



**SAP SuccessFactors** 

Implementation Guide | PUBLIC  
Document Version: 2H 2020 – 2020-10-29

## Employee Central Master

# Content

<b>1</b>	<b>Overview Employee Central . . . . .</b>	<b>10</b>
1.1	Employee Central Overview . . . . .	10
1.2	Recommended Implementation Sequence . . . . .	10
	Recommended Implementation Sequence with Other Modules . . . . .	14
1.3	Where to Find Information about Other Employee Central Topics . . . . .	14
1.4	Components for SAP SuccessFactors Employee Central . . . . .	16
1.5	Assignment ID . . . . .	17
	Differentiating Between Person ID, UUID, User ID, and Assignment ID . . . . .	19
1.6	Data Protection and Privacy for Employee Central . . . . .	21
<b>2</b>	<b>Configuring the System . . . . .</b>	<b>25</b>
2.1	Provisioning . . . . .	25
2.2	Initial Configuration Tasks . . . . .	25
	Setting Up the Basics . . . . .	26
	Setting Up Attachment Options . . . . .	28
	Creating a Super Admin User in Provisioning . . . . .	28
	Configuring Company System Settings . . . . .	30
	Configuring Employee Central Settings . . . . .	31
	Keeping the User Directory and Org Chart Up to Date . . . . .	32
	Enabling the Employee Central SOAP APIs . . . . .	33
	Enabling the Legacy Home Page . . . . .	33
	Enabling the To-Do List . . . . .	34
	Configuring the Internal Job History Block . . . . .	35
	Setting Up the Currency Exchange Rate . . . . .	37
	Generating User IDs . . . . .	39
	Creating the Employment Settings Configuration Object . . . . .	41
	Creating the Hire/Rehire Configuration Object . . . . .	43
<b>3</b>	<b>Managing Role-Based Permissions . . . . .</b>	<b>45</b>
3.1	Permissions for Employee Central . . . . .	45
	User Permissions for Employee Central . . . . .	45
	Administrator Permissions for Employee Central . . . . .	58
	Adding Value Help Permissions for Everyone . . . . .	62
<b>4</b>	<b>Ad Hoc Report Types . . . . .</b>	<b>64</b>
4.1	Permissions for Ad-Hoc Reports . . . . .	68
4.2	Ad Hoc Report Query Trimming . . . . .	69

<b>5</b>	<b>Data Models</b>	<b>70</b>
<b>6</b>	<b>Working with Foundation Objects</b>	<b>71</b>
6.1	Introduction to Foundation Objects.	71
	Characteristics of Foundation Objects.	73
	MDF vs. Legacy Foundation Objects.	73
	Deleting a Foundation Object.	75
6.2	Associations.	76
	Using Associations to Structure Your Business.	76
	Examples of Foundation Object Associations.	78
	Configuring Associations.	79
6.3	Effective Dating.	98
	Changing a Legacy Foundation Object.	98
6.4	Foundation Objects for Structuring Your Business.	100
6.5	Foundation Objects for Handling Job-Related Areas.	102
6.6	Foundation Objects for Handling Pay-Related Areas.	104
6.7	Other Foundation Objects.	106
<b>7</b>	<b>Working with MDF Foundation Objects</b>	<b>108</b>
7.1	MDF Foundation Objects.	108
7.2	Adding Search Criteria.	109
7.3	Working with Associations, Field Criteria and Value Help.	110
	Adding Legal Entity to the Cost Center Object.	115
7.4	Behavior Changes to Importing Foundation Objects.	116
7.5	Behavior Changes to Workflows.	117
7.6	Optional: Editing the Country/Region and Currency Objects.	117
7.7	Adding a New Country/Region and Related Fields to LegalEntity.	119
	Step 1: Creating a New MDF Object for the Country/Region-Specific Fields.	119
	Step 2: Assigning the New Object to Legal Entity.	120
7.8	Adding a New Country/Region and Related Fields to Job Classification.	121
	Step 1: Creating a New MDF Object for the Country/Region-Specific Fields.	121
	Step 2: Assigning the New Object to JobClassificationCountry.	122
7.9	Configuring Standard Fields.	122
<b>8</b>	<b>Working with Generic Objects</b>	<b>124</b>
8.1	Generic Objects.	124
	Standard Generic Objects.	124
	Characteristics of Generic Objects.	125
8.2	Creating Customer-Specific Foundation Objects.	126
8.3	Example: Configuring Workflows for Legacy Foundation Objects.	127
8.4	Example: Automatic Generation of Legacy Foundation Object Codes.	129
<b>9</b>	<b>Working with Picklists</b>	<b>130</b>

9.1	Introduction to Picklists. . . . .	130
9.2	Working with ECV2 (Legacy) Picklists. . . . .	131
	Creating ECV2 Picklists. . . . .	131
	Modifying ECV2 Picklists. . . . .	132
	Deleting ECV2 Picklists. . . . .	133
	About the Picklist Import File. . . . .	134
9.3	Working with MDF Picklists. . . . .	136
9.4	Working with Cascading Picklists. . . . .	137
9.5	Keeping ECV2 and MDF Picklists in Sync. . . . .	142
	Step 1: Preparing the ECV2 Picklist. . . . .	142
	Step 2: Creating the MDF Picklist. . . . .	143
	Step 3: Mapping the ECV2 Picklist to the MDF Picklist. . . . .	144
<b>10</b>	<b>Events and Event Reasons. . . . .</b>	<b>146</b>
10.1	Events. . . . .	146
10.2	Event Reasons. . . . .	147
10.3	Creating Event Reasons for Employee Central. . . . .	149
10.4	Setting Up Country/Region-Specific Picklists for Event Reasons. . . . .	151
<b>11</b>	<b>Creating Rules. . . . .</b>	<b>154</b>
11.1	Event Reason Derivation Business Rules. . . . .	154
11.2	Business Rules in Employee Central. . . . .	155
	Optimizing Business Rules. . . . .	157
	Standard and Model Base Objects for Employee Central. . . . .	158
	Mapping of Employee Central Data Types and Business Rule Field Types. . . . .	161
	Overview: Rule Events in Employee Central. . . . .	162
	Common Problems for Business Rules in Employee Central. . . . .	164
	Cross-Block Rules. . . . .	170
	Rule Scenarios for Employee Central Core. . . . .	174
	Adding Contexts for Business Rules. . . . .	175
	Example Employee Central Business Rules. . . . .	178
<b>12</b>	<b>Human Resource Information System (HRIS) Synchronization. . . . .</b>	<b>190</b>
12.1	What Triggers HRIS Sync? . . . . .	190
12.2	HRIS Sync Job. . . . .	191
	Configuring an HRIS Sync Job. . . . .	194
	Sync Order for an HRIS Sync Job. . . . .	196
12.3	HRIS Element and Fields Sync Logic. . . . .	197
12.4	Hard-Coded Sync Mapping. . . . .	199
12.5	HRIS Sync Mapping ( <code>hris-sync-mappings</code> ) and Configuration Validation Rules. . . . .	203
12.6	Picklist Configuration for Employee Status and Job Relationship Type. . . . .	205
12.7	Special Handling for Syncing Fields . . . . .	206

12.8	Syncing the Termination Date Between Employee Central and Standard User Fields. . . . .	207
<b>13</b>	<b>Block-Specific Information. . . . .</b>	<b>209</b>
13.1	Additional Information for Addresses. . . . .	209
13.2	Additional Information for Biographical Information. . . . .	211
13.3	Additional Information for Employment Information. . . . .	211
13.4	Additional Information for Job Information . . . . .	212
13.5	Additional Information for Job Relationships. . . . .	213
13.6	Additional Information for National ID. . . . .	215
13.7	Additional Information for Personal Information. . . . .	215
13.8	Additional Information for Primary Emergency Contact. . . . .	217
13.9	Additional Information for Work Permits. . . . .	218
<b>14</b>	<b>Forward Propagation in Employee Central. . . . .</b>	<b>220</b>
<b>15</b>	<b>Deep Links in Employee Central. . . . .</b>	<b>224</b>
<b>16</b>	<b>Mobile Employee Central. . . . .</b>	<b>226</b>
16.1	Setting Up Mobile . . . . .	226
16.2	Mobile To-Do Items in Employee Central. . . . .	227
<b>17</b>	<b>Retired Functionality. . . . .</b>	<b>228</b>
17.1	Direct Deposit or Old Payment Information block. . . . .	228
17.2	HRIS Propagation Configuration Template. . . . .	228
17.3	Event Reason Derivation from XML. . . . .	235

# What's New in Employee Central

Learn about changes to this guide in recent releases.

## 2H 2020

What's New	Description	More Information
Data Models	<p>The data models section was removed.</p> <p>There is new guide that covers all data model information in SAP SuccessFactors.</p> <p>For more information about data models, refer to the <a href="#">SAP SuccessFactors Data Model Reference Guide</a>.</p>	<a href="#">Data Models [page 70]</a>
Embedded Analytics	This topic was removed since it was only valid for Employee Central v12.	
Assignment ID	Topics about assignment ID and other IDs used in the SAP SuccessFactors HXM Suite were added.	<a href="#">Assignment ID</a> <a href="#">Differentiating Between Person ID, UUID, User ID, and Assignment ID</a>
Rules	<p>Topics removed from the Implementing Business Rules guide and added to the Master guide.</p> <p>A new topic about Cross-Block rules was added.</p> <p>A new topic to list the Employee Central Core rule scenarios was added.</p>	<a href="#">Standard and Model Base Objects for Employee Central [page 158]</a> <a href="#">Properties of Model Base Objects [page 159]</a> <a href="#">Overview: Rule Events in Employee Central [page 162]</a> <a href="#">Event Types for HRIS Elements and HRIS Fields [page 162]</a> <a href="#">Cross-Block Rules [page 170]</a> <a href="#">Rule Scenarios for Employee Central Core [page 174]</a>
Internal Job History Block	New topic added for configuring the Internal Job History block in the People Profile.	<a href="#">Configuring the Internal Job History Block [page 35]</a>
Hire/Rehire Configuration	New topic added about configuration settings for the duplicate check in hire/rehire scenarios.	<a href="#">Creating the Hire/Rehire Configuration Object [page 43]</a>

What's New	Description	More Information
Event Reason Derivation XML Rules	<p>The XML-based event reason derivation is now deprecated and replaced by business rules.</p> <p>Topic added:</p> <ul style="list-style-type: none"> <li>• Event Reason Derivation from XML</li> </ul> <p>Topic updated:</p> <ul style="list-style-type: none"> <li>• Event Reason Derivation Business Rules</li> </ul> <p>Topics deleted:</p> <ul style="list-style-type: none"> <li>• Setting Up XML Event-Reason Derivation Rules</li> <li>• XML Examples for Event-Reason Derivation Rules</li> </ul>	<a href="#">Event Reason Derivation from XML [page 235]</a> <a href="#">Event Reason Derivation Business Rules [page 154]</a>

## 1H 2020

What's New	Description	More Information
Terminology	The term "portlet" has been replaced by the term "block" in Employee Central.	
Event Reason Derivation XML Rules	<p>Topic moved to the Workflows guide:</p> <ul style="list-style-type: none"> <li>• Limitations for Workflow and Event-Reason Derivation OnSave Rules</li> </ul>	<a href="#">Limitations for Workflow and Event-Reason Derivation OnSave Rules</a>
Employment Settings Configuration	You can now choose business rules for the Enforce New Employment setting.	<a href="#">Creating the Employment Settings Configuration Object [page 41]</a>
Set OnView rules for the fields in non-effective dated blocks	If necessary, you can use OnView rules to control who can see which fields in the blocks that refer to non-effective dated entities, as you cannot use role-based permissions to set field-level <i>View</i> permissions for these blocks.	<a href="#">Employee Data Permissions - HR Information [page 46]</a> <a href="#">Example Employee Central Business Rules [page 178]</a>
Effective-Dated Entities	You can show the user name rather than the user ID in the History pages of effective-dated HRIS elements.	<a href="#">Employee Central Effective Dated Entities Permissions [page 55]</a>

What's New	Description	More Information
Country/Region Specific Picklists for Pay Components and Event Reasons	<p>This topic has been split up so there is one topic for event reasons and one topic for pay components.</p> <p>The event reasons topic is moved to the Events and Event Reasons section.</p> <p>The pay components topic is moved to the Employee Compensation Data guide.</p>	<a href="#">Setting Up Country/Region-Specific Picklists for Event Reasons [page 151]</a> <a href="#">Setting Up Country/Region-Specific Picklists for Pay Components</a>
HR Information Permission	You can use the permission, National ID (Restricted to only country/region of legal entity), to restrict administrators so that they can only access the national ID information of an employee relevant to the country or region of the legal entity where the employee is currently employed.	<a href="#">Employee Data Permissions - HR Information [page 46]</a>
Mobile Employee Central	This section has been renamed and updated. The user topic was removed.	<a href="#">Setting Up Mobile [page 226]</a>
Changing or Translating Block Help Content	Topic moved to the People Profile guide.	<a href="#">Changing or Translating Block Help Content</a>
MDF Foundation Objects	New topic for associations	<a href="#">Adding Legal Entity to the Cost Center Object [page 115]</a>
Check Tool	<p>Section removed.</p> <p>For more information, refer to the Check Tool guide.</p>	<a href="#">Using the Check Tool</a>
Introduction to Using MDF-Based Business Rules for Event-Reason and Workflow Derivation	This section was removed in the continuing improvement of the guide.	
Activating Embedded Analytics	This topic was removed.	

## Q4 2019

What's New	Description	More Information
User ID Generation	<p>Information added about generating user IDs with business rules as well as how user IDs are generated for global assignments and concurrent employment.</p> <p>A new topic was added about creating number ranges for user ID sequences.</p>	<a href="#">Generating User IDs [page 39]</a> <a href="#">Optional: Creating Number Ranges for User ID Sequences [page 41]</a>

What's New	Description	More Information
Contexts for Business Rules	Updated with the Report No-Show UI.	<a href="#">Adding Contexts for Business Rules [page 175]</a>
HRIS Sync	Rearranged the child topics under this main topic and updated some content.	<a href="#">Human Resource Information System (HRIS) Synchronization [page 190]</a>
Employment Settings	Topic updated with information about assignment ID.	<a href="#">Creating the Employment Settings Configuration Object [page 41]</a>
Events and Event Reasons	New topics to better explain event and event reasons.	<a href="#">Events [page 146]</a> <a href="#">Event Reasons [page 147]</a>
Primary Emergency Contact	Updated information about default mandatory field validation and added information about avoiding duplicate emergency contacts.	<a href="#">Additional Information for Primary Emergency Contact [page 217]</a>

# 1 Overview Employee Central

## 1.1 Employee Central Overview

Employee Central (EC) provides comprehensive, integrated, searchable people and organizational information.

Information is natively stored in our product so other modules can access the information. It captures information about a company's organization, pay, job structure, and employees. It drives a lot of the information that you see in Employee Profile and talent information.

Employee Central data is smart because it allows you to capture history, create associations, use effective-dated objects, define automated workflows, and automatically configure options for on-screen selections.

## 1.2 Recommended Implementation Sequence

This is the recommended implementation sequence for Partners and Consultants. We strongly recommend that you follow this sequence for the first few implementations and discuss any variations with your Team Lead.

To help you with your implementation, we recommend following sequence of steps.

### → Remember

As a customer, you don't have access to Provisioning. To complete tasks in Provisioning, contact your implementation partner. If you're no longer working with an implementation partner, contact SAP Cloud Support.

For information on this step...	See...
Step 1: Setting Up a New Account in Provisioning	<a href="#">Initial Configuration Tasks [page 25]</a>  This section describes the steps to get started with your implementation, including the different options you need to select in Provisioning to enable Employee Central.
Step 2: Creating the Super Admin	<a href="#">Initial Configuration Tasks [page 25]</a>  The section <i>How do you create the Super Admin</i> describes the different steps required to create the Super Admin.
Step 3: Defining the Corporate Data Model	Refer to the <a href="#">SAP SuccessFactors Data Model Reference Guide</a> for information about how to set up the Corporate Data Model.

For information on this step...	See...
Step 4: Defining the Country/Region-Specific Corporate Data Model	Refer to the <a href="#">SAP SuccessFactors Data Model Reference Guide</a> for information about how to set up the Country/Region-Specific Corporate Data Models.
Step 5: Setting up MDF Foundation Objects	<p><a href="#">MDF Foundation Objects [page 108]</a></p> <p><a href="#">Generic Objects [page 124]</a></p> <p>These sections describe how you can configure your MDF FOs and custom GOs. These sections also describe how you can set up the country/region-specific configurations for MDF Foundation Objects.</p>
Step 6: Configuring the Succession Data Model	<p>Refer to the <a href="#">SAP SuccessFactors Data Model Reference Guide</a> for information about how to set up the Succession Data Model.</p> <p>If you have defined associations in steps 3 and 4, make sure that you update the field criteria in the Succession Data Model.</p>
Step 7: Configuring the Country/Region-Specific Succession Data Models	Refer to the <a href="#">SAP SuccessFactors Data Model Reference Guide</a> for information about how to set up the Country/Region-Specific Succession Data Models.
Step 8: Importing the Picklist	<p><a href="#">Introduction to Picklists [page 130]</a></p> <p>This section describes how to import the different values that a customer sees when they click a dropdown menu. It also provides information on MDF picklists and cascading picklists.</p>
Step 9: Creating Foundation Objects	<p><a href="#">Introduction to Foundation Objects [page 71]</a></p> <p>This section describes what foundation objects are including how you can define them.</p>
Optional: Configuring Position Management	<p><a href="#">Employee Central Position Management</a></p> <p>This guide explains how to set up position management in Employee Central. If you plan on using position management, then it should be set up before creating business rules since many business rules and event/workflow derivation are based off of Position Management configuration (sync, position change, and on).</p>
Step 10: Configuring Business Rules	<p><a href="#">Business Rules in Employee Central [page 155]</a></p> <p>This section describes how to set up the different rules for your system.</p>

<b>For information on this step...</b>	<b>See...</b>
Step 11: Creating Event-Reason Derivation Rules	<a href="#">Event Reason Derivation Business Rules [page 154]</a>  This section describes how to set up the different rules. Depending on the attributes that change, the system automatically determines the rule to apply.
Step 12: Creating Workflow Derivation Rules	<a href="#">Implementing and Configuring Workflows in Employee Central</a> guide in the SAP Help Portal.  This guide describes what workflows are, when to use them, and how to set them up.
Step 13: Setting Role-Based Permissions	<a href="#">Permissions for Employee Central [page 45]</a>  This section describes which permissions are specific to Employee Central and how you manage them.  Please refer to the <a href="#">Role-Based Permissions</a> guide on the SAP Help Portal for details on how to set up permission roles, permission groups, and permission assignments.
Step 14: Importing Employee Data	<a href="#">Employee Central Imports</a>  This guide describes everything you need to know about importing employee data.
Step 15: HRIS Sync	<a href="#">Human Resource Information System (HRIS) Synchronization [page 190]</a>  This section describes how you can sync data from Employee Central to other modules.
Step 16: Setting up Leave of Absence	You need to set up Time Off to use leave of absence. Note that you need to decide first whether you want to use leave of absence as standalone or together with other Time Off features. Depending on this decision, the setup varies.  You can find more information about how to set up leave of absence in the <a href="#">Implementing Employee Central Time Off</a> guide on the SAP Help Portal.
Optional: Setting Up Payment Information	<a href="#">Employee Central Payment Information</a>  This guide explains how to set up the MDF-based payment information for users in the system.
Optional: Setting Up People Profile	<a href="#">People Profile</a>  This guide explains how to set up the People Profile in your instance.

For information on this step...	See...
Optional: Setting Up Global Assignments	<p><a href="#">Implementing and Configuring Global Assignments in Employee Central</a></p> <p>This guide explains how to set up global assignments in Employee Central.</p>
Optional: Setting Up Concurrent Employment	<p><a href="#">Implementing and Configuring Concurrent Employment in Employee Central</a></p> <p>This guide explains how to set up concurrent employment in Employee Central.</p>
Optional: Setting Up Contingent Workforce	<p><a href="#">Contingent Workforce Management</a></p> <p>This guide explains how to set up contingent workers in Employee Central.</p>
Optional: Setting Up Document Generation	<p><a href="#">Employee Central Document Generation</a></p> <p>This guide explains how to set up document generation in Employee Central.</p>
Optional: Setting Up Employee Central Advanced Reporting	<p><a href="https://help.sap.com/viewer/7acb210c5f7f47aea70754cb158d2951/latest/en-US">https://help.sap.com/viewer/7acb210c5f7f47aea70754cb158d2951/latest/en-US</a></p> <p>This guide explains how to set up standard reports for the different areas in Employee Central.</p>
Optional: Setting Up Mobile	<p><a href="#">SAP SuccessFactors Mobile Deployment Guide</a>.</p> <p>This guide describes how to set up &amp; use Employee Central on your mobile devices.</p>
Optional: Setting Up Employee Central Payroll	<p><a href="#">Employee Central Payroll using Point-to-Point Integration</a></p> <p>This guide describes how to set up Employee Central Payroll.</p>
<div style="background-color: #e0e0e0; padding: 10px;"> <p><b>i Note</b></p> <p>If you want to have the same IDs in the Employee Central and Employee Central Payroll systems, we recommend that you use numeric employee IDs in Employee Central, because the PERNR is numeric in Employee Central Payroll. Therefore, an alphanumeric <i>ID</i> cannot be used across all processes in the Employee Central Payroll system.</p> </div>	

## 1.2.1 Recommended Implementation Sequence with Other Modules

This is information about the recommended sequence for partners and consultants to integrate Employee Central with other SAP SuccessFactors modules.

It is recommended to either start with Employee Central or end with Employee Central.

► [Employee Central](#) ► [Recruiting](#) ► [Onboarding](#) ▶

► [Recruiting](#) ► [Onboarding](#) ► [Employee Central](#) ▶

Before implementation, consider the following topics and how they impact other modules:

- Employee ID generation
- Foundation Objects
- Company Structure
- Job Structure
- Global Assignment
- Concurrent Employment

## 1.3 Where to Find Information about Other Employee Central Topics

This document covers the core of Employee Central, but there are many parts to Employee Central. Hopefully, this will help you find all the information that you need.

All guides for Employee Central can be found on the SAP Help Portal at [https://help.sap.com/hr\\_ec](https://help.sap.com/hr_ec)

Topic	Link
Advances	<a href="#">Employee Central Advances</a>
Alternative Cost Distribution	<a href="#">Implementing and Configuring Alternative Cost Distribution in Employee Central</a>
Apprentice Management	<a href="#">Employee Central Apprentice Management</a>
Business Configuration UI	<a href="#">Setting Up and Using Business Configuration UI (BCUI)</a>
Business Rules	<a href="#">Implementing Business Rules in SAP SuccessFactors</a>
Check Tool	<a href="#">Using the Check Tool</a>
Company Structure Overview	<a href="#">Employee Central Company Structure Overview</a>

Topic	Link
Compensation	<a href="#">Implementing and Configuring Employee Compensation Data in Employee Central</a>
Concurrent Employment	<a href="#">Implementing and Configuring Concurrent Employment in Employee Central</a>
Contingent Workforce Management	<a href="#">Contingent Workforce Management</a>
Country/Region Specifics	<a href="#">Employee Central Country/Region Specifics</a>
Data Object Tables	<a href="#">Data Object Tables in Employee Central</a>
Deductions	<a href="#">Implementing and Configuring Deductions in Employee Central</a>
Dependents	<a href="#">Implementing and Configuring Dependents in Employee Central</a>
Document Generation	<a href="#">Document Generation in Employee Central</a>
Imports	<a href="#">Employee Central Imports</a>
Integration with SAP ERP	<a href="#">SAP SuccessFactors Employee Central Integration to SAP Business Suite</a>
Global Assignments	<a href="#">Implementing and Configuring Global Assignments in Employee Central</a>
Global Benefits	<a href="#">Implementing and Configuring Global Benefits in Employee Central</a>
Managing Employment	<a href="#">Managing Employment in Employee Central</a>
Metadata Framework (MDF)	<a href="#">Implementing the Metadata Framework (MDF)</a>
Mobile	<a href="#">SAP SuccessFactors Mobile Deployment Guide</a> <a href="#">SAP SuccessFactors Mobile Features</a>
Payment Information	<a href="#">Employee Central Payment Information</a>
Payroll	<a href="#">SAP SuccessFactors Employee Central Payroll</a>
Pension Payouts	<a href="#">Employee Central Pension Payouts</a>
Position Management	<a href="#">Employee Central Position Management</a>
Service Center	<a href="#">Employee Central Service Center</a>
Time Off	<a href="#">Implementing Employee Central Time Off</a>

Topic	Link
Time Sheet	<a href="#">Implementing Employee Central Payroll Time Sheet</a>
Workflows	<a href="#">Implementing and Configuring Workflows in Employee Central</a>

## 1.4 Components for SAP SuccessFactors Employee Central

Here is a list of the most used components for Employee Central

Component	Description
LOD-SF-EC	Employee Central
LOD-SF-EC-TIM	Time Off
LOD-SF-EC-POS	Position Management
LOD-SF-EC-WFL	Workflows
LOD-SF-EC-GBF	Global Benefits
LOD-SF-EC-RUL	Business Rules and Event Derivation
LOD-SF-EC-HIR	Hire, Rehire
LOD-SF-EC-JOB	Job Information and Propagation
LOD-SF-EC-PAY	Payment Information
LOD-SF-EC-TMS	Time Sheet
LOD-SF-EC-MDF	Metadata Framework
LOD-SF-EC-BCI (BCUI)	Business Configuration UI
LOD-SF-EC-LOC	Localization
LOD-SF-EC-CMP	Compensation Information
LOD-SF-EC-PER	Person Information
LOD-SF-EC-INT	Integration - EC to RCM, ONB, CVP
LOD-SF-EC-EDP	Employee Data Imports
LOD-SF-EC-DOC	Document Generation

Component	Description
LOD-SF-EC-RBP	Role-Based Permissions
LOD-SF-EC-CWF	Contingent Workforce Management
LOD-SF-EC-CGA	Concurrent Employment &Global Assignment
LOD-SF-EC-EMP	Employment Information
LOD-SF-EC-HRS	HRIS Sync
LOD-SF-EC-PP3	People Profile
LOD-SF-EC-SRV	Employee Central Service Center
LOD-SF-EC-FOO	Foundation Objects
LOD-SF-EC-AAD	Advances and Deductions
LOD-SF-EC-MOB	Mobile
LOD-SF-EC-ALR	Alerts and Notifications
LOD-SF-EC-DPD	Dependents Management
LOD-SF-EC-ADM	Admin Tools
LOD-SF-EC-APM	Employee Central Apprentice Management
LOD-SF-EC-CSO	Company Structure Overview

## 1.5 Assignment ID

Assignment ID is an identifier assigned to the work relationship between a person and the company. The relationship could be an employment relationship, contingent relationship, pensioner relationship, intern, global assignment, or others. A person can have one or many work relationships with a company at the same time, for example, concurrent employments or home and host assignment in a global assignment.

### i Note

Currently, assignment ID is **not** supported in some SAP SuccessFactors areas, for example, Learning, Compensation, Onboarding 1.0, and data protection and privacy features. This might cause display inconsistencies across the HXM Suite. See [Important Notes about Assignment ID](#) to find the specific areas impacted by assignment ID as well as the areas where assignment ID is **not** supported. This document will be regularly updated to reflect the latest development of assignment ID.

### Caution

Before you change assignment IDs, we recommend that you evaluate the risks associated with the inconsistencies. If assignment ID is not supported in the SAP SuccessFactors areas you've enabled, please don't make any changes to assignment ID at this time.

Assignment ID (assignment\_id\_external) is unique, case-sensitive, visible, and can be given to an employee, a contingent worker, or an intern. Assignment ID is used to identify users across the HXM Suite, in import and export tools, in the user interface, in APIs, and in reports. User ID (users\_sys\_id) still exists, but we recommend that you use it for system integration.

The system automatically generates assignment IDs for users created prior to the Q3 2019 release, and their default values are the same as the current user IDs. However, in the Employee Central-enabled instances, if you have used a business rule to generate assignment IDs, the system then creates assignment IDs based on the rule and the assignment IDs might be different from the user IDs. When you create new users using the user management tools such as [Employee Import](#), [Manage Users](#), or OData APIs, assignment IDs for these users are also added to the system.

Assignment ID can be changed **ONLY** through the [convertAssignmentIdExternal](#) function import.

## Why Assignment ID?

Previously, when you wanted to change user IDs in some cases, such as employee relocation or going live on Employee Central or another HRIS system, a support ticket was needed. The user ID conversion process was costly and time-consuming. In addition to this, user ID conversion wasn't supported in Employee Central, Metadata Framework, or SAP HANA database.

Now, you can use assignment ID to identify users and change it if needed.

## Assignment ID in Employee Central Integration

Assignment ID is a unique identifier in Employee Central and assigned to the Employee Central object employment. It is a multiple purpose field. Currently assignment ID supports two main scenarios. One is the platform use case of managing users with the Manage Users, Employee Import, Import Extended User Information admin tools, and OData APIs. The other is the integration use case of the Employee Central to SAP ERP system or SAP S/4HANA (SAP ERP/S/4). In the Employee Central integration use case, the assignment ID is equal with the SAP ERP/S/4 PERNR (personnel number). Employee Central is responsible for ensuring the assignment ID matches the SAP ERP/S/4 PERNR format and determines an assignment ID by using rules during all processes where a new employment is created. As a result, the assignment ID (8 digit max) is generated and replicated to the integrated SAP ERP/S/4 system.

See [Using Assignment ID in Employee Central Integration with SAP ERP HCM](#) for more details.

### Note

You must decide for one scenario and are not allowed to switch between the two scenarios.

## 1.5.1 Differentiating Between Person ID, UUID, User ID, and Assignment ID

Read the following table to find the differences and relationships between person ID, UUID, user ID, and assignment ID.

Field	Description	Can this ID be changed?	Relationship between these IDs
Person ID (person-id-external)	A unique identifier of a person in Employee Central. Person ID identifies a natural person. An employee generally has only one person ID throughout their time at the company, since this ID is associated to each person.	Yes	UUID and person ID are in a one-to-one relationship. User ID and assignment ID are in a one-to-one relationship.
UUID (per-person-uuid)	This identifier is generated when person data is created in the system. UUID is introduced for integrating person data in Employee Central with other modules. UUID is stored at a database level only and is not visible on the UI.	No	One person ID is associated to one or more user IDs and assignment IDs. One UUID is associated to one or more user IDs and assignment IDs.
User ID (users-sys-id)	A unique identifier of user entity. A person might have one or more user IDs, for example, in the case of global assignments or concurrent employments. If a customer maintains only one employment per person in SAP SuccessFactors, the user ID can serve as the person's unique identifier in the company.  We recommend that you adopt user ID for system integration.	No	

Field	Description	Can this ID be changed?	Relationship between these IDs
Assignment ID (assignment-id-external)	<p>Assignment ID is actually the "mutable user ID". It is visible to customers and can be used to identify users. A person might have one or more assignment IDs, for example, in the case of global assignments or concurrent employments. If a customer maintains only one employment per person in SAP SuccessFactors, assignment ID can serve as the person's unique identifier in the company.</p> <p>In Employee Central, the assignment ID field can be used to store a unique identifier. For example, in the Employee Central integration scenarios, customer store the SAP ERP PERNR (personnel number) in this field.</p>	Yes	

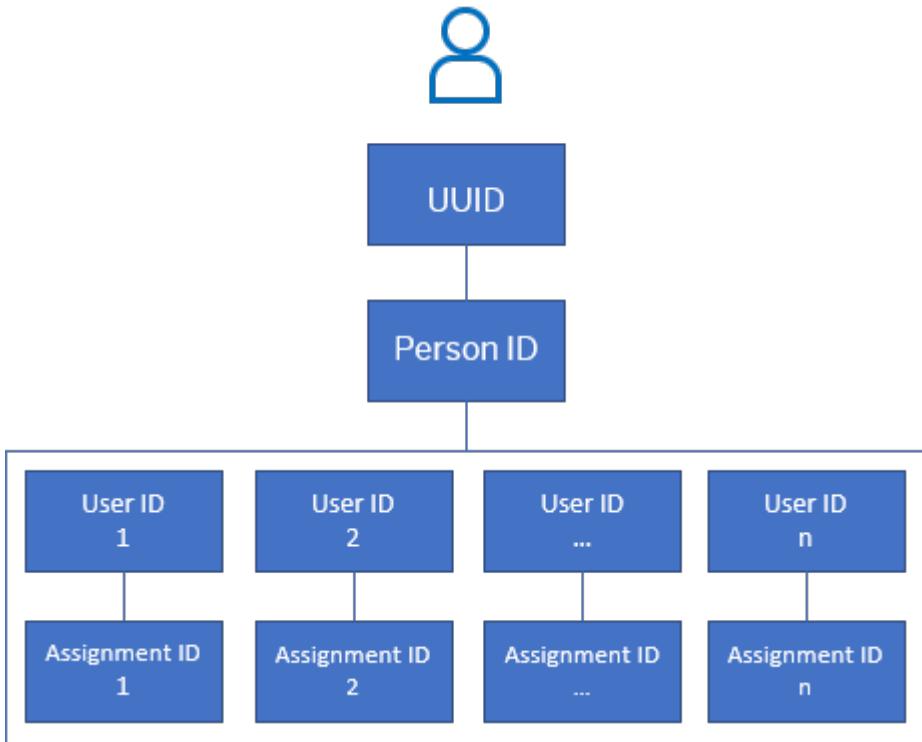


Figure 1: Relationship Between Person ID, UUID, User ID, and Assignment ID

## 1.6 Data Protection and Privacy for Employee Central

Here are some data protection and privacy features specific to Employee Central rather than the entire HXM Suite.

