only one reference chromosome.

→ Tip

You can also use the *Auto Map* feature to automatically map all chromosomes in the sample file to the corresponding chromosomes in the reference files. The system identifies chromosomes as matching if one of the following criteria is the same for both the sample and the reference chromosomes:

- Chromosome size
- Chromosome name

i Note

You can't have two chromosome files with the same names in a folder, or different folders within the source VCF file.

i Note

Any sample chromosomes that you don't map will not be imported.

i Note

The *All* node contains the list of all the chromosomes from the sample files. You can use the *All* node feature to map sample chromosomes to the reference chromosome. If you map a specific chromosome file from the *All* view, the system automatically maps all occurrences of that file in the sample files to the corresponding reference genome.

8. From the *Filter* view, select which filters you want to apply to the variant data model, and therefore, use during data import.

i Note

The filters must exist in the source VCF files.

- a. Select a filter from the sample files.
- b. Drag and drop the filter onto the *Add* entry under *Variant Data Model*.
- c. When prompted, specify the name and description you want to apply to the filter.

➔ Tip

You can also use the *Auto Map* feature to automatically map all filters where the name of the filter in the sample file matches the name of the corresponding filter in the variant data model.

i Note

The *All* node contains the list of all the filters from the sample files. You can choose a filter from the *All* node and the system automatically maps all occurrences of that filter in the sample files to the corresponding filter in the variant data model.

- 9. From the *Attribute* view, select the variant attributes and genotype attributes you want to import.
 - a. Select an attribute from the sample files.
 - b. Drag and drop the attribute onto the *Add* entry under *Variant Data Model*.
 - c. When prompted, specify the name and description you want to apply to the attribute.

➔ Tip

You can also use the *Auto Map* feature to automatically map all attributes where the group, name, allele type, and array size of the attribute in the sample file matches that of the corresponding attribute in the variant data model.

i Note

The *All* node contains the list of all the attributes from the sample files. You can choose an attribute from the *All* node, and the system automatically maps all occurrences of that attribute in the sample files to the corresponding attribute in the variant data model.

i Note

Attribute mappings are validated based on data type and array size.

The following validations are applied:

Table 46: Data Type Validations

Source	Target
Char	Char
Char	String

Source	Target
String	String
String	Structured
Flag	Flag
Integer	Integer
Integer	Float
Float	Float

Table 47: Array Size Validations

Source	Target
0	0
1	1
Allele	Allele
Unlimited	Unlimited
>1	Unlimited
>1	>1

10. From the [Sample](#) view, assign the samples you want to import to the relevant patients.
 - a. Choose [Add new patient](#) or [Add existing patient](#).

➔ Recommendation

If you want to integrate generic or test VCF data, we recommend that you create a new patient and name it accordingly. For VCF data from real samples or studies, we recommend that you map the samples only to existing patients, which are created during clinical data integration.

- b. Drag and drop the samples onto the corresponding interactions for each patient.

i Note

Samples relate to patients indirectly through interactions.

- c. When prompted, specify the sample type such as; Germline, Tumor, and Primary Tumor.
11. From the [Integration Settings](#) view, specify the relevant options to apply during data import.

The following options are available:

Table 48:

Option	Description
Variants	Specifies whether to import variants. This option is selected by default.
Genotype	Specifies whether to import genotypes. If you select this option, you must map your samples.
Haplotype	Specifies whether to import haplotypes. If you select this option, you must map your samples. <div style="background-color: #fff9c4; padding: 5px;"> <p>i Note</p> <p>If you want to import haplotypes, you must import genotypes.</p> </div>
Minimum variant quality	Specifies the minimum quality a variant must have to be included in the data import. You must specify a positive number. Variant quality values are specified within the source VCF files.
Annotation Method	Specifies which imported feature annotation you want to apply.
Variant Annotation Profile	Displays the list of active VCF annotation profiles.

- On the *Summary* tab, review the summarized details of the genomic data you want to integrate and choose *Submit* to start the data integration run.

➔ **Tip**

The *Valid for Integration* column indicates whether the mapped files will be included in the data integration run based on your settings and the validation rules.

- To monitor the status of the data integration run, choose the *Monitor Profile Runs* view to see the start time, duration, and status for each applied profile.

i Note

To cancel the data integration run, choose *Cancel* from the *Actions* column.

- To monitor any impacts on the related tables in the clinical data warehouse, use the *Clinical Data Warehouse Cockpit* app.

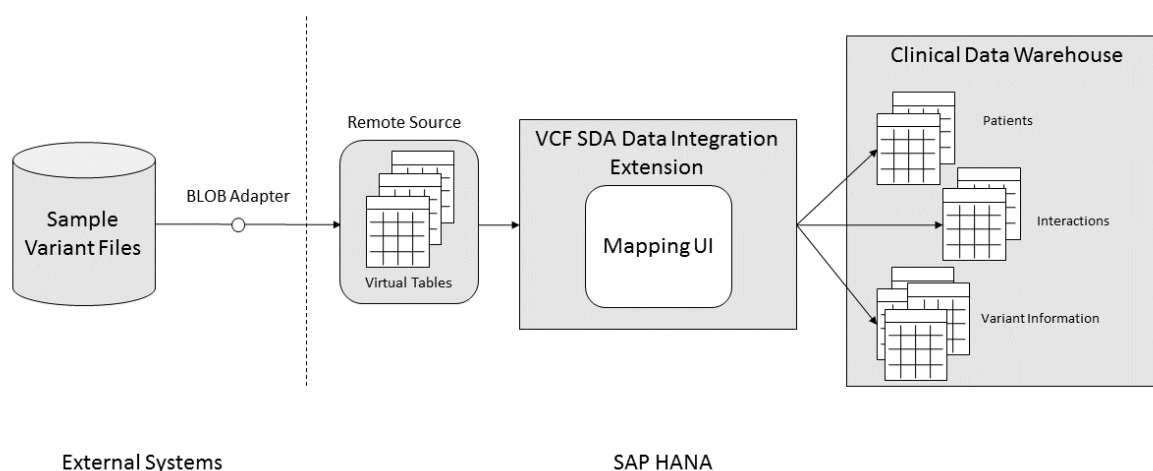
Related Information

[VCF SDA Extension \[page 66\]](#)

[VCF Table Extension \[page 69\]](#)

2.3.3.1 VCF SDA Extension

SAP Connected Health provides a data integration extension for integrating sample variant sequences in VCF format. You configure this extension in the *Data Integration Cockpit*.



This data integration extension uses the deployed BLOB adapter to retrieve VCF data from the source system and automatically creates virtual tables to store that data in SAP HANA. Before the VCF data can be integrated into the clinical data warehouse, an administrative user must manually map the VCF data by using the provided *VCF Data Integration Mapping* user interface.

VCF Format

VCF format is a very complex text format used to capture sequenced genome information taken from patient samples. This format encodes a multi-dimensional data model with extension options on all levels. This requires many tables to model it efficiently and most of these must be able to store optional attributes. Furthermore, this format might contain large amounts of data, which could impact the performance of the data integration extension.

Sample Code

```
##fileformat=VCFv4.2
##fileDate=20090805
##source=myImputationProgramV3.1
##reference=file:///seq/references/1000GenomesPilot-NCBI36.fasta
##contig=<ID=20,length=62435964,assembly=B36,md5=f126cdf8a6e0c7f379d618ff66beb2da,species="Homo sapiens",taxonomy=x>
```

```

##phasing=partial
##INFO=<ID=NS,Number=1,Type=Integer,Description="Number of Samples With Data">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Total Depth">
##INFO=<ID=AF,Number=A,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##FORMAT=<ID=HQ,Number=2,Type=Integer,Description="Haplotype Quality">
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT NA00001 NA00002 NA00003
20 14370 rs6054257 G A 29 PASS NS=3;DP=14;AF=0.5;DB;H2 GT:GQ:DP:HQ 0|
0:48:1:51,51 1|0:48:8:51,51 1/1:43:5:.,.
20 17330 . T A 3 q10 NS=3;DP=11;AF=0.017 GT:GQ:DP:HQ 0|0:49:3:58,50 0|
1:3:5:65,3 0/0:41:3
20 1110696 rs6040355 A G,T 67 PASS NS=2;DP=10;AF=0.333,0.667;AA=T;DB
GT:GQ:DP:HQ 1|2:21:6:23,27 2|1:2:0:18,2 2/2:35:4
20 1230237 . T . 47 PASS NS=3;DP=13;AA=T GT:GQ:DP:HQ 0|0:54:7:56,60 0|
0:48:4:51,51 0/0:61:2 20 1234567 microsat1 GTC G,GTCT 50 PASS NS=3;DP=9;AA=G
GT:GQ:DP 0/1:35:4 0/2:17:2 1/1:40:3

```

Profile Parameters

When you integrate VCF data using the VCF SDA extension, you can create profiles to define how data is integrated. The following parameters are available when creating profiles for the VCF SDA extension:

Table 49:

Parameter	Description
Remote Source	External source containing VCF files You define the name of the remote source during deployment of the related data integration adapter.
OrgID	The organizational unit ID used when searching for patients during sample mapping.

Target Tables

When you integrate VCF data using either the VCF SDA extension or the VCF Table extension, the following SAP Connected Health tables are populated:

Table 50:

Table	Purpose
"SAP_HPH". " sap.hc.hph.cdw.db.models::DWEEntities.Interactions_Key"	Stores the business key for the interaction.
"SAP_HPH". " sap.hc.hph.cdw.db.models::DWEEntitiesEAV.Interaction_Details"	Stores key/value pairs taken from the interaction detail.
"SAP_HPH". " sap.hc.hph.cdw.db.models::DWEEntities.Patient_Key"	Only required if a new patient is created during data integration
"SAP_HPH". " sap.hc.hph.cdw.db.models::DWEEntities.Patient_Attr"	Only required if a new patient is created during data integration
"SAP_HPH". "sap.hc.hph.genomics.db.models::General.Samples"	Imported samples and references to clinical data warehouse
"SAP_HPH". "sap.hc.hph.genomics.db.models::SNV.Headers"	Header content of VCF file
"SAP_HPH". "sap.hc.hph.genomics.db.models::SNV.Variants"	Variant specific data, scalar INFO attributes and filters
"SAP_HPH". "sap.hc.hph.genomics.db.models::SNV.VariantAlleles"	Reference and alternative alleles and allele-related INFO attributes
"SAP_HPH". "sap.hc.hph.genomics.db.models::SNV.VariantMultiValueAttributes"	Variant specific multi-dimensional INFO attributes
"SAP_HPH". "sap.hc.hph.genomics.db.models::SNV.VariantStructuredAttributes"	Attributes extracted from annotation attributes
"SAP_HPH". "sap.hc.hph.genomics.db.models::SNV.Genotypes"	Genotype specific data and scalar FORMAT attributes
"SAP_HPH". "sap.hc.hph.genomics.db.models::SNV.GenotypeAlleles"	Genotype specific allele counts and allele-related FORMAT attributes
"SAP_HPH". "sap.hc.hph.genomics.db.models::SNV.GenotypeMultiValueAttributes"	Genotype specific multi-dimensional FORMAT attributes

Table	Purpose
"SAP_HPH"."sap.hc.hph.genomics.db.models::SNV.Haplotypes"	Phased haplotype alleles
"SAP_HPH"."sap.hc.hph.genomics.db.models::SNV.VariantAnnotations"	Variant annotations based on internal annotation algorithm

2.3.3.2 VCF Table Extension

As an alternative to the VCF SDA extension, SAP Connected Health provides a data integration extension for integrating sample variant sequences in VCF format using tables. You configure this extension in the [Data Integration Cockpit](#).

You can use the VCF Table Extension if the VCF SDA Extension either fails or it isn't supported in your implementation scenario.

Profile Parameters

When you integrate VCF data using the VCF Table Extension, you can create profiles to define how data is integrated. The following parameters are available when creating profiles for the VCF Table Extension:

Table 51:

Parameter	Description
Schema name	Schema containing the input virtual table
Table Name	Name of the input virtual table
OrgID	The organizational unit ID used when searching for patients during sample mapping

Target Tables

When you integrate VCF data using either the VCF SDA extension or the VCF Table extension, the following SAP Connected Health tables are populated:

Table 52:

Table	Purpose
"SAP_HPH". " sap.hc.hph.cdw.db.models::DWEEntities.Interactions_Key"	Stores the business key for the interaction.
"SAP_HPH". " sap.hc.hph.cdw.db.models::DWEEntitiesEAV.Interaction_Details"	Stores key/value pairs taken from the interaction detail.
"SAP_HPH". " sap.hc.hph.cdw.db.models::DWEEntities.Patient_Key"	Only required if a new patient is created during data integration
"SAP_HPH". " sap.hc.hph.cdw.db.models::DWEEntities.Patient_Attr"	Only required if a new patient is created during data integration
"SAP_HPH". "sap.hc.hph.genomics.db.models::General.Samples"	Imported samples and references to clinical data warehouse
"SAP_HPH". "sap.hc.hph.genomics.db.models::SNV.Headers"	Header content of VCF file
"SAP_HPH". "sap.hc.hph.genomics.db.models::SNV.Variants"	Variant specific data, scalar INFO attributes and filters
"SAP_HPH". "sap.hc.hph.genomics.db.models::SNV.VariantAlleles"	Reference and alternative alleles and allele-related INFO attributes
"SAP_HPH". "sap.hc.hph.genomics.db.models::SNV.VariantMultiValueAttributes"	Variant specific multi-dimensional INFO attributes
"SAP_HPH". "sap.hc.hph.genomics.db.models::SNV.VariantStructuredAttributes"	Attributes extracted from annotation attributes
"SAP_HPH". "sap.hc.hph.genomics.db.models::SNV.Genotypes"	Genotype specific data and scalar FORMAT attributes
"SAP_HPH". "sap.hc.hph.genomics.db.models::SNV.GenotypeAlleles"	Genotype specific allele counts and allele-related FORMAT attributes
"SAP_HPH". "sap.hc.hph.genomics.db.models::SNV.GenotypeMultiValueAttributes"	Genotype specific multi-dimensional FORMAT attributes

Table	Purpose
"SAP_HPH"."sap.hc.hph.genomics.db.models::SNV.Haplotypes"	Phased haplotype alleles
"SAP_HPH"."sap.hc.hph.genomics.db.models::SNV.VariantAnnotations"	Variant annotations based on internal annotation algorithm

2.4 Integrating Unstructured Documents

Related Information

[Integrating Pathology Reports \[page 71\]](#)

[Integrating Medical Documents \[page 73\]](#)

[Integrating Documents Without Text Analysis \[page 75\]](#)

[Integrating Documents Using the BLOB Adapter and the Basic Extension \[page 79\]](#)

[Extension for Processing Unstructured Documents \[page 84\]](#)

2.4.1 Integrating Pathology Reports

Using the provided data integration adapters, the plug-in for pathology text processing, the extension for document processing and the original document service, you can integrate pathology documents into SAP Connected Health where users can search the content, analyze the entities extracted from the content, and open the document from the user interface.

Procedure

1. Create the required document types for your pathology documents in the *Document Type Configuration* application.

For more information, see .

2. Import documents into SAP Connected Health.

To do so, you have the following options:

- Import the documents as embedded binary content in HIX files.
In this scenario, documents are imported with the patient and interaction data, so you don't need to establish the connection between patients and documents.

For more information, see [Integrating HIX Data \[page 26\]](#).

- Use any ETL tool to write the document content to the source tables of the basic extension for clinical data.

In this scenario, you can import documents and patient data together or separately. If you import documents separately, you must ensure the `InteractionID` assigned to the documents matches that of the patients. If not, the system won't know what patients the documents belong to.

For more information, see [Integrating Data Using the Basic Extension for Clinical Data \[page 10\]](#).

- Use the BLOB adapter to write the document content to the source tables of the basic extension for clinical data.

In this scenario, you can import documents and patient data together or separately. If you import documents separately, you must ensure the `InteractionID` assigned to the documents matches that of the patients. If not, the system won't know to match the patients with their corresponding documents.

For more information, see [Integrating Documents Using the BLOB Adapter and the Basic Extension \[page 79\]](#).

i Note

This step only brings the documents into the database. It doesn't process the documents, for example, to generate indexes or extract entities.

3. Set up text analysis by integrating ontology content and creating text analysis dictionaries.

For medical documents, you must integrate ontology content from the following providers:

- SAP Vocabularies
- National Cancer Institute Metathesaurus (NCIm)
- National Center for Biotechnology Information (NCBI) Gene

For more information, see .

For information about what pathology entities are extracted during text analysis, see the SAP Connected Health Platform Text Analysis Language Reference.

4. Open the *Plug-In Cockpit* application.
5. Select the plug-in `SAP Connected Health Platform Pathology Text Processing Plug-In` and choose *Configure*.
6. Assign the document types you created for the pathology documents.
7. Activate the plug-in.
8. Set up original document retrieval based on the relevant archiving scenario.

If you store your documents in an external document store, see .

If your documents are included only in structured data as embedded binary content and you don't have a dedicated external document store, see .

9. Open the *Data Integration Cockpit* application.
10. Create a profile for data integration.
 - a. Open the *Run and Manage Profiles* view.
 - b. Choose *Create Profile*.
 - c. Choose the extension `Data Integration Extension for Processing Unstructured Documents`.
 - d. Choose a DW Source.

DW Source is an optional attribute used in the clinical data warehouse to filter data during integration. This means, only data matching the specified DW Source will be integrated during the profile run.

You can create a new DW Source by choosing the **+** icon.

- e. Choose a schedule for running the profile.

The schedule is based on an existing schedule configuration. You can create a new schedule configuration by choosing the **+** icon.

- f. Specify the required parameters.

For information about the specific parameters, see [Extension for Processing Unstructured Documents \[page 84\]](#).

11. Activate and run the profile.

Results

After the profile run has completed, you can confirm the data was correctly integrated by either viewing the data in the target tables of the data integration extension or searching for the data in the end-user applications.

2.4.2 Integrating Medical Documents

Using the provided data integration adapters, the plug-in for medical text processing, the extension for document processing and the original document service, you can integrate medical documents into SAP Connected Health where users can search the content, analyze the entities extracted from the content, and open the document from the user interface.

Procedure

1. Create the required document types for your medical documents in the [Document Type Configuration](#) application.

For more information, see .

2. Import documents into SAP Connected Health.

To do so, you have the following options:

- Import the documents as embedded binary content in HIX files.
In this scenario, documents are imported with the patient and interaction data, so you don't need to establish the connection between patients and documents.
For more information, see [Integrating HIX Data \[page 26\]](#).
- Use any ETL tool to write the document content to the source tables of the basic extension for clinical data.

In this scenario, you can import documents and patient data together or separately. If you import documents separately, you must ensure the `InteractionID` assigned to the documents matches that of the patients. If not, the system won't know what patients the documents belong to.

For more information, see [Integrating Data Using the Basic Extension for Clinical Data \[page 10\]](#).

- Use the BLOB adapter to write the document content to the source tables of the basic extension for clinical data.