For information about data protection and privacy in your SAP SuccessFactors system, refer to [Setting Up and Using Data Protection and Privacy](#) on the SAP Help Portal.

### Purge

### Data Blocking

For HRIS workflows in Employee Central as well as MDF workflows, data blocking is only available for completed workflows. This means, for workflows that have the status approved, rejected, or canceled. The completed workflows can only be viewed by users with the correct permissions.

## Read Audit

For Employee Central, we recommend not only marking fields to be flagged for the read audit, but also flagging fields as sensitive, which masks them on the UI. Both of these fields can be added to the HRIS elements in the Business Configuration UI. This helps to prevent that a log is written every time that UI is accessed.

Note that attachments cannot be tagged as sensitive information. If an error occurs for a field with an attachment, then the system will not show that block.

The following core areas support read audit. Note that Address and Deductions do not support read audit.

Area	Sub-area
Personal Information	Person Information
Personal Information	Personal Information
Personal Information	Global Information
Personal Information	Email Information
Personal Information	Phone Information
Personal Information	Social Account Information
Personal Information	Primary Emergency Contact
Personal Information	National ID Information
Personal Information	Work Permit Information
Job Information	Job Information
Job Information	Job Relationship
Job Information	Employment Information
Job Information	Compensation Information
Job Information	Pay Component Recurring
Job Information	Pay Component Non-Recurring

## Change Audit

The following core areas support the change audit. This includes both effective and non-effective dated entities.

<b>Area</b>	<b>Sub-area</b>
Employment Information	Employment Info
Employment Information	Job Information
Employment Information	Job Relationship
Employment Information	Compensation Info - Pay Component Recurring & Spot Bonus
Personal Information	Address
Personal Information	Biographical
Personal Information	Dependents
Personal Information	Email Info
Personal Information	Emergency Contact
Personal Information	IM Info
Personal Information	National Id
Personal Information	Personal Information & Person Global
Personal Information	Phone Info
Personal Information	Work Permit Info

The following changes are included in the Change Audit report:

<b>Entity</b>	<b>Change</b>	<b>Abbreviation</b>
Non-Effective-Dated	Create	I
Non-Effective-Dated	Change	U
Non-Effective-Dated	Delete	D
Effective-Dated	Create	I
Effective-Dated	Insert of a new time slice in an existing record	EDU
Effective-Dated	Change to an existing record	COR
Effective-Dated	Delete	EDD

There are 2 limitations:

- If an admin changes data for users in different countries with different retention times, then the system applies the lesser of the retention times, for example, 3 months instead of 6 months.
- The system does not check whether target users are Employee Central users or not, for example, a user could be from Onboarding 1.0 or other modules.

## **Information Report**

# 2 Configuring the System

## 2.1 Provisioning

Provisioning is an internal tool that Professional Services consultants and partners use to set up SAP SuccessFactors modules for a customer. You can access each customer instance from within Provisioning.

→ Remember

As a customer, you don't have access to Provisioning. To complete tasks in Provisioning, contact your implementation partner. If you're no longer working with an implementation partner, contact SAP Cloud Support.

## 2.2 Initial Configuration Tasks

To get started with the customer implementation, you need to do a number of initial configuration tasks.

The tasks listed below are the minimum required provisioning settings. You will make further Provisioning settings based on the customer's requirements as you progress through the implementation.

→ Remember

As a customer, you don't have access to Provisioning. To complete tasks in Provisioning, contact your implementation partner. If you're no longer working with an implementation partner, contact SAP Cloud Support.

### Prerequisite

An instance has already been created for the customer.

### Tasks

Do the initial configuration tasks in the following sequence.

[Setting Up the Basics \[page 26\]](#)

[Setting Up Attachment Options \[page 28\]](#)

[Creating a Super Admin User in Provisioning \[page 28\]](#)

---

[Configuring Company System Settings \[page 30\]](#)

---

[Configuring Employee Central Settings \[page 31\]](#)

---

[Keeping the User Directory and Org Chart Up to Date \[page 32\]](#)

---

[Enabling the Employee Central SOAP APIs \[page 33\]](#)

---

[Enabling the Legacy Home Page \[page 33\]](#)

---

[Generating User IDs \[page 39\]](#)

---

[Setting Up the Currency Exchange Rate \[page 37\]](#)

---

[Creating the Employment Settings Configuration Object \[page 41\]](#)

---

Additional features you can enable based on your requirements:

- [Check Tool](#)

## 2.2.1 Setting Up the Basics

Here is an overview of the basic options that can be selected for Employee Central.

### Procedure

1. Log on to Provisioning with your user name and password, and select the company from the list shown or through the initial letter of the company ID.

#### → Remember

As a customer, you don't have access to Provisioning. To complete tasks in Provisioning, contact your implementation partner. If you're no longer working with an implementation partner, contact SAP Cloud Support.

2. Select  [Edit Company Settings](#) .
3. Enable the company languages by selecting the checkboxes of the relevant language packs.  
Make sure you select a minimum of one language pack.
4. Select the following Employee Central checkboxes:
  - Enable the Attachment Manager
  - Employee Profile Data Audit
  - Employee Central Foundation Objects
  - Enable Translation of Employee Central Foundation Objects - required Employee Central Foundation Objects, Enable Generic Object and Enable the Attachment Manager
  - Effective Dated Data Platform
  - Enable Effective-Dated Fields in Basic Import
  - Employee Central V2 (Event Reason Derivation) - requires Effective Dated Data Platform

- Enable Business Configuration in Admin Tools - requires Enables Generic Objects, Employee Central V2 (Event Reason Derivation), Enable the Attachment Manager, Effective Dated Data Platform, Employee Profile Data Audit
    - Enable Generic Objects - requires Enable the Attachment Manager
5. Select the *Role-based Permission* checkbox:
- Role-based Permission (This will disable Administrative Domains)
  - Dynamic Groups V2 (My Groups)
6. Select the data retention management checkbox:
- Enable Data Retention Management  
Enter the minimum number of approvers required by the company
- This allows the admin to purge inactive users. For more information about purging users, refer to the [Setting Up and Using Data Protection and Privacy](#) guide.
7. Optional: For a new customer, if you want to use the new Payment Information block (MDF-based, effective-dated, and employment-specific), select the following checkbox. You don't have to set up the HRIS elements `directDeposit` and `paymentInfo` in Succession Data Model. For more information, refer to the *Implementing and Configuring Payment Information in Employee Central* guide on the SAP Help Portal.
- Enable New Payment Information (MDF-based, effective-dated, and employment-specific). [CAUTION: For existing customers, by switching on this feature via Upgrade Center, the old direct-deposit-based UIs, APIs and objects will be irreversibly deactivated. New Payment Information is integrated into Employee Central Payroll. Integration scenarios towards 3rd party systems utilizing the old direct deposits APIs might no longer work. Please check in advance and inform customers that they might need to migrate existing 3rd party integration scenarios to the new APIs, for example, compound employee API or OData API.] — requires Employee Central V2 (Event Reason Derivation), Enable Generic Objects, Effective Dated Data Platform, Employee Profile data audit and Enable the Attachment Manager

#### i Note

For an existing customer that is using the old *Payment Information* or *Direct Deposit block*, if you want to enable the new *Payment Information* block, please use the Upgrade Center instead. For more information, see the *Implementing and Configuring Payment Information in Employee Central* guide on the SAP Help Portal.

8. Optional: If you want to hide the user name in the value help in the system, select the following checkbox:
- Hide user name from UI
9. Scroll back up to the top and select *Save Feature*.

## 2.2.2 Setting Up Attachment Options

Here is an overview of the attachment options that can be selected for Employee Central.

### Context

→ Remember

As a customer, you do not have access to Provisioning. To complete tasks in Provisioning, contact SAP Support.

### Procedure

1. In *Provisioning*, on the *Company Settings* page, scroll to the section *Document Attachment*.
2. Specify the attachment settings as required by the customer.

If the customer requirements are not known at this time, make the following settings:

Attachment	Setting
Attachment Storage Allocation	1G
Attachment User Limit	No Limit
Attachment Max Size	5M
Attachment Limit Notification Monitor Period	Never

3. Save your settings.

## 2.2.3 Creating a Super Admin User in Provisioning

Create a super admin user in the Provisioning application, for a specific customer instance, so that you can access the system and grant necessary permissions to other users.

### Prerequisites

You have Provisioning access to the instance.

→ Remember

As a customer, you don't have access to Provisioning. To complete tasks in Provisioning, contact your implementation partner. If you're no longer working with an implementation partner, contact SAP Cloud Support.

## Procedure

1. Log into Provisioning and select the company instance you wish to access.
2. Go to ► *Edit Company Settings* ► *Company Settings* ▶.
3. Search or scroll down the page to find the section with "super admin" settings.
4. Provide **all** of the following information.

Setting	Description
<b>Admin Username</b>	Determines both Username and User ID of the super admin user.
<b>Admin Password</b>	Password with which super admin can access your system.
<b>Admin First Name</b>	First name of super admin as it appears in the system.
<b>Admin Last Name</b>	Last name of super admin as it appears in the system.
<b>Admin Email</b>	Email address to which super admin receives notifications.
<b>Use PWD to log in to SAP SuccessFactors</b>	Once it is checked, the newly created super admin can log into the system using username and password.
<p><b>i Note</b></p> <p>This ONLY applies to the instance that has enabled Partial Organization SSO.</p>	
<b>Confirmation of customer approval</b>	Provisioning user must check a box confirming that they have received approval from the affected customer for the creation of a super admin user account.
<p><b>i Note</b></p> <p>As a Provisioning user, it is your responsibility to obtain this approval before creating a super admin. You cannot proceed without confirming that you have done so.</p>	
<b>Customer Email Address</b>	Customer email address that receives notification when the super admin account is created.
<p><b>i Note</b></p> <p>This should be the email address of one person who provided the customer approval. You can only send notification to one address.</p>	

Setting	Description
	As a Provisioning user, it is your responsibility to notify the customer and share this information with more people if necessary.

5. Select [Create Admin](#).

**i Note**

You can only proceed to create a super admin if you have provided all of the required information. If not, this action is disabled.

6. Save your changes.

## Results

The super admin user account is created and the customer is notified at the email address provided.

## 2.2.4 Configuring Company System Settings

Allow the admin access to the [Company System and Logo Settings](#) link in the Admin Center, which has many Employee Central relevant settings.

### Prerequisites

Ensure that the permission for  [Administrator Permission](#)  [Manage System Properties](#)  [Company System and Logo Settings](#) is enabled.

### Procedure

1. Go to  [Admin Center](#)  [Company System and Logo Settings](#).
2. In the [Company System Setting](#) section, select what is required for the company. Go through the list carefully. Many settings enable validations or affect search results.

We recommend that at least the following validations be selected:

- [Enable Address Validations](#)
- [Enable National ID Validations](#)
- [Enable Bank Account Validations](#)

- *Enable Payment Information Validations*
3. **Optional:** If you use contingent workers, select the *Enable target group based filtering for Worker fields* checkbox.
- This means that, if checked, then the values in the dropdown list for Worker fields will be based on the target group settings assigned in permissions. If not checked, then all users will be available in the dropdown list.
4. Save your settings.

## 2.2.5 Configuring Employee Central Settings

Enable areas within Employee Central from this page.

### Prerequisites

You must have the required permissions to view the page: ► [Permission Settings](#) ► [Manage System Properties](#) ► [Employee Central Feature Settings](#) ▶

### Context

Manage the areas of Employee Central using the Admin Center, for example:

- Time and Attendance Management
- Person, Employment and Worker Type
- Position Management
- Deductions Management
- Advances
- Fiscal Year
- IT Declarations
- Cost Distribution

### Procedure

1. Go to ► [Admin Center](#) ► [Manage Employee Central Settings](#) ▶

#### i Note

If you are unable to see this page, it is recommended that you log out and log back in to the Admin Center. Doing so will trigger the changes in permission immediately. You should then be able to search for the [Manage Employee Central Settings](#) page.

2. Enable your changes.
3. Save your changes.

## 2.2.6 Keeping the User Directory and Org Chart Up to Date

The User Directory and Org Chart use data that is maintained and stored in Employee Central. You need to make sure that the data in the different systems is current and consistent.

### Context

The user directory and org chart pull the data from User Data File. This User Data File holds the standard employee data that Directory, Org Charts and Talent Modules use as well.

Employee Central updates the user data in real time using HRIS Sync.

Employee Central needs to update the mini-master in real time, which is why we need to schedule the sync job. Once it is scheduled, every time a field in Employee Central that needs to flow to User Data File (for example, First Name, Last Name,... gets updated), the job launches automatically and updates the file.

If there are extra fields to update, you can configure the HRIS Sync mappings in the Succession Data Model.

#### → Remember

As a customer, you don't have access to Provisioning. To complete tasks in Provisioning, contact your implementation partner. If you're no longer working with an implementation partner, contact SAP Cloud Support.

#### i Note

This is an SAP SuccessFactors Business Beyond Bias feature. Use it to support processes that detect, prevent, or eliminate the influence of bias, helping you achieve your diversity and inclusion goals.

### Procedure

1. In *Provisioning*, navigate away from the *Company Settings* page by selecting the company name on the top left.
2. Scroll down and select *Manage Job Scheduler* *Manage Scheduled Jobs* .
3. Select *Create New Job* (above the *Job Type*) field.
4. Enter a suitable job name (such as **EC Effective Dates Sync**).
5. As the job owner, specify the Super Admin you created in the previous task. You can use the *Find User* function to do this.
6. From the drop-down list, select the job type *HRIS Sync*.

7. Based on the customer requirements, specify the time and frequency of the job.
8. Specify additional email recipients and job start notifications as required.
9. On the bottom right, select *Create Job*.
10. Save your changes.

## 2.2.7 Enabling the Employee Central SOAP APIs

Turn the API setting to on so that you can use them to integrate data.

### Procedure

1. Go to  [Admin Center](#)  [Edit Company Settings](#) .
2. Under *Web Services*, select the *Employee Central SOAP API* setting.
3. Save your changes.

### Results

With this option turned on, you can use Employee Central SOAP APIs.

## 2.2.8 Enabling the Legacy Home Page

Enable the legacy home page (released Q2 2016) in the Upgrade Center.

### Prerequisites

- You have access to the *Upgrade Center*.
- You can manage role-based permissions.

### Context

#### Caution

The legacy home page isn't the most recent user experience. It was released in Q2 2016 and is eventually being replaced by a newer user experience released in 2H 2020.

## Procedure

1. Go to  [Admin Center](#)  and find the [New Home Page](#) upgrade.
2. Follow the on-screen instructions to complete the upgrade.

## Results

### Caution

As part of the upgrade, we create a default permission role called [Homepage3 Tile View](#) that is assigned to everyone. That means that, by default, all users of the system can see all sections of the home page. This may result in some users seeing blank sections where no tiles are visible or sections that do not apply to their role.

## Next Steps

Proceed immediately to grant role-based view permissions for each home page section. Otherwise, the page is blank.

As an administrator setting up the home page, grant yourself permission to view all sections. Later, when the home page is set up, you can assign different view permissions to different roles.

## 2.2.9 Enabling the To-Do List

Enable the To-Do block so that it will show up in the Home screen for users.

## Context

As the name suggests, to-dos are tasks you are expected or required to carry out. For example, as a manager, you might be expected to approve an employee's rating or assessment.

## Procedure

1. In Provisioning, choose   [Company Settings](#)  [Home Page Settings](#) .

### → Remember

As a customer, you don't have access to Provisioning. To complete tasks in Provisioning, contact your implementation partner. If you're no longer working with an implementation partner, contact SAP Cloud Support.

2. Select the *Show ToDo block* option.
3. Save your changes.

## Results

Once you have done this, the block appears in your profile. You can manage how your To-Dos look using the To-Do List Manager.

For example, you can:

- Choose how long completed tasks continue to be displayed.
- Choose how long before they're due pending tasks are displayed.
- Generate lists of to-dos covering a given period, such as a month or a year.
- Opt to drop "supplemental tasks" — that is, tasks you actually create from the list.

## 2.2.10 Configuring the Internal Job History Block

Configure the Internal Job History block to view the internal career history of an employee. This history can be shared with a broader audience with the company.

### Prerequisites

- Ensure that you have permissions for People Profile from ► [Administrator Permissions](#) ► [Manage System Properties](#) ► [Manage Employee Files](#) ▶
- Create a rule using the *Internal Job History Block* rule scenario. You do this by going to ► [Admin Center](#) ► [Configure Business Rules](#) ► [Employee Central Core](#) ► [Internal Job History](#) ▶  
This rule scenario only supports Job Information as the base object. In the rule, you only add the If condition, for example, to show job changes in the People Profile. You can't change the Set condition in the rule, and it is not shown in the rule.

Here is an example for a rule configured for an event reason for promotion. Rules can be configured and customized based on customer requirements.

● IJH\_Event Reason (IJH\_Event\_Reason)

Scenario: Internal Job History [Change Scenario](#)

Basic Information

Start Date	01/01/1900
Description	

Parameters

Name	Object
Context	System Context
Job Information	Job Information

[Insert New Record](#) [Collapse All](#) | [Expand All](#)

Variables

If

**Job Information.Event Reason is equal to Promotion - Pay Change (PROWP)**

Then

This job information record will be displayed in the Internal Job History Block on the People Profile page.

- Ensure that you have permissions for this block from **User Permissions > Employee Data > Background Widgets > Internal Job History**.
- Ensure that you have permissions to view Job Information records from **User Permissions > Employee Central Effective Dated Entities > Job Information**.

## Context

This is a read-only block, which is really a filtered version of the job history for an employee.

## Procedure

1. Navigate to the **Admin Center > Configure People Profile**.
2. Select **Custom Blocks > Internal Job History**.
3. Drag and drop the block to where it should show in the profile.

This is a double block, meaning it needs 2 block spaces. The system does not allow you to place it in a single block.

4. Select the block to edit its properties.
5. Edit the block.
  - a. Enter the block title and block description.
  - b. Select the Job Information fields from the visible fields present in the SDM and CSF-SDM that you would like to see in the block.

### i Note

Transient fields, for example, time in position, time in job, time in company, are not supported.

### i Note

Any fields with the label **NA** do not show data in the block.

- c. Select the rule you created using the *Internal Job History* rule scenario from the *Rules* dropdown.

### i Note

Only the Internal Job History rule scenario rules are shown. If no rule is configured, then the Rule dropdown will not be seen.

6. Save your changes.

## Results

To view the Internal Job History Block, navigate to Employee Profile for the employee whose history you would like to see. The new block appears on the Employee Profile. You will see the filtered version of the job history for an employee based on the rule scenario configured and the Job Information fields selected.

### 2.2.11 Setting Up the Currency Exchange Rate

Set up the currency exchange rate to show pay component group values as well as to be able to calculate Compa Ratio/Range Penetration.

## Context

### i Note

From the Q2 2016 release, the Currency Conversion Rate Tables are deprecated for Employee Central and have been replaced by the Currency Exchange Rate MDF Object.

For Employee Central, this means you no longer have to update the *ECT\_CONV\_TABLE* currency conversion table. Note however, that this is not the case for Compensation and Variable Pay. These components still use *ECT\_CONV\_TABLE*.

If you use **both** *ECT\_CONV\_TABLE* for Compensation and/or Variable Pay and MDF Object Currency Exchange rate, then you will have to **manually sync** the currency tables.

To check the table:

1. Go to  [Admin Center](#)  [Manage Currency Conversion Rate Tables](#) .
2. Select the *ECT\_CONV\_TABLE* from the list.
3. When you try to update this table, an information message appears informing you that the table has been migrated. Select *OK*.
4. It is still possible to update this table. Make your changes and save.

For more information about this, refer to the [Instance Sync: Implementation and Administration](#) guide on the SAP Help Portal.

## Procedure

1. Ensure the correct permissions are granted.

Permission	Description
► <a href="#">Manage Compensation</a> ► <a href="#">Manage Currency Conversion Rates Tables</a> ▶	Allows users the right to manage the currency exchange rate tables.
► <a href="#">Metadata Framework</a> ► <a href="#">Import Permission on Metadata Framework</a> ▶	Allows users to import the MDF object into the system.
► <a href="#">MDF Foundation Objects</a> ► <a href="#">Currency Exchange Rate</a> ▶	Allows users to see and use the Currency Exchange Rate MDF object.

2. Check the picklists.

If you are a new customer, check whether there is already an existing picklist with ID = "CurrencyExchangeRateType" and a picklist value = "DEFAULT". If it is not available, create the picklist and the default value.

However, if you are an existing customer, the picklist will be created automatically during the migration. Note that the picklist CurrencyExchangeRateType with the value "DEFAULT" must be made available since it corresponds to the table ECT\_CONV\_TABLE.

- a. Go to ► [Admin Center](#) ► [Configure Object Definitions](#) ▶.
  - b. Select ► [Create New](#) ► [Picklist](#) ▶.
  - c. Enter at least the following:
    - o **Code:** CurrencyExchangeRateType
    - o **Effective Start Date:** 01/01/1900
    - o **External Code:** DEFAULT  
DEFAULT must be in capital letters.
  - d. Save your changes.
3. Check the default records.
    - a. Go to ► [Admin Center](#) ► [Manage Data](#) ▶.
    - b. In the **Search** field, select *Currency Exchange Rate*.
    - c. Update the default records for the currency exchange rates.

### ⚠ Caution

Only exchange rates with the default exchange rate type, DEFAULT, are used in Employee Central currency conversion.

For mass updates of currency exchange rates, you can use the standard import and export UI. For more information, refer to [Importing MDF Data](#) topic in the *Implementing the Metadata Framework (MDF)* guide.

- d. Save any changes.

## 2.2.12 Generating User IDs

Set up your system to automatically assign user IDs to employees once they are created in the system.

### Procedure

1. Go to the  [Admin Center](#)  [Company Settings](#).
2. Select [Company System and Logo Settings](#).
3. Select [Next Person Id Assigned](#).

Let's say a company with 10.000 employees acquires another company. When merging the employees into one company, 5.000 new employees are added with a CSV file import to the system. You would then need to set the ID for the next person that is hired to be the current number of employees plus 1, so you would enter **15.001** in the [Next Person Id Assigned](#) field.

#### Note

We recommend selecting a number that does not start with a leading zero. For example, for a company with 10.000 employees, you could start with the following number "100000", which would give a capacity of 1 million employees.

#### Note

If Position Management is active in your system, remember to update the Employee Number in this section as well as the Position Sequence Number during data migration.

4. Optionally, you can also select [Enable Database Sequence to generate next Employee Id](#) to change the process of how employee IDs are generated.

Instead of creating the IDs based on the `SYS_CONFIG` table, they are generated based on a database sequence. We recommend using this option if you receive application errors caused by duplicate employee IDs, which can happen in the following cases:

- Hires are processed by concurrent API calls
- Inactive number ranges
- During peak times when a high volume of hires are processed by multiple users at the same time In certain cases, gaps in the row of assigned IDs cannot be avoided, such as aborted hire processes. For example, if 2 hires are canceled, employees 1020 – 1040 exist, 1041 – 1042 are skipped, and employee IDs continue with 1043.

5. Save your changes.

## Results

The *Maximum Number already used as Employee ID for existing employees* field contains the last number already used for employees in the system. To start the next employee ID from this number, please enter this number in *Next Person ID Assigned*.

In certain cases, gaps in the row of assigned IDs cannot be avoided, such as aborted hires. After enabling the sequence, you may get the error with the text “Unable to generate unique Employee ID. Sequence needs to be updated. Please contact your Admin.”. This error occurs because the next number provided by database sequence is already used for other employees. The system has a set number of attempts to find a valid number, but if it can't find one, then you will get the error. To resolve this issue, you can update the sequence to *Maximum Number already used as Employee ID for existing employees* value. To do this, enter the *Maximum Number already used as Employee ID for existing employees* in the field *Next Person ID Assigned* and save the changes.

When you enable the database sequence, employee ID generation starts with the value in field *Next Person ID Assigned*. To start the sequence with different number, update the number in the field *Next Person ID Assigned* and save the changes.

If you decide to disable *Database Sequence*, the next employee ID generation logic will switch to the older approach. While switching, by default the old approach will start with the number in field *Next Person ID Assigned*. To start from a different sequence, update this number.

## Using Business Rules to Generate User IDs

As of Q3 2018, it is also possible to generate user IDs for hires or rehires using business rules.

## User IDs for Global Assignment and Concurrent Employment

When a global assignment is created in the system, a new user ID is automatically generated for the user. This is based on settings in the Global Assignment Configuration object. You can **never** influence the user ID with business rules.

## Related Information

- [Generating User IDs with Business Rules](#)
- [Different IDs in SAP SuccessFactors HXM Suite](#)

## 2.2.12.1 Optional: Creating Number Ranges for User ID Sequences

You can create multiple sequence ranges for customers who want to place certain types of employees in those ranges.

### Procedure

1. Go to  [Admin Center](#) .
2. In [Manage Sequence](#), create a new sequence.
3. Enter the required information:
  - External code
  - External name
  - Start
  - Step

For a range that starts at 3000, fill in the fields as such:

- External code: SetPositionExternalCode
  - External name: SetPositionExternalCode
  - Start: 3000
  - Step: 1
4. Save your changes.

### Next Steps

You can then create a business rule with an If/Else condition to call the respective sequence based on the employee category.

## 2.2.13 Creating the Employment Settings Configuration Object

Configure employment settings to generate the assignment ID automatically using business rules.

### Prerequisites

You created a rule to generate the assignment ID.

## Context

### i Note

Since the system creates assignment IDs based on the rule, this means that the assignment IDs might be different from the user IDs.

## Procedure

1. Go to  [Admin Center](#)  [Manage Data](#).
2. Create the *Employment Settings Configuration Object*.
3. Enter the required data:

Field	Setting
External Code	EMPLOYMENTSETTINGSCONFIG
Status	Active
Activate "Generate Assignment ID External" by Using a Business Rule	<p>No Choose No for the system to create the assignment ID based on the user sys ID.</p> <p>Yes Choose Yes for the system to create the assignment ID within a set number range. This is needed for integration with SAP S/4HANA on-premise, SAP ERP, and Employee Central Payroll so that the assignment ID reflects the personnel number (PERNR) from the SAP S/4HANA on-premise, SAP ERP system., and Employee Central Payroll.</p>
Enforce New Employment	<p>On change of Legal Entity</p> <p>Based on Business Rules</p>

4. Add the rule.
5. Save your settings.

## Related Information

### Assignment ID

[Optional: Prohibiting Change of Legal Entity in Any Employment](#)

[Enforcing New Employment Using Business Rules](#)

## 2.2.14 Creating the Hire/Rehire Configuration Object

Configure the hire and rehire for various optional settings

### Context

The search settings are for the duplicate check (match popup) used in the [Add New Employee](#) page.

If you don't configure this object, system will run for duplicate check based on the default criteria:

- Users with matching first name, last name, and date of birth
- Users with matching first name and last name
- Users with matching national ID

### Procedure

1. Go to [Admin Center](#) [Manage Data](#).
2. Select [Create New](#) [Hire/Rehire Configuration](#).
3. Enter the external code `MATCHINGUSERCONFIG`.

#### Note

You will get an error message if you enter a code other than MATCHINGUSERCONFIG.

4. Set the status to [Active](#).
5. Choose the configuration settings based on your requirements:

For this setting...	Choose...
<i>Enable search for employees using first name and last name</i>	The default is set to <a href="#">Yes</a> for the duplicate check.
<i>Enable search for employees using first name and date of birth</i>	The default is set to <a href="#">Yes</a> for the duplicate check.
<i>Enable search for employees using last name and date of birth</i>	The default is set to <a href="#">Yes</a> for the duplicate check.
<i>Enable search for employees using first name, last name, and date of birth</i>	The default is set to <a href="#">Yes</a> for the duplicate check.
<i>Enable search for employees using national ID</i>	The default is set to <a href="#">Yes</a> for the duplicate check.
<i>Default Personal Information during rehire</i>	The default is set to <a href="#">Yes</a> for rehire with previous employment scenarios.

<b>For this setting...</b>	<b>Choose...</b>
<i>Default Job Information during rehire</i>	The default is set to <b>Yes</b> for rehire with previous employment scenarios.
<i>Default Compensation Information during rehire</i>	The default is set to <b>Yes</b> for rehire with previous employment scenarios.
<i>Event reason for end-dated payments</i>	Select an event reason for the end-dated payments feature in the Compensation block.
<i>Rule to generate next employee ID</i>	Select an existing rule or choose the <b>Plus</b> icon to add a new rule.  For example, you can create and assign a Next User ID rule to control the sequence of IDs.
<i>Forward background elements from the previous (terminated) employment to the next when rehiring (with new employment)</i>	The default is set to <b>No</b> .  Background elements are part of Employee Profile and are tied to the userSysId. So, when employee is rehired using new employment, then these background elements such as Compensation Data are not visible in new employment. If this flag is set to <b>Yes</b> , then during rehire with new employment from UI, the system copies these background elements from the old userSysId to new userSysId and thus, the old background elements will be visible in the new employment record.

6. Save your settings.

# 3 Managing Role-Based Permissions

## 3.1 Permissions for Employee Central

You can use role-based permissions (RBP) to control access to who sees what with regard to employee information.

Role-based permissions allow you to grant different levels of read or write access depending on the role of the employee. For example, an employee is only allowed to read their own compensation information, but an HR Admin is allowed to edit it. You define these kinds of permissions by managing permission roles.

The blocks seen by users in the employee profile are directly related to permissions and roles granted to those users.

The permission categories are divided in [User Permissions](#) and [Admin Permissions](#), which are further subdivided, for example, [Employee Data](#) or [Miscellaneous Permissions](#). Once selected, the list of permissions associated with this category is displayed on the right side and in some areas, further divided into groups. For example, the [HR Information](#) section contains groupings, for example, for [Biographical Information](#).

### 3.1.1 User Permissions for Employee Central

You can use role-based permissions (RBP) to control access to who sees what with regard to what users can see and do in the system.

The blocks seen by users in the employee profile are directly related to permissions and roles granted to those users.

The permission categories are divided in [User Permissions](#) and [Admin Permissions](#), which are further subdivided, for example, [Employee Data](#) or [Miscellaneous Permissions](#). Once selected, the list of permissions associated with this category is displayed on the right side and in some areas, further divided into groups. For example, the [HR Information](#) section contains groupings, for example, for [Biographical Information](#).

Here is a list of the user permission categories.

Permission Category	Sections Relevant for Employee Central
Employee Data	<ul style="list-style-type: none"> <li>• <a href="#">HR Information</a></li> <li>• <a href="#">Employment Details</a></li> <li>• <a href="#">Global Assignment Details</a> Only available if you have activated global assignments in the Admin Center.</li> <li>• <a href="#">Pension Payout Details</a> Only available if you have activated pension payouts in the Admin Center.</li> <li>• <a href="#">HR Actions</a></li> <li>• <a href="#">Future Dated Transaction Alert</a></li> <li>• <a href="#">Transactions Pending Approval</a></li> <li>• <a href="#">View Workflow Approval History</a></li> <li>• <a href="#">Pay Component Groups</a></li> <li>• <a href="#">Pay Components</a></li> </ul>
Employee Central Effective Dated Entities	Set field-level permissions for effective-dated blocks and fields. These blocks are effective dated: <ul style="list-style-type: none"> <li>• Addresses</li> <li>• Compensation Information</li> <li>• Dependents</li> <li>• Job Information</li> <li>• Job Relationships</li> <li>• Personal Information</li> </ul>
Employee Views	Define whether the user can access the following pages from the Employee Files: <ul style="list-style-type: none"> <li>• Employment Information</li> <li>• Personal Information</li> <li>• Pending Requests</li> </ul> For workflows requested by you or for which you are the approver.

### 3.1.1.1 Employee Data Permissions - HR Information

Assign permissions for blocks that refer to non-effective dated entities. Non-effective dated means that the history for the changes will not be maintained (for example, for Phone Information).

The entries listed here refer to the different blocks that have been defined as HRIS elements in the Succession Data Model. You can choose these permissions:

- ***View***: The user can see the block.

→ Tip

If necessary, you can use OnView rules to control who can see which fields in the blocks listed here, since you cannot use role-based permissions to set field-level ***View*** permissions for these blocks. For

more information about how to create such rules, refer to the [Example Employee Central Business Rules \[page 178\]](#).

- **Edit:** The user can edit the block on the Personal Information or Employment Information page by clicking the [Edit](#) link in the block:

Note that the labels depend on the labels defined in the Succession Data Model. If you have taken over the standard Succession Data Model, the following entries are displayed under HR Information:

This HR Information entry...	...refers to this HRIS element:
Biographical Information	personInfo
National ID Information	<code>nationalIdCard</code> Select this permission to define whether users can view or edit national ID information for all countries or regions.
National ID (Restricted to only country/region of legal entity)	<code>nationalIdCard</code> Select this permission to restrict administrators so that they can only access the national ID information of an employee relevant to the country or region of the legal entity where the employee is currently employed. For example, an administrator responsible for an employee currently employed in the United States can't view or add national ID information related to other countries or regions for the employee.  For an employee with multiple assignments or employments in different countries or regions, if the responsible administrators with this permission can access an assignment, they only view and edit the national ID information relevant to the assignment.  Note that selecting only <a href="#">View</a> has the same impact as selecting both <a href="#">View</a> and <a href="#">Edit</a> , that is, administrators can both view and edit the relevant national ID information.
Phone Information	phoneInfo
Email Information	emailInfo
Business Email Address	This entry is an exception: It refers to the <a href="#">Email Type</a> field of the <code>emailInfo</code> HRIS element, where you can select the type <a href="#">Business</a> .  It is listed here because normally every employee needs a business email address. If a company assigns the email addresses to the employees and does not want them to be editable by the employees, select only <a href="#">View</a> permission here.
Social Accounts Information	imInfo

<b>This HR Information entry...</b>	<b>...refers to this HRIS element:</b>
Primary Emergency Contact	emergencyContactPrimary
Spot Bonus	payComponentNonRecurring
	Here you define the permissions for the manager:
	<ul style="list-style-type: none"> <li>• Select <a href="#">View</a> to allow the user to view the Spot Bonus block on the Employment Information page.</li> <li>• Select <a href="#">Edit</a> so that the user can navigate from the Employment Information page to the <a href="#">Update Employee Records</a> page using the <a href="#">Take Action</a> menu.</li> </ul>
	<p><b>i Note</b></p> <p>You can assign approval workflows for changes done on the <a href="#">Update Employee Records</a> page.</p>
Spot Bonus Edit Action	payComponentNonRecurring
	Here you define the permissions for the employee for changes done on the Employment Information page.
Payment Information	paymentInfo
Work Permit Info	workPermitInfo
Global Assignment Details	globalAssignmentInfo
	This entry is only relevant if you have activated Global Assignments Management in the Admin Center.
	For more information, refer to the <a href="#">Implementing and Configuring Global Assignments in Employee Central</a> guide on the SAP Help Portal.
	Select <a href="#">Edit</a> to allow the user to manage global assignments on the <a href="#">Update Employee Records</a> page using the <a href="#">Take Action</a> menu.
	<p><b>i Note</b></p> <p>You can assign approval workflows for changes done on the <a href="#">Update Employee Records</a> page.</p>

This HR Information entry...

Pension Payout Details

...refers to this HRIS element:

pensionPayoutsInfo

This entry is only relevant when you have activated pension payouts in the Admin Center.

You can find more information in the [Implementing and Configuring Pension Payouts in Employee Central](#) guide on the SAP Help Portal.

- Select [View](#) to allow the user to view the *Pension Payout Details* block on the Employment Information page.
- Select [Edit](#) to allow the user to manage pension payouts on the *Update Employee Records* page using the *Take Action* menu.

**i Note**

You can assign approval workflows for changes done on the *Update Employee Records* page.

### 3.1.1.1 Employee Data Permissions - Global Assignment Details

Assign permissions for the *Global Assignment Details* block.

You can set field-level permissions for the fields from the Succession Data Model for the HRIS element `globalAssignmentInfo`.

In addition, you can set the following permissions:

For this Global Assignment Details entry...

*Global Assignment View block*

...select this permission:

[View](#) to allow the user to view the *Global Assignment Details* block on the *Employment Information* page.

Only [View](#) is applicable here; [Edit](#) has no function.

For this Global Assignment Details entry...

...select this permission:

*Global Assignment Edit Link*

*Edit* to allow the user to make changes to the *Global Assignment Details* block directly on the *Employment Information* page.

You must also select the *Global Assignment View block* permission.

**i Note**

You **cannot** add approval workflows to changes done using the *Edit* link.

*Global Assignment Add*

*Edit* to allow the user to add a global assignment by navigating from the *Employment Information* page to the *Update Employee Records* page using the *Take Action* button.

*Global Assignment Edit/MSS*

*Edit* to allow the manager to edit a global assignment by navigating from the *Employment Information* page to the *Update Employee Records* page using the *Take Action* button.

**i Note**

You can assign approval workflows for changes done on the *Update Employee Records* page.

*Global Assignment End*

*Edit* to allow the manager to end a global assignment by navigating from the *Employment Information* page to the *Update Employee Records* page using the *Take Action* button.

*Global Assignment Delete*

*Edit* to allow the manager to delete a global assignment by navigating from the *Employment Information* page to the *Update Employee Records* page using the *Take Action* button.

### 3.1.1.2 Employee Data Permissions - Pension Payout Details

Assign permissions for the *Pension Payout Details* block.

The fields from the Succession Data Model for the HRIS element `pensionPayoutsInfo` are listed in the Permissions screen. Only the HRIS fields with visibility "both" or "view" are available for setting permissions.

There is one exception:

For this Pension Payout Details entry...

...select this permission:

Pension Payout Edit Link

- *View* to allow the user to view the *Pension Payout Details* block in *Employment Information*.
- *Edit* to allow the user to update the block on the *Employment Information* page by clicking the *Edit* link. Note that workflows cannot be assigned for changes done this way.

### 3.1.1.2 Employee Data Permissions - Employment Details

Assign permissions for the *Employment Details* block.

The fields from the Succession Data Model for the HRIS element `employmentInfo` are listed in the Permissions screen. Only the HRIS fields with visibility "both" or "view" are available for setting permissions. Termination-related fields are also included.

There are these exceptions:

For this Employment Details entry...

...select this permission:

Employment Details MSS

For the manager:

- *View* to allow the manager to view the *Employment Details* block.
- *Edit* to allow the manager to navigate from the Employment Information page to the *Update Employee Records* page using the *Take Action* button.

#### i Note

You can assign approval workflows for changes done on the *Update Employee Records* page.

Employment Details Edit

For the employee:

- *View* to allow the user to view the *Employment Details* block on the *Employment Information* page.
- *Edit* to allow the user to edit the block on the *Employment Information* page by clicking the *Edit* link in the block.  
Note that workflows cannot be assigned for changes done this way.

<b>For this Employment Details entry...</b>	<b>...select this permission:</b>
Add New Employment	For the user who is allowed to add multiple employments for one employee:  <i>Edit.</i> Please note that <i>Concurrent Employment Management</i> needs to be enabled in the Admin Center to use this function.
Bonus Pay Expiration Date	This field is listed here because of a functional limitation of the role-based permissions framework. Hide this field from the user interface by <b>deselecting</b> <i>View</i> and <i>Edit</i> .  It is called a functional limitation because it actually belongs to the Termination Information block. The permissions are included because it required field permission rather than permission for the whole block.
Change primary Employment	The field defines whether the admins are allowed to change the employment classification of an employee in the <i>Employment Details</i> rather than in the <i>Manage Data</i> UI.

### 3.1.1.3 Employee Data Permissions - HR Actions

Assign permissions for the *Update Employee Records* page.

The HR Actions section controls mainly who has access to the *Update Employee Records* page for actions defined in the Succession Data Model.

<b>This HR Action...</b>	<b>...defines this permission:</b>
Update Employment Records (displayed as <i>Take Action</i> button)	This option overrules all other permissions in this section. It controls whether the user can see and use the <i>Take Action</i> button from the <i>Employment Information</i> page.
View Higher Grades	This option defines if a manager can view an employee's job classification if it is higher than the manager's.  To restrict a manager's view, leave the permissions unchecked. Make sure that the job classifications are assigned to a pay grade, and have a <i>paygradeLevel</i> maintained. When the manager goes to the <i>Update Employee Records</i> page for <i>Job Information</i> , the list of job classifications in the dropdown is limited to those whose <i>paygradeLevel</i> is less than the manager's. Note that this function is limited to this page; it does not have any effect on the <i>Job Info History</i> page.

This HR Action...	...defines this permission:
hireAction	This is an hrис-action from the Succession Data Model. It defines if the user can access the <a href="#">Add New Employee</a> link in the Admin Center.
reHireAction	This is an hrис-action from the Succession Data Model. It defines if the user can access the <a href="#">Rehire Inactive Employee</a> link in the Admin Center.
terminateAction	This is an hrис-action from the Succession Data Model. It defines if the user can access the <a href="#">Take Action</a> button on the <a href="#">Employment Information</a> page and select <a href="#">Terminate</a> from the dropdown menu.
Terminate/Retire	This option defines whether the admin has permission to terminate every single concurrent employment. If yes, the <a href="#">Terminate All</a> option will be visible in the Terminate/Retire screen.

**i Note**

Permissions to access the [Update Employee Records](#) page for Global Assignments are set in HR Information.

### 3.1.1.4 Employee Data Permissions - Future-Dated Transaction Alert

Define whether a user has the permission to view future changes for effective-dated entities.

You define if a user has the permission to view future changes for effective-dated entities by clicking on the [Pending future change...](#) link:

Only the [View](#) permission is applicable here. The [Edit](#) permission has no function. Future-dated transaction alerts can be set for the following blocks:

This block...	...refers to this HRIS element:
Personal Information	personalInfo
Addresses	homeAddress

This block...	...refers to this HRIS element:
Dependents	personRelationshipInfo
	This entry is only relevant if you have activated the <i>Dependents Management</i> feature in the Admin Center.
	You can find more information in the <a href="#">Implementing and Configuring Dependents in Employee Central</a> guide on the SAP Help Portal.
Job Information	jobInfo
Compensation Information	compInfo
Job Relationships	jobRelationsInfo

### 3.1.1.5 Employee Data Permissions - Transactions Pending Approval

Define whether a user can see if a workflow has been initiated, but not yet approved.

The *Pending Approval* link:

- *View* means the pending approval link is shown, but you cannot click on it to get to the details of the workflow request.
- *Edit* means you can view and click on the pending approval link.

You can set the permission for the following blocks:

This block...	...refers to this HRIS element:
Personal Information	personalInfo
Employment Details	employmentInfo
Job Information	jobInfo
Compensation Information	compInfo
Spot Bonus	payComponentNonRecurring
Job Relationships	jobRelationsInfo

### **3.1.1.6 Employee Data Permissions - View Workflow Approval History**

Define the permissions to view the workflow history from the [History](#) page of certain effective-dated entities..

Only the [View](#) permission is relevant, [Edit](#) has no function. The user with [View](#) permission can select [View Approval History](#) from the [Take Action](#) dropdown menu from the [History](#) page.

This block...	...refers to this HRIS element:
Personal Information	personalInfo
Job Information	jobInfo
Compensation Information	compInfo
Job Relationships	jobRelationsInfo

### **3.1.1.7 Employee Data Permissions - Event Reasons**

Assign View or Edit permissions for individual event reasons. This helps distribute different functions within the company to the correct people.

Here are a few examples, of why distribution is important:

- HR admins can be the only ones given access to data changes and this action has no workflow attached.
- HR admins have access to transfers outside the team.
- Managers only have access to transfer to/from their team.
- Payroll admins only have access to out-of-cycle salary increases.

There are many types of event reasons, for example, data changes, termination, job changes, global assignment, benefits, paid or unpaid leave, hire or rehire, transfer, and so on.

### **3.1.1.8 Employee Central Effective Dated Entities Permissions**

Set field-level permissions for effective-dated blocks and fields.

These permissions include country/region-specific fields that are prefixed by the 3-letter ISO code (for example, FRA for France, DEU for Germany, and so on). There are 5 different permissions you can select for effective-dated entities:

- [Delete](#): The user to delete an effective-dated entity. This is only applicable at element or block level, not at field level.
- [View Current](#): The user can see only the current field value of an effective-dated entity. When the user looks at the [History](#) page, the past data record for this field is not displayed.

- ***View History***: The user can see past values on the *History* page. This permission also includes the *View Current* permission, so that the user can also see the current field value.
- ***Edit/Insert***: The user can edit an effective-dated entity by inserting a new data record for it which is effective as of a certain date. As the user does not really change the data record itself (then it would just overwrite the past data record), past data records are still available in the *History*. The field is also available for editing when a new data record is inserted.
- **Correct**: The user can make corrections to a field from the *History* page:

The following sections are relevant for Employee Central:

For this block...	...you can set permissions for fields from:
Personal Information	<p>personalInfo</p> <p>plus globalInfo fields from the country/region-specific Succession Data Model</p>
Addresses	<p>homeAddress</p> <p>This is an exception: Here you can only set permissions on block level, but not on field level.</p>
Dependents	<p>personRelationshipInfo</p> <p>This entry is only relevant if you have activated the <i>Dependents Management</i> feature in the Admin Center.</p> <p>You can find more information in the <a href="#">Implementing and Configuring Dependents in Employee Central</a> guide on the SAP Help Portal.</p>
Job Information	<p>jobInfo</p> <p>plus jobInfo fields from the country/region-specific Succession Data Model</p> <p>Note: The field <i>FTE</i> is a calculated field and thus read-only; select only <i>View Current</i> and/or <i>View History</i>.</p>
Compensation Information	<p>compInfo</p> <p>Note: The fields <i>range penetration</i> and <i>compa-ratio</i> are calculated fields and thus read-only; select only <i>View Current</i> and/or <i>View History</i>.</p>
Job Relationships	jobRelationsInfo

For a complete list of all listed fields, refer to the fields listed in your Succession Data Model and country/region-specific Succession Data Model.

## Show/Hide User

You can show the name of the user next to the user name in the *Last modified by* field in the History pages of effective-dated HRIS elements. Showing the user name rather than the user ID in the History page makes it easier to identify employees who last changed the record.

If the  *Platform Feature Settings*  setting is active in the system, then the person who made the latest changes is shown with their full name.

If the  *Platform Feature Settings*  setting is not active in the system, then the person who made the latest changes is shown with their full name and their user ID.

## Further Block Permissions

In addition, for each of the sections, you can set these two permissions on block level:

- [block] *Actions*: This permission allows the user to:
  - Insert a new record from the *History* page
  - Add a new record using the *Take Action* dropdown menu, or add a new record using the *Edit* link in the block for those blocks that are listed on the *Personal Information* page

### i Note

Use this option when you want to associate an approval workflow with the changes done in this block.

Select *Correct* as minimum if you want to grant this permission to a user.

- *Edit Link*: Allows the user to edit fields in the block by clicking the *Edit* link in the block on the *Employment Information* or *Personal Information* page.

The remaining entries refer to the fields listed in the Succession Data Model and country/region-specific Succession Data Model. If a field is configured in both the Succession Data Model and the country/region-specific Succession Data Model, only the field from the Succession Data Model is shown in this list.

### 3.1.1.9 Employee Views Permissions

Define whether the user can access pages from the *Employee Files* using the drop-down menu.

Under *Employee Views*, you define whether the user can access pages from the *Employee Files* using the drop-down menu.

The following employee views are relevant for Employee Central:

- *Employment Information*: Grants access to the *Employment Information* page
- *Personal Information*: Grants access to the *Personal Information* page
- *Pending Requests*: Grants access to the *Pending Requests* page for workflows requested by you or for which you are the approver.

## 3.1.2 Administrator Permissions for Employee Central

You can use role-based permissions (RBPs) to control access with regard to which admin can view or edit which data.

Role-based permissions allow you to grant different levels of read or write access depending on the role of the employee. For example, an employee is only allowed to read their own compensation information, but an HR Admin is allowed to edit it. You define these kinds of permissions by managing permission roles.

Under [Administrator Permissions](#), the following permission categories are relevant for Employee Central:

Permission Category	Description
Manage System Properties	These permissions ensure that access and validations are properly set up.
Manage Foundation Objects	These permissions ensure that users can import and work with foundation objects and translations for Job Codes.
Manage Foundation Object Types	These permissions are control what the admin is allowed to do on the <a href="#">Manage Organization, Pay and Job Structures</a> page. Grant permissions for each individual foundation object.
Manage User	These permissions ensure that users have the correct access to all they need in Employee Central. This is especially important for the integration between Recruiting, Onboarding 1.0, and Employee Central.
Metadata Framework	These permissions ensure that users can work with generic objects in the Metadata Framework (MDF).  For more information, refer to the <a href="#">Implementing the Metadata Framework</a> guide on the SAP Help Portal.

### i Note

What is the difference to the [Manage Data](#) permission?

Without the read/write permission, the user cannot see or manage generic objects on any page in the system. Without the [Manage Data](#) permission, the user cannot access the [Manage Data](#) page, but is still able to manage data from other pages, such as the [Configure Business Rules](#) page (if the [Configure Business Rules](#) permission is granted).

Permission Category	Description
Manage Business Configuration	<p>These permissions ensure that users can work with the Business Configuration UI, which allows them to access the Succession Data Model as well as the country/region-specific Succession Data Model from the UI rather than having to go through Provisioning.</p> <p>This entry is only displayed if you have activated the <i>Business Configuration in Admin Tools</i> feature in Provisioning.</p> <p>For more information, refer to the <a href="#">Setting Up and Using Business Configuration UI (BCUI)</a> guide on the SAP Help Portal.</p>
Employee Central API	<p>These permissions ensure that users can work with the SOAP-based application programming interfaces (APIs) for Employee Central. These are relevant for integrating Employee Central with other software products.</p> <p>The Foundation APIs are relevant for foundation data, the HRIS APIs for person and employment data.</p> <p>For more information, refer to the <a href="#">SAP SuccessFactors Employee Central OData API: Reference Guide</a> on the SAP Help Portal.</p>
Manage Time Off Manage Time	<p>These permissions ensure that users can work with Time Off and the Time Sheet.</p> <p>For more information about Time Off, refer to the <a href="#">Implementing Employee Central Time Off</a> guide on the SAP Help Portal.</p> <p>For more information about Time Sheet, refer to the <a href="#">Implementing Employee Central Payroll Time Sheet</a> guide on the SAP Help Portal.</p>
Manage Positions	<p>These permissions ensure that users can work with Position Management.</p> <p>For more information, refer to the <a href="#">Employee Central Position Management</a> guide on the SAP Help Portal.</p>
Manage Compensation Manage Pay Scale Manage Deductions Manage Spot Awards	<p>These permissions ensure that users can work with Employee Central compensation data.</p> <p>For more information, refer to the <a href="#">Implementing and Configuring Employee Compensation Data in Employee Central</a> guide on the SAP Help Portal.</p>

### 3.1.2.1 Main Admin System Permissions

Set permissions to ensure admins have access to the correct pages to complete their work.

Here you define permissions for the admin that cover many aspects of the system, for example, creating & updating company settings as well as processes. Allowing admins the rights to update settings for mobile and security areas is also done here.

Ensure that at least the following are selected:

- Employee Central *Feature Settings*  
This allows admins to turn on features in Employee Central themselves without having to create an SAP Cloud Support ticket.
- *Company System and Logo Settings*  
This allows admins to make enable or disable further company settings, such as validations for sensitive data.

### 3.1.2.2 Permissions for Foundation Objects

Set permissions to ensure users can work with foundation objects.

Here you define permissions for working with foundation objects.

- *Import Foundation Data*: Grants access to the *Import Foundation Data* link in the Admin Center.
- *Import Translations*: Allows the admin to import translations for the `jobCode` foundation object, using the *Import Translations* link in the Admin Center. For more information, refer to [Translating Foundation Data](#).

### 3.1.2.3 Permissions for Foundation Object Types

Set permission to ensure users can work with foundation object types.

You can define permissions for the admin that refer to the different types of foundation objects. Foundation objects are created, edited, and deleted in the Admin Center. To access the page, in the *Tools Search* field, select [Manage Organization, Pay and Job Structures](#).

The following permissions are relevant here and refer to what the admin is allowed to do on the [Manage Organization, Pay and Job Structures](#) page:

- *View*: The admin can only view the corresponding foundation object type.
- *Create*: The admin can create a foundation object of the selected type.
- *Insert*: The admin can create a new data record for a foundation object type, by selecting [Insert New Record](#).
- *Correct*: The admin can correct foundation objects by selecting [Take Action](#) in the [History](#) page.
- *Delete*: The admin can delete foundation objects by selecting [Take Action](#) in the [History](#) page.

### 3.1.2.4 Permissions for User Management

Set permissions to ensure that users have the correct access to all they need in Employee Central. This is especially important for the integration between Recruiting, Onboarding 1.0, and Employee Central.

The following scenarios may be relevant for you to help you make the correct selections:

- Employee Central Only  
The employee is created manually by selecting [Add New User](#) and/or by importing the data manually (10+ Imports per User)
- Onboarding 1.0 to Employee Central  
The employee data flows from Onboarding 1.0, either fully or partially depending on the panels selected in Onboarding. This will be introduced using the Pending Hires Process.
- Recruiting to Employee Central  
All the information captured in Recruitment will flow to Employee Central. It may be that there are lots of details from the employee that still need to be completed using the Pending Hires Process.
- Recruiting to Onboarding 1.0 to Employee Central  
All information flows from either from Recruiting or Onboarding 1.0 and the HR Admin will need to just review that all the information is correct using Pending Hires Process.

Here, the following checkboxes are relevant for Employee Central:

- [Add New User](#): Grants access to the [Add New Employees](#) link in the Admin Center.

#### i Note

The [Add New Employee](#) screen does not respect the role-based permissions you set up here. Instead it respects the settings from the data models with regards to whether a field or block is visible or editable.

- [Rehire Inactive Employee](#) or [Rehire Inactive Employee with New Employment](#): Grants access to the [Rehire Inactive Employee](#) link in the Admin Center.
- [Rehire Inactive Employee with New Employment \(by 'match' in New Hire\)](#) or [Rehire Inactive Employee \(by 'match' in New Hire\)](#): Grants access to the [Match](#) pop-up in the [New Hire](#) screen.
- [Include Inactive Employees in the search](#): Enables the search for inactive users on the [Employee Files](#) page and in the directory search.
- [Import Employee Data](#): Grants access to the [Import Employee Data](#) link in the Admin Center.
- [Restrict fields of type Worker](#)  
Fields of the type Worker (for example, supervisor in Job Information or HR/matrix manager in Job Relationship, and so on) respect target groups defined in permissions. This means that, if configured, users can only add managers that are included in the target group defined in the permissions.  
For example, you may want to restrict the access of a user to all managers of a legal entity.
- [Manage Workflow Requests](#): Grants access to the [Manage Workflow Requests](#) link in the Admin Center, for example, to change the approver for a particular workflow.

#### i Note

The admin can only access the workflow requests for the target population to which the admin role has been granted access.

- [Manage Workflow Groups](#): Grants access to the [Manage Workflow Groups](#) link in the Admin Center.

### 3.1.2.4.1 Permissions for Rehiring an Employee

There are several ways in which you can rehire a previous employee.

For data privacy reasons, you can create a new employment record for them that doesn't contain their previous Job Information and Compensation Information but only refers to their person ID.

There are 4 ways to rehire a previous employee: 2 types of rehire (re-activate existing inactive employment, create new employment) and 2 places to rehire (Match pop-up in New Hire, Rehire inactive employee)

UI/Type of Rehire	Re-activate existing inactive employment	Add new employment
Rehire inactive employee	Recommended when it is acceptable that the users in the new organizations can access the old employment records.	Recommended if the existing inactive employment should be shielded from users of the rehired employees new organization.
Match pop-up in New Hire	Recommended when it is acceptable that the users in the new organizations can access the old employment records.	Recommended if the existing inactive employment should be shielded from users of the rehired employees new organization.

If the [Rehire Inactive Employee with New Employment](#) permission is granted, then admins will see a drop-down list in the [Rehire Inactive Employee](#) screen.

### 3.1.3 Adding Value Help Permissions for Everyone

Set the read-only permission for the name and external code fields of the Metadata Framework (MDF) Foundation Objects (FOs) and set the rest of the fields to No Access. This grants Value Help Read Only permissions for everyone.

#### Context

These permission settings allow all users to view and select information on the New Hire, Employee Self Service (ESS), and Manager Self Service (MSS) pages.

Without these settings, users will not be able to view or select values from a drop-down list associated with a field of the MDF FO (also referred to as value help). For example, if the setting is not applied to the Legal Entity MDF FO, the user will not be able to view or select any Legal Entity value from the drop-down lists.

#### Procedure

1. Go to  [Admin Center](#)  [Manage Permission Roles](#)

2. Select *Create New*.
3. In the *Role Name* field, specify a name for the role. For example, *MDF Foundation Objects Value Help Read*.
4. Select the *Permission* button.
5. Select *MDF Foundation Objects*. The permission settings page is displayed listing all the MDF FOs.
6. Define settings for *BusinessUnit*, for example. Apply *Read Only* permissions to the *externalCode* field.
  1. We will now set *Read Only* permission for the *name* field. In this example, select *Business Unit Name* from the *Field* dropdown and apply the *Read Only* permission.
  2. Now, repeat this for all other fields displayed in the *Field* dropdown but set *Permission* to *No Access*, as shown below.
    - a. Select the *View Current* checkbox.
    - b. Select the *Field Level Overrides* checkbox. Additional options are displayed.
    - c. From the *Field* drop-down, select the field corresponding to name. In this case, *Business Unit Code*.
    - d. From the *Permission* drop-down next to the field name, select *Read Only*.
    - e. We will now set *Read Only* permission for the *name* field. In this example, select *Business Unit Name* from the *Field* drop-down and apply the *Read Only* permission.
    - f. Now, repeat this for all other fields displayed in the *Field* drop-down but set *Permission* to *No Access*.
    - g. Select *Done* to save your changes.
7. Repeat step 7 for all migrated MDF Foundation Objects.
8. After the permission settings are done, in Permission Groups, assign the role to *Everyone*.

# 4 Ad Hoc Report Types

All Ad Hoc reports are now enabled automatically when the corresponding module is enabled. There are several report schemas available for Employee Central ad hoc reports. Some are end-user reports and others are purely meant for admins.

## Basic Information

- Some reports do not support role-based permissions
- Reports will time out after 5 minutes. This means that, for the more complex or large ad-hoc reports, we recommend that you run these reports offline. Alternatively, if they are to be run on a regular basis, they can be scheduled to run and output to SFTP
- Always use filters in Date Range and As Of Date reports

## Person and Employment (As of Date)

This report shows employee HR data as of a given date (by default, it shows today unless specified), for example, reporting all employees hired as of a certain date. This report can be run based on future dates as well. For example, you could run a Termination report on Jan 01, 2013 to see how many future dated terminations are set to take place As Of Date Jan 31, 2013.

Make sure to use filters to limit the size and scope of the report - such as filtering on a particular Legal Entity, or Country.

This report respects Cell Level Permissions.

Be mindful of the number of Column Sets (JOINS) you include in one report - for example, if you include Compensation or Pay Component data (as employees tend to have more than one), you could end up with duplicate rows in the report .

The report results return only as numbers. This is for performance reasons. If you want the corresponding labels or external codes that match, select the columns in the relevant entity to generate a report with codes.

### i Note

Since the data displayed in the Compensation Information block is transient (calculated when the page loads), the displayed value is not stored in the database, and therefore not directly available in ad hoc reports. To display this information in the Person and Employment (as of date) ad hoc report, you must have the [HRIS PayComponentGroup Sums Sync](#) job created and scheduled in your instance. This job must be created by Product Support or the Implementation Consultant configuring the instance.

When the job runs for the first time, it will likely take some time to complete. However, once completed, all subsequent jobs that run (advised as once daily) will be much faster. Once the job is completed, the Person and Employment (As of Date) report displays the calculated values when selecting the column set "Employee Pay Group Sums" and one of the Pay Component Groups, such as AnnualizedSalary.

## **Job Information (Date Range)**

This report shows an employee's Job Info for a range of dates; for example, reporting all Job Information and Status changes within the give date period. For example, all Job Information and Status changes between Jan 01, 2012 and July 01, 2013 (in mm/DD/yyyy format).

This schema reports data based on the effective dates of the employees Job Information records. If you report on Compensation Information, the report generates one row per Job Info effective-dated record the employee has, and NOT based on one row per Compensation Info effective-dated record the employee has. For example, if the employee has three Compensation Info records but six Job Info records, and you report on Job Information using this report, you will see six rows for Compensation Information, because the Compensation Information records are reported on based on the Job Info record effective dates, within the date range you specify.

This report does not respect Cell Level Permissions.

The report always generates based on Job Information data structure.

If you add multiple column sets to the report, this increases the complexity of the report and you may need to run the report offline for it to complete successfully.

## **Recurring Compensation (Date Range)**

This report shows an employee's Compensation Information for a range of dates; for example, reporting on salary changes between 01/01/2012 and 07/01/2013 (mm/DD/yyyy).

This schema reports on data based on the effective dates of the employee's Compensation Information records. If you report on Job Information, the report generates one row per Comp Info effective dated record the employee has, and NOT based on one row for each Job Information effective dated record the employee has. For example, if the employee has three Job Information records but six Compensation Information records, and you report on Job Information using this report, you see six rows for Job Information, because the Job Information is reported on based on the Compensation Information record effective dates within the date range you specify.

This report does not respect Cell Level Permissions.

Do not include too many complex joins. For example, do not include Pay Component Non-Recurring data if there is no need.

If you are getting multiple (duplicate) rows - please ensure for each Effective Dated column set, you include also the Start Date and Sequence Number fields - this makes the report easier to understand when mashing a lot of different table data together.

## **Non-Recurring Compensation (Date Range)**

This report shows the non-recurring pay Components within a Date Range specified by the user; for example, reporting bonus payments within a certain date range. You should only use this report to identify Spot Bonus/ One-Time Bonus information for a period.

This report does not respect Cell Level Permissions.

Do not include too many complex joins. For example, do not include Pay Component Recurring data if there is no need.

If you are getting multiple (what looks like duplicate) rows, please ensure that, for each Effective Data column set, you also include the Start Date and Sequence Number fields. This makes the report easier to understand when a lot of different table data is being mashed together.

## **Person and Employment (Audit)**

This report shows all the inserts and corrections of an employee's information in Employee Central, including who made changes and when. An example would be reporting employee movements and flagging any historical changes.

We recommend that you use this report to determine who inserted, deleted, or edited a record in the employee's data in Employee Central. This is a very powerful report that shows one row per Insert/Update/Delete of data for each record that is reported on. Run this on only one area of Employee Central data at a time, for example: Job Information (do not include Compensation Information, or other data). Make a separate report for Compensation Information audit, or Personal Information audit, and so on.

This report does not respect Cell Level Permissions.

### Action Types in the Audit Report

- Insert = Represents the change was made using 'Take Action' or Inserting a record into the history
- Update = Can happen from either the 'Pencil' icon or from editing an existing record from the Employee History
- Delete = The record was deleted. Please note that you can view any Deleted record with this report

You must filter the report to ensure that you do not get too much data returned. No filter results in ALL audit data, but will most likely cause the report to fail (since it is a LOT of data).

The report should only ever be run on a one-column set. Do not mix fields from different columns sets. Doing so will skew the report when the tables are joined.

Only admin users should have access to this report.

This report should not be used for any headcount or functional user reporting. It is purely an admin report used to check who changed what and when.

You can see deleted records in this report.

## **Person and Employment (Export)**

You can export employee data so that it can be updated and reimported with any need to format it in an import-friendly way. For example, if you needed to update Job Information records for multiple users, you would use this report to extract the data.

If you need to create multiple export files on a regular basis, you can create a Multi Data Set report using only the P&E Export schema, and include one column set for each domain. Then you can extract data for Job Info, Comp Info, Home Address, and so on, into separate tabs.

This report does not respect Cell Level Permissions.

Do not include multiple column sets (such as Job Information, Compensation Information, and so on). This report is meant to be run against one column set at a time (all fields in Job Information column list, for example).

This report should not be used for any headcount/functional reporting. This report is purely an admin report to allow export of data in an easy-to-use format for data imports.

Ensure that this report is always run with filters. It will likely fail when run with no filters if the employee/employment population in the instance is very high.

## Foundation Objects

You can export information directly from the Foundation Object tables that have been loaded to the system or manually entered. For example, reporting directly on one or more particular Foundation Objects, such as returning the details of all Locations and linked Organization Units (showing the relationship).

Use this report to export **only** the legacy Foundation Object data, in an import-friendly format. If you need to export MDF-based Foundation Object data for import, please use [Import and Export Data](#) instead.

## Multi-Data Set Reporting

Do not mix and match the report schemas when creating multi-data set reports. For example, if you create a multi-data set report using a Date Range and an As Of Date schema, the system generates the report based on the As Of Date schema. The same is true if you include the Export schema within the above scenario (Export, Date Range and “As of Date” schemas), then the report will actually run based on the Export report, and date range/as of date will not be possible with the reports. You will also have unexpected results and behavior, as the system is not designed to work in this way. If you do need to create multi-data set reports, please ensure you use the same schema type for each domain you add to the report.