In this scenario, you can import documents and patient data together or separately. If you import documents separately, you must ensure the `InteractionID` assigned to the documents matches that of the patients. If not, the system won't know to match the patients with their corresponding documents.

For more information, see [Integrating Documents Using the BLOB Adapter and the Basic Extension \[page 79\]](#).

i Note

This step only brings the documents into the database. It doesn't process the documents, for example, to generate indexes or extract entities.

3. Set up text analysis by integrating ontology content and creating text analysis dictionaries.

For medical documents, you must integrate ontology content from the following providers:

- SAP Vocabularies
- National Cancer Institute Metathesaurus (NCIm)
- RxNorm

For more information, see .

For information about what medical entities are extracted during text analysis, see the SAP Connected Health Platform Text Analysis Language Reference.

4. Open the *Plug-In Cockpit* application.
5. Select the plug-in `SAP Connected Health Platform Medical Text Processing Plug-In` and choose *Configure*.
6. Assign the document types you created for the medical documents.
7. Activate the plug-in.
8. Set up original document retrieval based on the relevant archiving scenario.

If you store your documents in an external document store, see .

If your documents are included only in structured data as embedded binary content and you don't have a dedicated external document store, see .

9. Open the *Data Integration Cockpit* application.
10. Create a profile for data integration.
 - a. Open the *Run and Manage Profiles* view.
 - b. Choose *Create Profile*.
 - c. Choose the extension `Data Integration Extension for Processing Unstructured Documents`.
 - d. Choose a DW Source.

DW Source is an optional attribute used in the clinical data warehouse to filter data during integration. This means, only data matching the specified DW Source will be integrated during the profile run.

- You can create a new DW Source by choosing the **+** icon.
- e. Choose a schedule for running the profile.

The schedule is based on an existing schedule configuration. You can create a new schedule configuration by choosing the **+** icon.

- f. Specify the required parameters.

For information about the specific parameters, see [Extension for Processing Unstructured Documents \[page 84\]](#).

11. Activate and run the profile.

Results

After the profile run has completed, you can confirm the data was correctly integrated by either viewing the data in the target tables of the data integration extension or searching for the data in the end-user applications.

2.4.3 Integrating Documents Without Text Analysis

Context

In SAP Connected Health, you integrate unstructured clinical data as either binary content embedded within structured clinical data or as source documents, such as Microsoft Word documents or PDF documents.

i Note

To link unstructured documents to interactions and, therefore, patients, you must integrate the documents as embedded binary content along with the related structured data.

Procedure

1. Create the required document types for your medical documents in the [Document Type Configuration](#) application.

For more information, see .

i Note

To integrate documents without text analysis, you must ensure not to assign any text-processing plugin to the document type. In addition, if you choose to use an existing document type for your medical

documents, ensure to remove the text-processing plug-in, if any. Else, system will initiate and implement text analysis process.

2. Import documents into SAP Connected Health.

To do so, you have the following options:

- Import the documents as embedded binary content in HIX files.
In this scenario, documents are imported with the patient and interaction data, so you don't need to establish the connection between patients and documents.
For more information, see [Integrating HIX Data \[page 26\]](#).
- Use any ETL tool to write the document content to the source tables of the basic extension for clinical data.
In this scenario, you can import documents and patient data together or separately. If you import documents separately, you must ensure the `InteractionID` assigned to the documents matches that of the patients. If not, the system won't know what patients the documents belong to.
For more information, see [Integrating Data Using the Basic Extension for Clinical Data \[page 10\]](#).
- Use the BLOB adapter to write the document content to the source tables of the basic extension for clinical data.
In this scenario, you can import documents and patient data together or separately. If you import documents separately, you must ensure the `InteractionID` assigned to the documents matches that of the patients. If not, the system won't know to match the patients with their corresponding documents.
For more information, see [Integrating Documents Using the BLOB Adapter and the Basic Extension \[page 79\]](#).
- Use the BLOB adapter and the extension for staging unstructured documents to import the documents.
In this scenario, there is no link between patients and documents. This scenario is intended for testing purposes only.
For more information, see [Extension for Staging Unstructured Documents \[page 77\]](#).

i Note

This step only brings the documents into the database. It doesn't process the documents, for example, to generate indexes.

3. Set up original document retrieval based on the relevant archiving scenario.

If you store your documents in an external document store, see .

If your documents are included only in structured data as embedded binary content and you don't have a dedicated external document store, see .

4. Open the *Data Integration Cockpit* application.

5. Create a profile for data integration.

- a. Open the *Run and Manage Profiles* view.
- b. Choose *Create Profile*.
- c. Choose the extension `Data Integration Extension for Processing Unstructured Documents`.
- d. Choose a DW Source.

DW Source is an optional attribute used in the clinical data warehouse to filter data during integration. This means, only data matching the specified DW Source will be integrated during the profile run.

- You can create a new DW Source by choosing the **+** icon.
- e. Choose a schedule for running the profile.

The schedule is based on an existing schedule configuration. You can create a new schedule configuration by choosing the **+** icon.

- f. Specify the required parameters.

For information about the specific parameters, see [Extension for Processing Unstructured Documents \[page 84\]](#).

6. Activate and run the profile.

Results

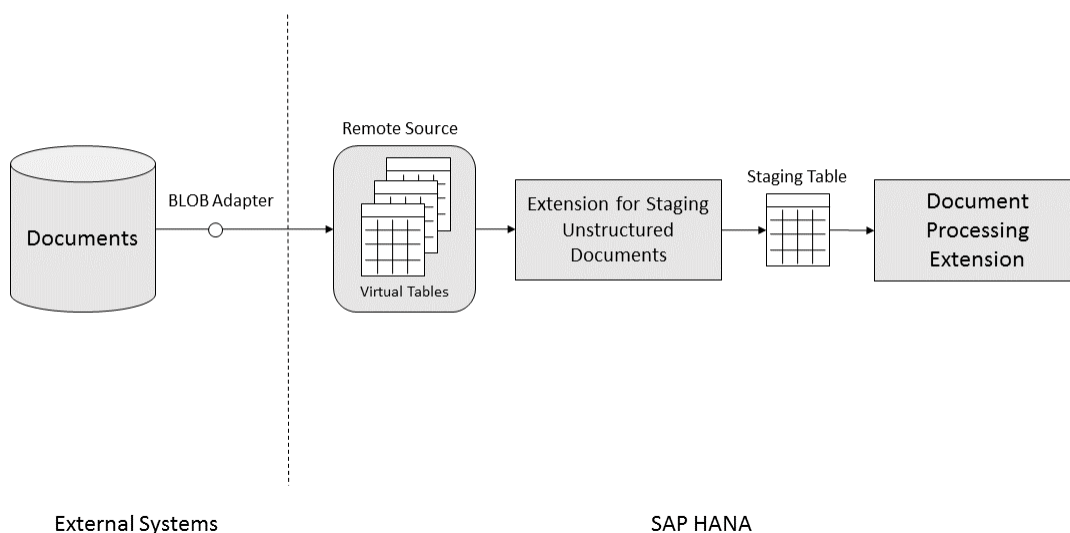
After the profile run has completed, you can confirm the data was correctly integrated by either viewing the data in the target tables of the data integration extension or searching for the data in the end-user applications.

Related Information

[Extension for Staging Unstructured Documents \[page 77\]](#)

2.4.3.1 Extension for Staging Unstructured Documents

SAP Connected Health provides a data integration extension for importing binary documents from an external file system into the document processing staging table. You configure this extension in the [Data Integration Cockpit](#).



This data integration extension uses the deployed BLOB adapter to retrieve the unstructured documents from the source system and store them in the corresponding SAP HANA virtual tables, which were created during system implementation. The extension then stores the documents in the document processing staging table, from which the Document Processing Extension can retrieve the documents for further processing.

i Note

This data integration extension does not bind integrated documents to any interactions, and therefore, to any patients. This extension serves mainly to test your implementation of text analysis. If you want documents to relate to interactions and patients, you must integrate documents using an alternate data integration extension, such as the Basic Extension for Clinical Data.

Profile Parameters

The following parameters are available when creating profiles for the extension for staging unstructured documents:

Table 53:

Parameter	Description
Document Type	Logical type of the document being imported. For a list of available Document Types, see the Document Type Configuration application. All the documents being imported in one profile run are given the same document type.
Language Code	An optional ISO 639-1 language code for the documents being imported.
Name of Virtual Table Schema	Name of the Schema to which the Virtual Table belongs to.
Name of Virtual Table	Name of the virtual table that is used for accessing the external documents This virtual table was created during installation with the technical name <code>SAP_HPH_VT_BLOB</code> .

Target Tables

After the binary documents are imported into the system using this extension, the table `"SAP_HPH"."sap.hc.hph.di.documents.staging.db.models::DocumentStage.ProcessingStage"` is populated.

To start processing the documents from this table, use the Extension for Processing Unstructured Documents.

2.4.4 Integrating Documents Using the BLOB Adapter and the Basic Extension

This procedure describes how to integrate documents using the SAP HPH BLOB adapter and the `Basic Data Integration Extension for Clinical Data`. The steps described also ensure users can search for patients based on the content of the doctor's letters and then open the original versions from the [Related Documents](#) view.

Prerequisites

- SAP HANA Enterprise Information Management data provisioning server is installed and configured on the SAP HANA system.
- SAP HANA Enterprise Information Management data provisioning agents are installed and configured on external data source systems.
- You have installed the data integration adapters for SAP Connected Health. For more information, see .
- You have the privileges required to install and configure SAP HANA Enterprise Information Management data provisioning agents and create the related remote sources. For more information, see the SAP HANA Enterprise Information Management Administration Guide.
- You have administrative privileges for the systems the data provisioning agents are running on.

Context

In the standard scenarios for integrating documents, you can either embed documents in HIX files and use the standard tools for integrating HIX data, or you can use any ETL tool, such as SAP Data Services, to import your documents into the staging tables of the `Basic Data Integration Extension for Clinical Data` and then run that extension.

However, if neither of these two options is feasible for you, you can make use of the other standard components provided by SAP Connected Health. In this example, we'll see how to use the existing SAP HPH BLOB adapter and the `Basic Data Integration Extension for Clinical Data` to integrate doctor's letters (or any other document type).

Procedure

1. If you haven't already, set up the SAP HPH BLOB adapter and create virtual tables for it.
 - a. Register the SAP HPH BLOB adapter with SAP HANA.

To do so, follow the steps described in .

➔ Tip

As an easy alternative, execute the following SQL statement:

```
CREATE AGENT '<agent name>' PROTOCOL 'TCP' HOST '<host where agent was installed>' PORT '<port specified during agent installation>'
CREATE ADAPTER 'SAP HPH Blob Adapter' AT LOCATION AGENT '<agent name>'
```

- b. In your SAP HANA system, create and configure a remote source for the SAP HPH BLOB adapter.

To do so, follow the steps described in .

➔ Tip

As an easy alternative, execute the following SQL statement:

```
CREATE REMOTE SOURCE "REMOTE_SOURCE_NAME" ADAPTER "SAP HPH Blob Adapter"
AT LOCATION AGENT "AGENT_NAME" CONFIGURATION
'<?xml version="1.0" encoding="UTF-8"?><ConnectionProperties
name="configuration">
  <PropertyEntry name="SourceFolder">FILE_LOCATION</PropertyEntry>
  <PropertyEntry name="SourceFilter"></PropertyEntry>
  <PropertyEntry name="AccessMode">Read-Write</PropertyEntry>
  <PropertyEntry name="AccessPath">Absolute</PropertyEntry>
  <PropertyEntry name="AutoUncompress">False</PropertyEntry>
  <PropertyEntry name="BatchSize">10</PropertyEntry>
</ConnectionProperties>'
WITH CREDENTIAL TYPE 'PASSWORD' USING '***';
```

When you configure the remote source, you define the following parameters:

Table 54:

Parameter	Mandatory	Description
Documents Folder	X	Directory of unstructured documents in Data Provisioning Agent instance
Documents Filter		An optional string to filter the files, such as ".xml"
Access Mode	X	A flag indicates mode of access for the BLOB adapters. Allowed values: <ul style="list-style-type: none">○ Read-Only: Read access to the documents folder○ Read-Write: Read and write access to the documents folder

Parameter	Mandatory	Description
Access Path	X	A flag to enable absolute or relative paths. This property will be used only in case of Read-Only mode. Allowed values: <ul style="list-style-type: none"> ○ Absolute: Absolute path will be fetched [Default] ○ Relative: Path relative to Documents Folder will be fetched
Auto Uncompress	X	A flag to uncompress the archive while parsing. Allowed values: True or False (default)
Batch Size	X	Defines how many files to be processed in one batch. Default value is 100.

- c. Create the required virtual tables corresponding to the remote source you've created and assign them to your schema.

By default, the system suggests the following virtual tables:

- SAP_HPH_VT_BLOB
- SAP_HPH_VT_LOG_TRACE
- SAP_HPH_VT_LOG_DETAIL

→ Tip

You can rename these virtual tables to anything you want.

- d. For your virtual table corresponding to the SAP_HPH_VT_BLOB virtual table, assign object privileges (SELECT) to the technical user and to the SYS_REPO user.
- e. To avoid early timeouts during complex or extensive data integration runs, increase the time allowed before your SAP HANA EIM data provisioning server session times out.

To do so, you can use the following example SQL statements, which increase the timeout to 86,400 seconds:

```
ALTER SYSTEM ALTER CONFIGURATION ('dpserver.ini', 'SYSTEM') SET ('framework', 'usePrefetch') = 'true' WITH RECONFIGURE;
ALTER SYSTEM ALTER CONFIGURATION ('dpserver.ini', 'SYSTEM') SET ('framework', 'prefetchLimit') = '1' WITH RECONFIGURE;
ALTER SYSTEM ALTER CONFIGURATION ('dpserver.ini', 'SYSTEM') SET ('framework', 'prefetchTimeout') = '86400' WITH RECONFIGURE;
```

For more information, see the SAP HANA Enterprise Information Management Administration Guide.

2. If the related patients and interactions don't already exist in the system, integrate the required patient and interaction data using the Basic Data Integration Extension for Clinical Data.
 - a. Use any ETL tool to bring the patient data into the staging table `"SAP_HPH"."sap.hc.hph.plugins.basiccdm.db.models::Stage.Patient"` and the interactions into the staging table `"SAP_HPH"."sap.hc.hph.plugins.basiccdm.db.models::Stage.Interactions"`.
 - b. In the *Data Integration Cockpit*, create and run a profile for the Basic Data Integration Extension for Clinical Data.

➔ Remember

For the `Patient Source` and `Interactions Source` parameters, make sure you specify the default tables `"SAP_HPH"."sap.hc.hph.plugins.basiccdm.db.models::Stage.Patient"` and `"SAP_HPH"."sap.hc.hph.plugins.basiccdm.db.models::Stage.Interactions"` respectively.

i Note

In this example, we assume you are integrating doctor's letters and assigning them to one interaction. If, however, you want to integrate different documents and assign them to different interactions, you must create the corresponding mapping information in either a table or view and then ensure that documents are correctly mapped to interactions during integration. This scenario is not currently supported by the standard SAP Connected Health.

Alternatively, you could simply repeat the steps described in this tutorial for each document-interaction mapping.

3. Create the temporary table `BLOB_DOC_STAGE` and populate it with the documents from virtual table. To do so, execute the following SQL statement:

```
CREATE COLUMN TABLE "BLOB_DOC_STAGE"
AS
(
  SELECT * FROM <virtual table schema>.<virtual table name>
);
```

4. In the *Document Type Configuration* application, create the document type `sap.hph.DoctorLetter`. For more information, see .

5. Load the documents to the staging table `sap.hc.hph.plugins.basiccdm.db.models::Stage.Document`, which is used by the Basic Data Integration Extension for Clinical Data.

To do so, execute the following SQL statement:

```
INSERT INTO "SAP_HPH"."sap.hc.hph.plugins.basiccdm.db.models::Stage.Document"
("PackageID", "_LoadTimeStamp", "_SourceID", "DocumentID", "FileName",
"MIMETYPE", "Content", "Title")
SELECT
  TO_INTEGER(1) AS "_PackageID",
  TO_TIMESTAMP(now()) AS "_LoadTimeStamp",
  LEFT(TO_NVARCHAR(<Source ID>), 5) AS "_SourceID",
  LEFT(TO_NVARCHAR("DOCUMENTID"), 1024) AS "DocumentID",
  LEFT(TO_NVARCHAR("DOCUMENTNAME"), 256) AS "FileName",
  LEFT(TO_NVARCHAR("MIMETYPE"), 256) AS "MIMETYPE",
```

```
"DOCUMENTCONTENT" AS "Content",
LEFT (TO_NVARCHAR("DOCUMENTNAME"), 1024) AS "Title"
FROM "BLOB_DOC_STAGE";
```

- Assign the doctor's letters to interactions by defining the following attributes in the staging table:

Table 55:

Column	Value	Details
ParentEntityType	sap.hc.hph.cdw.Interaction	This is a constant.
ParentEntityID	<Interaction ID to which the document is to be assigned>	This should be same as the column "InteractionID" of the interactions stage table "sap.hc.hph.plugins.basiccdm.db.models::Stage.Interactions" which is used for loading interactions.
_ParentEntitySourceID	<Source ID of the Interaction to which the document is to be assigned>	This refers to the "_SourceID" column of the interactions stage table "sap.hc.hph.plugins.basiccdm.db.models::Stage.Interactions" which is used for loading Interactions. If the source id of the document and interactions is same, this column can be left blank.

➔ Tip

As an easy alternative, if you are only assigning documents to one interaction, execute the following SQL statement:

```
UPDATE "SAP_HPH"."sap.hc.hph.plugins.basiccdm.db.models::Stage.Document"
set "_ParentEntitySourceID" = <Interaction Source> , "ParentEntityID" =
<InteractionID>, "_ParentEntityType" = 'sap.hc.hph.cdw.Interaction';
```

- Delete the temporary table BLOB_DOC_STAGE.

To do so, execute the following SQL statement:

```
DROP TABLE "BLOB_DOC_LET_STAGE";
```

- In the *Data Integration Cockpit*, create and run a profile for the Basic Data Integration Extension for Clinical Data.

➔ Remember

For the Document Source parameter, make sure you specify the default table "SAP_HPH"."sap.hc.hph.plugins.basiccdm.db.models::Stage.Document".

- To enable the doctor's letters to be retrieved from the *Related Documents* view, configure the original document service so that the BLOB adapter virtual table is used for retrieving the original content.

For more information, see .