## Cross-Domain Reporting

It is currently not possible to use this ad hoc Report feature with Employee Central 2.0 ad hoc report schemas.

## Scheduled Reporting

All Employee Central ad hoc Reports can be scheduled to run and export to SAP SuccessFactors or external FTP folders. To set up scheduled Employee Central ad hoc reports, please create the report you wish to have

scheduled, and then raise a support ticket with the Customer Success team, who will help schedule the report for you. Please be sure to provide Customer Success with the timing the report should run under (what date/time should the report be scheduled to run, how often), and also the name of the report and the username of the owner of the report.

## 4.1 Permissions for Ad-Hoc Reports

There are different options that can be enabled for Ad Hoc Reports.

### Row Level Permissions

Row-level permissions are enabled by default and cannot be disabled. This layer of security restricts the user running the report, to be able to report on the target populations assigned by role-based permissions (RBP). Please note that this will mean the user running the report will be able to report on any data for any user in their target population.

This is specific to the report schema you are creating. The As of Date will limit 1 row for the Effective Date you report on. If cell-level permission is turned off, then only row-level permission is applied, meaning the report will include everyone in the target population of the user running the report. If cell-level permission is enabled, then the row level will still include all the users in your target population, but then restrict what data you see in that cell in the row for the targeted user.

Row-level permissions include historical and future-dated data. For period reporting = Date Range reports should be used if you want to see all records in a period. For example, you have not enabled cell-level permission, and a manager wants to run a Compensation report to see the pay component data of their direct reports. If the manager's manager is in the target population, then they will see their manager's pay component data. If, however, cell-level permission is enabled, then further restricting based on RBP, the manager will still see the columns but for the row where their manager comes, there will be no values returned in those cells.

### Cell and Field Level Permissions

In Table reports, the cell level and field level permissions are supported only for the Employee Profile domain and not for the other domains of the Employee Central schema.

The cell level and field level permissions are supported for the Employee Central schema only in Canvas reports (Advanced Reporting).

## 4.2 Ad Hoc Report Query Trimming

All Ad Hoc reports are now enabled automatically when the corresponding module is enabled.

To reduce the size of SQL query, which helps reduce the query parsing time, the system is set for Ad Hoc Query Trimming, which is enabled by default. It can be disabled in Provisioning if required.

AdHoc Query Trimming supports only the following Ad Hoc Reports (Employee Central):

- Person and Employment (As of Date)
- Job Information (Date Range)
- Recurring Compensation Information (Date Range)
- Non-Recurring Compensation (Date Range)

# 5 Data Models

Data Models describe how data elements are structured in a database. They also define the properties these elements possess and their relationships to each other.

For more information about data models, refer to the [SAP SuccessFactors Data Model Reference Guide](#).

For more information about fields in the data model, refer to the [Data Object Tables in Employee Central](#).

# 6 Working with Foundation Objects

## 6.1 Introduction to Foundation Objects

Foundation objects are used to set up data that can be shared across the entire company, such as job codes, departments, or business units. Foundation objects are sometimes referred to as “foundation tables”.

Foundation objects are the first objects you should load because some of the lists of values proposed in employment information come from the Foundation Objects.

You can use Foundation Objects to populate data at the employee level. For example, if you assign a job code to an employee, that employee's record is then populated with all information based on the attributes of the job code. Additionally, the relationships that are configured between the Foundation Objects can be used to filter the lists of values in Employment Information. For example, the list of pay components that are selectable on an employee's record can be filtered based on the country the employee is associated with as determined by the employee's Legal Entity.

Some Foundation Objects are predelivered for you in the Corporate Data Model. For a list of these object, refer to [Predelivered Foundation Objects in Corporate Data Model](#) in the SAP SuccessFactors Data Model Reference Guide on the SAP Help Portal.

## Importing Foundation Objects

You need to import the data into the system using different methods and in a specific order. The import methods are as follows:

- Foundation Data Imports
  - Managed through the Corporate Data Model using the [Import Foundation Data](#) page
  - Managed through MDF using the [Import and Export Data](#) [Import Data](#) page
- Position Management Imports
- Employee Data Imports

The order of imports into Employee Central is critical. The Basic Import gets the employee started in the system and each of the subsequent imports populate a different block in the employee's file. For example, the third import, Employment Details Import, populates the Employment Details block. At minimum you have to import the six following files:

1. Basic Import
2. Biographical Information Import
3. Employee Details Import
4. Job History Import
5. Compensation Information Import
6. Personal Information Import

## Accessing Foundation Objects

- You create and update Legacy Foundation Objects in the Corporate Data Model. To manage Legacy Foundation Object data, choose  [Admin Center](#)  [Manage Organization, Pay, and Job Structures](#).
- You create and update MDF Foundation Objects using  [Admin Center](#)  [Configure Object Definitions](#). To manage MDF Foundation Object data, choose  [Admin Center](#)  [Manage Data](#).
- Ad-hoc reports work based on both the migrated and Legacy Foundation Objects. For Advanced Reporting (ODS), the reports will be migrated when you first invoke the reports after migration.

## Migration to MDF (Existing Customers Prior to 2015)

Starting with the November 2014 release, Foundation Objects are being migrated to the Metadata Framework (MDF) in a phased manner and will now be referred to as MDF Foundation Objects or Generic Objects (GO). The original Foundation Objects will now be referred to as Legacy Foundation Objects. Migrated MDF Foundation Objects are no longer configured using the Corporate Data Model. Instead, the [Configure Object Definition](#) and [Manage Data](#) pages in the Admin Center are used.

For more information on the different MDF Foundation Objects, refer to the *Migrating to MDF Foundation Objects Guide* found on the SAP Help Portal.

Table 1: Foundation Objects Migrated to MDF

Foundation Object	Migrated in...
Cost Center	November 2014 Release
Business Unit	Q2 2015 Release
Department	Q2 2015 Release
Division	Q2 2015 Release
Legal Entity	Q2 2015 Release
Legal Entity Local	Q2 2015 Release
Job Function	Q4 2015 Release
Pay Group	Q4 2015 Release
Pay Calendar	Q4 2015 Release
Job Family (Deprecated)	Q4 2015 Release
Job Classification	Q4 2015 Release
Job Classification Local	Q4 2015 Release

## Related Information

[Importing Foundation Data](#)  
[Employee Data Import Process](#)

### 6.1.1 Characteristics of Foundation Objects

Here's a summary of the features available in foundation objects.

#### Features

- Each foundation object consists of one or more fields. Some of them are required if you use the relevant object.
- Each foundation object has a technical ID, called an hris-element-id. You cannot change this.
- Each standard field within a foundation object also has a technical field ID. You cannot change this.
- However, you can change the labels of the foundation objects and the fields each object contains. The label is the descriptor that appears on the user interface (UI).
- The order in which the fields are displayed on the UI is the same as the order in which you list them in the setup of the foundation object.

For Legacy FOs only: The start date will always appear at the top of the screen.

- You can decide whether a field actually appears on the UI and, if so, whether:
  - It is required or optional
  - It is read-only or whether users can change or edit it
- Every foundation object contains custom fields. These are empty fields you can use to handle data not covered by the standard fields.
- Many, but **not** all foundation objects, are "effective dated".
- For each foundation object, you can determine the relationship to other foundation objects through the use of "associations".
- The search criteria for foundation objects can only be string texts. They cannot be picklists or generic objects.

#### i Note

For example, if you configure the city field in the Corporate Data Model as a picklist for a Country X, you can't use city in the search criteria for location. If you do, you won't be able to search locations by city for Country X.

### 6.1.2 MDF vs. Legacy Foundation Objects

There are many similarities between MDF Foundation Objects and Legacy Foundation Objects. Both serve to provide foundational data that organizations can use to structure their companies. Both provide the ability to

store attributes on the object level that can be referenced or propagated to the employee's job and compensation records.

However, MDF and Legacy Foundation Objects are built on two separate platforms, which result in different ways of accessing, configuring, and managing the objects and corresponding data. Below is a table which summarizes the key differences between the two object types.

	Legacy Foundation Object	MDF Foundation Object
Configuring the Object	<a href="#">▶ Provisioning</a> <a href="#">▶ Corporate Data</a> <i>Model and CSF Corporate Data Model</i> <a href="#">▶ Definitions</a>	<a href="#">▶ Admin Center</a> <a href="#">▶ Configure Object</a> <i>Definitions</i>
Managing the Object Values/Data	<a href="#">▶ Admin Center</a> <a href="#">▶ Manage</a> <i>Organization, Pay, and Job Structures</i>	<a href="#">▶ Admin Center</a> <a href="#">▶ Manage Data</a>
Importing Object Values/Data	<a href="#">▶ Admin Center</a> <a href="#">▶ Import Foundation</a> <i>Data</i>	<a href="#">▶ Admin Center</a> <a href="#">▶ Import and Export</a> <i>Data</i>
Exporting Object Values/Data	<a href="#">▶ Ad Hoc Report</a> <a href="#">▶ Report Definition</a> <i>Type "Foundation Objects"</i>	<a href="#">▶ Admin Center</a> <a href="#">▶ Import and Export</a> <i>Data</i>
Mass Deleting Object Values/Data	Manual deletion only using <a href="#">▶ Admin Center</a> <a href="#">▶ Manage Organization, Pay, and Job Structures</a> Alternatively, you can import the foundation data and switch the "status" from "Active" to "Inactive"	<a href="#">▶ Admin Center</a> <a href="#">▶ Import and Export</a> Use the "Operation" column to indicate which data should be deleted upon import.
Permissions for the objects and data	Manage Foundation Object Types	MDF Foundation Objects
Custom Fields	You are limited to 20 of each type: <ul style="list-style-type: none"> <li>• String</li> <li>• Date</li> <li>• Decimal</li> <li>• Long</li> <li>• Number</li> </ul>	There is no limit to the number of custom fields you can create for MDF objects. In addition to the data types supported for Legacy FOs, there are additional field types available. For more information, refer to the <i>Implementing the Metadata Framework</i> guide on the SAP Help Portal.

In addition to the differences in maintaining the tables and data, there are vast differences in the supported functionality and capabilities of the two object types. All functionality that is supported for Legacy Foundation Objects is supported for MDF Objects (associations, field-level configuration, picklists, and so on). However, the opposite is not true – all functionality that is supported for MDF Objects is NOT supported for Legacy Foundation Objects. MDF Objects contain a plethora of additional supported capabilities, including the support of business rules, field-level permissions, and more. MDF Foundation Objects and MDF Generic Objects are created and maintained under the overall MDF platform.

For more information, refer to the *Migrating to MDF Foundation Objects* guide on the SAP Help Portal.

## Related Information

[Implementing the Metadata Framework \(MDF\)](#)  
[Migrating to MDF Foundation Objects](#)

### 6.1.3 Deleting a Foundation Object

To correct an error, you may need to delete a foundation object or a foundation object value at some point.

#### Context

If a Foundation Object value is required to be removed from the UI, it is vital to consider the following points:

- Do **not** delete a Foundation Object value that at some stage has been used in an employee's data. If a value should be removed from the UI, it is recommended to set the "Status" of the value to "Inactive" rather than delete it from the system. This allows for a proper audit of the data to be maintained in the system, and eliminates the risk of unexpected system behavior.
- If a value needs to be deleted from the system, first run a Person and Employment Export Ad Hoc report to determine if any users have the value associated with their Employee Central data. Certain objects, such as the Pay Component FO, will not allow for deletion of values if they have been added to employee data (both current and historical records).

#### i Note

SAP SuccessFactors does not support mass deletion of Legacy Foundation Objects. Legacy Foundation Objects must be deleted individually on the User Interface or mass-inactivated through import.

#### Procedure

1. To manually delete a foundation object value, complete the following:
  - a. Go to  [Admin Center](#)  [Manage Data](#).
  - b. Search for the object and select it.
  - c. Under *Take Action*, select *Permanently Delete Entry*.
2. To mass delete MDF Foundation Objects, use the Import and Export Data tool.
  - a. Go to  [Admin Center](#)  [Import and Export Data](#).
  - b. Choose *Export Data*, then select the object. Remove all values that do not need to be deleted.
  - c. Select *Export*.
  - d. For any remaining values that need to be mass-deleted, insert the key "Delete" in the [OPERATOR] column.
  - e. Validate the data and then import the data back to the system.

## 6.2 Associations

Associations define relationships between foundation objects.

For example, a business unit consists of several departments, so you would create an association of one business unit to many departments — a ONE-TO-MANY relationship. Whereas a location can only have one geozone associated with it — this is a ONE-TO-ONE association. The type of association restricts what the user can display or enter in Employee Central — for a ONE\_TO\_ONE association from location to geozone, for example, the user can enter exactly one geozone for a location on the UI.

The standard XML file for the Corporate Data Model already contains some associations. You can add more ONE\_TO\_MANY associations, or change the existing associations in the XML file if needed. Each association has a “driving object” that acts as the basis for the association.

Associations should always be configured to go from the lower-level object (the child object) to the higher-level object (the parent object). In other words, the association should be placed on the object you expect to be filtered based on the parent value selected. For example - Location (Child) to Legal Entity (Parent) requires that the association be placed on the Location object. Following this, in the UI, after selecting the Legal Entity, a filtered list of Locations will be available, based on the selection made in Legal Entity.

For Foundation Objects, you can only define a ONE\_TO\_MANY association and not a MANY\_TO\_ONE association. In most cases, the one object typically filters the many object. However, it is recommended that associations be modeled on the many object rather than the one object to achieve the required filtering behavior.

As an example, if we assume the job codes ENG01 and ENG02 are applicable to “Philadelphia” and you would like to filter job codes by location. Logically, this would be a MANY-TO-ONE relationship from the jobCode to the location. However, as only ONE-TO-MANY associations are supported, this would need to be configured as a ONE-TO-MANY association from jobCode to location. Once this association has been defined, the valid locations can be attached to the job codes in Employee Central when setting up the job codes on the UI.

### 6.2.1 Using Associations to Structure Your Business

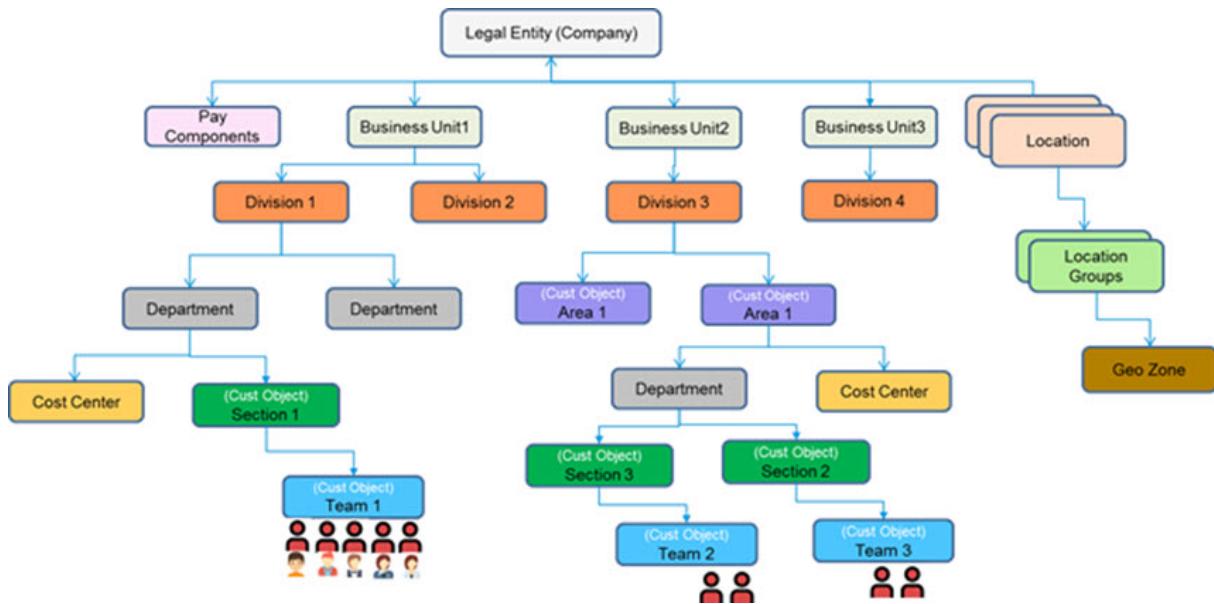
In this example of a company’s organization structure, you can see a range of different options for configuring and customizing the associations to accommodate different hierarchies.

In this example, we can see that there is a ONE-TO-MANY association between the following objects:

- Business Unit and Legal Entity
- Division and Business Unit
- Department and Division

We can also see the ONE-TO-ONE associations between the following objects:

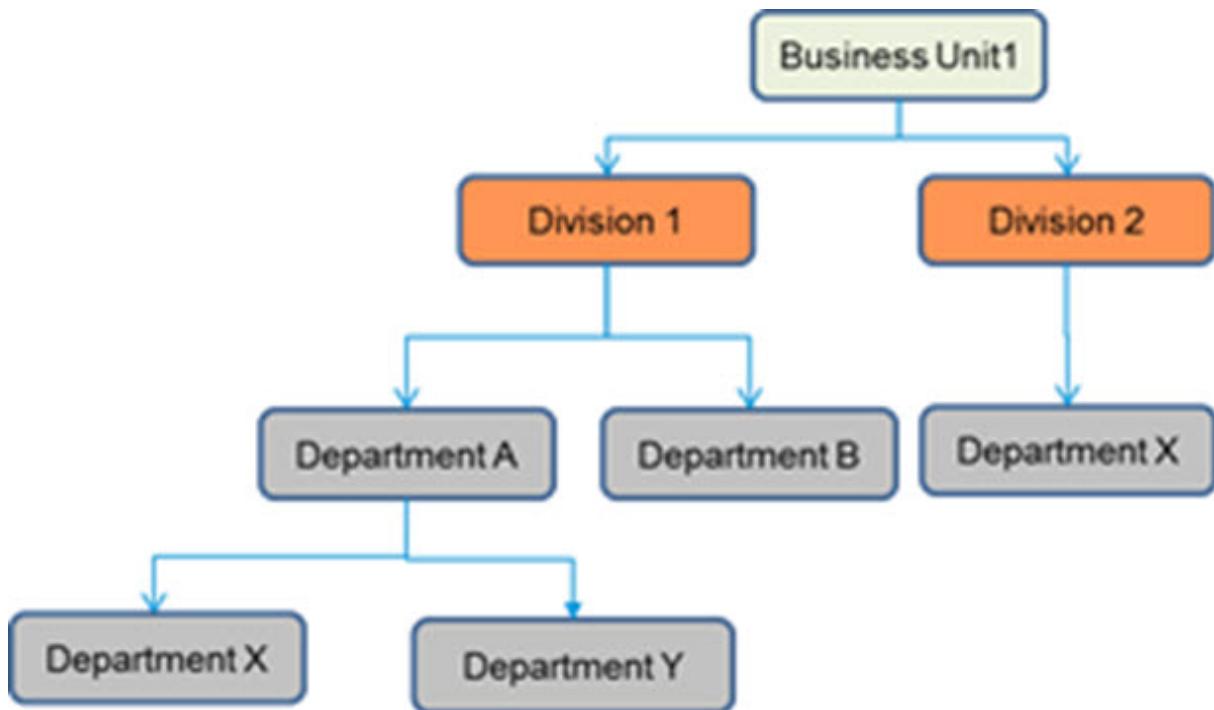
- Location Group and Location
- Geo Zone and Location Group



Should the standard Foundation Objects not account for the number of organization units that a company needs to define their org hierarchy, additional levels can be created through the use of custom MDF Generic Objects. In this example, we can see the following ONE-TO-MANY associations to custom objects:

- Area to Division
- Section to Department

An additional option available in constructing the Foundation Object associations is to build an association against the same object. For example, if a larger department is divided into sub-departments, a parent-child association can be created against the department object. The benefit of constructing this parent-child relationship is that it does not drive any restrictions when drilling down the hierarchy.



## 6.2.2 Examples of Foundation Object Associations

This table lists several examples of associations to show the relationships between foundation objects.

### i Note

References to Department, Division, Legal Entity and Business Unit in these examples now point to the MDF foundation objects.

Source	Target	Multiplicity	Description
Location	Geozone	ONE_TO_ONE	A location can only belong to one geozone. (Location is the driving object.)
Location	Legal Entity	ONE_TO_MANY	Several companies can have the same location. (Legal Entity is the driving object.)
Division	Business Unit	ONE_TO_MANY	A division can be associated with several business units. (Business Unit is the driving object.)
Department	Division	ONE_TO_MANY	A department can be associated with multiple divisions. (Division is the driving object)
Job Code	Business Unit	ONE_TO_MANY	A job code can be used across several business units. (Business Unit is the driving object.)
Pay Range	Geozone	ONE_TO_ONE	Companies generally have different pay ranges for each combination of Legal Entity, Job Code, and Geozone.
Pay Range	Pay Grade	ONE_TO_ONE	A pay range is generally associated with one pay grade.
Pay Range	Legal Entity	ONE_TO_ONE	Companies generally have different pay ranges for each combination of Legal Entity, Job Code, and Geozone.
Pay Component Group	Pay Component	ONE_TO_MANY	A pay component group can contain multiple pay components.

## 6.2.3 Configuring Associations

You need to fully understand the relationships between foundation objects in order to define them correctly in the system.

With the migration of Foundation Objects to MDF FOs, the HRIS elements of the migrated objects are no longer available in the Corporate Data Model XML as an association destination. This requires a solid understanding of the association requirements, in order to configure the association in the correct manner. Associations from Legacy FOs to other Legacy FOs are defined in the Corporate Data Model, whereas associations from MDF FOs to Legacy FOs or to other MDF FOs (or GOs) are defined in MDF (Configure Object Definitions). For associations from an MDF FO to a Legacy FO, associations cannot be directly defined. Instead, a wrapper MDF FO is used. A wrapper is not required for associations to custom FOs as these are considered to be GOs.

For more details of how to configure associations based on the object destination, see [Working with Associations, Field Criteria and Value Help \[page 110\]](#) topic.

## Creating Associations Between Different Blocks

You can also add an association to a field that is not part of the same block; for example, to filter the pay components on the compensation info block based on job info criteria. To do this, you have to add a prefix of the corresponding object as destination field value as in this example:

### Sample Code

```
<hris-element id="payComponentRecurring">
    <label>Compensation</label>
    <hris-field id="pay-component" visibility="both" required="true">
        <label>Pay Component</label>
        <field-criteria destinationFieldValue="jobInfo.payScaleGroup"
            sourceFieldName="PayScaleGroup"/>
    </hris-field>
    ...

```

Here, the pay component that is part of the payComponentRecurring block is filtered based on the field payScaleGroup from the job info block. To achieve this, you add the prefix `jobInfo.` to the destination field value.

The next few sections describe how you can get associations to work for the following scenarios.

- Legacy Foundation Object filtering another Legacy Foundation Object
- Foundation Object filtering a Generic Object
- Generic Object filtering a Foundation Object
- Generic Object filtering another Generic Object

The steps also describe optional configuration that is required only if you have position management enabled and need associations to work on the position and the employee record.

### 6.2.3.1 Filtering Custom Fields in Foundation Objects

Disable the filter for custom fields of type Foundation Object in the system.

#### Context

Custom fields using the attribute type="foundationObject" take over the association settings of the corresponding foundation object. For example, if you have a custom field of type="location", and you have associated the location FO with the legal entity FO, the custom field would only show a restricted list of values (where the legal entity defines which locations are displayed). However, if you prefer custom fields with type="foundationObject" to show all the FOs available in the system, you need to check the following setting:

#### Procedure

1. Go to  Admin Center  Company Settings  Company System and Logo Settings .
2. Under *Company System Settings*, select the checkbox *Disable filter for custom fields of type Foundation Object*. *To activate this setting, upload any data model in provisioning.*
3. Save your settings.

### 6.2.3.2 Configuring a Generic Object to Filter Another Generic Object

It is possible to use one generic object as a filter for another.

#### Context

To understand the steps involved, consider the following example of the Generic Object `cust_MarketCategory` being filtered by the Generic Object `cust_FunctionalArea`.

External Name Of Generic Object doing filtering	<code>cust_FunctionalArea</code>
Technical Name of field on Position referencing Generic Object doing filtering	<code>cust_FunctionalArea_field</code>
Technical Name of field on Position referencing Generic Object being filtered	<code>cust_jobMarketCategory_field</code>
External Name Of Generic Object to be filtered	<code>cust_jobMarketCategory</code>

## 6.2.3.2.1 Step 1: Configuring the Association to the Generic Object to be Filtered

Configure the association on the Generic Object to be filtered.

### Context

This allows you to attach the parent Generic Object doing the filtering (`cust_FunctionalArea`) to the child Generic Object being filtered (`cust_jobMarketCategory`). The steps are as follows:

### Procedure

1. Create the Generic Objects that will do the filtering and be filtered.
2. Go to  [Admin Center](#)  [Configure Object Definitions](#)
3. From the *Search* dropdown, select *Object Definition* and then select *Job Market Category* (the Generic Object to be filtered) from the dropdown next to it.  
The *Configure Object Definitions* page is displayed.
4. From the *Take Action* dropdown, select *Make Correction*.
5. Scroll to the *Associations* section at the bottom.
6. Select *Details*. We will now set the association for the child object, meaning the object to be filtered.
7. In the *Name* field, specify a name for the association.
8. In the *Multiplicity* field, select the type of association, either *One to One* or *One to Many*.
9. In the *Destination Object* field, select the Generic Object that will filter the values for this Generic Object on the UI.

### Results

A sample completed entry is shown below.

<code>mdfSystemProxyUser</code>	<code>proxyUser</code>	255	String	<a href="#">Details</a>
<code>mdfSystemRecordStatus</code>	<code>recordStatusStr</code>	255	Enum	<a href="#">Details</a>
<code>mdfSystemEffectiveEndDate</code>	<code>effectiveEndDate</code>	255	Date	<a href="#">Details</a>
<code>mdfSystemTransactionSequence</code>	<code>transactionSequence</code>	255	Number	<a href="#">Details</a>
<code>mdfSystemVersionId</code>	<code>versionId</code>	255	Number	<a href="#">Details</a>

#### Associations

Name	Multiplicity	Destination Object	Type	(8) More
<code>cust_functionalArea_to_jobMarketCategory</code>	One To Many	Functional Area	Valid When	<a href="#">Details</a>

#### Searchable Fields

## 6.2.3.2.2 Step 2: Attaching the Relevant Parent Generic Object Values to the Child Generic Object

Attach the relevant Parent Generic Object values to the Foundation Object.

### Context

This sets up which parent Generic Object values filter the child Generic object values.

### Procedure

1. Go to Admin Center Manage Data.
2. From the Create New dropdown, select *Job Market Category*.
3. In the *externalCode* field, specify the name of the child Generic Object value to be filtered.
4. In the *externalName* field, specify the external name for this object.
5. In the *effectiveStartDate* field, specify the start date.
6. From the *functionalArea* dropdown, select the parent Generic Object value doing the filtering. In this case, HR.
7. Save your changes.

### Results

The screenshot shows the 'Manage Data' screen for creating a new 'Job Market Category'. The 'externalCode' field is populated with 'HR1' and has a yellow callout pointing to it labeled 'The value to be filtered'. The 'functionalArea' dropdown is set to 'HR (HR)' and has a yellow callout pointing to it labeled 'The value(s) doing the filtering'. Other fields shown include 'externalName' (HR1) and 'effectiveStartDate' (01/01/1900).

### 6.2.3.2.3 Step 3: Defining Field Criteria for the Generic Object Field Being Filtered

In the Succession Data Model, define field criteria for the Generic Object field being filtered.

#### Context

This tells the system for this field what Generic Object is doing the filtering and the field that references it on Job Information.

#### Procedure

1. Define the field criteria for the Generic Object being filtered in the Succession Data Model.
2. Save your changes.

#### Results

```
<hris-field maxLength="256" id="custom-string19" visibility="both" type="cust_functionalArea">
  <label xml:lang="en-US">Functional Area</label>
</hris-field>
<hris-field maxLength="256" id="custom-string20" visibility="both" type="cust_jobFamily">
  <label xml:lang="en-US">Job Market Category</label>
<field-criteria destinationFieldValue="custom-string19" sourceFieldName="cust_functionalArea_to_jobMarketCategory.mdfSystemInternalCode"/>
</hris-field>
```

This is the technical name of the field on Job Information doing the filtering

This is the name of the association you configured on the Generic Object to be filtered.

This is the fieldname of the internal code on the Generic Object to be filtered.

#### i Note

The field name of the internal code on the parent Generic Object can be derived from the `<internalCode>` database field named found in the [Configure Object Definition](#) page.

#### i Note

The field-criteria attribute is not supported for the Country/Region-Specific Succession Data Model.

## 6.2.3.2.4 Step 4: Defining the Field Criteria for the Child Generic Object Being Filtered

Define field criteria for the child Generic Object being filtered.

### Context

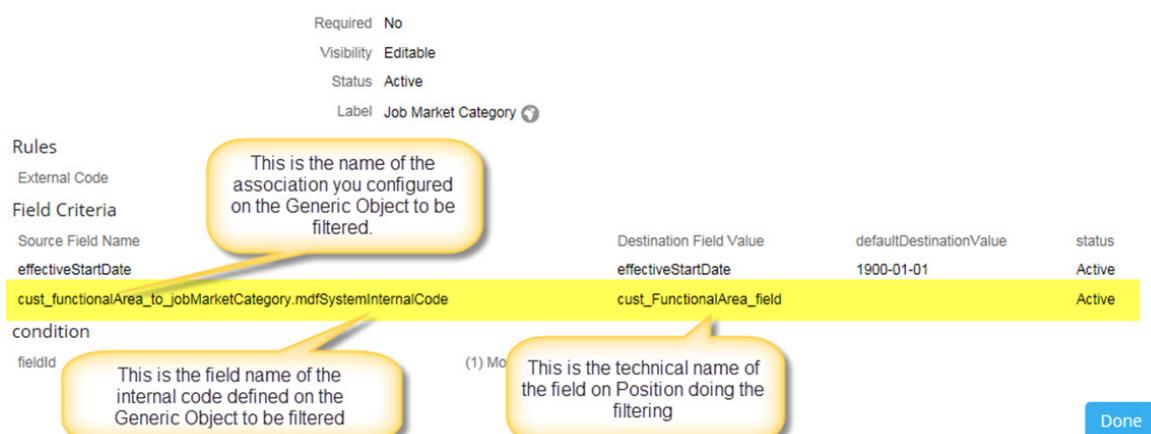
This step is to be done **only when Position Management is enabled**. This tells the system for this field what Generic Object is doing the filtering and the field that references it on the Position object.

#### i Note

Refer to the previous step for information on how to derive the internal code.

### Procedure

1. Go to  [Admin Center](#)  [Configure Object Definitions](#).
  2. From the *Search* drop-down list, select *Object Definition* and then select *Position* from the drop-down list next to it.
- The *Configure Object Definitions* page is displayed.
3. From the *Take Action* drop-down list, select *Make Correction*.
  4. In the *Fields* section, scroll to the *Generic Object* field to be filtered. In this case, *cust\_jobMarketCategory\_field*.
  5. Select the *Details* link to view the configuration.



### **6.2.3.3 Configuring a Generic Object to Filter a Foundation Object**

It is possible to use one generic object as a filter for a foundation object.

#### **Context**

To understand the steps involved, consider the following example where Legal Entity (Generic Object) is required to filter Location (Foundation Object) on Job Information and Position.

External Name of Generic Object doing filtering	LegalEntity
Technical name of field on Position referencing the Generic Object doing filtering	company
Technical name of Foundation Object field on Position being filtered	location
Technical name of Foundation Object field on Job Information being filtered	location
Field name on Job Information referencing Generic Object doing the filtering	company
Foundation Object to be filtered	location

#### **6.2.3.3.1 Step 1: Adding an Association to the Generic Object in the Foundation Object Element to Be Filtered**

In the Corporate Data Model, add an association to the Generic Object in the Foundation object element that is to be the subject of filtering.

#### **Context**

In this example, the association is added to the location.

```
<hris-associations>
| <association id="id" multiplicity="ONE_TO_MANY" destination-entity="LegalEntity" required="true" />
</hris-associations>
```

## **6.2.3.3.2 Step 2: Attaching the Parent Generic Object Values to the Foundation Object**

Attach the relevant parent Generic Object values to the Foundation Object.

### **Context**

This sets up Generic Object values that will filter the Foundation Object values.

### **Procedure**

1. Go to  [Admin Center](#)  [Manage Organization, Pay and Job Structures](#) .
2. From the *Search* drop-down list, select *Location* (the object whose values will be filtered) and then select the relevant location from the drop-down next to it. For this example, let's select *London*. The *Configure Object Definitions* page is displayed.