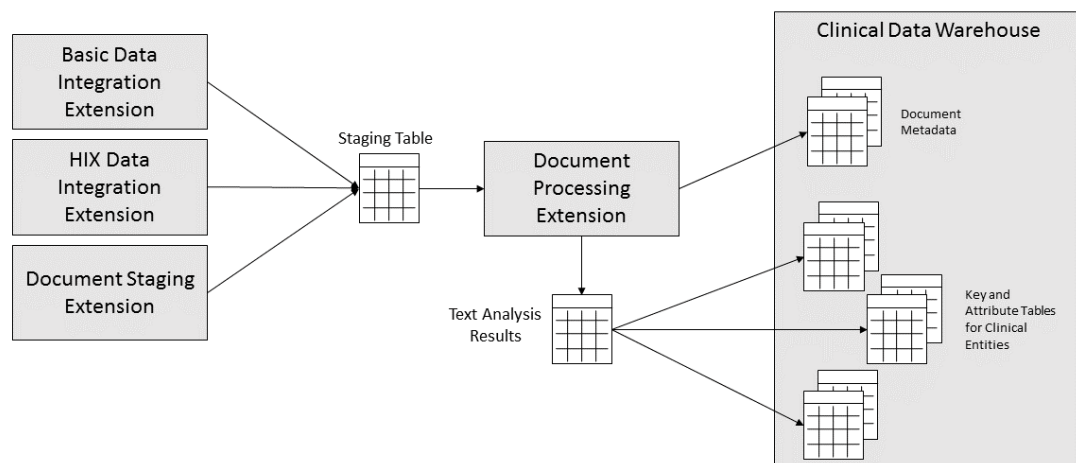
10. In the *Data Integration Cockpit*, create and run a profile for the `Data Integration Extension for Processing Unstructured Documents`.

While creating the profile, you can specify the parameter `Enable Document Archive` as `False` because the original documents are available through the BLOB adapter virtual tables. For more details on the other available parameters, see [Extension for Processing Unstructured Documents \[page 84\]](#).

11. Verify that the document has been correctly integrated.
 - a. In the *Patient Search* application, search for the related patient by using key terms you know appear in the document.
 - b. Select the related patient and open the *Related Documents* view.
 - c. Find the document you integrated and open it.

2.4.5 Extension for Processing Unstructured Documents

SAP Connected Health provides a data integration extension for processing the binary documents available in the document processing staging table. You configure this extension in the *Data Integration Cockpit*.



This data integration extension retrieves and processes documents from the document processing staging table. Document metadata is then extracted and stored in the clinical data warehouse for use in the *Patient Search* app to enable search for documents related to patients.

In addition, this extension can perform text analysis on the documents to extract clinical facts from the documents, which can be stored in the clinical data warehouse as clinical entities and attributes.

After processing, documents are removed from the processing stage.

Profile Parameters

The following parameters are available when creating profiles for the extension for staging unstructured documents:

Table 56:

Parameter	Description
Update the Clinical Data Warehouse	Set to <code>True</code> if the document metadata needs to be imported into the clinical data warehouse.
Enable Search for Documents	Set to <code>True</code> if the document should be enabled for free text search. The corresponding SAP HANA full text index will be available in the table <code>"SAP_HPH"."sap.hc.hph.cdw.db.models::DWDocuments.DocumentSearchIndex"</code>
Store Documents for Original Document Service	Set to <code>True</code> if the document should be stored for Original Document Service Configuration.
Enable Text Analysis for Documents	Set to <code>True</code> if you want to apply text analysis to the document. Based on which text processing plug-in you selected, the final results of text analysis are stored in one of the following post-processing tables: <ul style="list-style-type: none"> <code>"SAP_HPH"."sap.hc.hph.plugins.textProcessing.medical.postProcessing::Entities.TA_FILTERED"</code> <code>"SAP_HPH"."sap.hc.hph.plugins.textProcessing.pathology.postProcessing::Entities.TA_FILTERED"</code>
Clear Staging Area	Set to <code>True</code> if the document should be deleted from the processing stage after the processing. The pre-processing text analysis results are also deleted from the table <code>"SAP_HPH"."sap.hc.hph.plugins.documents.db.models::TA.Results"</code> .
Batch Size for Parallel Processing	
Document Type	This type shall be used for the documents that do not have any type set in the processing stage.

2.5 Integrating Content from Ontology Content Providers

Import and transform ontologies from external sources by using the [Data Integration Cockpit](#).

Prerequisites

- You have the required licenses for the ontology sources you want to use.
- You have activated the plug-ins for ontology services in the [Plug-In Cockpit](#).
- You have deployed and configured the file adapter on the SAP HANA Enterprise Information Management data provisioning agent.

Context

To ensure that clinical data created in different external systems using different coding systems is correctly interpreted and harmonized, SAP Connected Health supports the integration of ontology content from external providers. To integrate this content into the platform, you must import it into the system and then transform and optimize it.

Regardless which ontology content providers you want to use, you must first run the `SAP Connected Health Platform Vocabulary ID Mappings Ontology Import Extension`.

i Note

To integrate ontology content, you must ensure you have the proper licenses from the content providers.

i Note

Based on the volume and complexity of the data, the import and transformation of ontology content can take up to several hours. Please plan accordingly.

Procedure

1. If you haven't already, set up the SAP EIM File Adapter.
 - a. Using the SAP Data Provisioning Agent Configuration Tool, register the SAP EIM File Adapter with SAP HANA.

→ Tip

As an alternative to using the SAP Data Provisioning Agent Configuration Tool, you can execute the following SQL statements to register both the data provisioning agent and the adapter with SAP HANA:

```
CREATE AGENT '<agent name>' PROTOCOL 'TCP' HOST '<host where agent was installed>' PORT '<port specified during agent installation>'
CREATE ADAPTER "SAP EIM File Adapter" AT LOCATION AGENT '<agent name>'
```

- b. In your SAP HANA system, create remote sources for the SAP EIM File Adapter.

For more information about how to create remote data sources, see .

i Note

You don't need to create any virtual tables.

2. Download ontology content directly from the relevant content provider and save the file to the appropriate directory.

These directories are created in the sub-directories of the SAP HANA Enterprise Information Management file adapter's root directory.

→ Remember

For the vocabularies content, the required file is provided with the plug-in *Vocabulary ID Mappings for Ontology Services* in the file folder ► *sap* ► *hc* ► *hph* ► *plugins* ► *ots* ► *vocabularies* ► *data* ►.

3. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

4. Open the *Data Integration Cockpit* app.
5. Create a profile for the SAP Connected Health Platform Vocabulary ID Mappings Ontology Import Extension.
 - a. Open the *Run and Manage Profiles* view.
 - b. Choose *Create Profile*.
 - c. Choose the relevant data integration extension.

→ Remember

Data integration extensions are only available if the corresponding plug-in is activated.

- d. Skip DW Source selection, as it isn't required.
- e. Skip schedule selection, as plug-ins for ontology services do this automatically.
- f. Specify the required parameters.

For more information about the profile parameters, see [Vocabularies Import Extension \[page 94\]](#).

6. Activate the profile and choose *Run Profile*.
7. To monitor the status of the profile run, open the *Monitor Profile Runs* view.

8. After the profile run is complete, create and run a profile for the SAP Connected Health Platform Vocabulary ID Mappings Ontology Import Extension.

For more information about the data integration extension for transforming vocabularies, see [Vocabularies Transform Extension \[page 95\]](#).

9. Repeat the process for any ontology content providers you want to support.

i Note

If you want to use CMS GEMs ontology content, you must first integrate NCIIm ontology content.

Related Information

[NCIIm Import Extension \[page 88\]](#)

[NCIIm Transform Extension \[page 89\]](#)

[Value Sets Import Extension \[page 90\]](#)

[Value Sets Transform Extension \[page 91\]](#)

[CMS GEMs Import Extension \[page 92\]](#)

[CMS GEMs Transform Extension \[page 92\]](#)

[RxNorm Import Extension \[page 93\]](#)

[RxNorm Transform Extension \[page 94\]](#)

[Vocabularies Import Extension \[page 94\]](#)

[Vocabularies Transform Extension \[page 95\]](#)

2.5.1 NCIIm Import Extension

You integrate ontology content from the National Cancer Institute Metathesaurus (NCIIm) to provide SAP Connected Health with information about biomedical concepts and their relationships.

NCIIm is derived from the Unified Medical Language System (UMLS). For more information, go to <http://ncimeta.nci.nih.gov/> .

Profile Parameters

The following parameters are available when creating profiles for this extension:

Table 57:

Parameter	Description
Remote Source	External source containing ontology files You define the name of the remote source during deployment of the file adapter.
Path	Specify the full directory path of the content source. This information is provided in the configuration of the registered file adapter on the data provisioning agent instance.

Target Tables

When you integrate ontology data using this extension, the following SAP Connected Health tables are populated:

Table 58:

Table	Purpose
"SAP_HPH"."sap.hc.hph.plugins.ots.ncim::Entities.MR*"	Stores the imported content as is from the reference source.

2.5.2 NCIIm Transform Extension

You integrate ontology content from the National Cancer Institute Metathesaurus (NCIIm) to provide SAP Connected Health with information about biomedical concepts and their relationships.

NCIIm is derived from the Unified Medical Language System (UMLS). For more information, go to <http://ncimeta.nci.nih.gov/>.

Profile Parameters

The following parameters are available when creating profiles for this extension:

Table 59:

Parameter	Description
Use Cancer-specific Terms	If you use ontology services and the related applications only in the context of cancer research, you can eliminate redundant terms from labels, such as "neoplasm" or "cancer", by using ICD-O labels.

Target Tables

When you integrate ontology data using this extension, the following SAP Connected Health tables are populated:

- "SAP_HPH"."sap.hc.hph.ots.core::Entities.ConceptAttributes"
- "SAP_HPH"."sap.hc.hph.ots.core::Entities.RelationAttributes"
- "SAP_HPH"."sap.hc.hph.ots.core::Entities.TermAttributes"
- "SAP_HPH"."sap.hc.hph.ots.services::Entities.ConceptClassification"
- "SAP_HPH"."sap.hc.hph.ots.services::Entities.Log"

2.5.3 Value Sets Import Extension

You integrate ontology content from the Value Set Authority Center to provide SAP Connected Health with vocabulary value sets contained in the Clinical Quality Measures.

For more information, go to <https://vsac.nlm.nih.gov/> .

Profile Parameters

The following parameters are available when creating profiles for this extension:

Table 60:

Parameter	Description
Remote Source	External source containing ontology files You define the name of the remote source during deployment of the file adapter.

Parameter	Description
Path	Specify the full directory path of the content source. This information is provided in the configuration of the registered file adapter on the data provisioning agent instance.

Target Tables

When you integrate ontology data using this extension, the following SAP Connected Health tables are populated:

Table 61:

Table	Purpose
"SAP_HPH"."sap.hc.hph.plugins.ots.valuesets::Entities.ValueSetCodes"	Stores the imported content as is from the reference source.
"SAP_HPH"."sap.hc.hph.plugins.ots.valuesets::Entities.ValueSetDefinitions"	Stores the imported content as is from the reference source.

2.5.4 Value Sets Transform Extension

You integrate ontology content from the Value Set Authority Center to provide SAP Connected Health with vocabulary value sets contained in the Clinical Quality Measures.

For more information, go to <https://vsac.nlm.nih.gov/> .

Target Tables

When you integrate ontology data using this extension, the following SAP Connected Health tables are populated:

- "SAP_HPH"."sap.hc.hph.ots.core::Entities.Concepts"
- "SAP_HPH"."sap.hc.hph.ots.core::Entities.Relations"
- "SAP_HPH"."sap.hc.hph.ots.services::Entities.Log"

2.5.5 CMS GEMs Import Extension

You integrate ontology content from the Centers for Medicare & Medicaid Services to provide SAP Connected Health with general equivalence mappings to translate ICD9CM codes into ICD10CM codes.

For more information, go to <http://www.cms.gov>.

Profile Parameters

The following parameters are available when creating profiles for this extension:

Table 62:

Parameter	Description
Remote Source	External source containing ontology files You define the name of the remote source during deployment of the file adapter.

Target Tables

When you integrate ontology data using this extension, the following SAP Connected Health tables are populated:

Table 63:

Table	Purpose
"SAP_HPH"."sap.hc.hph.plugins.ots.cmsgems::Entities.ICD9CMTtoICD10CM"	Stores the imported content as is from the reference source.
"SAP_HPH"."sap.hc.hph.plugins.ots.cmsgems::Entities.ICD9CMTtoICD10CMRaw"	Stores the imported content as is from the reference source.

2.5.6 CMS GEMs Transform Extension

You integrate ontology content from the Centers for Medicare & Medicaid Services to provide SAP Connected Health with general equivalence mappings to translate ICD9CM codes into ICD10CM codes.

For more information, go to <http://www.cms.gov>.

Target Tables

When you integrate ontology data using this extension, the following SAP Connected Health tables are populated:

- "SAP_HPH"."sap.hc.hph.ots.core::Entities.Concepts"
- "SAP_HPH"."sap.hc.hph.ots.core::Entities.Relations"
- "SAP_HPH"."sap.hc.hph.ots.core::Entities.Terms"
- "SAP_HPH"."sap.hc.hph.ots.services::Entities.ConceptClassification"
- "SAP_HPH"."sap.hc.hph.ots.services::Entities.Log"

2.5.7 RxNorm Import Extension

You integrate ontology content from RxNorm to provide SAP Connected Health with normalized names for clinical drugs and map the names to many of the common drug vocabularies.

For more information, go to <http://rxnav.nlm.nih.gov/>.

Profile Parameters

The following parameters are available when creating profiles for this extension:

Table 64:

Parameter	Description
Remote Source	External source containing ontology files You define the name of the remote source during deployment of the file adapter.
Path	Specify the full directory path of the content source. This information is provided in the configuration of the registered file adapter on the data provisioning agent instance.

Target Tables

When you integrate ontology data using this extension, the following SAP Connected Health tables are populated:

Table 65:

Table	Purpose
"SAP_HPH"."sap.hc.hph.plugins.ots.ncim::Entities.RXN*"	Stores the imported content as is from the reference source.

2.5.8 RxNorm Transform Extension

You integrate ontology content from RxNorm to provide SAP Connected Health with normalized names for clinical drugs and map the names to many of the common drug vocabularies.

For more information, go to <http://rxnav.nlm.nih.gov/>.

Target Tables

When you integrate ontology data using this extension, the following SAP Connected Health tables are populated:

- "SAP_HPH"."sap.hc.hph.ots.core::Entities.Concepts"
- "SAP_HPH"."sap.hc.hph.ots.core::Entities.Relations"
- "SAP_HPH"."sap.hc.hph.ots.core::Entities.Terms"
- "SAP_HPH"."sap.hc.hph.ots.services::Entities.ConceptClassification"
- "SAP_HPH"."sap.hc.hph.ots.services::Entities.Log"

2.5.9 Vocabularies Import Extension

This data integration extension imports vocabulary lists provided by SAP, which serve as the foundation for integrating other ontology content.

This vocabulary content is provided with the plug-in *Vocabulary ID Mappings for Ontology Services* in the file folder `sap > hc > hph > plugins > ots > vocabularies > data`.

i Note

Before importing this vocabulary content, you must copy the file to a location that your deployed SAP HANA Enterprise Information Management file adapter can access.

Profile Parameters

The following parameters are available when creating profiles for this extension:

Table 66:

Parameter	Description
Remote Source	External source containing ontology files You define the name of the remote source during deployment of the file adapter.
Path	Specify the full directory path of the content source. This information is provided in the configuration of the registered file adapter on the data provisioning agent instance.

Target Tables

When you integrate ontology data using this extension, the following SAP Connected Health tables are populated:

Table 67:

Table	Purpose
"SAP_HPH"."sap.hc.hph.plugins.ots.vocabularies::Entities.Vocabularies"	Stores the imported content as is from the reference source.

2.5.10 Vocabularies Transform Extension

This data integration extension transforms the imported vocabulary lists provided by SAP, which serve as the foundation for integrating other ontology content.

Target Tables

When you integrate ontology data using this extension, the following SAP Connected Health tables are populated:

- "SAP_HPH"."sap.hc.hph.ots.core::Entities.Vocabularies"

2.6 Creating Profiles for Data Integration Extensions

To integrate data using either a delivered or custom data integration extension, you must create and configure profiles for that data integration extension in the *Data Integration Cockpit*.

Context

Each profile has its own unique settings and parameters, which means you can create multiple profiles for one extension to, for example, integrate data from several different data sources or specify how integrated data is handled in the system based on the profile.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Open the *Plug-In Cockpit* app.
3. Select and activate the required plug-in.
4. Open the *Data Integration Cockpit* app.
5. Create a profile for data integration.
 - a. Open the *Run and Manage Profiles* view.
 - b. Choose *Create Profile*.
 - c. Choose the relevant data integration extension.

➔ Remember

Data integration extensions are only available if the corresponding plug-in is activated.

- d. Optionally, choose a DW Source.

DW Source is an optional attribute used in the clinical data warehouse to filter data during integration. This means, only data matching the specified DW Source will be integrated during the profile run.

You can create a new DW Source by choosing the **+** icon.

- e. Optionally, choose a schedule for running the profile.

The schedule is based on an existing schedule configuration. You can create a new schedule configuration by choosing the **+** icon.

i Note

You can't create schedule configurations for the following data formats:

- GFF3
- VCF
- BED
- PED

- f. Specify the required parameters.
6. Activate the profile and choose [Run Profile](#).
7. To monitor the status of the profile run, open the [Monitor Profile Runs](#) view.

Results

After the profile run has completed, you can confirm the data was correctly integrated by either viewing the data in the target tables of the data integration extension or searching for the data in the end-user applications.

2.7 Rolling Back and Cleaning Up Integrated Data

You can roll back or clean up integrated data in the [Data Integration Cockpit](#).

Rolling Back Profile Runs

After you run a profile to integrate data, you may realize something is incorrect and want to delete the data you just integrated from the target tables. You can roll back this profile run and delete the integrated data by going to the [Monitor Profile Runs](#) view, selecting the profile run you want to roll back, and choosing [Roll Back](#).

➔ Recommendation

To avoid potential dependency issues, we recommend only rolling back the most recent profile run.

Cleaning Up Staging Areas

After a successful data integration run, you may want to delete the data from the tables used for staging so that you reduce memory consumption. This is supported by specific data integration extensions. To allow staging areas to be cleaned up, go to the profile parameters for the relevant data integration extensions, and specify the parameter for cleaning up SAP HANA staging or source tables.

2.8 Connecting to New Data Sources

Context

During implementation of SAP Foundation for Health, it is assumed that the deployment and configuration tasks for all relevant data sources will be completed. However, over the life of the platform, you may want to add additional data sources to your system. This typically involves one of the following scenarios:

- You want to integrate supported data types from a new source located in an existing system that has already been configured for SAP Foundation for Health.
- You want to integrate supported data types from a new source located outside the existing SAP Foundation for Health landscape.

i Note

If you want to integrate a type of data that is not supported by any available data integration extension, you must develop a custom plug-in. For more information, see the SAP Foundation for Health Plug-In Developer Guide.

For the existing system scenario, you simply have to go to the [Data Integration Cockpit](#), select the appropriate data integration extension, and create a new profile that specifies the new data source.

For the new system scenario, proceed as follows:

i Note

For any aspects relating to SAP HANA Enterprise Information Management (EIM), see the SAP HANA Enterprise Information Management Administration Guide for more details.

Procedure

1. Install an SAP HANA EIM data provisioning agent on the source system.
2. Using the SAP Data Provisioning Agent Configuration Tool, connect to SAP HANA and register the data provisioning agent.
3. Deploy the appropriate data integration adapter.

For more information, see the SAP Connected Health Platform Implementation Guide.

4. In the [Data Integration Cockpit](#) application, select the appropriate data integration extension, and create a new profile that specifies the new data source.

3 Operations

SAP HANA platform provides most of the operations tools you require, for example, to monitor and audit your system. However, specifically for the clinical data warehouse and data integration, SAP Connected Health provides additional features for monitoring and auditing.

Related Information

[Monitoring Profile Runs \[page 99\]](#)

[Monitoring the Clinical Data Warehouse \[page 99\]](#)

[Viewing and Deleting Patient Data \[page 102\]](#)

3.1 Monitoring Profile Runs

You can monitor the progress of individual profile runs by going to the *Monitor Profile Runs* view in the *Data Integration Cockpit*.

All runs are listed in the *Profile Runs* table, sorted by their start times. You can choose any run to look at the details, which includes overview information and a list of the specific files that were integrated.

From the details view of any run, you can re-run the profile run or roll it back if supported. When you roll back a profile run, the integrated files are removed from the clinical data warehouse.

All profile runs have one of the following statuses:

- Running
- Queued
- Completed
- Failed
- Rolled Back

3.2 Monitoring the Clinical Data Warehouse

In addition to the standard administrative tools of SAP HANA platform, SAP Connected Health provides the *Clinical Data Warehouse Cockpit* application, specifically designed for monitoring and auditing data stored in the clinical data warehouse.

The *Clinical Data Warehouse Cockpit* includes both a technical view and a semantic view, as well as a load history view.

Technical View

This view offers a holistic view of the memory consumption of SAP Connected Health. For deeper analysis, this view also offers an overview of all table namespaces in your implementation of SAP Connected Health and provides specific measures of the number of tables and memory consumed. For easy reference, namespaces are grouped by system component.

In the technical view, you can select a namespace to view a breakdown of all tables in the namespace and then select a specific table to view a breakdown of its columns. Of all the details shown for tables and columns, the following are particularly noteworthy:

Table 68:

Detail	Description
Total Size	Main Size + Delta Size (Shown as "0" if the table / column load status is "No")
Main Size	Size of main storage in memory (Shown as "0" if the table / column load status is "No")
Delta Size	Size of delta storage in memory (Shown as "0" if the table / column load status is "No")
Size on Disk	Total size of the table on disk
Record Count (in delta)	Total number of records in delta storage that haven't been merged yet
Data (For columns only - appears in Data/Dict/Index/Misc)	Current memory consumption of the data part of the column
Dict (For columns only - appears in Data/Dict/Index/Misc)	Current memory consumption of the dictionary part of the column
Index (For columns only - appears in Data/Dict/Index/Misc)	Current memory consumption of the index part of the column
Misc (For columns only - appears in Data/Dict/Index/Misc)	Current memory consumption of unspecified parts of the column


For each table and column, you can also see whether it is currently loaded in memory. For this, the following load statuses are shown:

Table 69:

Load Status	Description
Yes	A table / column is completely loaded in memory.

Load Status	Description
No	A table / column is not loaded in memory.
Partially	If a table consists of a mixture of columns with both <i>Yes</i> and <i>No</i> status, the table receives the status <i>Partially</i> .
Always	Rowstore tables are always loaded in memory and are, therefore, flagged as "Always".

→ Tip

In the technical view, you can choose  to see an overview of all namespaces presented in a bar chart. You can also edit the table settings to specify the type of information you want to see.

Semantic View

This view offers an overview of the real-world entities stored in the clinical data warehouse. For each entity, you can see the number of records and their total memory consumption. This information is retrieved from the physical tables that belong to the respective entities.

You can also see how any entity was loaded by selecting an entity and choosing the button to switch to the load history view for that specific entity.

Load History

i Note

To view the load history, you must be assigned the role `sap.hc.hph.cdw.cockpit.roles::DATA_ADMIN`.

In the load history view, you can see exactly what data was added to the tables of the clinical data warehouse during data integration. You specify what data you want to see by selecting one of the following dimensions:

- Data integration extension ID
- Profile ID
- Source ID
- Entity

→ Recommendation

When selecting a data range, we recommend selecting a range smaller than one month. Anything larger than a month will result in over-compression of the chart columns, making the charts difficult to read.

For each dimension, the overview timeline chart displays the number of data loads by date. You can then drill down by selecting a date to see the specific data integration runs by audit ID and duration. By further drilling

down, you can see the entity table chart which shows specific entities and number of records integrated during a data integration run.

From the entity table, you can open a table to see the exact data added during the data integration run. For tables, only the first 100 rows are shown. If you want to see the entire table, you must query the table in SAP HANA studio.

3.3 Viewing and Deleting Patient Data

In most cases, data is automatically deleted during the data replication and synchronization process; however, if required, you can also manually delete patient data.

Context

When considering whether to delete all data associated with a patient, you can first view a list of all tables containing data associated with that patient. After determining the impacts of deleting a patient, you can delete the patient from the relevant central tables. By doing this, all instances of that patient and all related data and documents in dependent tables are also automatically deleted.

SAP provides the stored procedures required to view tables associated with patients and delete patients in the SAP HANA database.

i Note

You should manually delete patient data in exceptional cases only.

➔ Recommendation

We recommend backing up any critical data before proceeding.

Procedure

1. Determine the impacts of deleting a patient by finding out which tables contain data from that patient.
 - a. To view patient data relating to genomics services, in the repository workspace, execute the stored procedure "SAP_HPH"."sap.hc.hph.genomics.db.procedures.model::GetPatientInfo".

When executing the procedure, to specify the patient you want to view, use the DWID parameter for the patient. To identify the DWID, execute the following SQL statement:

```
select DWID from
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities.Patient_Key" where
"DWSource" = <DWSource> and "PatientID" = <PatientID>;
```

- b. To view patient data in the clinical data warehouse, in the repository workspace, execute the stored procedure

```
"SAP_HPH"."sap.hc.hph.cdw.patientdeletion.procedures::CDWDisplayPatient".
```

When executing the procedure, use the `PatientID` and `DWSource` parameters to specify the patient you want to view.

2. Delete the patient and all associated data.

- a. To delete patient data relating to genomics services, in the repository workspace, execute the stored procedure `"SAP_HPH"."sap.hc.hph.genomics.db.procedures.model::DeletePatient"`.

When executing the procedure, to specify the patient you want to delete, use the `DWID` parameter for the patient. To identify the `DWID`, execute the following SQL statement:

```
select DWID from
"SAP_HPH"."sap.hc.hph.cdw.db.models::DWEntities.Patient_Key" where
"DWSource" = <DWSource> and "PatientID" = <PatientID>;
```

- b. To delete patient data in the clinical data warehouse, in the repository workspace, execute the stored procedure

```
"SAP_HPH"."sap.hc.hph.cdw.patientdeletion.procedures::CDWDeletePatient".
```

When executing the procedure, use the `PatientID` and `DWSource` parameters to specify the patient you want to delete.

4 User Management

The user concept for SAP Connected Health is based on the standard security concepts of SAP HANA platform; however, it is limited to SAP HANA restricted users who access data indirectly through a technical user.

In addition, access to applications and application content is controlled by application configurations, which you can assign to individual users or user groups in the form of organizational units.

The relevant parameters and attributes for user management are provided by either standard or custom plug-ins. SAP Connected Health provides the following standard plug-ins for user management:

- User Attributes Plug-In (`sap.hc.hph.um.user`)
- Org Attributes Plug-In (`sap.hc.hph.um.org`)

Related Information

[User Management Configuration \[page 104\]](#)

[Creating Users \[page 107\]](#)

[Creating Organizational Units \[page 110\]](#)

[Reassigning Organizational Units \[page 111\]](#)

[Deactivation of Users and Organizational Units \[page 112\]](#)

[Integrating Users and Organizational Units from LDAP Directory \[page 112\]](#)

4.1 User Management Configuration

Before a system administrator can create users and organizational units, you must configure the parameters and attributes for users and organizational units. This includes specifying which attributes are available and which are required.

i Note

In addition, note that you will have to reconfigure the user management settings if you choose to update your User Management application version.

Related Information

[Configuring User Accounts \[page 105\]](#)

[Configuring Organizational Units \[page 106\]](#)

[E-Mail Server Configuration \[page 107\]](#)

[User Management \[page 104\]](#)

4.1.1 Configuring User Accounts

You configure the parameters and attributes available for user accounts in the *User Management Configuration* application.

Prerequisites

You have configured the password policy for your SAP HANA system for password authentication.

i Note

For more information, see the SAP HANA Administration Guide.

Context

i Note

The specific parameters and attributes described in this guide assume you use the user attributes plug-in provided by SAP.

For more information, see *Configuring SAP HANA for User Authentication and Single-Sign On* in the SAP HANA Administration Guide.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Open the *User Management Configuration* app.
3. Under *Generic Parameters*, specify how users are authenticated.
4. From the list of user account attributes, select which attributes are available when creating users.

From these attributes, you can also select which are required attributes.

i Note

The standard SAP plug-ins for user attributes and org attributes require specific attributes for the system to operate correctly. These attributes cannot be changed.

Results

After you configure the parameters and attributes for user accounts, you can create user accounts in the *User Management* application.

4.1.2 Configuring Organizational Units

You configure the parameters and attributes available for organizational units in the *User Management Configuration* application.

Context

i Note

The specific parameters and attributes described in this guide assume you use the org attributes plug-in provided by SAP.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Open the *User Management Configuration* app.
3. From the list of organizational unit attributes provided by the org attribute plug-in, select which attributes are available when creating organizational units.

From these attributes, you can also select which are required attributes.

i Note

The standard SAP plug-ins for user attributes and org attributes require specific attributes for the system to operate correctly. These attributes cannot be changed.

Results

After you configure the parameters and attributes for organizational units, you can create organizational units in the *User Management* application.

4.1.3 E-Mail Server Configuration

When you create new users, these users receive passwords to gain access to SAP Connected Health applications. The system sends these passwords by e-mail. Therefore, you must ensure your SAP HANA system is configured with a valid connection to an e-mail server.

In addition, you can replace the default email template with custom template based on the organizational preferences.

In SAP HANA, you configure such a connection by creating an SMTP configuration.

For more information, see *Creating an SMTP Configuration* in the SAP HANA Administration Guide.

4.2 Creating Users

You create users in the *User Management* application in the *All Users* view.

Prerequisites

You have defined the generic parameters and available attributes for user accounts in the SAP Connected Health application *User Management Configuration*.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Open the *User Management* application and go to the *All Users* view.
3. To create a user, choose *Create User*.

To edit an existing user, search for a user by name or e-mail, choose the user, and choose *Edit User*.

➔ Tip

Alternatively, you can use CSV files to create multiple users. To do so, choose *Import Users from CSV File*.

For more information about the required structure of the CSV file, see [Using CSV File for Multiple Users and Organizational Units \[page 109\]](#).

4. Specify the user account attributes.

To enable SAML authentication method, select *SAML* and choose identity provider from *SAML Identity Provider*.

➔ Remember

You won't be able to select SAML and choose a relevant identity provider, if you haven't configured the SAML settings in the SAP HANA XS application.

For more information, see [SAML Configuration \[page 120\]](#).

5. Select an available organizational unit to assign the user to.

i Note

Within a single organizational unit hierarchy, you can assign a user to multiple organizational units.

6. Assign the user the appropriate business roles.
7. Save the user and, from the *All Users* view, activate the user.

Results

After you create a user, you can assign configurations to the user in the *Configuration Assignment* application, which will enable the user to view the corresponding applications.

Related Information

[Using CSV File for Multiple Users and Organizational Units \[page 109\]](#)

4.2.1 Using CSV File for Multiple Users and Organizational Units

You create multiple users and organizational units in the *User Management* application by choosing [Upload CSV File](#) to upload a list of users and units.

You must use either the recommended CSV file template or your own CSV file that uses the same structure as the template. To download CSV template, choose download link in the [Import Users from CSV File](#) dialog box.

The following table describes the column names specific to the CSV file template for multiple users:

Table 70:

Column Name	Description
Username	The unique login ID of the user.
Firstname	The first name of the user.
Lastname	The last name of the user.
Email	The email ID of the user.
Role	The user role specific to SAP Foundation for Health. For more information about user roles, see User Roles [page 117] .

The following table describes the column names specific to the CSV file template for multiple organizational units:

Table 71:

Column Name	Description
OrgID	The unique ID of an organization.
Name	The specific name of an organization
ParentOrgID	The unique numeric ID of an organization that serves as the parent organization for the organizational unit hierarchy. i Note You can use this field to create desired organizational unit hierarchy based on your business needs.

i Note

All column values are mandatory. You can't delete or modify the column names. However, you can update the column details such as; Users, First Name, and Org ID in the CSV file for both users and organizational units.

→ Tip

You can separate multiple roles and organizational IDs by using a semi-colon (;).

If the volume of data in the CSV file is huge, you must ensure you configure a maximum timeout value for your system accordingly in the SAP HANA XS Administration Tool.

For example, if you import 500 users at a time from the CSV file, the `<max_request_runtime>` value must be 600 seconds. Similarly, if you import more than 500 users, the `<max_request_runtime>` value must be adjusted accordingly. For information about configuring the timeout value, see *SAP HANA XS Configuration Parameters* in the SAP HANA Administration Guide.

4.3 Creating Organizational Units

You create organizational units in the *User Management* application in the *Organizational Units* view.

Prerequisites

You have defined the generic parameters and available attributes for organizational units in the SAP Connected Health application *User Management Configuration*.

Context

Rather than implementing your organization's security concept on a user-by-user basis, you can reflect the structure of your organization in the system by creating organizational units and assigning users to those organizational units. You can then assign configurations and privileges to the organizational units, ensuring that only users assigned to those organizational units can access the corresponding applications and content.

By placing your organizational units in a hierarchy, you can enable organizational units, and therefore the assigned users, to inherit configurations and privileges granted to the parent organizational unit. With multiple levels within a hierarchy, you could build up a structure that would, for example, go from a nation-wide clinical association down to individual teams within a specific department of a practice.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

-
2. Open the *User Management* application and go to the *Organizational Units* view.
 3. If you want to create an organizational unit within a hierarchy, select the existing organizational unit that is the direct parent of the organizational unit you want to create.
 4. Create the organizational unit by choosing *Create Organizational Unit*.
 5. Specify the organizational unit attributes.
 6. Save the organizational unit.

Results

After you create an organizational unit, you can assign configurations to it in the *Configuration Assignment* application, which will enable users assigned to the organizational unit to view the corresponding applications.

4.4 Reassigning Organizational Units

You reassign organizational units in the *User Management* application in the *Organizational Units* view.

Prerequisites

You have defined the generic parameters and available attributes for organizational units in the SAP Foundation for Health *User Management Configuration* application.

Context

You can reassign users from one parent organizational unit to the other by transferring users to a new position in the organizational structure. This allows you to recreate a new organizational unit structure based on your business needs.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Open the *User Management* application and go to the *Organizational Units* view.

3. To reassign the parent organization unit for a specific organizational unit, select that unit and click the right arrow symbol in the *Details* column.
4. Choose *Edit Organization*.
5. In the *Organizational Unit Details* view, choose *Reassign Parent*.
6. In the *Reassign Organizational Unit* view, select the parent organization to reassign and choose *OK*.
The parent organization is now reassigned for the chosen organizational unit.
7. In the *Organizational Unit Details* view, save the changes in the organizational unit structure.

Results

After you save the changes, you can see the renewed hierarchy in the *Organizational Units* view.

4.5 Deactivation of Users and Organizational Units

In the *User Management* application, you can deactivate individual users or entire organizational units.

i Note

If you deactivate an organizational unit, any existing child organizational units are also automatically deactivated. In addition, any users assigned to the organizational unit or its children are automatically deactivated as well. However, if a user is still assigned to another active organizational unit, this user is not deactivated.

For auditing reasons, users and organizational units are not deleted; they are simply deactivated. Anything that the deactivated user created in the system, such as notes or cohorts, will still remain in the system.

4.6 Integrating Users and Organizational Units from LDAP Directory

Integrate the list of users and organization units residing in the LDAP directory into SAP Connected Health using an LDAP adapter and LDAP data integration extension. You configure this extension in the *Data Integration Cockpit*.

Prerequisites

- An SAP HANA Enterprise Information Management data provisioning agent is installed and configured and has access to the LDAP directory with the details of users and organizational units.

- The LDAP adapter is deployed and configured and you have created the required virtual tables and remote sources.
For more information, see the SAP Connected Health Platform Implementation Guide.
- You have activated the relevant plug-in in the *Plug-In Cockpit*.
- You have configured the relevant data integration extensions in the *Data Integration Cockpit*. For more information, see [Creating Profiles for Data Integration Extensions \[page 96\]](#).
- You are assigned the role `HC_HPH_PLUGINS_ADMIN`.

Procedure

1. If you haven't already done so, set up the LDAP adapter and create virtual tables for it.
 - a. Register the LDAP adapter with SAP HANA.

To do so, follow the steps described in .

→ Tip

As an easy alternative, execute the following SQL statement:

```
CREATE AGENT '<agent name>' PROTOCOL 'TCP' HOST '<host where agent was installed>' PORT '<port specified during agent installation>'
CREATE ADAPTER 'SAP HPH XML Adapter' AT LOCATION AGENT '<agent name>'
```

- b. In your SAP HANA system, create and configure a remote source for the LDAP adapter.

To do so, follow the steps described in .

When you configure the remote source, you define the following parameters:

Table 72:

Parameter	Mandatory	Description
Server Host	X	Host address for LDAP server .
Server Port	X	Connection port for LDAP server.
Root Schema	X	Parent organization selected for import.

Parameter	Mandatory	Description
Full DN as ID		DN stands for distinguished name for an organizational unit or user ID in the LDAP directory. Default values are "True" or "False". When this value is set to true, the system uses the full distinguished name from the LDAP directory. When the value is set to False, the system uses only the last part of the DN.
File Path		LDAP attribute configuration file path. This JSON file stores user attribute details that provide mapping between LDAP server and SAP Connected Health tables.
Username	X	Username to access the LDAP server.
Password	X	Password for the LDAP server account.

- c. Create the required virtual tables corresponding to the remote source you've created and assign them to your schema.

By default, the system suggests the following virtual tables:

- SAP_HPH_VT_UM_ORG
- SAP_HPH_VT_UM_USER
- SAP_HPH_VT_UM_ORG_USER
- SAP_HPH_VT_LOG_DETAIL
- SAP_HPH_VT_LOG_TRACE

i Note

Virtual tables for the LDAP adapter must be created in the `SAP_HPH` schema.