## Location: London (UK\_LON)

\* Effective as of 06/08/2015 \* Required Fields

Blue indicates that the item changed on this date

Foundation Object Value  
being filtered

\* Code UK\_LON

Standard Hours Click or focus to edit

Name London



Description London, UK



\* Status Active

Location Group EMEA (EMEA)

Timezone Europe/London (GMT+00:00)

company String1 No Selection

department String2 No Selection

division String3 No Selection

businessUnit String4 No Selection

CostCenter String5

Generic Object Value  
doing filtering

Legal Entity +

Ace UK (ACE\_GBR)

Business Address

\* Country United Kingdom

### **6.2.3.3.3 Step 3: Defining Field Criteria in the Succession Data Model for the Foundation Object Field being Filtered (For Job Information)**

Define field criteria for the foundation object field being filtered in the Succession Data Model.

#### **Context**

This specifies which Generic Object is doing the filtering and the field that references it on Job Information.

```
<hris-field maxLength="128" id="location" visibility="both">
  <label>Location</label>
  <label xml:lang="fr-CA">Emplacement</label>
  <label xml:lang="es-ES">Ubicación</label>
  <label xml:lang="fr-FR">Site</label>
  <label xml:lang="de-DE">Standort</label>
  <field-criteria destinationFieldValue="company" sourceFieldName="LegalEntity"/>
</hris-field>
```

Specify the external code of the Generic Object type doing the filtering

Type the field name specified in Job Information (field referencing the Generic Object doing the filtering)

#### **i Note**

The field-criteria attribute is currently not supported for the Country/Region-Specific Succession Data Model.

### **6.2.3.3.4 Step 4: Defining Field Criteria for the Foundation Object Field to be Filtered (For Position)**

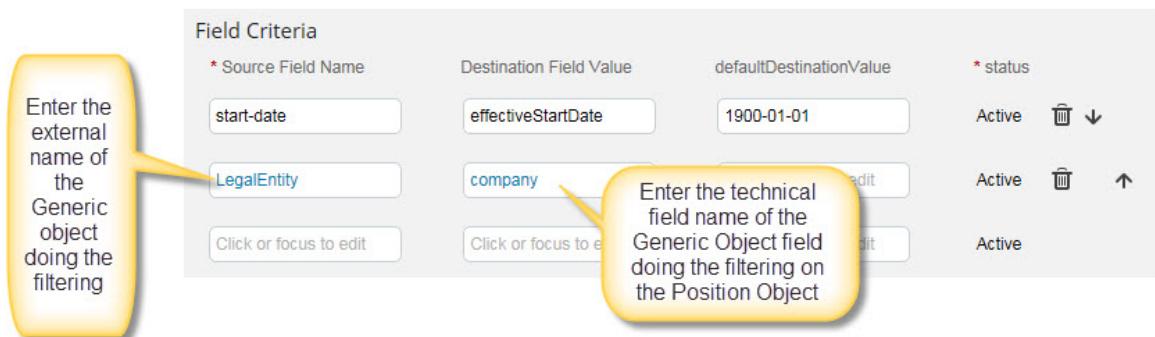
Define field criteria on the Foundation Object that is to be the subject of the filtering field on the Position Object.

#### **Context**

This step is to be done when the object doing the filtering is an MDF Object. This tells the system for this field which Generic Object is doing the filtering and the field that references it on Position.

## Procedure

1. Go to  [Admin Center](#)  [Configure Object Definitions](#).
  2. From the *Search* drop-down list, select *Object Definition* and then select *Position* from the drop-down next to it.
- The *Configure Object Definitions* page is displayed.
3. From the *Take Action* drop-down list, select *Make Correction*.
  4. In the *Fields* section, scroll to the Generic Object field to be filtered. In this case, *location*.
  5. Select the *Details* link to view the configuration.
  6. Scroll to the *Field Criteria* section.
  7. In the *Source Field Name* field, enter the external name of the Generic Object doing the filtering.
  8. In the *Destination Field Value* field, enter the technical field name of the Generic Object field doing the filtering on the Position object, for example, the technical name for company (GO doing the filtering)



Field Criteria			
* Source Field Name	Destination Field Value	defaultDestinationValue	* status
<input type="text" value="start-date"/>	<input type="text" value="effectiveStartDate"/>	<input type="text" value="1900-01-01"/>	Active  
<input type="text" value="LegalEntity"/>	<input type="text" value="company"/>	<input type="text" value="Click or focus to edit"/>	Active  
<input type="text" value="Click or focus to edit"/>	<input type="text" value="Click or focus to edit"/>	<input type="text" value="Click or focus to edit"/>	Active

9. Save your changes.

### 6.2.3.4 Configuring a Foundation Object to Filter Another Foundation Object

It is possible to use one foundation object as a filter for another foundation object.

To understand the steps involved, consider an example where: Location Group (Foundation Object) is required to filter Location (Foundation Object) on Job Information and Position.

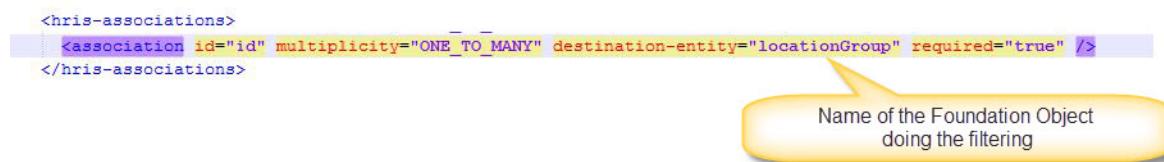
Name of the Foundation Object doing the filtering	Location Group
Name of the Foundation Object being filtered	Location

### 6.2.3.4.1 Step 1: Adding an Association to the Foundation Object to be Filtered

In the Corporate Data Model, add an association to the Foundation Object to be filtered.

#### Context

In this example, an association is added to the location.



### 6.2.3.4.2 Step 2: Attaching the Parent Foundation Object Values to the Foundation Object

Attach the relevant parent Foundation Object values to the Foundation Object.

#### Context

This attaches the relevant parent Foundation Object values to the child Foundation Object and allows you to specify which parent values filter which child values.

#### Procedure

1. Go to [Admin Center](#) [Manage Organization, Pay and Job Structures](#).
2. From the *Search* drop-down list, select *Location* (the object to be the subject of filtering) and then select the relevant Location values that will filter the chosen Location from the drop-down list next to the *Location* field. For this example, let's select ACE\_STO\_BE.  
The *Foundation Object* page is displayed.
3. Select *Insert New Record*.
4. Specify the *Location Group* for the child foundation object. This will update the page.

**Location: ACE\_STO\_BE (ACE\_STO\_BE)**

**Effective as of** 01/01/1990

Blue indicates that the item changed on this date

Code	ACE_STO_BE	Child Foundation Object to be filtered
Name	ACE_STO_BE	
Description	Berlin Office	
Status	Active	
Location Group		
	Timezone	Europe/Berlin (GMT+01:00)
	Geo Zone	
Location Group		
APAC (APAC)	Parent Foundation Object doing the filtering	
Business Address		
	Country	Germany
	Address 1	Unter den Linden 10
	Address 2	
	Town	Berlin
	Postal Code	10000

5. Save your changes.

### 6.2.3.4.3 Step 3: Defining Field Criteria in the Succession Data Model

Define field criteria in the Succession Data Model (SDM) for the Foundation Object field being filtered (in this case, location).

This step is to be done for Job Information as it is maintained in the Succession Data Model. For the Position object, see the next step.

Here, we are using a custom field in the field criteria, <custom-string2>, to refer to <locationGroup> as locationGroup is not a standard field of Job Information.

```
<hris-field maxLength="128" id="location" visibility="both">
    <label>Location</label>
    <label xml:lang="fr-CA">Emplacement</label>
    <label xml:lang="es-ES">Ubicación</label>
    <label xml:lang="fr-FR">Site</label>
    <label xml:lang="de-DE">Standort</label>
    <field-criteria destinationFieldValue="custom-string2" sourceFieldName="locationGroup"/>
</hris-field>
```

## 6.2.3.4.4 Step 4: Defining Field Criteria for the Foundation Object Field to Be Filtered (For Position)

Define field criteria for the Foundation Object that is to be the subject of the filtering field on the Position Object.

### Context

This step is to be done when the object doing the filtering is an MDF Object. We just defined the field criteria for the FO that is the subject of filtering for Job Information. We will now do the same for the Position Object. This tells the system for this field which Generic Object is doing the filtering and the field that references it on Position.

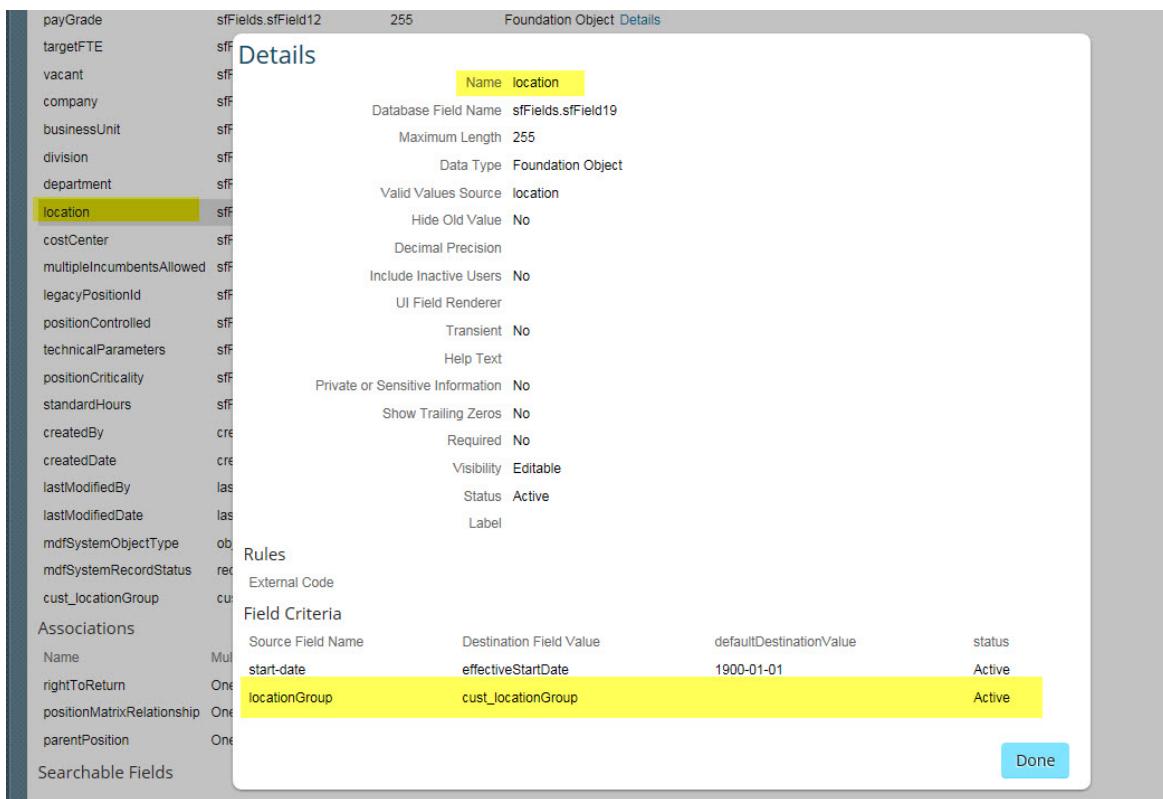
### Procedure

1. Go to  [Admin Center](#)  [Configure Object Definitions](#).
2. From the *Search* drop-down list, select *Object Definition* and then select *Position* from the drop-down next to it.  
The *Configure Object Definitions* page is displayed.
3. From the *Take Action* drop-down, select *Make Correction*.
4. In the *Fields* section, scroll to the *Generic Object* field to be filtered. In this example, *cust\_locationGroup*.

#### Note

For the example, we assume that you have already created a custom field by the name of *cust\_locationGroup* which is of type Foundation Object.

5. Select the *Details* link to view the configuration.
6. Scroll to the *Field Criteria* section.
7. In the *Source Field Name* field, enter the external name of the Foundation Object doing the filtering.
8. In the *Destination Field Value* field, enter the technical field name of the Foundation Object field doing the filtering on the Position object.



- Save your changes.

### 6.2.3.5 Configuring a Foundation Object to Filter a Generic Object

It is possible to use one foundation object as a filter for a generic object.

#### Context

To understand the steps involved, consider the following example where the Foundation Object Pay Grade is required to filter the Generic Object Grade Level on Job Information and Position.

Name of the Generic Object to be filtered	cust_GradeLevel
Technical Name of Field on Position of Generic Object being filtered	cust_GradeLevel_field
Technical Name of Field on Position of Generic Object doing filtering	payGrade

Name of the wrapper Generic Object required to connect the Generic Object to be filtered to the Foundation Object (Note that this is a pre-delivered wrapper. It is important that you do not create or use a custom wrapper.)	FOWPayGrade
Name of the association to the object doing the filtering configured on the Generic object to be filtered (Note that this association name follows the recommended naming guideline. For ease of use, it is suggested that you follow the same protocol: cust_to<Object External Code>.	cust_toFOWPayGrade
Element name of the Foundation Object doing the filtering in the Corporate Data Model	payGrade

### 6.2.3.5.1 Step 1: Associating the Generic Object Wrapper to the Child Generic Object

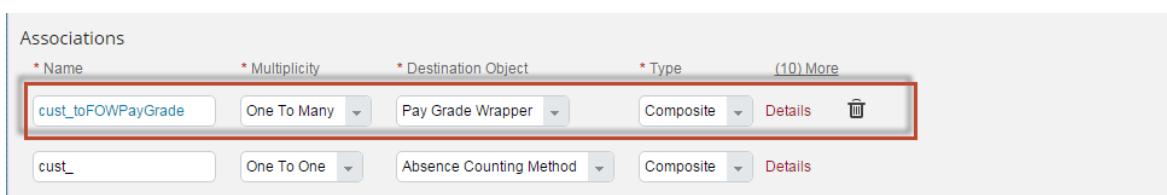
Associate the wrapper to the child generic object, which means the field that should have its values filtered.

#### Context

In the example, we associate the wrapper to cust\_GradeLevel\_field.

#### Procedure

1. Go to [Admin Center](#)  [Configure Object Definitions](#) .
2. In the *Search* field, select *Object Definition* and then select *Grade Level* (the object whose values will be filtered) from the drop-down next to it.
3. From the *Take Action* drop-down, select *Make Correction*.
4. Scroll to the *Associations* section at the bottom and create an association between the wrapper (cust\_toFOWPayGrade) and the Generic Object (Grade Level).



The screenshot shows the 'Associations' configuration screen. At the top, there are fields for 'Name' (containing 'cust\_toFOWPayGrade'), 'Multiplicity' (set to 'One To Many'), 'Destination Object' (set to 'Pay Grade Wrapper'), 'Type' (set to 'Composite'), and buttons for 'Details' and a trash can icon. Below this, another row of fields shows 'cust\_' in the 'Name' field, 'One To One' in 'Multiplicity', 'Absence Counting Method' in 'Destination Object', 'Composite' in 'Type', and again 'Details' and a trash can icon. A red box highlights the first row of fields.

5. Save your changes.

## 6.2.3.5.2 Step 2: Associating Values to be Filtered

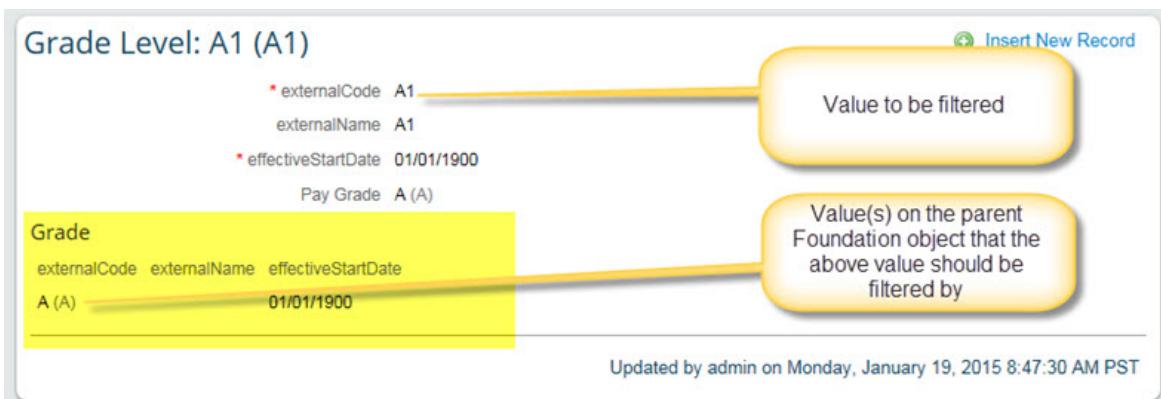
Associate values to be filtered to the values doing the filtering on the child Generic Object to be filtered.

### Context

With this, you configure the child values (Generic Object) that can be selected for specified parent values (Foundation Object).

### Procedure

1. Go to  [Admin Center](#)  [Manage Data](#).
2. Choose the relevant values to be filtered and attach the values doing the filtering to this object.



The screenshot shows the 'Grade Level: A1 (A1)' screen. On the left, there is a table with one row:

Grade	externalCode	externalName	effectiveStartDate
A (A)			01/01/1900

To the right of the table, two yellow callout boxes point to specific fields:

- A box labeled "Value to be filtered" points to the "externalCode" field of the Grade record, which contains "A1".
- A larger box labeled "Value(s) on the parent Foundation object that the above value should be filtered by" points to the "effectiveStartDate" field of the Grade record, which contains "01/01/1900".

At the top right of the screen is an "Insert New Record" button. At the bottom right, it says "Updated by admin on Monday, January 19, 2015 8:47:30 AM PST".

3. Save your changes.

## 6.2.3.5.3 Step 3: Defining the Field Criteria for the Generic Object Field Being Filtered

In the Succession Data Model, define the field criteria for the Generic Object field being filtered.

### Context

This configuration ensures that the association works on Job Information.

```

<hris-field maxLength="256" id="pay-grade" visibility="both">
  <label>Pay Grade</label>
  <label xml:lang="en-GB">Pay Grade</label>
  <label xml:lang="es-ES">Grado de remuneración</label>
</hris-field>
<hris-field maxLength="256" id="custom-string20" visibility="both" type="cust_GradeLevel">
  <label>Custom String 20</label>
  <label xml:lang="en-GB">Custom String 20</label>
  <label xml:lang="es-ES">Cadena personalizada 20</label>
  <field-criteria destinationFieldValue="pay-grade" sourceFieldName="cust_toFOWPayGrade.externalCode"/>
</hris-field>

```

This is the technical field name (in the Succession Data Model) of the field doing the filtering

This is the name of the association you previously created on the child Generic Object.

This is the field on the wrapper Generic Object which connects to the Foundation Object doing the filtering.

#### i Note

The field-criteria attribute is currently not supported on the Country/Region-Specific Succession Data Model.

### 6.2.3.5.4 Step 4: Defining Field Criteria for the Generic Object Field being Filtered

Define field criteria for the Generic Object field being filtered on the Position Generic Object.

#### Context

This step is to be done **only when Position Management is enabled**.

#### Procedure

1. Go to  [Admin Center](#)  [Configure Object Definitions](#)
2. From the *Search* drop-down, select *Object Definition* and then select *Position* from the drop-down next to it.  
The *Configure Object Definitions* page is displayed.
3. From the *Take Action* drop-down, select *Make Correction*.
4. In the *Fields* section, scroll to the *Generic Object* field to be filtered. In this example, *cust\_GradeLevel\_field*.
5. Select the *Details* link to view configuration.

**Details**

UI Field Renderer: Click or focus to edit

Transient: No

Help Text: Click or focus to edit

Private or Sensitive Information: No

Show Trailing Zeros: No

Default Value: Click or focus to edit

Required: No

\* Visibility: Editable

\* Status: Active

Label: Click or focus to edit

Rule: \* E

No Selection

Field Criteria

* Source Field Name	Destination Field Value	defaultDestinationValue	* status
cust_toFOWPayGrade.externalCode	payGrade	Click or focus to edit	Active
Click or focus to edit	Click or focus to edit	Click or focus to edit	Active

'cust\_toFOWPayGrade' is the name of the association you previously created on the child Generic Object to be filtered

'externalCode' here is the field on the wrapper Generic Object which connects to the Foundation Object doing the filtering

Things to note for this scenario:

- When adding an association from a migrated GO to a Foundation Object, you must use one of the pre-delivered wrappers like FOWPayGrade (Pay Grade Wrapper) instead of creating a custom wrapper. However if OData or API features are enabled, it is important that you do NOT use a pre-delivered wrapper for associations from other MDF objects to Foundation Objects. Pre-delivered wrappers can be identified by their names: '<FO Name> Wrapper' and their external code 'FOW<FOName>'.
  - If OData or API features are not enabled, it is possible to use pre-delivered wrappers as association destination at more than one MDF object types. To do this you must manually remove the field criteria for effective start date at the field "parent" of the FO wrapper type. There is the restriction that the FO instances cannot be filtered by the parent's effective start date.
  - If a pre-delivered wrapper type is not used as association destination of CostCenter by an OData or API customer, the pre-delivered wrapper type must be used to configure a Foundation Object to filter a Generic Object.
6. Save your changes.

## 6.3 Effective Dating

Effective dating means that information records capture time as part of the data that is stored in SAP SuccessFactors and the time element can be edited.

In the application, the HRIS fields “start-date” and “end-date” are used for effective dating. The “start-date” is usually uppermost on the UI. This is where the user has to enter the date from which the changes are effective. Whether an HRIS element is effective-dated or not is defined by the system.

The HRIS field “end-date” does not appear on the UI but is used for reporting purposes. For example, if you change an effective-dated field such as *Pay Grade* and set the date when the change should be effective to 01/01/2015, the system records 12/31/2014 as the end date in the background. If you run a report on the pay grade in the time from 01/01/2014 until 12/31/2014, the pay grade value that was valid in that time frame will be shown.

The system does not change the stored data. Instead, it creates a new row of data to track the new values from the effective date of the change, and continues to store the values that were effective before the change.

By default, the end date does not appear on the UI for MDF Foundation Objects, but it is possible to change the visibility of this field. To preserve the system functionality that automatically sets the end date, it is highly recommended to either leave the end-date field as hidden, or set to read-only. It is not recommended to manually set the end-date of Foundation Objects.

### 6.3.1 Changing a Legacy Foundation Object

Change a Foundation Object by inserting a new record to update the Foundation Object. You should never edit the object directly.

#### Context

Foundation objects are effective-dated in the same way as employee data. When updating an employee's job information, the process typically involves inserting a new record, effective on a specific date, updating the employee attributes, and then saving the record. The same applies to Foundation Objects.

This is an example of how you can update the name of a Location in your system.

#### Procedure

1. Go to  [Admin Center](#)  [Manage Organization, Pay and Job Structures](#).
2. The [Search](#) drop-down menu, select [Location](#)
3. Locate the Location object you want to amend and select it. Once it has loaded on the screen, select [Insert New Record](#).

- Set the start date for when the change should go into effect. By default, the date is today's date.

Ensure that the date you set is either today's date or a future-dated date. A best practice is to set these changes to happen over a weekend, when Job Information changes are not likely to be made.

**i Note**

It is important to know that if you set the date in the past, this could affect Job Information records that are using the older Location value but the effective start date of that Job Information record is after the Effective Start Date of the Location FO's changes.

- Save your changes.

## Next Steps

Once the Foundation Object has been updated, you will also need to add a new Job Information record to all employees that should have this updated Location. For example, if you updated the location's name from "Chicago" to "Chicago, USA", the system will not automatically propagate that change to Job Information, so you will need to update all users who have "Chicago" set as their Location in Job Information. This is because FO's are effective dated, and so are Job Information records.

**i Note**

If the employee's job information is not updated, you will still see the label update on view of the employee's job information. This is only a display feature. A new record should still be inserted in the Job Information of the user for the change to be reflected in Job History and synced to Employee Profile.

You can update the employee's Location manually using the UI, using the Mass Changes tool, or by importing a new record for the impacted employees.

Another example is the updating of a Business Address. For example, the company has moved their office at the location Chicago, from one address to a new address, and this address is shown in Employee Central or synced to Employee Profile. You would need to follow this full process to force the system to update the employee's Employee Central data.

## 6.4 Foundation Objects for Structuring Your Business

There are different types of foundation objects. You can use one of these types, called organization objects, to define how your business is structured.

### Organization Objects A-Z

Object	Object Type	Description
Legal Entity	MDF	<p>The legal entity table stores all the legal entities of a company. No legal entity can cover more than one country, so the country in the legal entity determines the country of employees assigned to the legal entity.</p> <p>This MDF FO can also store country/region-specific information for each legal entity in the country/region-specific child object. There are 5 pre-delivered countries: USA, DEU, ARG, ESP, FRA. For all other countries, custom child objects have to be defined by the name of <code>cust_LegalEntity&lt;country ISO code&gt;</code>.</p> <p>For more information on adding country/region-specific fields, refer to <a href="#">Adding a New Country/Region and Related Fields to LegalEntity [page 119]</a> in the next chapter of this guide.</p>
Business Unit	MDF	<p>The level of the organizational hierarchy lower than the Legal Entity. It is the Business area of the company, representing one operating unit or representing the business function within the Company (not geographical)</p>
Cost Center	MDF	<p>The cost center foundation object stores all the cost centers of a company. Cost Centers are usually defined in the ERP financial systems, and you simply load cost center info from those systems.</p>

Object	Object Type	Description
Division	MDF	The level of the organizational hierarchical structure lower than the Business Unit.
Department	MDF	It is usually necessary to divide a business into a number of departments, such as Sales and Marketing, Public Relations, and Dispatch. This Foundation Object enables you to do this.
GeoZone	Legacy	<p>You can group locations into one geo-zone. For example, you could create the GeoZoneEurope West, containing the locations UK, Netherlands, and Germany.</p> <p>This foundation object includes an <i>Adjustment Factor</i> field that where you enter a percentage to indicate the adjustment to the pay range for this GeoZone due, for example, to differences in the cost of living.</p> <p>For example, if you decide that people on the west coast of the US should be paid 10% more than those on the east coast and that the pay range for people on the east coast is \$100,000 - \$110,000, the pay range for people on the west coast is \$110,000 - \$121,000.</p>
Location	Legacy	The location Foundation Object stores address information for all the physical offices of a company. It supports international address formats.
		<p><b>i Note</b></p> <p>For customers using the Team Summary tile on the Home Page, the business address must also be configured so that an accurate location can be displayed.</p>

Object	Object Type	Description
Location Group	Legacy	<p>It is possible to combine locations into location groups under the Location Group Foundation Object. For example, you might want to group all your offices on the east coast of the United States in a location group labeled "US East Coast". By default, the Location Group object does not display on the employee's job information record, but is used strictly for reporting purposes.</p>

#### i Note

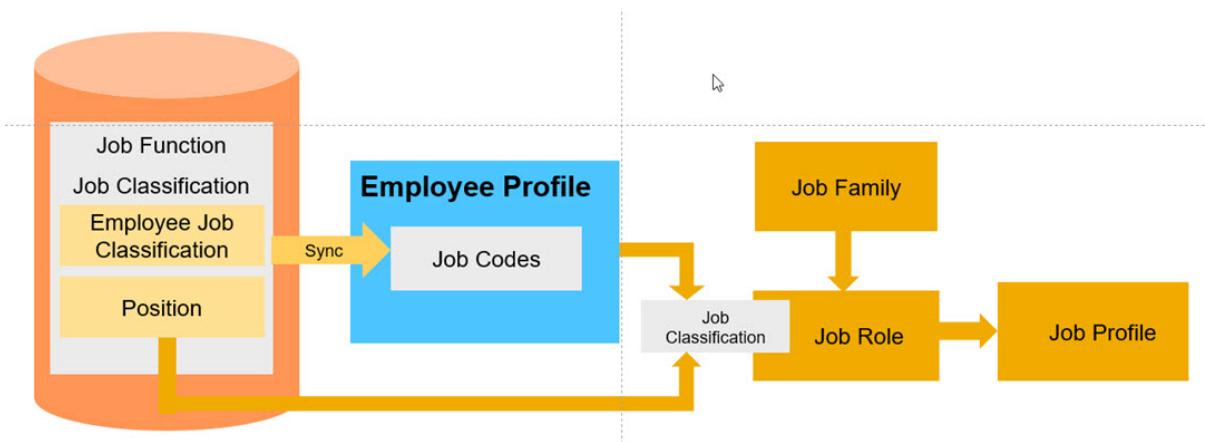
By default, Department, Division, and Location values are synced from Employee Central to Employee Profile for downstream talent processes. The values contained in these fields will also display in dashboards and UI views (such as the org chart and People Profile). For more information, see the [Human Resource Information System \(HRIS\) Synchronization \[page 190\]](#) topics.

## 6.5 Foundation Objects for Handling Job-Related Areas

Some of the foundation objects (FO) can be used to handle job-related issues.

### Job-Related Objects A-Z

Object	Object Type	Description
Job Classification	MDF	<p>The Job Classification object stores all job codes defined in a company and information associated with the job code. Companies can have one universal job classification, used for more than one country.</p>
Job Function	MDF	<p>Several Job Classifications can have the same Job Function. For example, the Job Classifications "Developer" and "Development Manager" could be associated to the same Job Function "Engineering". By default, the Job Function does not display on the employee's Job Information record, but is used strictly for reporting purposes.</p>



## 6.6 Foundation Objects for Handling Pay-Related Areas

Some of the foundation objects (FO) can be used to handle pay-related issues.

### Pay-Related Objects A-Z

Object	Object Type	Description
Pay Component	Legacy	<p>An employee's pay is comprised of more than one component, such as Basic salary, Target bonus, Company car allowance, and so on.</p> <p>For each component that will be maintained, a company needs to define attributes such as:</p> <ul style="list-style-type: none"><li>• Is the pay component recurring or one-time?<ul style="list-style-type: none"><li>◦ If the pay component is recurring, what is the frequency? This can be set directly on the pay component and propagated to the employee's record when the pay component is selected, or it can be derived from other attributes, such as the Pay Group.</li></ul></li><li>• Is the pay component an amount or a percentage?<ul style="list-style-type: none"><li>◦ If percentage, what is the percentage based on? For example, is it based on how much of a particular product the employee makes or sells?</li></ul></li><li>• Is the pay component actual pay or a target amount?</li><li>• Who has the ability to select or view the pay component? This can be controlled using RBP.</li><li>• Should the pay component be used by the Compensation module?</li><li>• Is the pay component taxable or non-taxable?</li></ul>

Object	Object Type	Description
Frequency	Legacy	Frequency is used by the PayComponent Foundation Object to determine how often a pay component is paid - for example, annually.
Pay Component Group	Legacy	It is possible to group pay components into pay component groups. The amount of a pay component group is equal to the sum of the pay components it includes. If the amounts in question are in different currencies or for periods of less than a year, the system automatically annualizes them and converts the currencies.
Pay Grade	Legacy	<p>Pay Grade is a Foundation Object related to Job Classification. A Job Classification is connected by default to a Pay Grade. This is optional and you can turn it off in the Corporate Data Model using the Grade field on the 'jobCode' element. To do this, simply set the visibility to "none".</p> <p>It can be used to identify when a transaction is lateral move, a promotion, or a demotion.</p>
Pay Range	Legacy	<p>Pay Range is primarily used for the calculation of Compa Ratio and Range Penetration. The system stores minimum, median, and maximum points of a pay range.</p> <p>Your company can define as many pay ranges as required. The range generally includes Pay Grade, Geozone, and Legal Entity and are updated every year.</p>
Pay Group	MDF	We recommend that you group people who share the same payroll-related attributes into one pay group. For example, employees in Europe who are all paid by SAP Payroll and paid bi-weekly can be grouped into one European Pay Group.

Object	Object Type	Description
Pay Calendar	MDF	The PayCalendar Foundation Object stores all the payroll periods within a year. For example, June 1 – June 15 2016 could be one payroll period.

For more information, see the [Implementing and Configuring Employee Payments in Employee Central](#) guide on the SAP Help Portal.

## 6.7 Other Foundation Objects

There are additional foundation objects that you can use.

### Overview

In addition to the organization objects, job-related objects, and pay-related objects, you can also use the following (most are related to workflows):

Object	Object Type	Description
Event Reason	Legacy	Employee Central uses event reasons to determine which HR event has taken place when employee data is changed and why. For example, the event "Termination" can take place either because the employee's performance was not satisfactory, or because the employee wanted to change company. In this case, two event reasons can be created: "Involuntary Termination -Performance Issues", or "Voluntary Termination – By Employee". You can create as many event reasons for an event as needed. These event reasons can then be used for reporting and analytics.
Workflow	Legacy	The Workflow FO allows for the ability to create and maintain workflow details, including approver steps, Contributors, and CC roles.

Object	Object Type	Description
Dynamic Role	Legacy	A Dynamic Role is one of the approver types available in setting up of approval workflows for changes to employee data. This allows the system to find the approver dynamically based on the employee's FO attributes.

# 7 Working with MDF Foundation Objects

## 7.1 MDF Foundation Objects

As part of the phased migration of Foundation Objects (FO) to the Metadata Framework (MDF), the following Foundation Objects are now MDF Foundation Objects (also referred to as GOs). Any organizational information configured using these FOs will now be configured using the corresponding MDF FO.

This Foundation Object...	Was migrated in this release...
Cost Center	Q4 2014 Release
Department	Q2 2015 Release
Division	Q2 2015 Release
Business Unit	Q2 2015 Release
Legal Entity	Q2 2015 Release
Legal Entity Local	Q2 2015 Release
Job Function	Q4 2015 Release
Pay Group	Q4 2015 Release
Pay Calendar	Q4 2015 Release
Job Family (Deprecated)	Q4 2015 Release
Job Classification	Q4 2015 Release
Job Classification Local	Q4 2015 Release

As part of the migration:

- The object definitions for these FOs have also been migrated from the Corporate Data Model to MDF. As a result, the migrated Foundation Objects will no longer be configured in the Corporate Data Model. Instead, the [Configure Object Definitions](#) page will be used to configure these MDF Foundation Objects and the [Manage Data](#) page will be used to manage these MDF Foundation Objects .
- The currency and country fields of the Legal Entity FO are now GOs. Any references to these fields will now refer to the corresponding GO.
- All instances of these objects and related data such as associations, translations, and audit data have been migrated to the respective entities in MDF.
- Picklists referenced by the fields on these objects are migrated to MDF picklists and kept in sync with the ECV2 picklist. Cascading picklists have also been migrated.

- Rules with references to the following FOs have been migrated to rules referencing their corresponding GOs: costCenter, businessUnit, division, department, company, jobCode, jobFunction, jobFamily, payGroup, and payCalendar. For example, a rule referring to the FO costCenter now refers to the GO CostCenter.
- Open workflows for the migrated FOs, mentioned above, have been migrated to MDF-based workflows for further processing.
- Permissions for the new migrated FOs have been migrated to the permission category for migrated Foundation Objects and stored as MDF permissions.
- Existing ad-hoc reports now work based on the migrated Foundation Objects. For Advanced Reporting (ODS), the reports will be migrated when first invoked after migration.
- If the FO jobFamily was previously configured in the Corporate Data Model, it has been migrated to the custom object type *cust\_JobFamily1511* to enable you to continue using the FO Job Family after migration. If you have not been using it, it will not be available. Note that there is no pre-delivered GO for Job Family since it has been deprecated.

## Related Information

[Implementing the Metadata Framework \(MDF\)](#)

[Migrating to MDF Foundation Objects](#)

## 7.2 Adding Search Criteria

You can add terms at the object under which they can be found in the search.

### Procedure

1. Go to  [Admin Center](#)  [Configure Object Definitions](#).
2. In the search field, select *Object Definition* and then select the object for which the criteria is to be defined. For this example, let's select *Business Unit*. The Object Definitions page now displays the current configuration for the GO BusinessUnit.
3. Select  [Take Action](#)  [Make Correction](#).
4. Scroll down to the *Searchable Fields* section.

Note that while the search criteria will appear blank, the *externalCode* and *name* fields are already implicitly defined as part of the search criteria. These are default search keys and do not need to be manually configured. In the empty text box, specify the name of any other field you would like to make searchable. You can choose from the list of fields mentioned in the *Fields* section.

### i Note

Default search-criteria for the fields **externalCode** and **name** are defined by default and do not need to be manually configured.

For the fields of type GO and Picklist, the field needs to be added in the search criteria. For example, if department is a field pointing to a Department GO, then department.name would need to be added to the search criteria.

5. Save your changes.

## 7.3 Working with Associations, Field Criteria and Value Help

With the migration of Foundation Objects (FOs) to MDF Foundation Objects (GOs), the HRIS elements of the migrated objects are no longer available in the Corporate Data Model XML as an association destination. Instead, associations from the GOs to another FO or GO are now defined in MDF. For associations from a GO to a FO, associations cannot be directly defined. Instead, a wrapper GO is used. A wrapper is not required for associations to custom FOs as these are considered to be GOs.

The table below describes the different associations possible. Here mFO refers to the MDF FO; cGO refers to a custom FO; FO refers to Foundation Objects defined in the Corporate Data Model.

Association before Migration	Is defined in	Association after migration	Is defined in	Details
mFO – cGO	Corporate Data Model	mGO – cGO	Metadata Framework	
mFO – FO	Corporate Data Model	mGO – FO using WrapperGO	Metadata Framework	Here you cannot have a direct association. Therefore, a WrapperGO is created during migration. The wrapper instances are created and association data is migrated.
cGO – mFO using custom WrapperGO	Metadata Framework	cGO – mGO using custom WrapperGO	Metadata Framework	The data type of the custom wrapper's external code is set to GO.
FO – mFO	Corporate Data Model	FO – mGO	Corporate Data Model	Here FO is changed to GO in the association definition.
mFO – mFO	Corporate Data Model	mGO – mGO	Metadata Framework	Defined in <a href="#">Configure Object Definitions</a>

Association before Migration	Is defined in	Association after migration	Is defined in	Details
mGO - mFO using wrapper GO	Metadata Framework	mGO - mGO	Metadata Framework	page. The association type is valid-when.

### ❖ Example

Association from FO costCenter to an FO or GO defined in the Corporate Data Model before the migration:

```
<hris-associations>
  <association id="id" multiplicity="ONE_TO_MANY" destination-entity="location"
required="false"/>
  <association id="id" multiplicity="ONE_TO_MANY" destination-
entity="cust_GOSubDivision"
required="false"/>
</hris-associations>
```

After the migration, the association to FO Location is migrated to the MDF association with name `cust_toFOWLocation` and destination object type `FOWLocation`. Here, `FOWLocation` is the wrapper GO for the FO Location. The association to the wrapper GO is modeled as Type "Composite" and Multiplicity "One To Many". The association to the custom FO Sub Division (`GOSubDivision`) will be modeled as an association of Type "Valid When" and Multiplicity "One To Many".

### ❖ Example

Association from FO to FO costCenter defined in Corporate Data Model before the migration:

```
<hris-associations>
  <association id="id" multiplicity="ONE_TO_ONE" destination-entity="geozone"
required="false"/>
  <association id="id" multiplicity="ONE_TO_MANY" destination-entity="company"
required="false" />
  <association id="id" multiplicity="ONE_TO_MANY" destination-
entity="costCenter" required="false" />
</hris-associations>
```

Association from FO to GO CostCenter defined in the Corporate Data Model after the migration:

```
<hris-associations>
  <association id="id" multiplicity="ONE_TO_ONE" destination-entity="geozone"
required="false" />
  <association id="id" multiplicity="ONE_TO_MANY" destination-
entity="LegalEntity" required="false" />
  <association id="id" multiplicity="ONE_TO_MANY" destination-
entity="CostCenter" required="false" />
</hris-associations>
```

## • Example

If you have implemented a GO with composite association to cost center, you must define an association from the GO to costCenter FO. For that you must implement a wrapper GO as proxy for the costCenter FO. After the migration, the wrapper GO will be the proxy for the GO CostCenter. If the wrapper GO is not used for other purposes, we recommend that you change the association definition at the GO and have GO CostCenter as the association destination instead of the wrapper GO.

Before the migration, cost center was an FO, and it is now a GO. If you want to maintain the GO CostCenter assignments, you can use the [Manage Data](#) page.

All the filtering on Job Information and Position works as before. Now you have a new element called [Field Criteria](#):

Source Field Name	Destination Field Value	defaultDestinationValue
start-date	effectiveStartDate	1900-01-01
CostCenter	costCenter	Click or focus to edit

Earlier, the value help on custom-defined fields would automatically filter out associated Generic Objects. After migration, the field criteria can be used to change the value help behavior as to which field shall be filtered as child field.

You can restrict the value list of the GO source depending on the GO/FO destination selection, while associating GO source to a GO/FO destination. If FO is the association destination, you perform this task using a GO Wrapper.

If you want to filter an FO-related field by a GO-related field, you define a [One-to-Many](#) association at the FO HRIS element type in the Corporate Data Model and enter the GO type as the association destination.

You must add field criteria in Job Information at the field that is filtered. Earlier, the element [Field Criteria](#) was not required, and the parent/child field relationship was reversed.

### • Example

On Job Information, the cost center field is filtered by business unit and a custom field `custom-string2` referring to GO `cust_GCC`:

```
<hris-field maxLength="256" id="cost-center" visibility="both">
    <label>Cost Center Account</label>
    <label xml:lang="de-DE">Kostenstellenkonto</label>
        <field-criteria
            sourceFieldName="cust_toBusinessUnit.internalId"
                destinationFieldValue="business-unit" />
        <field-criteria
            sourceFieldName="cust_tocust_GCC.mdfSystemInternalCode"
                destinationFieldValue="custom-string2" />
</hris-field>
```

### • Example

On Job Information, location field is filtered by cost center field:

```
<hris-field maxLength="128" id="location" visibility="both">
    <label>Location</label>
        <field-criteria sourceFieldName="CostCenter" destinationFieldValue="cost-
center"/>
</hris-field>
```

For the migrated FOs Business Unit, Division, Department and Legal Entity, FO Wrapper types are now deprecated. You must not use them anymore. If Cost Center has an association to an FO Wrapper, it will be migrated to the mapped GO and association type will be changed to [valid-when](#). This is applicable for associations to Business Unit, Division, Department and Legal Entity only.

A few scenarios are explained below.

### • Example

If the Department is restricted, the field criteria is always defined at the restricted field. The field criteria in this case will be as follows:

The source field name must be in the format `<association name>.internalId` and `destinationFieldValue` will be in the format `<filteringFieldID>`. The destination field will be the field name in the Succession Data Model.

For example, if the business unit filters the division, the field criteria defined on the division field looks as follows:

```
<field-criteria sourceFieldName="cust_toBusinessUnit.internalId"
    destinationFieldValue="business-unit" >
```

## • Example

If there is an association from Business Unit to Location, a wrapper will be required for the association. Additionally, if there is an association from Business Unit to Cost Center, it will be a direct association since this is an mGO - mGO association:

The screenshot shows the 'Associations' configuration interface. It displays two rows of association settings. The first row for 'cust\_toFOWLocation' has 'Name' as 'cust\_toFOWLocation', 'Multiplicity' as 'One To Many', 'Destination Object' as 'FOW Location', and 'Type' as 'Composite'. The second row for 'cust\_toCostCenter' has 'Name' as 'cust\_toCostCenter', 'Multiplicity' as 'One To Many', 'Destination Object' as 'Cost Center US', and 'Type' as 'Valid When'. There are also 'Details' and 'Delete' buttons for each row.

The field criteria defined on the business-unit field looks as follows:

```
<field-criteria sourceFieldName="cust_toFOWLocation.internalId"
destinationFieldValue="location" >
```

## Creating Associations Between Different Blocks

You can also add an association to a field that is not part of the same block; for example, to filter the pay components on the job information block. To do this, you have to add a prefix of the corresponding object as destination field value as in this example:

### ↳ Sample Code

```
<hris-element id="payComponentRecurring">
  <label>Compensation</label>
  <hris-field id="pay-component" visibility="both" required="true">
    <label>Pay Component</label>
    <field-criteria destinationFieldValue="jobInfo.payScaleGroup"
      sourceFieldName="PayScaleGroup"/>
  </hris-field>
  ...

```

Here, the pay component which is part of the payComponentRecurring block is filtered based on the field payScaleGroup from the job information block. To achieve this, you add the prefix `jobInfo.` to the destination field value.

## 7.3.1 Adding Legal Entity to the Cost Center Object

You can add the legal entity field to the cost center foundation object to create associations between these objects. This is for new customers from 2020.

### Prerequisites

You have set the visibility of the Legal Entity field to Yes in the object definition. By default the visibility of the legal entity field is set to No, but it must be set to Yes to use this field.

### Context

There is no need to create a custom legal entity field or a custom association to the legal entity object within cost center.

### Procedure

1. Go to  [Admin Center](#) .
2. Select the cost center object to have the association.
3. Select [Insert New Record](#) and then enter a date for the change to be effective in the system.
4. Add the legal entity and save your changes.

This creates the association between the specific cost center and the specific legal entity.

5. Go to  [Admin Center](#) .
6. Select  [Employee Central](#) .
7. For the <cost\_center> field, select the [Details](#) link.
8. In the [Field Criteria](#) section, add the required information:
  - Destination Field Value: company
  - Source Field Name: legalEntity
9. Save your changes.

### Results

In the Job Information block in the employee profile, the cost center objects displayed in the dropdown list are filtered by the selected legal entity. The user can view and select only the cost center objects that are assigned to the selected legal entity using the new legal entity field in the cost center object.

## Next Steps

You can repeat these steps for the Position object as well.

## 7.4 Behavior Changes to Importing Foundation Objects

There are a few areas where imports to now MDF foundation objects differ to legacy foundation objects.

You can import updates for all the migrated FOs such as Cost Center, Business Unit, Division, Department, Legal Entity, and Legal Entity Local.

For example, cost center import updates the GO CostCenter instead of FO costCenter. The [Import and Export Data](#) page continues to be the entry point for cost center imports.

Scheduled Jobs will run as before, without any change. There are a few areas with change in behavior, but for critical areas, backward compatibility is maintained.

You will notice a change in behavior in the following areas:

- You can use the standard MDF imports page as an alternate entry page to import the migrated FOs. In this case, you need to use the MDF import template for all the objects, which has a different structure than that of the standard import.
- Import template is enhanced to support translations. You are not required to import translations through a separate MDF import to the GO FOTranslations.
- MDF import does not support localized format for numbers. Use the format specified in the UI or use the ECv2 Import template.
- Translations for fields on all migrated FOs are not imported to the MDF object FO Translations.
- Threshold field is not honored in case of import of all the GOs. MDF imports always run asynchronously.
- Synchronous mode validation and import are not supported for the GOs.
- End Date is not exposed in the enhanced template.
- When future-dated associated values are entered in the import file, no error message appears. Instead, it shows blank value.
- Quick validations in MDF are not supported.
- Full purge is recommended over Incremental Load. See the examples below:
- On the [Monitor Jobs](#) page, the *Job Type* column displays content related to MDF now:
- Error messages are displayed with the new field IDs, for example, `effectiveStartDate` instead of `start-date`.

### • Example

Delimit Cost Center by inserting new associations:

Let's say Location was associated to the Cost Center in the Corporate Data Model as a [One-to-Many](#) relationship, and we import Cost Center value CC1 in Incremental Mode, with associated locations as Loc1, Loc2, and Loc3 on January 15, 2001.

If we import new record for CC1 with no location values for January 1, 2014, original associations with location, that is Loc1, Loc2 and Loc3, will be retained.

This is different from the ECv2 foundation data import, where the import delimits CC1 with no associated locations.

### • Example

Delimit Cost Center by removing existing associations:

Suppose Cost Center has a *Valid When* association with a customer GO `cust_GO` with values GO1 and GO2. We import a Cost Center record CC1 with association GO1 on January 1, 2014 and another import with association GO2 on January 2, 2014, through incremental load.

On January 2, 2014, Cost Center record has both GO1 and GO2 associations. Here GO1 is carried forward to the next record, which was not the case in ECV2 foundation data imports.

### • Example

Delimit Legal Entity by removing existing associations:

Suppose Legal Entity has a *Valid When* association with a customer GO `cust_GO` with values GO1 and GO2. We import a Legal Entity record LE1 with association GO1 on January 1, 2014 and another import with association GO2 on January 2, 2014, through incremental load.

On January 2, 2014, Legal Entity record has both GO1 and GO2 associations. Here GO1 is carried forward to the next record, which was not the case in ECV2 foundation data imports.

### → Recommendation

If you want to enhance, for example, the CostCenter object definition in MDF, use the MDF Import for cost center. For more information about MDF imports, refer to the [Implementing the Metadata Framework \(MDF\)](#) guide on the SAP Help Portal.

## 7.5 Behavior Changes to Workflows

Prior to the November 2014 release, you could create multiple workflows on the same cost center instance.

With the November 2014 release, if a workflow has already been initiated on the cost center, no other transaction can be performed for that cost center. The same applies for the following MDF Objects with effect from the Q2 2015 release: Business Unit, Division, Department, and Legal Entity.

For more information on changes as a result of the migration to MDF, see the [Migrating to MDF Foundation Objects](#) guide on the SAP Help Portal.

## 7.6 Optional: Editing the Country/Region and Currency Objects

With the Q2 2015 release, the `country` and `currency` fields of Legal Entity have been replaced with the `Country/Region` and `Currency` GOs. As a result, HRIS elements with fields referring to countries/regions or currencies are now based on the Country/Region and Currency GOs. With these new GOs, you can now add new countries/regions and currencies, set them to 'inactive' as well as maintain associations.

## Context

All currencies have been migrated to the GO Currency. The currencies from GO Currency are now visible in places where currencies are used (for example, PayComponent).

Instead of deleting countries/regions and currencies, we suggest setting them to Inactive.

You can import a full set of countries/regions as well as currencies (includes translations) through predelivered files available in the [SAP Help Portal](#) under *Implementation*.

## Procedure

1. Go to [Admin Center](#) *Manage Data*.
2. In the *Search* field, select *Country/Region*.
3. In the field next to it, select the country/region for which you would like to view information.
4. To view the translated country/region name, select the [View Translations](#) icon next to the country name.
5. To view the currency associated with a country/region, select the [View Currency](#) icon next to currency name.

Likewise, you can manage the currency using this page as well. Instead of country/region, select currency and proceed. Select the [View Translations](#) icon next to the currency name to get a list currency name translations.

6. To remove a currency from the drop-down list, enter the currency name. From [History](#), select *Take Action* *Make Correction*.
7. Change the status of the currency to *Inactive*:
8. Save your changes and repeat for all currencies and countries/regions.

## 7.7 Adding a New Country/Region and Related Fields to LegalEntity

If you want to add a new country/region and the fields related to it, you need to create a new MDF object for those country/region-specific fields. Then you have to assign the new object as a child object to LegalEntity.

### 7.7.1 Step 1: Creating a New MDF Object for the Country/Region-Specific Fields

Create a new MDF object for the country/region-specific fields needed for your company.

#### Procedure

1. Go to  Admin Center  Configure Object Definitions.
2. From the *Create New* drop-down, select *Object Definition*.
3. In the *Code* field, specify a code for the new object. It is a good idea to follow the following naming convention: **`cust_LegalEntity<Country Code>`**.
4. In the *Effective Dating* drop-down, select *From Parent*.
5. In the *Label* field, provide a unique identifier.
6. Set *API Visibility* as required.
7. In the *Fields* section, select *Details* against *externalCode*.
8. Change the *externalCode* data type to *Auto Number* and set its *Visibility* to *Not Visible*.
9. Select *Done* to go back to the previous page.
10. Now, specify settings for the *externalName* field. Select *Details* next to *externalName*.
11. Set the *externalName* field visibility to *Not Visible*.
12. Define the custom-specific fields.
13. Save your changes.

#### Results

This will add a number of predefined MDF fields.

## 7.7.2 Step 2: Assigning the New Object to Legal Entity

Assign the new country object to Legal Entity to associate the two objects in the system.

### Procedure

1. Go to  [Admin Center](#)  [Configure Object Definitions](#).
2. From the *Search* drop-down, select *Object Definition*.
3. Select *LegalEntity* from the field next to it.
4. Select  [Take Action](#)  [Make Correction](#).
5. Scroll down to the associations section and add a new association. We suggest following this naming convention: **`cust_toLegalEntity<CountryCode>`**.
  1. Multiplicity: One to One
  2. Destination Object: your newly created object
  3. Type: Composite
  4. Details:
    1. Condition fieldID: `countryOfRegistration.code`
    2. Condition Values: <Country Code>
6. Save your changes.

## 7.8 Adding a New Country/Region and Related Fields to Job Classification

If you want to add a new country/region and the fields related to it for job classification, you need to create a new MDF object for those country/region-specific fields. Then you have to assign the new object as a child object to JobClassificationCountry.

### 7.8.1 Step 1: Creating a New MDF Object for the Country/Region-Specific Fields

Create a new MDF object for the country/region-specific fields needed for your company.

#### Procedure

1. Go to  [Admin Center](#)  [Configure Object Definitions](#).
2. From the *Create New* drop-down, select *Object Definition*.
3. In the *Code* field, specify a code for the new object. It is recommended that you follow this naming convention: **cust\_JobClassification<Country Code>**.
4. From the *Effective Dating* drop -down, select *From Parent*.
5. In the *Label* field, specify a unique name.
6. Set *API Visibility* to *Editable*.
7. In the *Fields* section, select *Details* for the *externalCode*.
8. Change the *externalCode* data type to *Number* and set its *Visibility* to *Not Visible*.
9. Set *Default Value* to *1*.
10. Select *Done* to go back to the previous page.
11. We will now specify settings for the *externalName* field. Select *Details* for *externalName*.
12. Set the *externalName* field visibility to *Not Visible*.
13. Define the custom fields.
14. Add a label to display on the UI.
15. Save your changes.

#### Results

This will add a number of pre-defined MDF fields.

## 7.8.2 Step 2: Assigning the New Object to JobClassificationCountry

Assign the new country/region object to JobClassificationCountry to associate the two objects in the system.

### Procedure

1. Go to  [Admin Center](#)  [Configure Object Definitions](#) .
2. From the *Search* drop-down, select *Object Definition*.
3. Select *JobClassificationCountry* from the field next to it.
4. Select  [Take Action](#) .
5. Scroll down to the associations section and add a new association. We suggest following this naming convention: **`cust_toJobClassification<CountryCode>`**.
  1. Multiplicity: One to One
  2. Destination Object: your newly created object
  3. Type: Composite
  4. Details:
    1. Condition fieldID: country.code
    2. Condition Values: <Country Code>
6. Save your changes.

## 7.9 Configuring Standard Fields

This example shows you how you can configure a field. For this example, we'll be configuring the standard field **`glStatementCode`** as a picklist.

### Procedure

1. Go to  [Admin Center](#)  [Configure Object Definitions](#) .
  2. In the *Search* field, select *Object Definition* and then select *Cost Center*.
- The *Configure Object Definitions* page now displays the current configuration for the GO CostCenter.
3. Select  [Take Action](#) .
  4. In the *Fields* section, scroll down to *glStatementCode* and select *Details*.
  5. Set the *Visibility* of this field to *Not Visible*.
  6. Select *Done*.

7. Now, add a custom field of data type picklist. To do so, scroll to end of the fields list and select *Details* against the *cust\_* field.

The *Details* page comes up.

8. In the **Name** field, specify a name for the picklist. Note that the name is automatically prefixed with *cust\_* when you move to the next field.
9. From the **Data Type** drop-down, select *Picklist*.
10. In the **Valid Values Source** field, specify the ID of the MDF picklist.
11. Fill out the rest of the fields in the form and select *Done* to save your changes.

# 8 Working with Generic Objects

## 8.1 Generic Objects

When the standard Legacy and MDF Foundation Objects are not enough to structure an organization's business, it may be necessary to build custom objects to maintain additional information and attributes. These custom objects are referred to as Generic Objects. You use generic objects for information and settings relating to the people working in the company.

Generic objects are created using the Metadata Framework. This guide concentrates on the use of generic objects in the context of Employee Central. For more information about the Metadata Framework, refer to the [Implementing the Metadata Framework](#) guide.

What is the difference between Foundation Objects and Generic Objects?

The key difference between Foundation Objects (both Legacy and MDF) and Generic Objects is that Foundation Objects are standard, pre-delivered, and preconfigured objects that are used to structure a company's Organization, Job, and Pay areas. All Foundation Objects were originally Legacy Foundation Objects, created and maintained using the Corporate Data Model. As the Foundation Objects are slowly migrating to MDF, we are seeing the conversion from Legacy to MDF Foundation Objects. As Employee Central continues to expand, all future objects are being developed on the Metadata Framework platform. Objects that are created on the Metadata Framework are known as Generic Objects. In fact, all MDF Foundation Objects are Generic Objects, as they are built on the Metadata Framework platform, and it is common to see the term "Custom Foundation Object" and "Generic Object" used in place of "MDF Foundation Object".

### Related Information

[Data Models Used in Onboarding](#)

[Implementing the Metadata Framework](#)

### 8.1.1 Standard Generic Objects

In addition to the standard Foundation Objects available, as Employee Central continues to expand, additional objects are available to help organizations run their businesses effectively and efficiently.

Here are some examples of standard Generic Objects that are used to configure and support Employee Central functionality.

- Country/Region
- Pay Calendar
- Pay Scale

- Position

## 8.1.2 Characteristics of Generic Objects

Here's a summary of the features available in generic objects.

### Features

Here's a survey of the characteristics of generic objects.

- Each object has a technical ID, which you cannot change.

There are different types of technical ID. Here are some examples:

- Tab element ID: If you include a generic object with a tab element ID in your Succession Data Model, the relevant tab is available for use in your installation. You need to configure permissions for them though (see below).
- Field ID: If you include a generic object with a field ID, that field is available for use in your installation. Again, you need to give each user the permissions they need to use the field.

Each object has a label, which you can change to suit your requirements. It is possible to maintain this label in different languages if you need to.

- You have to enable generic objects in your system before you can use or see them. You do this by checking the Enable Generic Objects feature in Provisioning.

#### → Remember

As a customer, you don't have access to Provisioning. To complete tasks in Provisioning, contact your implementation partner. If you're no longer working with an implementation partner, contact SAP Cloud Support.

- You have to set permissions for generic objects, which determine who can use them and what they can do with them.
- In the case of field IDs, you can decide whether each field appears in your UI and, if so, whether it is for display only or whether users can change or edit the information in it.

## 8.2 Creating Customer-Specific Foundation Objects

Some customers may require additional foundation objects to be created to provide a holistic representation of their organization in Employee Central. For example, organizations with more levels in their organizational hierarchy may require the addition of a “Sub-Department”.

### Context

Customers transitioning from other SAP products may require the use of Generic Objects to store their “Personnel Area” and “Personnel Sub-Area” attributes, rather than using the standard “Employee Class” and “Employment Type” picklists.

### Procedure

1. Create the generic object in the system.

#### i Note

For information on how to create a generic object, refer to the *Implementing the Metadata Framework* guide on the SAP Help Portal.

2. Assign the Generic Object to the Corporate Data Model or Succession Data Model.

Download the Succession Data Model or Corporate Data Model from Provisioning and open it in an XML editor.

- a. If assigning the Generic Object to a Legacy Foundation Object

1. Download the Succession Data Model or Corporate Data Model from Provisioning and open it in an XML editor.

#### → Remember

As a customer, you don't have access to Provisioning. To complete tasks in Provisioning, contact your implementation partner. If you're no longer working with an implementation partner, contact SAP Cloud Support.

2. In the Corporate Data Model, add a customer-specific field as a custom-string and add the type attribute referencing the external code of the generic object.

```
<hris-element id="jobInfo">
<label>Job Information</label>
<hris-field max-length="256" id="custom-string5" visibility="both"
type="GO_Building" >
    <label>Building</label>
</hris-field>
```

### i Note

Use only a `custom-string` as customer-specific field when you use the `type` attribute with generic objects.

1. Save your changes and upload the data model in Provisioning.
- b. If assigning the Generic Object to an MDF Foundation Object or other Generic Object
  1. Go to the [Configure Object Definition](#) page and search for the destination object.
  2. Make a correction, and add a new custom field. In the [Details](#) link, at a minimum, fill out the following fields:
    - o Data Type = Generic Object
    - o Valid Values Source = the ID of the Generic Object
- c. If assigning the Generic Object to the Succession Data Model
  1. Go to [Manage Business Configuration](#), and select the relevant HRIS Element.
  2. Create a new custom string field and fill out the required information.
  3. Open the [Details](#) of the field, and update the following fields:
    4. o Type of Reference Object = Foundation Object
    - o Reference Object = the ID of the Generic Object
    - o Visibility = View or Edit
3. Save your changes.
4. Assign role-based permissions for the custom field you've added.

## Related Information

[Data Models Used in Onboarding](#)

[Implementing the Metadata Framework](#)

## 8.3 Example: Configuring Workflows for Legacy Foundation Objects

For this example, we configure a workflow for the Location Foundation Object. The workflow will be triggered when a new Location is created or an existing Location is edited.

1. Create a workflow for Location.

**Workflow: Workflow for Location (WFLocation)**

Workflow ID: WFLocation

Name: Workflow for Location

Description:

Remind in Days: 5

Is Delegate Supported: No

Alternate Workflow:

Redirect CC Users to Workflow Approval Page: No

**Step 1**

Approver Type	Approver Role	Context	Edit Transaction	No Approver Behaviour	Relationship to Approver	Respect Permission
Position	Account Manager (ACC_MGR_SC1)		Edit with Route Change	Stop the Workflow		No

Contributors:

No data for Contributors available or you do not have the necessary permission.

Cc Role:

No data for Cc Role available or you do not have the necessary permission.

For more information on how to configure workflows, see the [Employee Central Workflows: Implementation and Administration](#) guide on the SAP Help Portal.

## 2. Create a business rule that will trigger the workflow.

The base object must be the foundation object for which the workflow should be triggered. The parameter code FOWorkflow and the object FO Workflow must also be included.

**BRLocation (BRLocation)**

Basic Information		Parameters	
Start Date	01/01/1900	Name	Object
Rule Type		Context	System Context
Description		Location	Location
		FOWorkflow	FO Workflow

If

This rule is always true  
To add an expression please untick the Always True tick box.

Then

Set **FOWorkflow:Workflow Information** to be equal to **Workflow for Location(WFLocation)**

For more information on how to configure rules, see the [Implementing Business Rules in SAP SuccessFactors](#) guide on the SAP Help Portal.

## 3. Add the rule as an onSave event to the foundation object in the Corporate Data Model.

## 8.4 Example: Automatic Generation of Legacy Foundation Object Codes

It is possible for the system to generate foundation object codes by defining a sequence and then using this sequence in a business rule.

1. Navigate to the [Admin Center](#) [Manage Data](#).
2. In the *Create New* drop-down, select *Sequence*.  
Please ensure the sequence does not begin at 0 or with a negative value.
3. Create a business rule to set all position codes in the format: LOC\_seqCurrentNumber.  
Make sure to use the corresponding Foundation Object as the base object.

The screenshot shows the 'Location\_EC (Location\_EC)' configuration page. It includes sections for 'Basic Information' (Start Date: 01/01/1900, Rule Type: System Context, Description: Location), 'Parameters' (Name: Context, Object: System Context; Name: Location, Object: Location), and a 'Business Rule' section. The rule is defined as follows:

- If:** Location.Code is equal to Null
- Then:** Set Location.Code to be equal to Format()
  - Template: LOC\_%s
  - Argument: Get Next Value()
  - Sequence: location

4. Add the rule as an onInit event to the foundation object in the Corporate Data Model.

### i Note

The rule trigger needs to be set on the foundation object element, after the labels, but before the field configuration.

5. Save your changes.

# 9 Working with Picklists

## 9.1 Introduction to Picklists

Picklists determine the information that gets displayed in a drop-down menu.

Picklists allow you to restrict the values that can be specified for a field.

The master picklist can be downloaded from the Help Portal under  [Implementation](#)  [Master Picklists](#) .

A lot of research has gone into creating this picklist – it is quite comprehensive. As a single picklist is used across all modules, we suggest you exercise caution when modifying the file.

### i Note

You can add your own picklists, as well as modify the value of existing picklists with the exception of Employee Status, where you can change the label but not include additional status types.

Also note that the master picklist is available in XLS format on the Help Portal under  [Employee Central](#)  [Implementation](#) . Save this file in CSV format before you upload it to SAP SuccessFactors. You can do this using any spreadsheet editor.

In Employee Central, you could be using one of the following three picklists:

- ECV2 Picklists (also referred to as Legacy Picklists)
- MDF Picklists
- Cascading Picklists

### i Note

You can now manage all picklists in the suite (legacy, non-legacy, RBP and non-RBP) from a unified Manage Picklists user interface (UI). SAP SuccessFactors must first enable the MDF on your instance and then migrate your legacy picklist data to MDF picklists. After the data is migrated successfully, the Picklist option no longer appears on the Configure Object Definitions page in the Admin Center (you will be automatically redirected). After migration, MDF picklists are available for legacy picklist use. There is no need to deal with ECV2 picklists separate from MDF. Sync is not required as it is the same.

## 9.2 Working with ECV2 (Legacy) Picklists

Picklists determine the information that gets displayed when a dropdown menu is clicked for an object defined in the data model. A single picklist is used across all SAP SuccessFactors modules, so we suggest you exercise caution when making changes.

The master picklist can be downloaded from the Help Portal under  [Implementation](#) . Download this file to get started. Note that a lot of research has gone into creating this picklist – it is very comprehensive. Only company-specific values need to be added to the file. Review the picklist and upload.

### i Note

The master picklist is in XLS format. Save this file in CSV format before you upload it to SAP SuccessFactors. You can do this using any spreadsheet editor.

For picklists, the import engine expects the picklist label to be specified with the exception of the following where the external code defined in the data model needs to be specified:

- AddressType (in Address)
- Event in EventReason Foundation object Import
- JobRelationships

For information on how the different data models are set up and where you can find the picklist label and external code, refer to the Data Models section, [Data Models Used in Onboarding](#).

### i Note

You cannot create and modify picklists at the same time. Each import file must contain either new picklists or modified picklists, not both. Ideally you should export the existing picklists to verify whether or not a picklist exists already before uploading or modifying a picklist.