→ Tip

You can rename these virtual tables to anything you want.

2. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

3. Open the *Plug-In Cockpit* app.

4. Select and activate the SAP Connected Health Platform LDAP Plug-In.
5. Open the *Data Integration Cockpit* app.
6. Create a profile for data integration.
 - a. Open the *Run and Manage Profiles* view.
 - b. Choose *Create Profile*.
 - c. Choose the relevant data integration extension.

➔ Remember

Data integration extensions are only available if the corresponding plug-in is activated.

- d. Optionally, choose a DW Source.

DW Source is an optional attribute used in the clinical data warehouse to filter data during integration. This means, only data matching the specified DW Source will be integrated during the profile run.

You can create a new DW Source by choosing the **+** icon.

- e. Optionally, choose a schedule for running the profile.

The schedule is based on an existing schedule configuration. You can create a new schedule configuration by choosing the **+** icon.

i Note

You can't create schedule configurations for the following data formats:

- GFF3
- VCF
- BED
- PED

- f. Specify the required parameters.

For more information about the profile parameters, see [LDAP Extension \[page 116\]](#).

7. Activate the profile and choose *Run Profile*.
8. To monitor the status of the profile run, open the *Monitor Profile Runs* view.

Results

After the profile run has completed, you can confirm the data was correctly integrated by either viewing the data in the target tables of the data integration extension or searching for the data in the end-user applications.

Related Information

[LDAP Extension \[page 116\]](#)

4.6.1 LDAP Extension

SAP Connected Health provides data integration extension to integrate organizational units and users from an LDAP directory. You configure this extension in the [Data Integration Cockpit](#).

Profile Parameters

The following parameters are available when creating profiles for the LDAP extension:

Table 73:

Parameter	Description	Default
LDAP Remote Source	Name of the LDAP remote source table.	<null>
Import Entity	Specify the type of entity you want to import from the LDAP directory. For example, organizational units or users.	<ul style="list-style-type: none">• Organizations• Organizations and Users

Target Tables

When you have imported and integrated organizational units and users using the LDAP extension, the following SAP Connected Health tables are populated:

Table 74:

Table	Purpose
"SAP_HPH"."sap.hc.hph.um.user.db::userTable.USER_ATTRIBUTES"	Stores user details such as first name, last name, and external (LDAP) source name.
"SAP_HPH"."sap.hc.hph.cdw.db.models::Config.Org"	Stores organization specific details such as org name, address, and org ID.
"SAP_HPH"."sap.hc.hph.cdw.db.models::Config.UserOrgMapping"	Stores LDAP specific information such as external org ID, external source, and account validity.

5 Authorizations

For applications running on SAP Connected Health, users correspond to real people and are created as restricted users in SAP HANA. As restricted users, users can't create any objects in the database and they can't access any data in the database.

Security Logic

By default, the security logic for SAP Connected Health is incorporated into the database views.

➔ Recommendation

We recommend you use the provided default security logic. If you want to adjust the logic in any way, we recommend developing and implementing custom views.

Related Information

[User Roles \[page 117\]](#)

[Technical Users \[page 119\]](#)

[SAML Configuration \[page 120\]](#)

[Analytic Privileges \[page 120\]](#)

5.1 User Roles

User roles determine which applications or system components a user can access. User roles do not grant access to data.

➔ Remember

You must create users as SAP HANA restricted users.

SAP Connected Health provides standard business roles with the plug-in `sap.hc.hph.plugins.um`. This plug-in is active by default and provides the following roles:

i Note

The following table describes only the roles specific to SAP Connected Health. If you have any applications that run on top of this foundation, users may require additional application-specific roles. For more information, see the corresponding application guides.

In addition, SAP Connected Health provides specific roles for using and administering data integration plug-ins. The following roles are provided with the delivery unit `HCO_HC_HPH_CON`

Table 75:

Role	Technical Name	Description
SAP Connected Health Application Administrator	<code>sap.hc.hph.roles::HC_HPH_APP_APPLICATION_ADMIN</code>	<p>This role grants access to the following administrative and configuration applications:</p> <ul style="list-style-type: none"> • User Management • Patient Analytics Configuration <p>If you have installed other applications on the platform, such as SAP Medical Research Insights, this role also provides access to any corresponding administrative applications.</p>
SAP Connected Health Plug-in Administrator	<code>sap.hc.hph.plugins.roles::HC_HPH_PLUGINS_ADMIN</code>	<p>This role grants access to all administrative and configuration applications delivered with SAP Connected Health. In addition, this role also grants administration privileges for all default plug-ins delivered with SAP Connected Health.</p> <div style="background-color: #fff9c4; padding: 10px; border: 1px solid #ccc;"> <p>➔ Recommendation</p> <p>If you want to explicitly prevent your administrator from viewing personal patient data, we recommend that you add your own configuration administrator role. You can use this role to prevent administrators from assigning themselves the SAP Connected Health End User role and therefore obtaining rights to view personal patient data that end users can view. For more information, please contact your SAP technical consultant.</p> </div>

Role	Technical Name	Description
SAP Connected Health End User	<code>sap.hc.hph.plugins.roles::HC_HPH_PLUGINS_END_USER</code>	This role is granted to application users. It grants access to end-user applications, such as <i>Cohorts</i> , <i>My Profile</i> , and the <i>Patient Summary</i> application. In addition, this role grants privileges necessary for end users to retrieve documents.

The following roles are not available for new users accessing the SAP Connected Health applications. However, as an existing user, you may have already been assigned to these roles, you can continue to use the corresponding role privileges. For the new users, the role privileges associated with the roles in the following table are included in the SAP Connected Health Plug-in Administrator and SAP Connected Health End User roles:

Table 76:

Technical Name	Description
<code>sap.hc.hph.roles::HC_HPH_APP_PLATFORM_ADMIN</code>	This role grants access to all administrative and configuration applications delivered with SAP Connected Health.
<code>sap.hc.hph.roles::HC_HPH_APP_END_USER</code>	This role is granted to application users. It grants access to end-user applications, such as <i>Cohorts</i> , <i>My Profile</i> , and the <i>Patient Summary</i> application.

i Note

You should not grant additional roles to SAP Connected Health users, as this may bypass the authorization concept.

Roles for Other Applications:

You can use the user management application to assign administrative and end user roles specific to other applications such as SAP Medical Research Insights. To assign roles specific to other applications, you must have the corresponding plug-ins.

5.2 Technical Users

In SAP Connected Health, application users and data integration extensions can't access the database directly. Instead, such queries are executed indirectly by the technical user.

When SAP Connected Health is being implemented, a technical user must be created and assigned the following roles:

Table 77:

Technical Role	Description
<code>sap.hc.hph.roles::HC_HPH_TECH_CONNECTION</code>	Provides all SQL-level privileges for handling queries on the database from applications
<code>sap.hc.hph.plugins.roles::HC_HPH_PLUGINS_TECH_CONNECTION</code>	Required for data integration extensions to connect and write to the database This role includes the role <code>HC_HPH_TECH_CONNECTION</code> .

5.3 SAML Configuration

You can enable Security Assertion Markup Language (SAML) to authenticate users accessing SAP Connected Health applications.

To enable SAML authentication method for new and existing users on the SAP Connected Health platform, you must ensure you configure the [SAML](#) option in the [AUTHENTICATION](#) panel in the SAP HANA XS Administration Tool.

Additionally, you must do the following to enable SAML assertions to confirm the logon credentials of a user:

- Configure an SAP HANA system to act as a service provider.
- Add SAML Identity Providers for the required users in the SAP HANA XS Administration Tool.

For more information about SAML configuration, see *Maintaining SAML Providers* in the SAP HANA Administration Guide.

For more information about SAML as a logon authentication method for users in User Management application, see [Creating Users \[page 107\]](#).

5.4 Analytic Privileges

By default, the security logic for SAP Connected Health is incorporated into the database views. Analytic privileges are not supported by default.

If you already use custom analytic privileges, you can configure the system to use them. However, as soon as possible, you should adopt the approach of incorporating security logic into database views.

For more information, please contact your SAP technical consultant.

➔ Recommendation

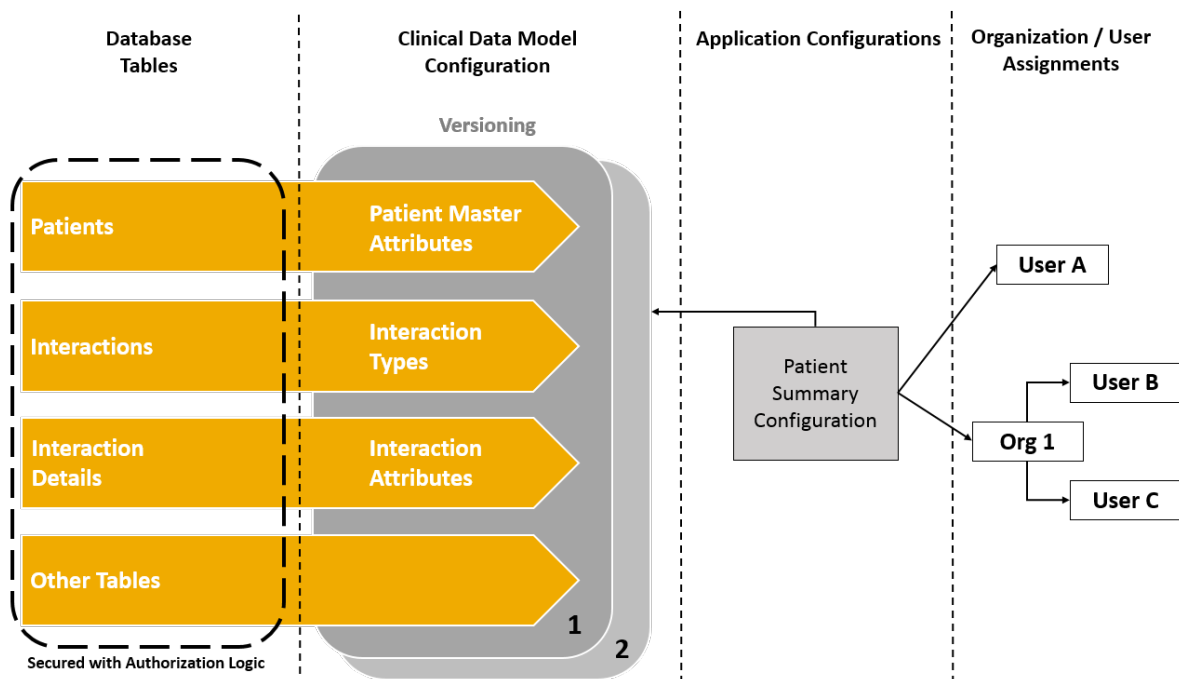
We recommend you use the provided default security logic. If you want to adjust the logic in any way, we recommend developing and implementing custom views.

For more information about implementing custom views, see [Customizing Data Consumption for Applications \[page 169\]](#).

6 Application Configuration

The application configuration concept for SAP Connected Health requires you to define one or more clinical data model configurations, which define, for example, which interaction types are relevant, what attributes apply, and which database tables and columns are used. Any application configurations, which define how the data appears on the user interface, are based on the underlying clinical data model configuration that you assign.

The following diagram uses the example of the *Patient Summary* application to depict how these different configurations work together to enable users to view patient data:



Related Information

- [Clinical Data Model Configuration \[page 123\]](#)
- [Assigning Configurations to Users \[page 153\]](#)
- [Configuring End-User License Agreements \[page 154\]](#)
- [Configuring Patient Summary \[page 156\]](#)
- [Configuring Search \[page 164\]](#)
- [Customizing Data Consumption for Applications \[page 169\]](#)

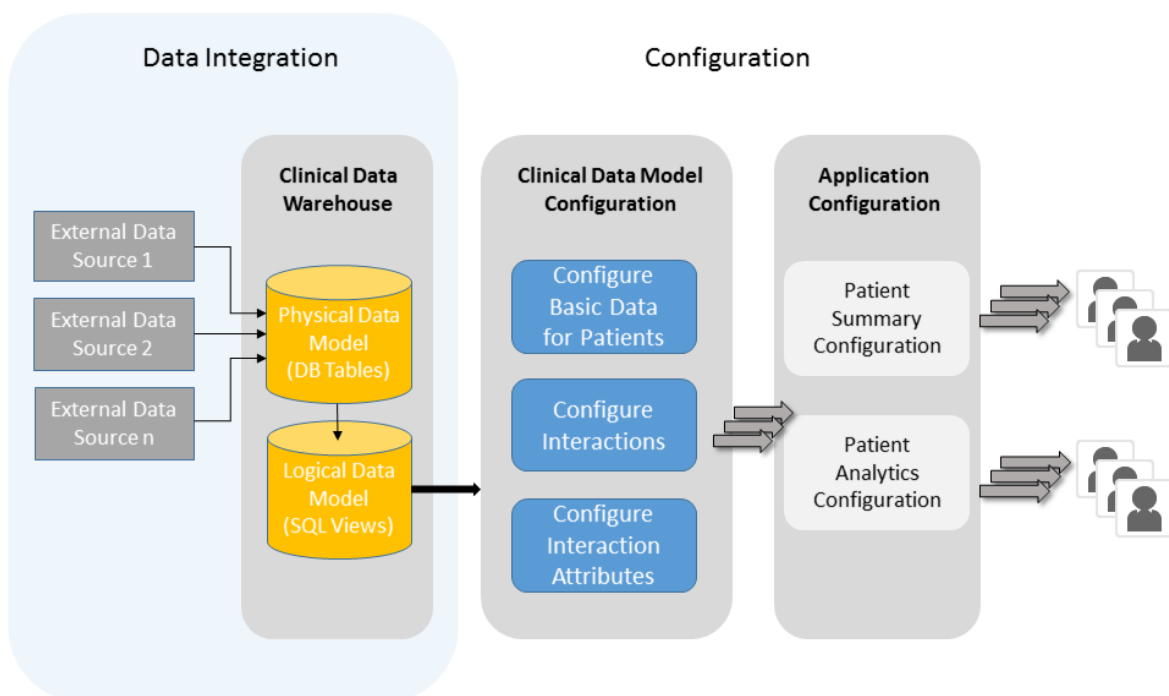
6.1 Clinical Data Model Configuration

Once you have integrated your clinical data into SAP Connected Health, you can control what data your users see in the applications. By defining what data is shown and how the data is combined and filtered, you can tailor the available data for different users and different use cases.

When you integrate clinical data from external data sources into SAP Connected Health, data integration extensions determine how your data is mapped to your clinical data model and stored in the clinical data warehouse. However, to use this data as content for applications running on SAP Connected Health, you need to define exactly which data is used, how it is filtered or combined, and how it is presented.

In SAP Connected Health, your clinical data model is based on the patient data from your organization. This data typically includes patient data, diagnosis data, genomic data, laboratory or biobank data, and data on treatments such as chemotherapy and radiotherapy. In the *Clinical Data Model Configuration* application of SAP Connected Health, you define each of these sets of data as an interaction entity. This configures how the system reads information from the views of the logical data model.

The following figure outlines the tasks you need to complete in clinical data model configuration:



This information captures:

- How the activities you complete in the *Clinical Data Model Configuration* application follow on from the data integration process
- How you use the logical data model as the basis for your clinical data model configurations
- Which activities you need to complete in the *Clinical Data Model Configuration* application to create your clinical data model configurations
- How you create clinical data model configurations with specific end users and use cases in mind

Once you have completed your activities in the *Clinical Data Model Configuration* application, you have created clinical data model configurations in your system that you can use as a basis for application-specific configurations. These typically define visual aspects of applications tailored to specific users and use cases. You use these clinical data model configurations as the basis for configurations in other configuration applications, such as the *Patient Summary Configuration* or *Patient Analytics Configuration* applications, for example.

Related Information

[Managing Configurations \[page 124\]](#)

[Configuring Basic Data for Patients \[page 125\]](#)

[Required Attributes \[page 129\]](#)

[Configuring Interactions \[page 130\]](#)

[Configuring Interaction Attributes \[page 136\]](#)

[Advanced Settings for Clinical Data Model Configurations \[page 142\]](#)

[Example: Configuring a Primary Tumor Diagnosis Interaction \[page 144\]](#)

6.1.1 Managing Configurations

When you open the *Clinical Data Model Configuration* application, you see an overview of all available clinical data model configurations. From this overview, you can manage your configurations, which includes creating new ones and selecting which versions to activate.

From the overview screen, you can perform the following actions to manage your configurations:

Table 78:

Action	Details
Create	You can create a configuration by simply choosing <i>Create Configuration</i> and giving it a unique name.
Delete	You can delete either entire configurations or only specific versions of configurations. If you delete a configuration, all versions are also deleted.
Import	You can import configurations that were edited and saved in the JSON file format by an external editor program.
Duplicate	You can create a new clinical data model configuration by duplicating an existing version of a clinical data model configuration and editing it.

Action	Details
Select version	Each configuration can have multiple versions. Any time you make changes to an existing configuration and save it, a new version is created. However, only one version of a configuration can be active at one time.
Activate	You define which version of a clinical data model configuration is the active version. The active version is the version that is recommended for use in application configurations. For each clinical data model configuration, there can only be one active version.
Export	You can export versions of clinical data model configurations and save them as JSON files.

6.1.2 Configuring Basic Data for Patients

Context

In the clinical data warehouse, basic patient data, such as date of birth, last name, and address, is usually mapped to the patient entity. In the clinical data model configuration, you define how the data from the database tables is combined and filtered so that it can be presented as attributes in the applications running on SAP Connected Health.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Open the *Clinical Data Model Configuration* application.
3. Create a new configuration or select a version of an existing configuration and edit the version.

➔ Recommendation

As your system can support numerous configurations, to avoid later confusion, we recommend that you choose a descriptive name that indicates the intent, purpose, or role of the related users.

4. Under *General Settings*, choose *Basic Data*.

- To generate suggestions for basic patient data attributes based on the tables in the clinical data warehouse, choose [Generate Suggested Interactions](#) and accept the suggested attributes that you want to keep.

i Note

Some basic data attributes, including required attributes, are accepted automatically.

- Add a new attribute or select an existing attribute.
- For the attribute, specify the following attribute details:

Table 79:

Details	Mandatory	Description
Name	Mandatory	Name of the attribute that appears on the user interface if a language-specific name hasn't been specified
Language / Title	Optional	Language-specific names of the attribute that appear on the user interface
Attribute ID to Use in Configuration	Mandatory	The attribute ID is generated automatically by the system. You only need to adjust the ID in this field in special situations.

i Note

An example of a special situation might be if you want to create your [Basic Data](#) attributes manually. The system requires some of these attributes to have fixed IDs. When you create a special attribute manually, you need to replace the automatically generated ID in this field with the fixed ID. Examples of attributes with fixed IDs are the [Basic Data](#) attributes [Patient ID](#) (pid) and [Patient Count](#) (pcount).