### 9.2.1 Creating ECV2 Picklists

Create picklists needed for your company that do not already exist in the system.

#### Procedure

1. Verify that the picklists do not exist yet in the system:
  - a. Navigate to the  [Admin Center](#)  [Picklists Management](#) .
  - b. Select [Import/Export](#).
  - c. Select [Submit](#).
  - d. Save and open the exported picklist file. Verify that the picklists you want to add do not exist yet in the system.

If it does already exist in the system, follow the description for [Modifying ECV2 Picklists \[page 132\]](#).

2. Create the new picklists import file by either:
  - Using a template by selecting *Export data format* from the *Picklists* page.
  - Using the standard picklists as a basis that you can download from the Help Portal under ► *Employee Central* ► *Implementation* ▶.

For more information about what the different columns of the picklist file mean, see *About the picklist import file*.

3. Import the picklist file:
  - a. Select *Import picklist(s)*.
  - b. Browse for the picklist file you want to upload.
  - c. For the question *Are all the picklists new?*, select *Yes*.
  - d. Select the relevant character encoding from the *Character Encoding* dropdown menu.
  - e. Select .

## 9.2.2 Modifying ECV2 Picklists

Export existing picklists in the system to make any required changes in the CSV file. Then upload the changes into the system.

### Procedure

1. Export the existing picklist:
  - a. Navigate to the ► *Admin Center* ► *Manage Picklist* ▶.
  - b. Select *Import/Export*.
  - c. Select *Submit*.
  - d. Save the exported picklist file.
2. Modify the picklist entries:
  - a. Open the exported picklist file.
  - b. Modify the picklist entries:
    - Remove any picklists that are correct and that you do not want to modify.
    - Make changes to existing picklists as required. Leave the option IDs since they are for the existing options, but leave it blank for new options.

#### i Note

The option ID must be retained for any existing picklist values. This ID is an internal reference. Any new picklist values being added must have an empty option ID (so the system recognizes them as new). The order picklist values render in is determined by the order they are imported in, **not** by the option ID value.

For more information about what the different columns of the picklist file mean, refer to [About the Picklist Import File](#).

- c. Save your changes to the picklist file.
3. Import the picklist file:
  - a. Select *Import picklist(s)*.
  - b. Browse for the picklist file you want to upload.
  - c. For the question *Are all the picklists new?*, select *No*.

This means that the uploaded file modifies existing picklists, and does not include new picklists.
  - d. Select the relevant character encoding from the *Character Encoding* drop-down menu.
  - e. Select .

### 9.2.3 Deleting ECV2 Picklists

You cannot remove or purge picklist items physically from the system; you can only change their status to DELETED or OBSOLETED.

#### Context

##### i Note

If you have duplicate records and you want to set one of these records to status OBSOLETED, make sure you set the **oldest** record (that is, the record with the smallest option ID) to OBSOLETED, and leave the most recent record as ACTIVE. If you don't do this, it can happen that the Employee Profile displays the ID value of the obsoleted record as it is the most recent, instead of displaying the label.

#### Procedure

- To set the status of a picklist entry to DELETED or OBSOLETED, see *Modifying ECV2 Picklists*.
- In the column *status*, change the relevant picklist entry from *ACTIVE* to *DELETED* or *OBSOLETED*.

#### Related Information

[Modifying ECV2 Picklists \[page 132\]](#)

## 9.2.4 About the Picklist Import File

Picklists are set up using a CSV file that contains specific columns. It is important to know this to avoid issues during import.

### ⚠ Caution

We recommend that you not use MS Excel to edit the CSV file, as this may break special characters. Instead, we recommend editing the downloaded CSV file in Notepad, Open Office, or any UTF-8 editor.

Column	Header	Required?	Description
A	<code>^picklistId</code>	Yes	The picklist value (or 'key') is used to map picklists to the Live Profile and the Succession Org Chart.  This picklist ID has to be the same as the picklist ID used in the data models.
B	<code>OptionId</code>	Assigned by system	The option ID value (or 'primary key') is used by the system to map edits to a previously established picklist value. This value is <b>assigned by the system</b> . Do not enter (for new picklists) or edit (for existing picklists) the option ID.  Note: For each system, the option IDs are different, so you always have to export the existing picklists to know the option ID of a picklist entry.
C	<code>minValue</code>	Yes	These columns are used to support ranges for future use. Set the <code>minValue</code> and the <code>maxValue</code> to the same value. Do not overlap <code>minValue/maxValue</code> with other <code>minValue/maxValue</code> in the same picklist.
D	<code>maxValue</code>	Yes	If you add new entries, you can enter <b>0</b> or <b>-1</b> as <code>minValue/maxValue</code> .

Column	Header	Required?	Description
E	<i>value</i>	No	This field is a placeholder for future use. (May be used in the future to find ranges of incremental values.) For fields that use values like rating scales (risk of loss, impact of loss, etc.) use the same value here as <i>minValue</i> and <i>maxValue</i> .
F	<i>status</i>	Yes	<p>Must be one of the following:</p> <ul style="list-style-type: none"> <li>• <b>ACTIVE</b>: The picklist value is available for use.</li> <li>• <b>DELETED</b>: Disables the picklist value from the system (it is not displayed on the UI anymore).</li> <li>• <b>OBSOLETE</b>: Disables the picklist value from being selected in new user records. Retain its selection for the Query Tool.</li> </ul>

**i Note**

It is important that you do not delete the entry. Use the DELETED and OBSOLETED states, instead.

Column	Header	Required?	Description
G	<code>external_code</code>	No	For standard picklist entries, use the external codes provided in the master picklist. The external code becomes relevant when you integrate with other systems, for example.
			<p><b>i Note</b></p> <p>Best Practice Tip: If you add customer-specific picklists, we recommend to add external codes even though it is not mandatory.</p>
H	<code>parentOptionId</code>	No	This field specifies the parent value for the child in a cascading picklist. This allows you to create links between values the user selects first (for example, "North America") and those that appear in the next picklist field (for example, "USA"). If you do not use cascading picklists, leave this field blank.
I	<code>en_US</code>	Yes	The <code>en_US</code> encoding for US English is the default and therefore required. Other locales (for example, <code>fr_FR</code> for French, and so on) are optional; each locale has to be in its own column.
J	[locale code]	No	

## 9.3 Working with MDF Picklists

MDF picklists are basic effective-dated picklists that can be used by other MDF objects.

For information about MDF picklists, check the [SAP Help Portal](#). Navigate to  [Development](#)  [Implementing the Metadata Framework \(MDF\)](#) 

## Related Information

[Keeping ECV2 and MDF Picklists in Sync \[page 142\]](#)

## 9.4 Working with Cascading Picklists

Cascading picklists allow you to restrict the value of a field based on a previous selection.

### i Note

With the Q2 2015 release, cascading picklists on the MDF Foundation Objects Legal Entity, Business Unit, Division, and Department have been migrated to MDF. They're automatically synchronized to the V2 picklists together with the 'cascading relationship'. If you had a cascading picklist migrated to an MDF picklist for CostCenter in the November 2014 Release, the parent relationship would have not been replicated as MDF picklists did not support cascading behavior. With the Q2 2015 release, cascading picklists are now supported. You can, however, now manually establish the parent-child relationship between picklists as shown in the example.

Consider an example of three picklists with cascading relationships: Country, State, and City. Assuming State can be derived from Country, Country acts as the parent of State. Likewise, assuming City can be derived from State, State is the parent of City.

The following example shows Country as the parent picklist that has different countries as options:

### i Note

Always set the *Effective Start Date* to 1/1/1900 to ensure that all the picklists and their relationships remain valid.

Back to: Admin Tools

### Configure Object Definitions

Search : Picklist ▾  Advanced Create New : No Selection ▾

**History** « 01/01/1900 Take Action ▾

Picklist: Country (country)

\* Code country  
parentPickList  
Display Order No Selection  
Name Country  
\* Status Active  
Effective Start Date 01/01/1900

**Values**

External Code	parentPickListValue	Label	Status
USA		United States	Active
IND		India	Active
DEU		Germany	Active
AUS		Australia	Active
ARG		Argentina	Active
country_61		Asia/Pacific Other	Active
country_59		Austria	Active
country_58		Bahamas	Active
country_57		Belgium	Active
country_56		Bermuda	Active
country_55		Brazil	Active
country_54		Canada	Active
country_53		Chile	Active
country_52		China	Active
country_49		Croatia	Active

In the State picklist, Country is shown as the parent picklist value:

Back to: Admin Tools

### Configure Object Definitions

Search : Picklist ▾  Advanced Create New : No Selection ▾

**History** « 01/01/1900 Take Action ▾

Picklist: State (state)

\* Code state  
parentPickList Country (country) ▾  
Display Order No Selection  
Name State  
\* Status Active  
Effective Start Date 01/01/1900

**Values**

External Code	parentPickListValue	Label	Status
AL	United States (USA) ▾	Alabama	Active
AK	United States (USA) ▾	Alaska	Active
AR	United States (USA) ▾	Arkansas	Active
AZ	United States (USA) ▾	Arizona	Active
CO	United States (USA) ▾	Colorado	Active
CA	United States (USA) ▾	California	Active
KA	India (IND) ▾	Karnataka	Active
HR	India (IND) ▾	Haryana	Active
DL	India (IND) ▾	Delhi	Active
CT	United States (USA) ▾	Connecticut	Active
DE	United States (USA) ▾	Delaware	Active
DC	United States (USA) ▾	District of Columbia	Active
FL	United States (USA) ▾	Florida	Active
GA	United States (USA) ▾	Georgia	Active

Likewise, the City picklist shows State as the parent picklist value:

**Picklist: City (city)**

External Code	parentPickListValue	Label	Status
IN_2	Karnataka (KA)	Bangalore	Active
IN_4	Delhi (DL)	New Delhi	Active
IN_1	Haryana (HR)	Gurgaon	Active
IN_3	Karnataka (KA)	Mysore	Active
US_CA_1	California (CA)	Los Angeles	Active
US_CA_2	California (CA)	East Palo Alto	Active
US_CA_3	California (CA)	San Fernando	Active
US_AL_1	Alabama (AL)	Geneva	Active
US_AL_2	Alabama (AL)	Hamilton	Active
US_AL_3	Alabama (AL)	Alexander City	Active

Now, assuming these picklists are used in one of the Generic Objects - say, *Division (Division)*:

**Object Definition: Division (Division)**

Name	Database Field Name	Maximum Length	Data Type	(16) More
effectiveStartDate	effectiveStartDate	255	Date	Details
externalCode	externalCode	32	String	Details
name	externalName	32	Translatable	Details
description	sffields.sffield1	128	Translatable	Details
effectiveStatus	effectiveStatusStr	255	Enum	Details
mdfSystemTransactionSequence	transactionSequence	255	Number	Details
headOfUnit	sffields.sffield2	255	User	Details
cust_Country	customFields.customField17		Picklist	Details
cust_State	customFields.customField18		Picklist	Details
cust_City	customFields.customField19		Picklist	Details
parentDivision	sffields.sffield3	255	Generic Object	Details

## i Note

The first criteria (parent.effectiveStartDate) as shown in the figure are added automatically. However, you must add the second field criteria manually to achieve the cascading property. Otherwise, it behaves as an independent picklist.

You need to add the relevant field criteria for State and City to achieve the cascading property.

Object Definition: Division (Division)

Take Action ▾

Fields	Details
Name	Name: cust_State Database Field Name: customFields.customField18 Maximum Length: Data Type: Picklist Valid Values Source: state Hide Old Value: No Decimal Precision: Include Inactive Users: No UI Field Renderer: Transient: No Help Text: Private or Sensitive Information: No Show Trailing Zeros: No Required: No Visibility: Editable Status: Active Label: State
effectiveStartDate	parent.effectiveStartDate
externalCode	externalC
name	externalN
description	sfFields.s
effectiveStatus	effectiveS
mdfSystemTransactionSequence	transaction
headOfUnit	sfFields.s
<b>cust_Country</b>	<b>customFi</b>
<b>cust_State</b>	<b>customFi</b>
cust_City	customFi
parentDivision	sfFields.s
cust_string5	customFi
mdfSystemRecordId	recordId
effectiveEndDate	effectiveE
mdfSystemObjectType	objectTy
mdfSystemEntityId	entityId

Rules

External Code

Field Criteria

Source Field Name	Destination Field Value	defaultDestinationValue	status
parent.effectiveStartDate	effectiveStartDate	1900-01-01	Active
<b>parentPickListValue</b>	<b>cust_Country</b>		Active

Done

Object Definition: Division (Division)

Take Action ▾

### Details

Name	<code>cust_City</code>
Database Field Name	<code>customFields.customField19</code>
Maximum Length	
Data Type	Picklist
Valid Values Source	city
Hide Old Value	No
Decimal Precision	
Include Inactive Users	No
UI Field Renderer	
Transient	No
Help Text	
Private or Sensitive Information	No
Show Trailing Zeros	No
Required	No
Visibility	Editable
Status	Active
Label	City

### Fields

Name	Database Field Name
<code>effectiveStartDate</code>	<code>effectiveStartDate</code>
<code>externalCode</code>	<code>externalCode</code>
<code>name</code>	<code>externalName</code>
<code>description</code>	<code>sfFields.description</code>
<code>effectiveStatus</code>	<code>effectiveStatus</code>
<code>mdfSystemTransactionSequence</code>	<code>transactionSequence</code>
<code>headOfUnit</code>	<code>sfFields.headOfUnit</code>
<code>cust_Country</code>	<code>customFields.cust_Country</code>
<code>cust_State</code>	<code>customFields.cust_State</code>
<code>cust_City</code>	<code>customFields.cust_City</code>
<code>parentDivision</code>	<code>sfFields.parentDivision</code>
<code>cust_string5</code>	<code>customFields.cust_string5</code>
<code>mdfSystemRecordId</code>	<code>recordId</code>
<code>effectiveEndDate</code>	<code>effectiveEndDate</code>
<code>mdfSystemObjectType</code>	<code>objectType</code>
<code>mdfSystemEntityId</code>	<code>entityId</code>

### Rules

External Code			
Field Criteria			
Source Field Name	Destination Field Value	defaultDestinationValue	status
<code>parent effectiveStartDate</code>	<code>effectiveStartDate</code>	<code>1900-01-01</code>	Active
<code>parentPickListValue</code>	<code>cust_State</code>		Active

**Done**

Now, on the [Manage Data](#) page for *Division (Division)*, select the required country from the *Country* dropdown. Then select the required state from the *State* dropdown. Only the states that are derived from the specified country are displayed in the *State* dropdown.

Similarly, only the cities that are derived from the specified state are displayed in the *City* dropdown:

Back to: Admin Tools

**Manage Data**

Search :   Advanced Create New :

**Division:**

* Start Date	05/27/2015 <input type="button" value="Calendar"/>
* Code	<input type="text" value="Click or focus to edit"/>
Name	<input type="text" value="Click or focus to edit"/>
Description	<input type="text" value="Click or focus to edit"/>
* Status	Active <input type="button" value="▼"/>
Head of Division	<input type="text" value="No Selection"/>
Country	<input type="text" value="United States (USA)"/>
State	<input type="text" value="California (CA)"/>
City	<input type="text" value="No Selection"/>
Parent Division	San Fernando (US_CA_3) East Palo Alto (US_CA_2) Los Angeles (US CA 1)
Custom String5	<input type="text" value="No Selection"/>
Custom String2	<input type="text" value="No Selection"/>

## 9.5 Keeping ECV2 and MDF Picklists in Sync

If you have different types of picklists in your system, you need to keep them in sync to avoid issues.

Prior to the November 2014 release, it was not possible to keep a V2 picklist and an MDF picklist in sync. With the November 2014 release, you can now do this for the migrated Foundation Objects and for all other V2 picklists that you want to use as MDF picklists. For the migrated foundation objects, this is done automatically during migration. For all other picklists, you would need to perform the steps below manually once to set up the sync process.

Broadly the steps for keeping your ECV2 and MDF picklists in sync are as follows:

1. Prepare your ECV2 picklist.
2. Create the MDF picklist.
3. Define the mapping between the ECV2 and the MDF picklist.

Once the mapping has been defined, any future changes to the MDF picklist will trigger a sync.

The subsequent sections describe these steps.

### How does the sync work?

All picklist entries that will be used by both legacy objects and MDF objects need to be identical in both the ECV2 picklist and the MDF picklist when the mapping is created. After the mapping is done, all changes to the MDF picklist will get synchronized. Picklist synchronization does **not** compare the ECV2 picklist and the MDF picklist; it only detects changes to the MDF picklist. Only changed MDF picklist values will be synchronized, unchanged picklist values are not synchronized. If the MDF picklist and the ECV2 picklist are out of sync at the time the picklist mapping entry is created, this is not "repaired" by triggering a picklist sync.

### 9.5.1 Step 1: Preparing the ECV2 Picklist

Assign a unique and meaningful external code to all your picklists listed in the ECV2 picklist file to avoid problems during the mapping to the MDF picklists.

As a first step, it is recommended that you assign a unique and meaningful external code to all your picklists listed in the ECV2 picklist file. As the external code was not mandatory with ECV2 picklists, it is possible that your picklist might not have external codes defined. If you do not perform this step, the system will automatically create unique external codes for each picklistId at the time of mapping. However, the names would not be meaningful. System-generated names are typically in the format `<picklist_id>_<n>`, where *n* is incremented with each additional entry associated with the picklistId.

#### i Note

For Foundation Objects that have been automatically migrated to MDF, unique external codes are automatically created for its associated picklists. For all others, you will have to manually assign a unique external code. This is a required to keep your ECV2 and MDF picklists in sync.

### i Note

Differences that exist before the mapping of the picklists will not be resolved during the sync. Therefore please make sure that before creating the mapping, the entries in the MDF picklist and the ECV2 picklist are **identical** for picklists that will be used by both legacy objects and MDF objects.

To start, download the ECV2 picklist. For more information, see *Modifying ECV2 Picklists*.

## Related Information

[Modifying ECV2 Picklists \[page 132\]](#)

### 9.5.2 Step 2: Creating the MDF Picklist

You can create an MDF picklist either in the UI or using the MDF import.

*Recommendation:* When you have cascading picklists, always create the parent picklist before creating the child picklist with a mapping to the parent picklist.

### i Note

Differences that exist before the mapping of the picklists will not be resolved during the sync. Therefore please make sure that before creating the mapping, the entries in the MDF picklist and the ECV2 picklist are **identical** for picklists that will be used by both legacy objects and MDF objects.

## Related Information

[Creating MDF Picklists Using the UI \[page 143\]](#)

[Creating MDF Picklists Using MDF Import \[page 144\]](#)

### 9.5.2.1 Creating MDF Picklists Using the UI

For small picklists, you can create them in the UI.

The steps to create an MDF picklist are described below.

1. Navigate to the  [Admin Center](#)  [Configure Object Definitions](#).

### i Note

If you've already been migrated, you'll be directed back to the [Manage Picklist](#) page.

2. From the *Create New* drop-down, select *Picklist*.  
The *Configure Object Definitions* page for the picklists is displayed.
3. In the *Code* field, specify the picklistID from the ECV2 picklist file.
4. In the *Name* field, specify the name for the picklist. This name is displayed on the UI.
5. From the *Status* drop-down, select *Active*.
6. In the *Effective Start Date* field, specify the date as start of time, for example, *January 1, 1900*. This is important to ensure picklist synchronization works. This is because MDF picklists are effective dated while ECV2 picklists are not effective-dated.
7. In the *Values* section, specify the different values that the picklist can take.
8. Save the values.  
For information on MDF picklists, refer to [Managing Your Picklists](#) in the MDF guide on the SAP Help Portal.

### 9.5.2.2 Creating MDF Picklists Using MDF Import

There is no automated process to convert your ECV2 picklist to an MDF picklist. You will need to create the MDF picklist from scratch using the Import/Export framework.

1. Start by exporting the existing MDF picklist. Navigate to the *Admin Center* *Import and Export Data*.
2. From the *Select the action to perform* drop-down, select *Export Data*.
3. Fill out the form. These settings will allow you to download the picklist file with all existing records.

For this field...	Select...
Select Generic Object	Picklist
Include Dependencies	Yes
Include Immutable IDs	No
Exclude reference objects	No
Select all data records	Yes

4. Select *Export*. This will begin a download of the picklist.
5. Use the *Monitor Jobs* option to track the status of the job. In the *Tools* search field, select *Monitor Job* and then **Enter**.  
This will bring up the *Monitor Jobs* page.
6. Select *Download Status* to download the zip file containing the picklist CSV.  
Update the picklist using the ECV2 picklist file as a reference. When you finish updating the CSV file, upload it using the *Import* option.  
You are now ready to trigger the sync between the picklists.

### 9.5.3 Step 3: Mapping the ECV2 Picklist to the MDF Picklist

Since certain foundation objects were migrated to MDF, you have to map the ECV2 picklist to the MDF picklist.

Once you have mapped the ECV2 picklist to the MDF picklist, the MDF picklist will become the leading picklist. It will no longer be possible to make changes to the ECV2 picklist. Changes will only be allowed to the MDF picklist.

1. Navigate to the  [Admin Center](#)  [Manage Data](#).
2. From the [Create New](#) drop-down, select *MDF Picklist to Legacy Picklist Map*.
3. From the *MDF Picklist* drop-down, select the MDF picklist you just created.
4. In the *Legacy Picklist ID* field, specify the picklist ID for the ECV2 picklist to be mapped.
5. Save your changes.

# 10 Events and Event Reasons

## 10.1 Events

Events are occurrences that span the various stages of an employee's lifecycle from hire to rehire. The event sets the user status.

Technically, events are defined in picklists. Events are predelivered by SAP SuccessFactors. You can't create new events or change existing ones, you can only change their labels.

This is a sample list of events delivered by SAP SuccessFactors:

- Additional Job
- Assignment
- Assignment Completion
- Job Change
- Completion of Probation
- Data Change
- Demotion
- Furlough
- Hire
- Job Reclassification
- Pay Rate Change
- Position Change
- Probation
- Promotion
- Rehire
- Return from Disability
- Return to Work
- Suspension
- Termination
- Transfer
- Add Global Assignment
- End Global Assignment
- Obsolete
- Start Pension Payout

The hire and rehire events set the employment status as active and that is the reason that the status is maintained as active where as the transfer/promotion/data change events get the status of the employment from the previous record and set it accordingly.

## 10.2 Event Reasons

Event reasons are defined by the customer. They are used to define more specifically the reason why an event has taken place.

When the manager or admin changes an employee's data, for example, by increasing the salary or changing the department information, the reason behind this change is normally that an event has taken place in that employee's professional life. For example, an event could be a promotion or a transfer to another department. The information about which event lies behind this change is stored in the system for reporting purposes. However, such a change might also include a change to the employee's status, for example, if the employee leaves the company, the employee status would be changed accordingly to reflect that the employee is no longer an active user in the system.

For example, the event "Termination" can take place either because the employee's performance was not sufficient, or because the employee wanted to change companies. In this example, if the company wants to differentiate between the two possibilities, you define two event reasons that you could call "Terminated-Performance Issues", or "Terminated – By Employee". You can create as many event reasons for an event as you like.

Technically, event reasons are foundation objects. This means that the admin can create event reasons in the  [Admin Center](#)  screen or by mass uploading data using a CSV file in the  [Admin Center](#) .

Event reasons are mandatory in the system. Even if a company decides not to create its own event reasons for the purpose of narrowing down the reasons why an event takes place, the admin has to create an event reason for each event that the company uses. A user cannot set an event. The event can only be set by the event reason. The event reason sets the status.

For a minimum setup, the admin should create one (or several) event reasons for the following:

- Hire event
- Rehire event
- Termination event
- Changes to Job Information and Compensation Information
  - You can associate the event reason for such changes to the Data Change event, or you create more specific event reasons for the events Promotion, Transfer, Pay Rate Change, and so on.
- If Leave of Absence is activated, you need to create event reasons for the events Leave of Absence and Return to Work.

## Expected Behavior

UI	Expected Behavior
Change Job and Compensation Information	<p>The <i>Event</i> and <i>Event Reason</i> fields are not shown on the UI if you use rules for event reason derivation.</p> <p>If rules are not used, you fields are shown on the UI for the user to manually select.</p> <ul style="list-style-type: none"><li>• All events are available and permissions are applied for event reasons.</li><li>• Only event reasons linked to a Data Change events can be selected.</li><li>• Only event reasons that have the employee status set to <i>No Selection</i> are available in the  <i>Take Action</i> <i>Job and Compensation Change</i> pages. Event reasons with any other status are not shown in the <i>Event Reason</i> drop-down menu.</li></ul>
Job History	<p>All events and event reasons are displayed in Job History.</p> <p>Only Data Change events should be inserted or created in Job History.</p>
Job History Import	The <i>Event Reason</i> field is a required field in a Job History template and must be included in the import file.
Add New Employee	All event reasons that have the event "Hire" (external_code H) and the employee status "Active" (external_code A) are displayed in the <i>Add New Employee</i> Wizard.
Add Concurrent Employment	All event reasons that have the event "Hire" (external_code H) and the employee status "Active" (external_code A) are displayed in the <i>Event Reason</i> field.
Add Global Assignment	All events reasons that have the event "Add Global Assignment" (external_code GA) and the employee status "Active" (external_code A) are displayed in the <i>Event Reason</i> field.
Termination	All event reasons that have the event "Termination" (external_code 26) and the employee status "Terminated" (external_code T) are displayed in the <i>Termination Reason</i> field..
Termination - Transfer of Direct Reports	<p>All event reasons are displayed in the <i>Transfer Event Reason</i> field of the Termination UI when terminating employees with direct reports.</p> <p>The event reasons are grouped by event and there is no way to change the order or to filter this list.</p>

## 10.3 Creating Event Reasons for Employee Central

You have to create event reasons for certain events in the employment cycle and set statuses for when they occur.

### Prerequisites

You have edit permissions for event reasons listed in  [Permission Settings](#)  [User Permissions](#)  [Employee Data](#).

### Procedure

1. Go to  [Admin Center](#)  [Manage Organization, Pay and Job Structures](#).
2. Select [Create New: Event Reason](#).
3. Create the event reason for the listed events. Enter the mandatory data and ensure you include the required employee status.

Event	Status
Hire	Active
Probation	
End of Probation	
Data Change	-
Assignment	
Assignment Completion	Terminated
Return from Disability	Active
Transfer	
Suspension	Suspended
Job Reclassification	
Job Change	-
Pay Rate Change	-

Event	Status
Demotion	-
Promotion	-
Additional Job	
Layoff	Furlough
Rehire	Active
Termination	Retired
	Terminated

4. Save your changes.

## Results

Event reasons can be sorted in alphabetical order by enabling the *Sort Picklist Columns Based On Labels* option in Provisioning .

## Next Steps

You can then use the Check Tool to verify that the event reasons are configured correctly.

If required, you can create country/region-specific event reasons.

Create event reason derivation rules in the system so that the system automatically selects the appropriate event reason for an event.

## Related Information

[Using the Check Tool](#)

[List of Role-Based Permissions](#)

[Setting Up Country/Region-Specific Picklists for Event Reasons](#)

## 10.4 Setting Up Country/Region-Specific Picklists for Event Reasons

For customers operating in multiple countries/regions that have several legal entities in one country or customers that have a high number of country/region-specific event reasons, country/region-specific picklists for event reasons filter the relevant values for admins and managers on all screens that have an event reason field.

### Prerequisites

- You use the generic object [Country](#). For the prerequisites of using generic objects, refer to [Generic Objects Overview](#).
- You have imported the Events picklist and have created event reasons in the system.

### Context

Customers that operate in multiple countries/regions often have event reasons that are very specific for a country/region. As the employee is always clearly assigned to one legal entity, and thus to one specific country/region, you can set up country/region-specific picklists for event reasons to filter and show only the values to those that are relevant for that employee.

Once the country/region-specific picklists for event reasons are set up, administrators and managers can use them on all screens that have an event reason field [Employment/Personal Information page; Update Employee Records page; History pages (for example, Job History); Add New Employee page, and so on].

#### This feature is useful for:

- Large customers operating in multiple countries/region that have several legal entities in one country
- Customers that have a high number of country/region-specific event reasons

### Procedure

1. You create an association from the Country/Region generic object to the Event Reason foundation object .
  - a. Create a wrapper generic object for event reason.
    1. Select  [Admin Center](#)  [Configure Object Definitions](#) .
    2. In the *Create New:* field, select [Object Definition](#) and create the wrapper generic object. Fill out the following fields:
      - **Code:** Enter the external code for this generic object to uniquely identify it in the system. Note that the prefix `cust_` is automatically added to the external code.
      - **Label:** Add a label for the wrapper generic object.
    3. Save your changes.

- b. Associate the Country/Region generic object with the wrapper generic object:
1. From the [Configure Object Definitions](#) page, choose [Object Definition](#) [Country/Region](#).
  2. Select [Take Action](#) .
  3. Under [Associations](#), in the [Destination Object](#) field, select the wrapper generic object that you have created in step 1:
- 
- | <b>Associations</b> | * Name               | * Type    | * Multiplicity | * Destination Object  |
|---------------------|----------------------|-----------|----------------|-----------------------|
|                     | Just_To_Wrapper_PayC | Composite | One To Many    | Wrapper Pay Component |
4. Save your changes..
- c. Link the wrapper generic object to the foundation object.
1. Navigate back to the wrapper generic object by selecting [Object Definition](#) in the [Search](#) field, and selecting the wrapper generic object.
  2. To edit the wrapper generic object, select [Take Action](#) .
  3. Select the [Details](#) link for the [externalCode](#) field and make the following entries:
    - o [Data Type](#): Select [Foundation Object](#).
    - o [Valid Values Source](#): Enter the HRIS-element ID of the foundation object as defined in the Corporate Data Model. For example, we want to have an association to the eventReason foundation object.
  4. Select [Done](#) and save your changes.
2. Import content for the Country/Region generic object:
- a. Download the Country.csv file. You can download the latest versions of the Employee Central master data models, master picklists, and other configuration files from the Help Portal under [Employee Central](#) .
  - b. Choose [Admin Center](#) .
  - c. Import the CSV file.
3. For each country/region, assign the event reason relevant for that country/region.
- a. Choose [Admin Center](#) .
  - b. In the [Search:](#) field, select the [Country/Region](#) generic object.
  - c. In the field next to [Country/Region](#), select the corresponding country/region, for example, [United States \(USA\)](#).
  - d. Select [Take Action](#) .
  - e. Under [Event Reasons](#), select the corresponding event reasons.

**Note**

- o Assign the country/region-specific event reasons to the relevant country only.
- o Assign the globally-applicable event reasons to **all countries/regions**.

- f. Save your changes..

## **Next Steps**

Repeat these steps for all relevant countries/regions.

# 11 Creating Rules

## 11.1 Event Reason Derivation Business Rules

You can create rules that define the event reason according to what change is done to an employee's data, so that the system automatically selects the appropriate event reason. Depending on the event reason, the employee status is updated, if necessary. These rules are for Job Information and Compensation Information only.

You can create event reason derivation rules using business rules.

If you don't create derivation rules, the user has to manually select the event-reason from the UI every time the user makes a change to the employee data that is linked to an event. However, this is time-consuming and more error-prone, since the employee status depends on the event reason that is selected.

The [Enable Business Rules for Event Reason Derivation](#) Provisioning setting must be enabled. When creating a business rule, you can select the event reason derivation rule scenario which restricts the base objects and SET condition of the rule to avoid configuration errors.

### Recommendations for Configuration

Here are a few recommendations to configure business rules for event reason derivation:

- Check if the event reason field's value is null before setting it through the business rule. This avoids overwriting the event reason accidentally.
- All onSave rules configured for an entity/element are evaluated. The value of the event reason set by the **last matching rule** is considered by system. Therefore, ensure that the rules are defined in an appropriate sequence. Workflow rules are triggered first, then the rules for event reason derivation.
- If the event reason is not set by a rule, the system issues an error and there is little that you can do to resolve this situation. Therefore, it's a good practice to configure a rule that checks whether the event reason is null or not and then set it with a default value (for example [Data Change](#)) if it's null.

### Migrated Rules

If you had XML rules in your system, they are now migrated to business rules. They can be found in the list on the [Configure Business Rules](#) page and contain the term "ERD\_migrated\_rule".

For any new event reason derivation rules, create them using business rules with the Event Reason Derivation rule scenario.

## Example: System Behavior

Consider a case where both the Job Information and Compensation Information entities are processed on the [Manager Self Service \(MSS\) Take Action](#) page.