- Select the configuration option you want to use to define where to get the values for your attribute: [RELATIONAL](#) or [ADVANCED](#).
- Depending on the configuration option you select, specify the following:

Configuration Option	Fields
RELATIONAL	<p>Data Source (Expression)</p> <p>Select an attribute stored in the patient table.</p>

Configuration Option	Fields
	<p>i Note</p> <p>You use this option when you want to access an attribute stored in the patient table.</p>
<i>ADVANCED</i>	<p>See the next step.</p> <p>i Note</p> <p>You use this option when you want to configure a more complex attribute. This type of attribute allows you to:</p> <ul style="list-style-type: none"> ○ Combine data from multiple sources ○ Apply custom formatting to your attribute ○ Define a custom aggregation, for example, to obtain a count or an average of a set ○ Refine the filter for your attribute

10. If you select *ADVANCED* as the configuration option, specify the following:

- a. If required, use the *Add New Custom Table* option to specify one or more custom tables or views.

You can use custom tables to adjust the mapping of placeholders and, therefore, retrieve information from different tables. Available placeholders are preceded by the @ sign. For example, you can refer to the OBSERVATIONS table using the @OBS placeholder. Placeholders for basic patient data attributes are limited to @PATIENT, @OBS, and @REF (for the reference catalog).

i Note

@PATIENT can only be changed in the *Advanced Settings for Clinical Data Model Configurations*.

- b. Specify whether the attribute is a normal attribute or an aggregate attribute.
c. For a normal attribute, specify the following:

Table 80:

Detail	Mandatory	Description
Data Source (Expression)	Mandatory	<p>An SQL expression used in the SELECT clause of an SQL statement, used to specify the source of the information for this attribute.</p> <p>➔ Tip</p> <p>Ensure you enter your SQL expression in uppercase.</p>

Detail	Mandatory	Description
Filter Expression	Optional	<p>An SQL expression used in the WHERE clause of an SQL statement, used as a filter for entries of the corresponding table.</p> <div style="background-color: #fff9c4; padding: 5px;"> <p>➔ Tip</p> <p>Ensure you enter your SQL expression in uppercase.</p> </div>

- d. For an aggregate attribute, specify a measure expression and optionally a filter expression. A measure expression is an SQL expression used in the SELECT clause of an SQL statement. This expression is an aggregation operation, for example, for obtaining a count or an average of a set.

11. Specify what type of data the attribute provides.

The following data types are supported:

Table 81:

Data Type	Typical Use
<i>Text</i>	For defining attributes based on data stored in text format
<i>Number</i>	For defining attributes based on data stored in integer format or other numeric formats
<i>Free text</i>	For defining attributes based on a database column that has a fuzzy search index and that contains more data than is relevant for your attribute, such as an entire paragraph. You might use this if you want to apply a fuzzy search on your data, for example.
<i>Date</i>	For defining attributes based on data stored with date information only (such as YYYY-MM-DD)
<i>Date and time</i>	For defining attributes based on data stored with date and time information (such as YYYY-MM-DD HH24:MI:SS.FF7)

➔ **Tip**

Make sure that the data type you select matches what is returned by your expression, particularly when you define attributes for dates and times. Consider whether you need to use a conversion function in your SQL expression in the *Data Source (Expression)* field (for a normal attribute) or *Measure Expression* field (for an aggregate attribute). For example:

- If you are defining an attribute for data stored as `TIMESTAMP` in your table column, and you select *Date* here as the data type for your attribute, you need to use the `TO_DATE` conversion function to convert the timestamp value into the `DATE` data type.
- If you are defining an attribute for data stored for a numeric format (such as integer) as `TEXT` in your table column, and you select *Number* here as the data type for your attribute, you need to use the `TO_INTEGER` conversion function to convert the string into the `INTEGER` data type.

12. Optionally specify any additional properties.

You can use this field to add snippets of JSON properties.

➔ Recommendation

If you want to add additional properties, please contact your SAP technical consultant.

13. Specify whether the attribute is a catalog attribute.

A catalog attribute is an attribute for which a set of possible values is defined. Ontology services terms can be used to define the set of possible values. In an application, a catalog attribute can offer the user a list of all possible values, in a dropdown list, for example. The catalog attribute can return information on all the defined possible values. Otherwise an attribute returns only information on the current values in the database.

If you want to define your attribute as a catalog attribute, specify the following:

Table 82:

Detail	Mandatory	Description
Reference Expression	Mandatory*	Here you specify an SQL expression used in the SELECT clause of an SQL statement. This is used to specify the source of the information for this attribute. This information is used to populate the texts displayed as lists for attribute fields on filter cards in <i>Patient Analytics</i> , for example.
Reference Filter	Mandatory*	Here you specify an SQL expression used in the WHERE clause of an SQL statement. This is used to select the correct reference catalog in your reference table.

* Only mandatory when you define your attribute as a catalog attribute

14. Choose *Test Attribute* to test that you've configured your attribute properly based on your data model.

i Note

If you're assigned the end-user role, the *Test Attribute* option also allows you to see sample data for the attribute.

6.1.3 Required Attributes

When you configure attributes manually for your clinical data model configuration, there are certain attributes that you need to ensure are configured and also assigned the intended IDs.

Depending on the intended use of your clinical data model configuration, there are certain attributes that you need to ensure are configured. These required attributes also need to be assigned specific IDs. These

attributes are all *Basic Data* attributes or *Default Attributes*. When you use the *Generate Suggested Interactions* option to create your interactions and attributes, these attributes are generated automatically and assigned the correct IDs where applicable. However, if you want to create your interactions and attributes manually, check the table below to ensure that you configure the required attributes correctly.

Table 83:

Basic Data Attribute	ID	Required For
Patient ID	pid	All applications that use CDM configurations
Patient Count	pcount	<i>Patient Analytics</i>
Date of Birth	dateOfBirth	<i>Patient Analytics</i> <i>Patient Summary</i>
Date of Death	dateOfDeath	<i>Patient Summary</i>

Table 84:

Default Attribute	ID	Required For
Start Date	start	<i>Patient Analytics</i> <i>Patient Summary</i>
End Date	end	<i>Patient Analytics</i> <i>Patient Summary</i>

i Note

When you create an attribute manually, an attribute ID is generated automatically by the system. You can adjust this manually by entering the correct attribute ID as outlined above in the *Attribute ID to Use in Configuration* field when you configure your attribute.

6.1.4 Configuring Interactions

Context

The interactions that you define in the clinical data model configuration are used as categories and filters in applications running on SAP Connected Health. For each interaction you define a name, specify which data from the clinical data warehouse tables is used for the interaction, and which attributes belong to the interaction.

You can choose *Generate Suggested Interactions* to generate suggested interactions and attributes and then accept the generated suggestions that you want to keep.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Open the *Clinical Data Model Configuration* application.
3. Create a new configuration or select a version of an existing configuration.

➔ Recommendation

As your system can support numerous configurations, to avoid later confusion, we recommend that you choose a descriptive name that indicates the intent, purpose, or role of the related users.

4. To create new interactions, you have two options:
 - To generate interaction definitions automatically based on the data available, choose *Generate Suggested Interactions* and then accept the suggested interactions that you want to keep.
 - To define a new interaction manually, choose *Add Interaction*.

i Note

If you want to use the Variant Browser, choose *Generate Suggested Interactions* and use the suggested *Genome Sequencing* interaction and *Genetic Variant* interaction.

5. Define the interaction.

Each interaction consists of the following elements, which you can define:

Table 85:

Definition Element	Mandatory	Description
Name	Mandatory	Name of the interaction that appears on the user interface if a language-specific name hasn't been specified
Language / Title	Optional	Language-specific name of the interaction that appears on the user interface

Definition Element	Mandatory	Description
Interaction ID to Use in Configuration	Mandatory	<p>The ID for your interaction is generated automatically by the system. You only need to adjust the ID in this field in special situations.</p> <div data-bbox="1023 465 1394 1032" style="background-color: #fff9c4; padding: 10px;"> <p>i Note</p> <p>An example of a special situation might be if you want to replace an existing interaction with a new one. If you create your new interaction by duplicating your existing interaction, the new interaction is assigned a new ID automatically by the system. When you adjust your duplicated interaction, you can copy the ID from the original interaction into this field in the new interaction once you've noted the ID and deleted the original interaction. This ensures that your configuration containing this interaction can still be used as before.</p> </div>
Filter Expression: Basic/Advanced	Mandatory	<p><i>Basic</i></p> <p>This is the default configuration option when you create a new interaction. When you choose this option, you select your interaction type from the dropdown list. This list contains the INTERACTION_TYPES from the INTERACTIONS table.</p> <p><i>Advanced</i></p> <p>Here, you specify an SQL expression used in the WHERE clause of a statement that is applied as a filter to the INTERACTIONS table and returns only the records that are included in this interaction definition. Usually this will be an equality condition on the INTERACTION_TYPE column of the INTERACTIONS table.</p>

Definition Element	Mandatory	Description
Custom Tables	Optional	If you choose the <i>Advanced</i> configuration option, you can use custom tables to adjust the mapping of placeholders. This allows you to retrieve information from different tables. Available placeholders are preceded by the @ sign. For example, you can refer to the INTERACTIONS table using the @INTERACTION placeholder.

Definition Element	Mandatory	Description
Condition Type	Optional	<p>You can specify a condition type if you want interactions of this type to be grouped based on the medical condition they are associated with (such as a specific tumor). If you define a condition type for an interaction, the interaction is then associated with a condition. When users filter data based on this interaction in <i>Patient Analytics</i> using multiple filter cards assigned to the same condition type, patients will only match the filter criteria if they have a matching interaction <i>belonging to the same condition</i> (that is, interactions that have the same condition ID in the interactions table).</p> <p>For example, you can enter Tumor Case if you want to use this condition type to group interactions that are related to the same tumor. If a patient undergoes two chemotherapy treatments for two different tumors and a tumor case is defined as the condition type, when users filter for patients with two chemotherapy treatments in <i>Patient Analytics</i>, this patient would not be included in the data returned. This is because the patient had just one chemotherapy treatment for each tumor. If you don't configure a condition type in this case, the patient would be a match when users filter for patients with two chemotherapy treatments.</p> <div data-bbox="1023 1585 1390 1944" style="background-color: #fff9c4; padding: 10px;"> <p>➔ Tip</p> <p>You can enter any arbitrary name when you create a condition type. Condition types are specific to the configuration you're configuring. Once you create a condition type, you can select it for any interaction within that specific configuration.</p> </div>

Definition Element	Mandatory	Description
		<p>i Note</p> <p>To be able to use this feature in <i>Patient Analytics</i>, users need to know which interactions have been defined with conditions types. It is recommended to only use this <i>Condition Type</i> feature when this information is available to users and if a condition ID was saved consistently for your data at the data modeling stage.</p>
Parent Interaction	Optional	<p>You can specify for your interaction whether there are any interactions you want to allow your users to define as parent interactions. This allows users to specify parent-child dependencies between interactions when they analyze their data in <i>Patient Analytics</i>, for example.</p> <p>You can specify any number of possible parent interactions for a child interaction. You specify these by selecting the relevant entries in the drop-down list.</p> <p>i Note</p> <p>If you want to use this feature, check your interaction attributes table</p> <pre>sap.hc.hph.cdw.db.models:: DWEntities:Interactions_At tr</pre> <p>and ensure that the <code>DWID_ParentInteraction</code> column is filled for this interaction.</p>

Definition Element	Mandatory	Description
Filter Control Label for Parent-Child Relationship	Optional	If you specify possible parent interactions in the <i>Parent Interaction</i> field, you specify here the type of parent-child relationship between the interactions. What you specify here appears as the label for the filter card attribute that users add to their child interaction filter card in <i>Patient Analytics</i> when they define a parent-child relationship.
Saved Attributes	Mandatory	Attributes of the interaction. Attributes contain all the information about an interaction.

6. Add and configure attributes for the interaction.

For more information, see [Configuring Interaction Attributes \[page 136\]](#).

6.1.5 Configuring Interaction Attributes

When you configure the interactions for your clinical data model configuration, you specify in detail the attributes you want to make available for this interaction.

Context

You can define interaction attributes for the following:

- Default attributes
Attributes applied to all interactions
- Defined interactions
Attributes applied to specific interactions you define
- Suggested interactions
Attributes applied to specific interactions that were generated automatically based on the tables of the clinical data warehouse

i Note

If you want to use the Variant Browser, do not modify the *Sample Id* attribute of the *Genome Sequencing* interaction and the *Location* attribute of the *Genetic Variant* interaction.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Open the *Clinical Data Model Configuration* application.
3. Select a version of an existing configuration.
4. Choose the interaction whose attributes you want to define or choose *Default Attributes*.
5. Add a new attribute or select an existing attribute.
6. For the attribute, specify the following attribute details:

Table 86:

Details	Mandatory	Description
Name	Mandatory	Name of the attribute that appears on the user interface if a language-specific name hasn't been specified
Language / Title	Optional	Language-specific names of the attribute that appear on the user interface
Attribute ID to Use in Configuration	Mandatory	The attribute ID is generated automatically by the system. You only need to adjust the ID in this field in special situations.

i Note

An example of a special situation might be if you want to create your *Default Attributes* manually. The system requires some of these attributes to have fixed IDs. When you create a special attribute manually, you need to replace the automatically generated ID in this field with the fixed ID. Examples of *Default Attributes* with fixed IDs are the *Start Date* (`start`) and *End Date* (`end`).

7. Select the configuration option you want to use to define where to get the values for your attribute: *EAV*, *RELATIONAL*, or *ADVANCED*.
8. Depending on the configuration option you select, specify the following:

Configuration Option	Fields
<i>EAV</i>	<i>Filter Expression</i>

Configuration Option	Fields
	Select an attribute stored in the tables for interaction details, measures, or text.
<i>RELATIONAL</i>	<i>Data Source (Expression)</i> Select an attribute stored in the interactions table.
<i>ADVANCED</i>	See the next step. <div style="background-color: #fff9c4; padding: 10px;"> <p>i Note</p> <p>You use this option when you want to configure a more complex attribute. This type of attribute allows you to:</p> <ul style="list-style-type: none"> ○ Combine data from multiple sources ○ Apply custom formatting to your attribute ○ Define a custom aggregation, for example, to obtain a count or an average of a set ○ Refine the filter for your attribute </div>

9. If you select *ADVANCED* as the configuration option, specify the following:

- a. If required, use the *Add New Custom Table* option to specify one or more custom tables or views.

You can use custom tables to adjust the mapping of placeholders and, therefore, retrieve information from different tables. Available placeholders are preceded by the @ sign. For example, you can refer to the INTERACTION_DETAILS table using the @CODE placeholder.

i Note

This sub-step is relevant for interaction attributes only and not for default attributes.

- b. Specify whether the attribute is a normal attribute or an aggregate attribute.
- c. For a normal attribute, specify the following:

Table 87:

Detail	Mandatory	Description
Data Source (Expression)	Mandatory	An SQL expression used in the SELECT clause of an SQL statement, used to specify the source of the information for this attribute. <div style="background-color: #fff9c4; padding: 10px;"> <p>➔ Tip</p> <p>Ensure you enter your SQL expression in uppercase.</p> </div>

Detail	Mandatory	Description
Filter Expression	Optional	<p>An SQL expression used in the WHERE clause of an SQL statement, used as a filter for entries of the corresponding table.</p> <p>→ Tip Ensure you enter your SQL expression in uppercase.</p>

- d. For an aggregate attribute, specify a measure expression and optionally a filter expression.

A measure expression is an SQL expression used in the SELECT clause of an SQL statement. This expression is an aggregation operation, for example, for obtaining a count or an average of a set.

10. Specify what type of data the attribute provides.

The following data types are supported:

Table 88:

Data Type	Typical Use
<i>Text</i>	For defining attributes based on data stored in text format
<i>Number</i>	For defining attributes based on data stored in integer format or other numeric formats
<i>Free text</i>	For defining attributes based on a database column that has a fuzzy search index and that contains more data than is relevant for your attribute, such as an entire paragraph. You might use this if you want to apply a fuzzy search on your data, for example.
<i>Date</i>	For defining attributes based on data stored with date information only (such as YYYY-MM-DD)
<i>Date and time</i>	For defining attributes based on data stored with date and time information (such as YYYY-MM-DD HH24:MI:SS.FF7)

→ Tip

Make sure that the data type you select matches what is returned by your expression, particularly when you define attributes for dates and times. Consider whether you need to use a conversion function in your SQL expression in the *Data Source (Expression)* field (for a normal attribute) or *Measure Expression* field (for an aggregate attribute). For example:

- If you are defining an attribute for data stored as `TIMESTAMP` in your table column, and you select *Date* here as the data type for your attribute, you need to use the `TO_DATE` conversion function to convert the timestamp value into the `DATE` data type.
- If you are defining an attribute for data stored for a numeric format (such as integer) as `TEXT` in your table column, and you select *Number* here as the data type for your attribute, you need to use the `TO_INTEGER` conversion function to convert the string into the `INTEGER` data type.

- Optionally specify any additional properties.

You can use this field to add snippets of JSON properties.

➔ Recommendation

If you want to add additional properties, please contact your SAP technical consultant.

- Specify whether the attribute is a catalog attribute.

A catalog attribute is an attribute for which a set of possible values is defined. Ontology services terms can be used to define the set of possible values. In an application, a catalog attribute can offer the user a list of all possible values, in a dropdown list, for example. The catalog attribute can return information on all the defined possible values. Otherwise an attribute returns only information on the current values in the database.

If you want to define your attribute as a catalog attribute, specify the following:

Table 89:

Detail	Mandatory	Description
Reference Expression	Mandatory*	Here you specify an SQL expression used in the SELECT clause of an SQL statement. This is used to specify the source of the information for this attribute. This information is used to populate the texts displayed as lists for attribute fields on filter cards in <i>Patient Analytics</i> , for example.
Reference Filter	Mandatory*	Here you specify an SQL expression used in the WHERE clause of an SQL statement. This is used to select the correct reference catalog in your reference table.

* Only mandatory when you define your attribute as a catalog attribute

- Specify whether you want to apply a classification from ontology services when you aggregate your data.

Ontology services translate clinical terms stored in codes into other codes or into natural language. You use this classification feature if you want to:

- Use data encoded in different code systems (vocabularies) for your attribute, for example, a diagnosis that for some patients is encoded in a specific code system (such as ICD-10-CM) and for other patients in another code system (such as ICD-9-CM)
- Aggregate data from within a code system hierarchy (such as ICD-10-CM)

This attribute brings together concepts that are encoded in different code systems and also allows you to specify as of which classification hierarchy level you want to aggregate your data.

If you want to apply a classification from ontology services, specify the following:

Table 90:

Detail	Mandatory	Description
Language	Mandatory*	You enter a two-letter language code indicating the language of the content into which the codes are translated, for example, en for English.
Subject	Mandatory*	You specify how you want to classify your data. Possible options you can enter are: <ul style="list-style-type: none"> ○ diseases.cmsgems: Classify diseases by ICD-10-CM including the translation from ICD-9-CM according to CMS ○ drugs.brandname: Classify drugs by their brand name according to RxNorm ○ drugs.ingredients: Drugs classified by their ingredients according to RxNorm ○ histology.morphology: Classify histology by ICD-O-3 on the morphology axis ○ histology.topography: Classify histology by ICD-O-3 on the topography axis
Starting Hierarchy Level	Mandatory*	You specify the catalog hierarchy level that you want to use as the starting level for your aggregation, such as 0 , 1 , or 2 , where level 0 is the highest level. If you want to aggregate from the highest level of the ICD-10 catalog, for example, you enter 0 . If we take C50 as an example here, all ICD-10 codes with C50, C50.x, C50.xx and C50.xxx would be included in your aggregation, for example.

Detail	Mandatory	Description
Term Context	Mandatory*	<p>This field allows different terms to be used for the same concept depending on the context. This means that in a specific context like 'cancer', terms can be shorter. For the context of a term, you can enter either default or cancer.</p> <p>i Note You can currently only specify cancer when you specify diseases.cmsgems as the <i>Subject</i>.</p>

* Only mandatory when you choose to apply a classification from ontology services

14. Choose [Test Attribute](#) to test that you've configured your attribute properly based on your data model.

i Note

If you're assigned the end-user role, the [Test Attribute](#) option also allows you to see sample data for the attribute.

6.1.6 Advanced Settings for Clinical Data Model Configurations

You define global settings for features based on clinical data model configurations in the [Advanced Settings for Clinical Data Model Configurations](#).

The settings defined in this application apply globally to all clinical data model configurations.

➔ Tip

This application is not included in the standard SAP Fiori launchpad catalogs for SAP Connected Health. To use this application, search for it in the tile catalog and add it manually to your SAP Fiori launchpad.

i Note

These settings apply only to features and applications, such as [Patient Timeline](#) or SAP Medical Research Insights, that use underlying clinical data model configurations.

Placeholder-to-View-Mappings

To simplify the user interface in the *Clinical Data Model Configuration* application, database views are represented by placeholders, such as @Interaction.

Each placeholder corresponds to a database view in the SAP Connected Health schema.

If you create custom entities for clinical entities, you can update the placeholder-to-view mappings.

Table 91:

Placeholder	Description
@INTERACTION	This points to the view for the interactions.
@OBS	This points to the view for the observations.
@CODE	This points to the view for the interaction details.
@MEASURE	This points to the view for the interaction measures.
@REF	This points to the view for the ontology services concepts.
@TEXT	This points to the view for the interaction texts.
@PATIENT	This points to the view for patients with no user-specific access restrictions applied. This is used for access to all patient data for aggregation purposes where no identifiable patient information is made available.
Guarded @PATIENT	This points to the view for patients with user-specific access restrictions applied. Users are automatically blocked from viewing entries that they are not authorized to see. This view is used for direct access to patient data by services that return data with identifiable patient information, such as the <i>Patient Summary</i> application.

Query Settings

- *Fuzziness Threshold*
Defines the grade of error tolerance for a search on an attribute. The value range here is between 0 and 1. The higher the threshold, the more precisely the search terms must match the text to produce a result.
- *Max Results Set Size*
Specifies the maximum number of results returned.

Available Languages

Specifies the languages that are available for selection in clinical data model configurations.

Related Information

[Customizing Data Consumption for Applications \[page 169\]](#)

6.1.7 Example: Configuring a Primary Tumor Diagnosis Interaction

Take a look at an example of the typical steps you follow in SAP Connected Health to configure an interaction and its attributes in the *Clinical Data Model Configuration* application.

Context

As a basic example of how to configure an interaction, let's take primary tumor diagnosis data as an interaction entity that you want to configure. Let's assume that the researchers within your organization require primary tumor diagnosis data to help identify a patient cohort for their research studies. To allow your researchers to later apply specific criteria and display specific information during their work, you define characteristics for your primary tumor diagnosis interaction. In SAP Connected Health, you configure these interaction characteristics as attributes. For the primary tumor diagnosis interaction, let's assume that these attributes are:

- ICD 10 code of the diagnosis
- Age of the patient at diagnosis
- Organization ID
The organization ID allows your researchers to see which departments the patients belong to and potentially identify patients based on this ID, for example.

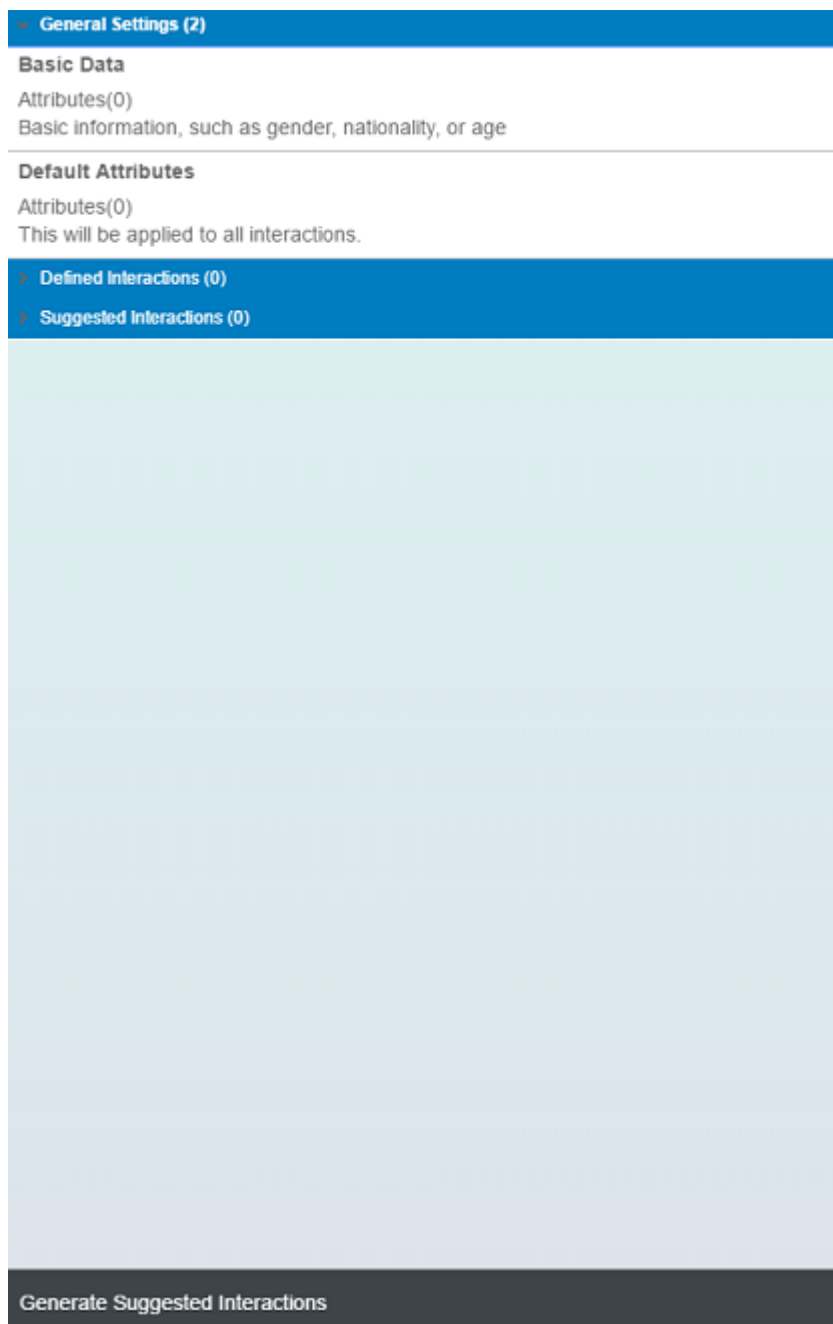
Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Open the *Clinical Data Model Configuration* application.
3. To create a new configuration, choose *Create Configuration*, enter a descriptive name for your configuration, such as **Researcher Config**, and choose *Create*.

The *Clinical Data Model* editor window appears and contains three collapsible sections to the left, which are *General Settings*, *Defined Interactions*, and *Suggested Interactions*:



4. Choose *Generate Suggested Interactions* to the bottom left to generate interactions based on the data in the system.
5. Under *Suggested Interactions*, identify the interaction related to your primary tumor diagnosis data, *ACME_M03*. Choose *Accept* for this interaction:

Suggested Interactions (10)		
Genome Sequencing Attributes(3)	✓ Accept	🗑️ Delete
Genetic Variant Attributes(7)	✓ Accept	🗑️ Delete
ACME_M03 Attributes(8)	✓ Accept	🗑️ Delete
ACME_M03TS Attributes(4)	✓ Accept	🗑️ Delete
ACME_M07_CHEMO Attributes(2)	✓ Accept	🗑️ Delete
ACME_M07_RADIO Attributes(4)	✓ Accept	🗑️ Delete
ACME_M07_SURGERY Attributes(2)	✓ Accept	🗑️ Delete
ACME_M16 Attributes(1)	✓ Accept	🗑️ Delete
BIOBANK Attributes(2)	✓ Accept	🗑️ Delete
Genomic Attributes(3)	✓ Accept	🗑️ Delete

➔ Remember

You can only work with suggested interactions once you've accepted them. When you accept an interaction, the interaction moves to the *Defined Interactions* section.

6. Define the details for your primary tumor diagnosis interaction.
 - a. Under *Defined Interactions*, select your interaction *ACME_M03*.
 - b. In the *Name* field in the editing area in the middle of the screen, replace *ACME_M03* with **Primary Tumor Diagnosis** as the interaction name.
 - c. Under *Language / Title*, choose *Add New Language* to add a German name for your interaction. Select *German* in the dropdown box for the language and enter **Primärtumor-Diagnose** in the field for the title.

➔ Tip

If you don't add language variants, the interaction name in the *Name* field is used as the default name regardless of the system language of the end user.

d. For this interaction, leave the suggested values in the following fields:

- *Interaction ID to Use in Configuration*
- *Filter Expression*
- *Condition Type*
- *Parent Interaction*
- *Filter Control Label for Parent-Child Relationship*

You've now defined the details for your interaction:

Name

Primary Tumor Diagnosis

Language / Title

+ Add New Language

German



Primärtumor-Diagnose



Interaction ID to Use in Configuration:

Primary_Tumor_Diagnosis_188e9bf9_f917_45a3_bea6_e4364e68b4f5

Filter Expression:

Basic

Advanced

@INTERACTION.INTERACTION_TYPE = 'ACME_M03'

Custom Tables:

+ Add New Custom Table

Condition Type:

acme



Parent Interaction:



Filter Control Label for Parent-Child Relationship

parent

7. Under *Saved Attributes*, delete the attributes that you don't need. The only listed attribute that you want to keep is the *ICD_10* attribute.
8. Configure your attribute for the ICD 10 code.
 - a. Under *Saved Attributes*, select *ICD_10*. In the *Attribute Details* section, enter **ICD-10-CM Code** in the *Name* field.
 - b. Under *Language / Title*, add a German language variant with **ICD-10-CM-Code** as the name.
 - c. Don't change the value in the *Attribute ID to Use in Configuration* field.
 - d. Define where you want to get the values for your attribute:
 1. Set *EAV* as the configuration option.

i Note

You use the *EAV* configuration option when you want to access an attribute stored in the tables for interaction details, measures, or text.

2. In the *Filter Expression* dropdown list, select *ICD_10*.
- e. In the *Data Type* dropdown list, leave *Text* as the data type for the attribute.
- f. Define the attribute as a catalog attribute. This allows you to link additional data to your attribute, such as descriptive texts:
 1. Activate the toggle switch for *Specify Attribute as a Catalog Attribute*.
 2. In the *Reference Expression* field, enter **@REF.CODE** as the SQL expression. For the link to the descriptive texts to be added, what is returned by this expression must match the value for the attribute.
 3. In the *Reference Filter* field, enter **@REF.VOCABULARY_ID = 'ICD03.1-TOPO-DIMDI'** as the SQL expression. This expression defines which concept you want to select.

➔ Tip

Choose *Test Catalog Attribute* to test that you've configured your attribute properly based on your data model.

i Note

If you're assigned the end-user role, the *Test Catalog Attribute* option also allows you to see sample data for the attribute.

You've now configured *ICD-10-CM Code* as the first attribute of your interaction:

Attribute Details:

Name

Language / Title + Add New Language

Attribute ID to Use in Configuration:

Filter Expression:

Data Type:

Additional Properties:

Specify Attribute as a Catalog Attribute:

Reference Expression:

Reference Filter:

9. Configure your attribute for the organization ID.
 - a. Under *Saved Attributes*, choose *Add Attribute* and select the new attribute added. In the *Attribute Details* section, enter **Organization ID** in the *Name* field.
 - b. Under *Language / Title*, add a German language variant with **Organisations-ID** as the name.
 - c. Define where you want to get the values for your attribute:
 1. Select *RELATIONAL* as the configuration option.

i Note

You use the *RELATIONAL* configuration option when you want to access an attribute stored in the interactions table.

2. In the *Data Source (Expression)* dropdown list, select *ORG_ID* as the SQL expression.
- d. In the *Data Type* dropdown list, select *Text* as the data type for the attribute.

You've now configured *Organization ID* as the second attribute of your interaction:

Attribute Details:

Name

Language / Title + Add New Language

Attribute ID to Use in Configuration:

Data Source (Expression):

Data Type:

10. Configure your attribute for the age of diagnosis.
 - a. Under *Saved Attributes*, choose *Add Attribute* and select the new attribute added. In the *Attribute Details* section, enter **Age at Diagnosis** in the *Name* field.
 - b. Under *Language / Title*, add a German language variant with **Alter bei Diagnose** as the name.
 - c. Don't change the value in the *Attribute ID to Use in Configuration* field.
 - d. Define where you want to get the values for your attribute:
 1. Select *ADVANCED* as the configuration option and *Normal Attribute* as the attribute type.

i Note

You use the *ADVANCED* configuration option when you want to configure a more complex attribute. This type of attribute allows you to:

- Combine data from multiple sources
- Apply custom formatting to your attribute
- Define a custom aggregation, for example, to obtain a count or an average of a set

- Refine the filter for your attribute

2. In the *Data Source (Expression)* field, enter

FLOOR(DAYS_BETWEEN(@PATIENT.DOB,@INTERACTION."START") / 365) as the SQL expression.

This expression computes the number of years between the date of birth of the patient and the date of the primary tumor diagnosis. It does this by calculating the days between these two events and expressing these as whole years.

3. Either clear the *Filter Expression* field or leave it empty. The reason for this is that you don't need to filter for certain values since you're not accessing EAV tables but accessing table columns directly.

→ Tip

Choose *Test Attribute* to test that you've configured your attribute properly based on your data model.

i Note

If you're assigned the end-user role, the *Test Attribute* option also allows you to see sample data for the attribute.

e. In the *Data Type* dropdown list, select *Number* as the data type for the attribute.

You've now configured *Age at Diagnosis* as the third attribute of your interaction:

Attribute Details:

Name

Language / Title + Add New Language

Attribute ID to Use in Configuration:

Custom Tables: + Add New Custom Table

Attribute Type:
 Normal Attribute

Data Source (Expression):

Aggregate Attribute

Measure Expression:

Filter Expression:

Data Type:

11. Choose *Validate* to test all the attributes in your configuration and confirm that your configuration is valid.
12. Choose *Save & Activate* to save and activate this configuration with your defined interaction and attributes in the back-end system.

Results

You've now created and activated a new clinical data model configuration with an interaction for primary tumor diagnosis in your system. You can use this clinical data model configuration as the basis for configurations in other configuration applications, such as the *Patient Summary Configuration* application.

6.2 Assigning Configurations to Users

If an application requires configuration, you must assign users at least one configuration to enable them to open the application.

Prerequisites

You have defined application configurations in the corresponding administrative application.

Context

Assigning configurations to users allows you to further control what users sees and how they see it.

For example, if you don't think genomic information is relevant for a specific user, you can configure the *Patient Summary* application to not include genomic interactions on the *Patient Timeline*, and then assign this configuration to that user.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Open the *Configuration Assignment* app.
3. To assign a configuration to a user, choose *Create Assignment*.
4. Specify the following attributes for the assignment:

Table 92:

Attribute	Description
Assignment Name	<p>Specifies the name of the assignment.</p> <p>➔ Recommendation</p> <p>As your system can support numerous configurations, to avoid later confusion, we recommend that you choose a descriptive name that indicates the intent, purpose, or role of the related users.</p>
Configuration Type	<p>Specifies which application the assignment relates to. <i>Patient Inspector</i> refers to the <i>Patient Summary</i> application.</p>
Configuration Name	<p>Specifies which configuration for the given application is assigned</p>
Entity Type	<ul style="list-style-type: none"> ○ <i>User</i> Assigns the configuration to a single user. ○ <i>Org</i> Assigns the configuration to all users belonging to a specified organization.
Entity Value	<p>Specifies to which user or organizational unit the assignment is made</p>

6.3 Configuring End-User License Agreements

If you include content in applications that is subject to specific licensing agreements, you can configure end-user license agreements (EULAs), which users have to accept before using the application.

Prerequisites

Your user is assigned the role `HC_HPH_APP_PLATFORM_ADMIN`.

Context

You can define multiple EULAs for an application, each with its own validity period and text. When a user starts an application for the first time, a list of all relevant EULAs appears for them to accept.

To configure EULAs, proceed as follows:

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Open the *End-User License Agreement Configuration* application.
3. From the left panel, select an existing EULA or choose  to create an EULA.

Note

If you want to edit an existing EULA, you can only edit the most recent version.

4. Specify the title of the EULA.

This title doesn't appear on the user interface and is for administrative purposes only.

5. Specify the start and end of the validity of the EULA.

If the EULA never expires, select the *Unlimited* checkbox.

6. Specify the text you want to be displayed on the user interface.

- a. Select a language for the text.

The languages available for EULAs are the same as those available for terminology management.

- b. Specify the name of the EULA.
- c. Enter a description of the EULA.

This description doesn't appear on the user interface and is for administrative purposes only.

- d. Enter the license text of the EULA.

Note

Only formatting created with the integrated rich text editor is supported.

- e. Repeat these steps for any other languages you want to support for the EULA.

7. Save the EULA.

When you save a new EULA or new version of a EULA, it is saved in draft status only.

8. Activate the EULA.

Note

If you've created a new version of an existing EULA and the validity of the new version overlaps with the previous version, the end date of the previous version is automatically overwritten with the start date of the new version.

Results

Whenever you activate a new EULA or a new version of a EULA, users will see and must accept the EULA the next time they try to log on to the system.

You can also terminate an active EULA at any time. When you choose *Terminate*, the end date of the EULA is changed to the current date. This does not affect the status of the EULA.

6.4 Configuring Patient Summary

Context

The *Patient Summary* application provides a chronological overview of the interactions contained in a patient's medical record.

To configure the features available on the *Patient Summary* interface, you select basic patient data, interactions, and attributes defined in a clinical data model configuration. As your organization probably has multiple clinical data model configurations, each with multiple versions, you can choose which configuration you want to use as basis for the configuration of your *Patient Summary* application.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Launch the *Patient Summary Configuration* application.
3. Select the data model configuration on which you want to base your configuration for the *Patient Summary* application.
4. Add a new configuration or choose an existing configuration for the *Patient Summary* application.

➔ Recommendation

As your system can support numerous configurations, to avoid later confusion, we recommend that you choose a descriptive name that indicates the intent, purpose, or role of the related users.

5. On the *Configuration Details* tab, specify the following:

Table 93:

Configuration Detail	Description
Name	Specify a name for the configuration
Data Model Version	Select a version of the data model configuration
Notes	You can enter notes on use and purpose of this configuration.

The data model version that is marked as *active* is the version that is recommended for use.

6. Configure patient details, patient summary tabs, and timeline lanes.
7. Activate the configuration for the *Patient Summary* application.

Related Information

[Configuring Patient Details \[page 157\]](#)

[Activating Patient Summary Tabs \[page 158\]](#)

[Configuring Timeline Lanes \[page 159\]](#)

[Enabling Users to View Genomic Content \[page 160\]](#)

[Activating Patient Summary Configurations \[page 161\]](#)

[Adapting to More Recent Versions of the Data Model Configuration \[page 162\]](#)

[Duplicating Patient Summary Configurations \[page 163\]](#)

[Importing Patient Summary Configurations in JSON Format \[page 163\]](#)

6.4.1 Configuring Patient Details

Context

At the top of the tabs in the *Patient Summary* application, the patient is identified by attributes that you select as header or subheader, for example, the patient ID or name, the patient's gender or year of birth.

To configure the patient details in the header, you select basic patient data attributes defined in a clinical data model configuration. As your organization probably has multiple clinical data model configurations, each with multiple versions, you which configuration you want to use as basis for your configuration for the *Patient Summary* application.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Launch the *Patient Summary Configuration* application.
3. Select the data model configuration and the *Patient Summary* configuration you want to configure.
4. Go to the *Patient Details* tab.
5. To configure a header, choose the attributes you want to use as header in the field on the right side.
6. Enter placeholders for the number of attributes you want to use in the following format: `{0}`, `{1}`, `{2}`.
If you want to use two attributes, use `{0}` and `{1}`.
7. You can enter punctuation marks, blanks, or text between, before, and after the placeholders.
8. You can repeat the procedure for additional lines of subheaders.
9. To save the settings, activate the configuration.

6.4.2 Activating Patient Summary Tabs

You specify which tabs you want to display in the *Patient Summary* application. You specify the tabs to be displayed by activating the tabs in the *Patient Summary Configuration* application. You also define which search view you want to use for the search feature on the *Related Documents* tab.

Context

In the *Patient Summary* application, the *Timeline* tab is always available. You can choose if you want to use two additional tabs:

- The *Overview* tab provides a vertical list of interactions between the patient and medical professionals in reverse chronological order.
- The *Related Documents* tab displays a list of documents associated with the patient.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Launch the *Patient Summary Configuration* application.

3. Select the data model configuration and the *Patient Summary* configuration you want to configure.
4. On the *Summary Tabs* tab, activate the *Overview* tab by turning the toggle switch to *ON*.
5. On the *Summary Tabs* tab, activate the *Related Documents* tab by turning the toggle switch to *ON*.
6. To specify which search view you want to be used on the *Related Documents* tab, select a search view in the *Related Documents* dropdown list.
7. To save the settings, activate the configuration.

6.4.3 Configuring Timeline Lanes


Context

On the *Timeline* tab of the *Patient Summary* application, interactions are grouped by category and each category is shown as a separate lane. You define the categories and choose which interaction attributes are visible in the timeline.

To configure the timeline lanes, you select interactions and attributes defined in a clinical data model configuration. As your organization probably has multiple clinical data model configurations, each with multiple versions, you which configuration you want to use as basis for your *Patient Summary* configuration.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`
2. Launch the *Patient Summary Configuration* application.
3. Select the data model configuration on which you want to base your configuration for the *Patient Summary* application.
4. Add a new configuration or choose an existing *Patient Summary* configuration.
5. On the *Lanes Definition* tab, choose *+ Add Lane*.
6. Enter a lane name for the category.
7. Enter language-specific names for the lane.
8. Select a lane color.
9. Select the interactions that you want to display in the lane by turning the toggle switch to *ON*.
10. For each interaction that you want to display in the lane, choose  to display a list of the attributes of the interaction.
11. Select the attributes that you want to use by turning the toggle switch to *ON*.

12. Specify which of the selected attributes are visible in the timeline. To do so, select which attribute should be displayed as first attribute and which as second attribute.

On the *Timeline* tab, only the values for the first and second attribute of an interaction can be seen initially. Attributes that you select in addition to the first and second attribute can be viewed in the detail view of an interaction and in the *Overview* tab.

If you want to use the Variant Browser, make the *Sample ID* attribute of the *Genome Sequencing* interaction visible. This way, the *Genome Sequencing* interaction provides a link to the Variant Browser view.

13. To define an additional lane, return to the lanes overview.
14. To save the settings, activate the configuration.

6.4.4 Enabling Users to View Genomic Content

For users to view genomic information from the *Patient Summary* application, you must enable the link between the *Patient Summary* and the *Variant Browser* and ensure that your clinical data model configuration supports genomic data.

Prerequisites

You have integrated genomic data into SAP Connected Health.

Context

For a clinical data model configuration to correctly support genomic data, we recommend you use the suggested interactions *Genome Sequencing* and *Genetic Variant* and then activate those interactions in the appropriate application configurations.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Open the *Clinical Data Model Configuration* application.
3. Open the configuration you want to apply or create a new configuration.
4. Choose *Generate Suggested Interactions*.
5. Accept the suggested *Genome Sequencing* interaction and the *Genetic Variant* interaction.

6. Optionally, if you want to edit the suggested details of these interactions, under *Defined Interactions*, choose one of these interactions and edit the available parameters and attributes.

➔ Recommendation

We recommend you don't delete the standard attributes.

7. Open the *Patient Summary Configuration* application.
8. Specify the clinical data model configuration containing the *Genome Sequencing* and *Genetic Variant* interactions.
9. Select an existing configuration or create a new configuration.
10. Under *Lanes Definition*, add a lane.
11. Under **▶ Interactions and Attributes > Genome Sequencing ▾**, make the attribute *Sample ID* visible.
12. Under **▶ Interactions and Attributes > Genetic Variant ▾**, make the attribute *Location* visible.
13. Activate the interactions.

Results

On the *Timeline*, users can now view genetic variant information and open the *Variant Browser* from genome sequencing events. This is assuming the patient has genomic data available and this application configuration is assigned to the user.

6.4.5 Activating Patient Summary Configurations

Context

Only an active configuration for the *Patient Summary* application can be assigned to a user or an organizational unit. When you activate a configuration, the system validates it to check that it doesn't contain inconsistencies and is compatible to the clinical data model. You can't activate an invalid configuration.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Launch the *Patient Summary Configuration* application.

3. Select the data model configuration.
4. Choose the *Patient Summary* configuration.
5. Before you activate the configuration, you can validate it to see if there are inconsistencies or incompatibilities that could prevent its activation. To do this, choose *Validate*. If necessary, correct the configuration settings.
6. To activate the configuration, choose *Activate*.

The configuration is active. Your user name appears as creator of the configuration.

6.4.6 Adapting to More Recent Versions of the Data Model Configuration

Context

You base your configurations for the *Patient Summary* application on a specific version of a data model configuration. A warning in the *Patient Summary Configuration* application informs you when a more recent version of this data model configuration is available. You can select the more recent version of the data model configuration and choose whether you want to adapt the *Patient Summary* configuration accordingly.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Launch the *Patient Summary Configuration* application.
3. Select the data model configuration on which the *Patient Summary* application is based.

If a more recent version of the data model version is available, you see a warning.

4. Select the most recent version of the data model configuration.

The data model configuration version that is marked as *active* is the version that is recommended for use.

5. Choose whether you keep the current settings or discard the current settings.

If you keep the current settings of the *Patient Summary* configuration, attributes and interactions that are changed in the new version of the data model configuration are no longer visible in the *Patient Summary* application.

If you discard the current settings of the *Patient Summary* configuration, you have to configure the patient details, patient summary tabs, and timeline lanes again.

6. To save the settings, activate the configuration.

6.4.7 Duplicating Patient Summary Configurations

Context

You can create new configurations for the *Patient Summary* application by editing duplicates of existing *Patient Summary* configurations.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Launch the *Patient Summary Configuration* application.
3. Select the data model configuration.
4. For the *Patient Summary* configuration that you want to duplicate, choose the **...** (More) icon.
5. In the menu, choose *Duplicate*.
A duplicate of the *Patient Summary* configuration is created.
6. Enter a name for the new configuration.
7. Edit the configuration settings for the configuration details, patient details, patient summary tabs, and timeline lanes.
8. Activate the new configuration.

6.4.8 Importing Patient Summary Configurations in JSON Format

Context

You can import configurations for the *Patient Summary* application that were saved in JSON file format by an external editor program.

You can also view each *Patient Summary* configuration in JSON format. To do so, you choose the **...** (More) icon on the *Patient Summary* configuration and then *Preview*. You can use this preview to copy and paste the configuration into an external editor program.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Launch the *Patient Summary Configuration* application.
3. Select the data model configuration on which you want to base your configuration for the *Patient Summary* application.
4. Choose *Import Configuration*.
5. Select a version of the data model configuration.
6. Paste a configuration in JSON format into the input field and choose *Import*.

6.5 Configuring Search

Configure how the *Search* application works by specifying which search view a search entity uses and how the results list of the search entity looks and behaves. You do this in the *Search Configuration* application.

Context


Search results are divided into different categories, which appear in the drop-down list next to the search field and on the search result tabs. The search categories are based on search entities. If you, for example, want to add a search entity or translate the name of the search entity on the user interface into different languages, you need to configure the *Search* application.

Procedure

1. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

2. Open the *Search Configuration* application.
3. Choose the search entity that you want to configure.

You can either configure an existing search entity or create a new one by choosing  on the footer toolbar.

4. Select the search view that the search entity uses.

If you want to use custom search views, see [Deploying Custom Search Views \[page 166\]](#).

5. Assign an order number to the search entity.

The order number specifies the order of the search entities in the drop-down list and on the search result tabs.

6. In the *Result List Implementation* fields, define how the results list of the search entity looks and behaves by specifying the view file and the controller file that create the results list.

i Note

You need to provide complete file paths.

➔ Tip

In SAP HANA studio, you can find the standard results list implementation files in the package `sap/hc/hph/search/ui/components/globalSearch/views`.

If you want to use custom results list implementation files, see [Customizing Search Results Lists \[page 167\]](#).

7. For each system language, assign a name to the search entity.

This is the name of the search entity that appears on the search result tab and in the drop-down list. However, by assigning a name to a search entity, you don't change how it's called in the *Search Entities* menu.

8. Save and activate your search entity configurations.

The activated search entities appear in the drop-down list and on the search result tabs of the *Search* application as well as in *Patient Summary Configuration* on the *Summary Tabs* tab in the *Related Documents Search View* list.

Related Information

[Deploying Custom Search Views \[page 166\]](#)

[Customizing Search Results Lists \[page 167\]](#)

[Changing the Search Background Image \[page 168\]](#)

6.5.1 Deploying Custom Search Views

In SAP HANA studio, create a custom search view for the *Search* application that is compatible with your data model. In the *Search Configuration* application, implement this custom search view by assigning it to a search entity.

Context

If a standard search view that generates search results for the *Search* application doesn't meet your requirements (for example, because it's not compatible with your data model), you need to deploy a custom search view.

Procedure

1. In SAP HANA studio, create a custom search view.
 - a. Outside the `sap` package, create a custom package.
 - b. In your custom package, create custom search view file.
 - c. Copy the standard search view that you want to adjust from either `sap/hc/hph/cdw/db/models/DWViews.hdbdd` or `sap/hc/hph/cdw/db/models/InterfaceViews.hdbdd` to your custom view file.

Note

You need to change the basic search view and the corresponding guarded search view.

Caution

Don't change the original files because changes to these files are lost when the system is updated.

- d. In the custom search view file, adjust the namespace to the location of the file in your custom package.
 - e. Edit the search view in the custom view file.
2. Set the search flag (`@Search`) and the enterprise search flag (`@EnterpriseSearch`) of your custom search view as shown in the following code block:

Source Code

```
@Search: {searchable: true}
@EnterpriseSearch.enabled: true
```

3. Enable the technical user to access the custom search view.
 - a. Copy the technical role `HC_HPH_CDW_TECH_CONNECTION` from the `sap/hc/hph/cdw/db/roles` package to your custom package.


- b. Rename the technical role to `HC_HPH_CUSTOM_TECH_CONNECTION`.
- c. Grant the technical role to the technical user.

For more information about how to configure the technical user, see *Setting up the Technical User* in the SAP Connected Health Platform Implementation Guide.

4. In the *Search Configuration* application, implement your custom search view by assigning it to a search entity.
 - a. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

- b. Open the *Search Configuration* application.
- c. In the *Search Entities* menu, choose the search entity that uses the custom search view.

You can either use an existing search entity or create a new one by clicking  on the footer toolbar.

- d. From the *Search View* drop-down list, select your custom search view.
5. Using test data with expected results, test your implemented custom search view.

➔ Tip

If the search results don't match what you expected, ensure that your authorization logic supports the custom search view.

If your custom search view doesn't support your analytic privileges, instead of those analytic privileges, you must manually implement your custom authorization logic directly in the search view.

6.5.2 Customizing Search Results Lists

Customize results lists by implementing custom results list implementation files. Create these files in SAP HANA studio and assign them to a search entity in the *Search Configuration* application.

Context

When you create a new search entity, you need to assign a controller file and a view file (results list implementation files) to it. The view file specifies the layout of the search results list while the controller file specifies the behavior of the search results list. If the existing results list implementation files don't meet your requirements, you need to create custom results list implementation files.

Procedure

1. In SAP HANA studio, in the `Content` folder, create custom results list implementation files.

- a. Outside the `sap` package, create a custom package.
- b. Create custom results list implementation files by copying and renaming existing results list implementation files.

i Note

You can find results list implementation files in the `sap/hc/hph/search/ui/components/globalSearch/views` package.

⚠ Caution

Don't change the original files because changes to these files are lost when the system is updated.

- c. Move your custom results list implementation files from the `sap/hc/hph/search/ui/components/globalSearch/views` package to your custom package.
 - d. In the custom results list implementation files, adjust the namespaces to the location of the files in your custom package.
 - e. Edit the custom results list implementation files.
2. Implement your custom results list implementation files by assigning them to a search entity.
 - a. Go to the SAP Fiori launchpad for SAP Connected Health.

The SAP Fiori launchpad is available at the following URL: `http://<WebServerHost>:80<SAPHANAinstance>/sap/hana/uis/clients/ushell-app/shells/fiori/FioriLaunchpad.html?siteId=sap.hc.hph.flp|hph#`

- b. Open the *Search Configuration* application.
- c. In the *Search Entities* menu, choose the search entity to which you want to apply the custom results list implementation files.
- d. Enter the names of the results list implementation files into the respective fields.

i Note

You need to provide complete file paths.

6.5.3 Changing the Search Background Image

You can change the default background image used in the *Search* application by creating a CSS file and ensuring the system can load it.

The CSS file must contain the following entry:

```
.sapHPH_MVBox.sapMVBox {
background-image: url(<File Path>/<Image File Name>);
}
```

6.6 Customizing Data Consumption for Applications

All applications consume data from database views, which comprise the logical data model. If the default views don't meet your business requirements, you can create and implement custom views to change how applications consume data.

i Note

Whenever implementing custom views, please contact your SAP technical consultant first.

The default views are designed to serve most use cases and are optimized for system performance. However, there may be specific use cases where they are unsuitable. For example, you may have a security concept that uses different logic than the standard security logic within the views. You might also want to modify how search works or incorporate custom entities and attributes into the system.

In general, to create and implement custom views, you must create the view in your own namespace using SAP HANA studio, bind the system to that view using the [Advanced Settings for Clinical Data Model Configurations](#) application, and then create clinical data model configurations based on the views. In the case of custom entities and attributes, you may have to create custom flowgraphs for data integration as well.

If your scenario is very specific and only requires custom attributes for individual use cases, you can leave the default views in the advanced settings and incorporate your custom views in specific clinical data model configurations. To do so, in the [Clinical Data Model Configuration](#) application, select the attribute of the interaction you want to customize and, under [Attribute Details](#), choose [Add New Custom Table](#).

Related Information

[Configuring Search \[page 164\]](#)

[Configuring Interaction Attributes \[page 136\]](#)

[Advanced Settings for Clinical Data Model Configurations \[page 142\]](#)

7 Troubleshooting

If any problems or errors occur, you must ensure that the system is configured correctly according to the information presented in this guide and the related implementation guide.

➔ Recommendation

To realistically re-create issues experienced by users, we recommend creating test users you can use instead of your system administrator user.

If problems persist, you can contact SAP Support by opening a support ticket under the component `IS-PMED-HPH`.

Related Information

[SAP HANA Troubleshooting and Performance Analysis Guide](#)

[SAP HANA Administration Guide](#)

Important Disclaimers and Legal Information

Coding Samples

Any software coding and/or code lines / strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended to better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, unless damages were caused by SAP intentionally or by SAP's gross negligence.

Accessibility

The information contained in the SAP documentation represents SAP's current view of accessibility criteria as of the date of publication; it is in no way intended to be a binding guideline on how to ensure accessibility of software products. SAP in particular disclaims any liability in relation to this document. This disclaimer, however, does not apply in cases of willful misconduct or gross negligence of SAP. Furthermore, this document does not result in any direct or indirect contractual obligations of SAP.

Gender-Neutral Language

As far as possible, SAP documentation is gender neutral. Depending on the context, the reader is addressed directly with "you", or a gender-neutral noun (such as "sales person" or "working days") is used. If when referring to members of both sexes, however, the third-person singular cannot be avoided or a gender-neutral noun does not exist, SAP reserves the right to use the masculine form of the noun and pronoun. This is to ensure that the documentation remains comprehensible.

Internet Hyperlinks

The SAP documentation may contain hyperlinks to the Internet. These hyperlinks are intended to serve as a hint about where to find related information. SAP does not warrant the availability and correctness of this related information or the ability of this information to serve a particular purpose. SAP shall not be liable for any damages caused by the use of related information unless damages have been caused by SAP's gross negligence or willful misconduct. All links are categorized for transparency (see: <http://help.sap.com/disclaimer>).

**[go.sap.com/registration/
contact.html](http://go.sap.com/registration/contact.html)**

© 2017 SAP SE or an SAP affiliate company. All rights reserved.
No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company. The information contained herein may be changed without prior notice.
Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.
These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.
SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies.
Please see <http://www.sap.com/corporate-en/legal/copyright/index.epx> for additional trademark information and notices.