The event reason will be derived as follows:

1. The system tries to get the event reason set on the Job Information entity first. If it is set on the Job Information entity, then the event reason is used.
  2. If no event reason is set on Job Information, the system tries to derive the event reason set on the Compensation Information entity. If it's set, then the event reason is used.
  3. If the system encounters a case where the event reason is neither set on the Job Information nor on the Compensation Information entity, it raises an error message and this is displayed on the screen.
- The list clearly show the event reason field and it is mandatory. Note that the value of the event reason set on these pages take precedence over the value of the event reason set by the onSave rule. In contrast, for the [History](#) page, the value of event reason set by the rule is considered and this overwrites the value specified on the UI.
1. New Hire
  2. Add/Edit/End Global assignment
  3. Leave of Absence

## Related Information

[Implementing Business Rules in SAP SuccessFactors](#)

[Event Reason Derivation from XML](#)

## 11.2 Business Rules in Employee Central

Business rules are a way to add application logic to determine the outcome of a change made to particular data in the system. This means that business rules can be set up to trigger certain actions when data is added, changed, or deleted from the system.

You can also set up business rules in Employee Central. Rules follow the logic 'If this data is changed in a certain way, then the system reacts in this way', for example, when changing a specific field or saving the job information for a newly hired employee.

The system also has rule scenarios to help configure the business rule in the correct way for certain scenarios. For example, for a rule for a hire or rehire, the rule scenario limits the base object to only either Employment Information or Employment Information model. This helps avoid issues later.

You can set up business rules to do the following:

- **Set default values**

You can define default values for specific fields.

Example: The [OK to rehire](#) field on the termination screen is always [Yes](#) by default.

- **Set conditional values**

You can define which default value is set when a specific condition is met: IF this condition is met, THEN this is how the system should react.

Example: When the admin selects the business unit *ENG*, the job classification is automatically set to *Engineering*.

- **Set field properties**

You can dynamically default a field as visible or required.

Example: If the company is *COMP\_USA*, the phone extension is always required.

**i Note**

However, hiding all fields in a block using a business rule is not supported and will potentially cause unexpected behavior in the system. You must have at least one field on this object enabled to avoid inconsistent behavior.

- **Display error messages**

You can define that an error message is displayed.

Example: The admin forgot to maintain the national ID for a new employee; the error message *National ID is required* is displayed.

- **Calculate transient fields**

You can define transient fields that are calculated "on the fly" when the user opens a page. The calculated values are not meant to be written to the database, as they are not fixed values.

Example: The user can see the employee's current age in the system.

- **Validate consistency of fields**

You can define that all relevant fields are provided.

Example: If an admin selects a *Contract Type* with fixed term validity, the *Contract End Date* needs to be provided. This is automatically checked.

Here is a list of the valid events supported by the system while associating a rule with an element or field.

This event...	Supports...
OnChange	Only for fields
OnInit	Only for elements and triggered only during the hire/rehire flow
OnSave	Only for elements
OnView	Only for elements
OnEdit	Only for elements
SaveAlert	Only for these elements: <ul style="list-style-type: none"><li>• Job Information</li><li>• Compensation Information</li><li>• Employment Details</li><li>• Work Permit Information</li></ul>

## 11.2.1 Optimizing Business Rules

Business rules are very important to keep business processes running, so making sure they are configured correctly is key. You can improve the system performance during rule execution by following some guidelines.

### Less is more

Before you create a new rule in the system, check the existing rules to see if any can be tweaked to cover any new business requirement.

For example, if you need to default a value on the Job Information block of the new hire process. Before immediately adding a new business rule, perform a quick assessment to understand how best to configure the requirement. Typically, there is at least one onInit new hire business rule that is already defaulting values on Job Information or setting the visibility of existing fields. This existing business rule can be tweaked to add the new requirement rather than creating a new one.

### Think Long Term

Remember that these rules are needed for to sustain long-running business processes and that customers don't want to create a support ticket for everything that may go wrong. Try to create meaningful rules that are set up to last.

### Order Matters

For complex business rules, it may be unavoidable to have a scenario where 30, 40, or more onSave business rules need to be configured. As the business rules begin to stack up, there may be performance issues with the application when saving transactions. To reduce the impact, prioritize IF conditions so that the broader conditions are processed first. For example, if you have a requirement that a field should default to a specific value for all union employees in the USA, the first IF statement should be the condition that narrows down the criteria the most. This allows the system to skip the subsequent IF conditions for non-US employees, therefore cutting down processing time for the business rule. While this simplistic example will not show a performance improvement, for customers with complex IF statement rules, the processing time grows exponentially.

### Optimize Where Possible

- Combine business rules wherever possible. The system processes IF/ELSE IF statements faster than processing multiple, separate business rules.
- Prioritize IF conditions.
- Access fields directly on the Base Object of the rule whenever possible, rather than navigating to another object to access the field. The latter approach takes the system longer to process the condition.

### Make Rules Scalable

Try to figure out ways to ensure that business rules don't need to be changed once set up.

Take the example of a customer who has a vastly different process for the termination of an employee, dependent on whether it is a voluntary termination or an involuntary termination. Besides a separate workflow that is triggered, there may be additional checks that need to happen based on the termination type or the defaulting of certain fields. For the workflow rule, a better design than listing all the voluntary terminations in the IF statement might be to have a flag on the Event Reason object that separates out a voluntary termination from an involuntary termination. By doing this, should the customer decide to add or remove termination event reasons in the future, they do not have to touch the business rule.

For more information about business rules, refer to the [Improving Performance](#) section of the *Implementing Business Rules in SAP SuccessFactors* guide.

## 11.2.2 Standard and Model Base Objects for Employee Central

For Employee Central objects, the base object defines what you can enter in the rule; for example, to set field properties, you have to choose a *Model* base object. At the same time, the base object defines what event types you can use in a later step when you assign the rule to the Employee Central object in the data model. For example, you cannot use *onView* events for changes done on the *Add New Employee* screen.

Here's an overview of how base objects, events and pages in the system belong together:

When the user is on this page:	And you want to trigger the rule when the user is...	Then choose this type of base object:
<b>Employee Files</b> Employment Information/Personal Information	<ul style="list-style-type: none"><li>Changing a field value (see <i>onChange</i> event)</li><li>Saving a page (see <i>onSave</i> event)</li><li>Viewing a transient field (see <i>onView</i> event)</li><li>Opening a page in Edit mode (see <i>onEdit</i> event)</li></ul>	Employee Central Object (Person or Employment Object)/[Employee Central Object] <i>Model</i>  For example: <ul style="list-style-type: none"><li><i>Compensation Information</i></li><li><i>Compensation Information Model</i></li><li><i>Job Information</i></li><li><i>Job Information Model</i></li><li><i>Dependents Model</i></li></ul>
<b>Add New Employee</b>	<ul style="list-style-type: none"><li>Opening a page (see <i>onInit</i> event)</li><li>Changing a field value (see <i>onChange</i> event)</li><li>Saving a page (see <i>onSave</i> event)</li></ul>	<i>Employee Information/Employee Information Model</i>  <b>i Note</b> Select a <i>Model</i> base object to set field properties in the rule (for more information, refer to <a href="#">About Model Base Objects [page 159]</a> ).
<b>Manage Organization,</b> <b>Pay and Job Structure</b>	<ul style="list-style-type: none"><li>Opening a page (see <i>onInit</i> event)</li><li>Changing a field value (see <i>onChange</i> event)</li><li>Saving a page (see <i>onSave</i> event)</li></ul>	Foundation object, for example: <ul style="list-style-type: none"><li><i>Location</i></li><li><i>Event Reason</i></li></ul>
<b>i Note</b> Beware that <i>onChange</i> rules are not supported for hidden fields.		

## Related Information

[Overview: Rule Events in Employee Central \[page 162\]](#)

[Event Types for HRIS Elements and HRIS Fields \[page 162\]](#)

### 11.2.2.1 Properties of Model Base Objects

Understand which field properties you can use for Employee Central model base objects.

For *Model* base objects, you can set the following properties:

- *Required*
- *Visibility*
- *Previous Value*
- *Value*

Here's some more information about the different properties:

- *Required*

You can make a field required or not by entering `true` or `false` accordingly.

#### i Note

Fields that are required in the data model should not be set to 'not required' in the rules. This would lead to errors. Here is a list of the required fields you should **not** override using rules:

For this HRIS element in the Succession Data Model...	...this HRIS field is always required:
comlInfo	currency-code
emaillInfo	email-address
	email-type
employmentInfo	end-date
	start-date
globalAssignmentInfo	company
	end-date
	assignment-type
	planned-end-date
imInfo	im-id
jobInfo	job-code
	company
	business-unit

**For this HRIS element in the Succession Data Model...      ...this HRIS field is always required:**

jobRelationsInfo	relationship-type rel-user-id
nationalIdCard	card-type national-id isPrimary country
payComponentNonRecurring	pay-component-code value pay-date
payComponentRecurring	pay-component frequency paycompvalue
pensionPayoutsInfo	company end-date
personallInfo	first-name last-name
personRelationshipInfo	relationship-type
phoneInfo	phone-type phone-number
workPermitInfo	issue-date
(for all HRIS elements where applicable)	start-date externalCode status

- **Visibility**

You can enter the following values:

- **both**: Field is visible and editable.
- **view**: Field is read-only.
- **none**: Field is not visible on the user interface.

- **Previous Value**

Use this property when you want to compare an old value with a new value, for example, when a rule is triggered only when a certain value is changed to a new value. You can also define that any data change to a specific field triggers the rule by setting up the rule as follows:

New value is not equal to previous value

For example: **FTE.Value** is not equal to **FTE.Previous Value**

### i Note

When you use *Previous Value* in the THEN condition, do not use *Set* as output type; it will be ignored by the system, as you cannot change a previous value using the *previous* property.

- **Value**

Use this property when you want to combine setting field properties with setting default or conditional values. When you select *Value*, you have to select the corresponding value in the dropdown menu when creating the rule.

### i Note

#### Current limitations for *Model* base objects:

- Consider if the *onSave* event makes sense to be used when you set field properties. For example, a field should be set to mandatory as soon as the user opens a page (then choose *onInit* event), or when the user makes certain changes (*onChange* event), but not when the user saves a change.

## 11.2.3 Mapping of Employee Central Data Types and Business Rule Field Types

The field type defines which function you can select in a business rule for a certain field. For Employee Central objects, this field type differs from the Employee Central object data type for the different HRIS elements.

Here is a mapping of the Employee Central object data types to the business rule field types, where the column defines the following:

- **Employee Central Object Data Type:** This is the data type that you can find in the Data Object Tables in Employee Central guide. This is based on the database field data type.
- **Business Rule Field Type:** This is the field type that is used on the *Configure Business Rules* page.
- **Manual Entry Value for Field Type:** This defines what the user can enter or choose from on the *Configure Business Rules* page.

Employee Central Object Data Type	Business Rule Field Type	Manual Entry Value for Field Type
String	Text	String
Boolean	Boolean	Yes/No
Date	Date	Date
Double	Decimal	Decimal
Decimal	Decimal	Decimal
Big Decimal	Decimal	Decimal
Long	Number	Integer
(Picklist)	Value	Dropdown list for user to select from
(Foundation object, for example: department, division)	Value	Dropdown list for user to select from

Employee Central Object Data Type	Business Rule Field Type	Manual Entry Value for Field Type
(Enum, for example: Gender, which has a picklist)	Enum	Dropdown list for user to select from

## 11.2.4 Overview: Rule Events in Employee Central

Find out which rule event you can use for a rule that you want to be triggered by a user action on a certain page in Employee Central. The following table gives an overview of the relationship between events and pages on the user interface in Employee Central.

Table 2: Overview: Relationship Between Rule Events and Pages in Employee Central

Rule Event Types	"Manage Org, Pay and Job Structures" Page	"Personal Information" Page	"Employment Information" Page	"Take Action" Button	"Add New Employee" Page
OnInit	Yes	No	No	No	Yes
OnSave	Yes	<ul style="list-style-type: none"> <li>• History</li> <li>• Edit</li> </ul>	<ul style="list-style-type: none"> <li>• History</li> <li>• Edit</li> </ul>	Yes	Yes
OnChange	Yes	<ul style="list-style-type: none"> <li>• History</li> <li>• Edit</li> </ul>	<ul style="list-style-type: none"> <li>• History</li> <li>• Edit</li> </ul>	Yes	Yes
OnView	No	<ul style="list-style-type: none"> <li>• History</li> <li>• Edit</li> <li>• Personal Information block</li> </ul>	<ul style="list-style-type: none"> <li>• History</li> <li>• Edit</li> <li>• Employment Information block</li> </ul>	Yes	No
OnEdit	No	Edit	No	No	No
OnPostSave	No	No	Yes	Yes	Yes

### 11.2.4.1 Event Types for HRIS Elements and HRIS Fields

There are different event types for HRIS element and HRIS fields in business rules. Events define which user action in the system triggers rule execution.

Note that the base object you've selected for creating a rule limits which event you can choose.

Here's an overview of the available event types and when to use which:

Rule is triggered when...	Use this event type:	Assign the rule event to:	Use this event to:
Page is loaded	<i>onInit</i>	HRIS element	Set field properties (for example, making fields mandatory, or hiding fields), or to default values that you want to be shown as soon as the user calls up a page

Rule is triggered when...	Use this event type:	Assign the rule event to:	Use this event to:
Page is saved	<a href="#">onSave</a>	HRIS element	<p>Validate user entries when the user wants to save the changes</p> <p>For example, if the user didn't make an entry in a mandatory field, an error message is displayed.</p>
Field value is changed	<a href="#">onChange</a>	HRIS field	<p>Trigger rules as soon as the user changes a field</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <b>i Note</b> <p>Beware that onChange rules are not supported for hidden fields.</p> </div>
Page with transient field is loaded	<a href="#">onView</a>	HRIS element	<p>Calculate fields that are transient (this means that the result is not a fixed value stored on the database, but is calculated during rule execution when the user calls up the page)</p> <p>For example, to calculate an employee's age.</p> <p>Note: Requires an additional <a href="#">onSave</a> rule that sets the transient field back to null.</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <b>i Note</b> <p>Check the <b>Limitations for onView Rules</b> section after the table.</p> </div>
Page is opened in edit mode	<a href="#">onEdit</a>	HRIS element	<p>Default values for editable fields</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <b>i Note</b> <p><a href="#">onEdit</a> is currently only supported for the paymentInfo EC object.</p> </div> <p>Such defaults could be derived from other fields in the user context.</p> <p>Please note that <a href="#">onEdit</a> is different from <a href="#">onInit</a>: While both are similar in use (to set defaults using rules), <a href="#">onInit</a> is used on the <a href="#">Add New Employee</a> page only (with a person or employment object as base object), whereas <a href="#">onEdit</a> is used on all other editable pages.</p>
Change to relevant employee records is saved	<a href="#">saveAlert</a>	HRIS element	Send alerts to remind users of coming system events
After changes to an object have been saved	<a href="#">onPostSave</a>	HRIS element	Trigger events for Intelligent Services

## Limitations for `onView` Rules

The following limitations apply to `onView` rules:

- You cannot use `onView` rules for the following objects:
  - Home Address
  - Primary Emergency Contact
- For Model base objects that represents data in rows and columns (also referred to as 'data grids'), you have to consider the following:
  - To set a column to invisible or required, create a rule without an If condition (*Always True* selected) and put only visibility and required changes in the Then statement of the rule. Do not set values here.
  - To set values for single rows or for the whole data grid: Create a rule with or without If conditions and set the values in the Then statement. Do not set the visible or required attributes in this rule.

### i Note

This limitation applies to the following Model base objects:

- Spot Bonus Model (for HRIS element `payComponentNonRecurring`)
- Job Relationships Model (for HRIS element `jobRelationsInfo`)
- Compensation Model (for HRIS element `payComponentRecurring`)
- National ID Information Model (for HRIS element `nationalIdCard`)
- Work Permit Info Model (for HRIS element `workPermitInfo`)
- Email Information Model (for HRIS element `emailInfo`)
- Social Accounts Information Model (for HRIS element `imInfo`)

## 11.2.5 Common Problems for Business Rules in Employee Central

Setting up business rules in the system can be tricky. Here are some answers to common questions to help you avoid any issues that may arise.

What you can do with rules	Do	Don't
Assign rules to all HRIS elements	You can assign rules only to HRIS elements contained in the Succession Data Model, the country/region-specific Succession Data Model, or the Corporate Data Model.	You cannot assign rules to the <code>userAccountInfo</code> HRIS element.
Assign more than one rule for the same HRIS element or HRIS field	Yes, you can assign several rules for the same HRIS element or HRIS field in the data model.	-

What you can do with rules	Do	Don't
Use correct base object	<p>Use the current correct entity for the base object.</p> <p>Rules with Employee Information and Employee Information Model are triggered only during new hire flow.</p> <p>If the rule base object is a model object, then always read/set the field properties for the value of that field, but not the direct field itself.</p> <p>The difference between Job Information and Job Information Model as base objects is that for Job Information, the system can only work with the field values in the rule. But with Job Information Model as the base object, in addition to field values, the system can also work with field properties such as Required or Visibility. Customers can choose depending on the requirement of whether field values or field properties are needed.</p>	Do not add additional base objects as parameters in the rule to access other elements in the same rule. To do this, add it to the If condition.
Create cross-block rules	<p>Cross-block rules can set values for fields in a different entity. Currently it is supported only for specific employment-related entities.</p> <ul style="list-style-type: none"> <li>• Job Information</li> <li>• Compensation Information</li> <li>• Employment Information</li> <li>• Job Relationships</li> <li>• Pay Component Recurring</li> <li>• Pay Component Non-Recurring</li> </ul> <p>These rules are triggered based on changes made in the <i>Take Action</i> menu as well as the <i>History</i> UI.</p>	Do not create <b>new</b> Compensation Information records using cross-block rules.

What you can do with rules	Do	Don't
Use rules to set field properties	<p>You can set field properties only with:</p> <ul style="list-style-type: none"> <li>• OnInit rules - to default field properties during new hire flow</li> <li>• OnChange rules - to change field properties based on value from a field</li> <li>• OnView rules - to hide fields in read-only mode of the block. Note that onView rules only work in View mode and not Edit mode.</li> </ul>	Do not use an OnSave rule to set field properties.
Use rules to set valid field properties	<p>You can use rules to change two properties for fields - visibility and required.</p> <p>Valid values for required are:</p> <ul style="list-style-type: none"> <li>• true</li> <li>• false</li> </ul> <p>Valid values for visibility are:</p> <ul style="list-style-type: none"> <li>• both</li> <li>• view</li> <li>• none</li> </ul> <p>Do not try to set the visibility to true or false.</p>	
Create country/region-specific rules	<p>Yes, you can create country/region-specific rules. The country/region-specific fields are listed under the corresponding Employee Central object, preceded by the country/region code (for example: IND for India).</p> <p>Make sure that the country/region-specific rules are in the Country/Region-Specific Succession Data Model.</p>	
Assign a rule in the Succession Data Model, and then a rule for the same field or the same element in the country/region-specific Succession Data Model	<p>Yes, you can assign a rule to the same HRIS field or HRIS element once in the Succession Data Model, and another rule in the Country/Region-specific Succession Data Model.</p>	

### i Note

You cannot set values for fields of a Foundation Object.

What you can do with rules	Do	Don't
Use pay component group sums in rules	<p>A pay component group sum is the total amount (sum) of the pay components that are part of a specific pay component group. You can use pay component group sums in rules, for example, to perform calculations.</p> <p>The pay component group sums are listed under the <i>Compensation Information</i> object, together with the other <i>Compensation Information</i> fields.</p> <p>Only the pay component groups are shown here that are displayed on the compensation UI (that means you have defined <i>Display on Comp UI = Yes</i> under <i>Manage Organization, Pay and Job Structures</i>).</p>	-
Rules for recurring pay components	-	Do not create OnChange rules for recurring pay components of type number to change the visibility of individual fields.
Set value of position and position entry date fields during rehire	<p>By default, the system always expects the user to select value for position field, and then the position entry date will be calculated in the code.</p> <p>You can, however, create an OnInit rule to default position entry date with rehire date. Set the base object of the rule to Job Information Model.</p>	-
Set onSave rules in the jobinfo element	-	<p>Do not create an onSave rule that defaults the event reason to "Other Data Change".</p> <p>The workflow derivation rule is triggered after this, and that rule checks if event reason = "termination". But since the event reason is changed in the earlier rule, this rule is never fulfilled, and does not trigger the workflow.</p>

What you can do with rules	Do	Don't
Create rules for alerts	<p>You can create alerts to be triggered for dates for certain events, for example, before a contract expires.</p>	<p>If you use Employee Information as the base object, note that the system can only read values for the fields of the same entity only.</p> <p>Similarly, if you use Job Information as the base object, you can only check for values in Job Information, and not other elements.</p>
Use previous values in rules	<p>The previous value property is available for all fields when the base object is a model object.</p> <p>However, in case of non-effective-dated entities such as Person Information, Employment Information, Email Information, and so on, the previous value is the value stored in the database before the transaction began.</p> <p>In case of effective-dated entities such as Job Information, when new records are inserted or updated with <i>Take Action</i>, then the value is taken from the previous effective-dated record in the database. In cases of updates made with <i>Make Correction</i>, then it is the value for the field on the same dated record from the database.</p>	-
Use country of company in rules	<p>Use the country listed in the <i>Legal Entity</i> field, since this is stored in the database.</p>	<p>Do not use the <code>&lt;country of company&gt;</code> field in the rule because it is a transient field, meaning it is not stored in the database.</p>
Trigger rules for Person	<p>In the <i>Business Configuration UI</i>, there are 2 nodes for the <i>Person</i> entity, <i>Employee</i> and <i>Dependent</i>. If a rule is configured for the <i>Person</i> entity, it will not be triggered in the system, due to the presence of the <i>Employee</i> node.</p> <p>To fix this problem, you can remove the <i>Employee</i> node to avoid confusion in the system.</p>	-

What you can do with rules	Do	Don't
Use picklists in rules	If you copy values from an MDF picklist to an Employee Central picklist, make sure that the external codes on both sides match.	-
OnInit rules	OnInit rules work only in the new Hire wizard and for foundation objects (in Manage Organization, Pay & Job Structures). Since these rules are for new hires, they do not work for existing users.	
Hire/Rehire rules	We recommend using the Hire/Rehire rule scenario. This limits the base object to either Employment Information or the Employment Information model. This ensures that all the objects involved in hire/rehire are available to the rule even though they are not yet saved in the system.	
Operation in rules for Job Information		It is no longer possible to set up a business rule that contains a CREATE or DELETE operation on a Job Information record.

## Troubleshooting

Here are some further tips to help you troubleshoot any issues with rules:

1. Enable the rule trace from the Admin Center, and make sure that the rule is actually triggered.
2. If the rule is not triggered, then check that it has the **proper base object** and that it is associated with correct event.

### i Note

Remember that rules for Hire and Rehire must only use Employment Information or the Employment Information Model as the base object.

3. Make sure no additional parameters are added to the rule that are used in one of the If/Set statements
4. Make sure that to group all If statements for Compensation Information before grouping the If statements for Job Information.
5. If the rule is accessing other entities with navigations from Employment Details, make sure those other objects are in the same object. For example, in case of new hires, other objects are not yet saved to the database, so those navigations will not work.
6. If the rule sets values to other entities, make sure cross-block rules are supported for those entities. As of now, cross-block rules are supported for employment-related entities only.
7. If the rule base object is Employee Information or Employee Information Model, those rules will be triggered on new hire/rehire only.

8. If both Job Information and Compensation Information are updated from [Take Action](#), then Job Information OnSave rules are executed before Compensation Information OnSave rules.
  9. For OnChange rules, you can enable browser extensions (such as F12 in Google Chrome), and then check for the call "triggerRule" to validate its response.
  10. OnSave rules are executed at the time of workflow initiation, not at the time of the approval during the final step of the workflow.
11.  **Caution**  
We recommend that every rule is tested once it is created to ensure that it works as required.
12. In some cases, you may need to change the order in which rules are executed to ensure that the rule works as required.

## 11.2.6 Cross-Block Rules

Cross-block rules can set values for fields in a different entity. Currently it is supported only for specific employment-related entities.

Cross-block rules are supported for these entities:

- Job Information
- Job Relationship Information
- Compensation Information
- Recurring Pay Component
- Non-Recurring Pay Component
- Employment Information

The source/target direction is very important. The source element must be the base object of the rule.

Common use cases for cross-block rules are

- Changes to Job Information (for example, company, location and/or, employee class) that then update Compensation Information
- Change to Job Information that then update Job Relationships
- Change to Job Information (for example, pay scale level, FTE) that then change (create, update, delete) Recurring Pay Components
- Change to Compensation Information (custom field with annual salary) to update amounts in a Recurring Pay Component

The History UI as well as Imports only support onSave rules for cross-block rules. Generally, onChange rules work when both entities are displayed on the UI, for example, in Manager Self-Service UIs.

Here is an overview of which elements support cross-block rules to other elements for the **creation** (or update) of new records:

### Note

Job Information and Compensation Information as the target element do not support updates to existing records. Cross-block rules with Job Information as the target must use the **Set** command and this always results in the creation of a new record.

Source Element	Target Element: Job Information	Target Element: Job Relationship Information	Target Element: Compensation Information	Target Element: Recurring Pay Component	Target Element: Non-Recurring Pay Component	Target Element: Employment Details
Job Information	-	Supported: • onSave • onChange	Not Supported	Supported: • onSave • onChange	Supported: • onSave • onChange	Not Supported
Job Relationship Information	Not Supported	-	Not Supported	Supported: • onSave • onChange	Supported: • onSave • onChange	Not Supported
Compensation Information	Not Supported	Supported: • onSave	-	Supported: • onSave • onChange	Supported: • onSave	Not Supported
Recurring Pay Component	Not Supported	Supported: • onSave	Not Supported	-	Supported: • onSave	Not Supported
Non-Recurring Pay Component	Not Supported	Not Supported	Not Supported	Supported: • onSave	-	Not Supported
Employment Details	Not Supported	Not Supported	Not Supported	Supported: • onSave	Supported: • onSave	-

Here is an overview of which elements support cross-block rules to other elements for the **update** of existing records.

These rules are triggered based on changes made in the *Take Action* menu as well as the *History* UI.

### i Note

Job Information and Compensation Information as the target element do not support updates of existing records. Cross-block rules with Job Information as the target must use the **Set** command and this always results in the creation of a new record.

Source Element	Target Element: Job Information	Target Element: Job Relationship Information	Target Element: Compensation Information	Target Element: Recurring Pay Component	Target Element: Non-Recurring Pay Component	Target Element: Employment Details
Job Information	-	Supported: • onSave • onChange	Supported: • onSave • onChange	Supported: • onSave • onChange	Supported: • onSave • onChange	Supported: • onSave • onChange

Source Element	Target Element: Job Information	Target Element: Job Relationship Information	Target Element: Compensation Information	Target Element: Recurring Pay Component	Target Element: Non-Recurring Pay Component	Target Element: Employment Details
Job Relationship Information	Supported: • onSave	-	Supported: • onSave • onChange	Supported: • onSave • onChange	Supported: • onSave • onChange	Supported: • onSave • onChange
Compensation Information	Supported: • onSave • onChange	Supported: • onSave	-	Supported: • onSave • onChange	Supported: • onSave	Supported: • onSave
Recurring Pay Component	Supported: • onSave	Supported: • onSave	Supported: • onSave	-	Supported: • onSave	Supported: • onSave
Non-Recurring Pay Component	Not Supported	Not Supported	Supported: • onSave	Supported: • onSave	-	Supported: • onSave

Source Element	Target Element: Job Information	Target Element: Job Relationship Information	Target Element: Compensation Information	Target Element: Recurring Pay Component	Target Element: Non-Recurring Pay Component	Target Element: Employment Details
Employment Details	Supported: • onSave	Not Supported	Supported: • onSave	Supported: • onSave	Supported: • onSave	-

**i Note**  
This only works when both Employment Information and Job Information are part of the transaction for Hire/Re-hire/Termination. It will not work if only Employment Information is changed.

Here is a general overview of which elements support cross-block rules to other elements to **delete** records:

### i Note

You cannot delete Job Information or Compensation Information records using the Delete function in a business rule.

Source Element	Target Element: Job Information	Target Element: Job Relationship Information	Target Element: Compensation Information	Target Element: Recurring Pay Component	Target Element: Non-Recurring Pay Component	Target Element: Employment Details
Job Information	-	Supported: • onSave • onChange	Not Supported	Supported: • onSave • onChange	Supported: • onSave • onChange	Not Supported
Job Relationship Information	Not Supported	-	Not Supported	Supported: • onSave • onChange	Supported: • onSave • onChange	Not Supported
Compensation Information	Not Supported	Supported: • onSave	-	Supported: • onSave • onChange	Supported: • onSave	Not Supported
Recurring Pay Component	Not Supported	Supported: • onSave	Not Supported	-	Supported: • onSave	Not Supported
Non-Recurring Pay Component	Not Supported	Not Supported	Not Supported	Supported: • onSave	-	Not Supported
Employment Details	Not Supported	Not Supported	Not Supported	Supported: • onSave	Supported: • onSave	-

## 11.2.7 Rule Scenarios for Employee Central Core

Rule scenarios help you create rules correctly, based on the rule context and parameters for a given scenario.

Rule Scenario	Description
Generate Assignment ID External	Use this scenario to create the rule that generates the value for Assignment ID External based on MDF Sequence objects. Create a single rule only based on this scenario. Please follow the detailed documentation in the SAP Help Portal under the search term 'Assignment ID External'.

Rule Scenario	Description
Generate Employee ID For Hire/Rehire	You can use this scenario to generate an Employee ID from the Metadata Framework Sequence (MDF) and assign it to the User ID field of the Employee Information object during the Hire/Rehire with new employment process. You must first register the rule in the Matching User Configuration object settings.
Rules For Hire/Rehire	You can use this scenario to configure rules on the Employee Information object during the Hire/Rehire with new employment process.
Event Reason Derivation	You can use this scenario to derive the event reason automatically for Job and Compensation Information entities.
Generate Alerts	You can use this scenario to create rules that generate alerts for Employee Central data, for example, alerts for job information changes.
Enforce New Employment for Rehire	You can use this scenario to configure a rule that validates the business requirements to enforce new employment and returns an error message if the conditions are not met.
Trigger Workflows	You can use this scenario to create a rule that triggers Employee Central workflows to approve data changes.
Internal Job History	You can use this scenario to configure the rule for the Internal Job History Block on the People Profile page.

## Related Information

[Rule Scenarios](#)

### 11.2.8 Adding Contexts for Business Rules

Add context to onSave and onChange rules in the Business Configuration UI to prevent triggering unnecessary rules in a given context and to improve system performance.

## Context

You do not have to add contexts to rules. If no contexts are set, then the rules are triggered when the parameters set in the rule are met. By adding context, you limit the situation where rules are triggered.

These contexts are currently only for HRIS elements, not for MDF objects. The contexts are only for onSave and onChange rules. If you select specific contexts, the rules will be exclusively triggered in the contexts checked. Rules in all contexts not checked will be ignored. Rules in contexts that are not explicitly listed on the screen will be triggered unaffected by any setting.

## i Note

Contexts **only** apply for onSave and onChange rules and do not apply for other types of rules (onView, postSave, init, and so on).

For more information about the BCUI, refer to the [Configuring Context for Business Rules](#) in the Setting Up and Using Business Configuration UI (BCUI) guide on the SAP Help Portal.

You can limit the triggering of rules to the following contexts:

- Edit
- History
- Imports
- Mass Changes
- Hire
- Onboarding
- Promotion from Compensation Planning
- Report No-Shows
- Off Cycle Event Batch
- Termination

Here are some recommendations for what situation the contexts are useful, for example:

The rules for Event Reason Derivation only make sense when making changes in ESS/MSS, so we recommended limiting such rules to the ESS/MSS context by setting it to **Yes** while switching all other contexts to **No**.

If validation rules are only made for specific purposes such as in the context of Termination or New Hire, we recommend setting only this exact context to **Yes** for such a rule.

Context/Rule Type	Workflow Derivation	Event Reason Derivation	Validation	Cross-Block
Edit (MSS/ESS)	Yes	Yes	Yes	Yes
History	No	No	Yes	Yes
Imports	No	No	Yes	Yes
	Only when workflows are disabled for im- ports.			
Mass Changes	No	No	No	No
	You would not want to create approval work- flows when making the same change for lots of people.			
Hire/Rehire	Yes	No	Yes	No
Onboarding	No	No	No	No
Promotion from Com- pensation Planning	No	No	No	No

Context/Rule Type	Workflow Derivation	Event Reason Derivation	Validation	Cross-Block
Report No-Shows	Yes	No	Yes	No
Off Cycle Event Batch	No	No	No	No
Termination	Yes	No	Yes	No

### i Note

The Onboarding rule context is applied on Onboarding data collection pages, for example, Personal Data Collection page.

The Promotion from Compensation Planning rule context is used so that the system can differentiate promotions done from within Compensation (on-cycle promotions) from those done from Employee Central (off-cycle promotions).

## Procedure

1. Go to  [Admin Center](#)  [Manage Business Configuration](#).
2. Under [Employee Central](#), select the relevant HRIS element.
3. For onSave rules, in the [Trigger Rules](#) section, select the [Details](#) link.

For onChange rules, find the relevant field and select the [Details](#) link. Scroll down to the [Trigger Rules](#) section and select the [Details](#) link.

4. In the [Details](#) pop-up, ensure that the [Event Type](#) is either onSave or onChange.
5. Select the [Plus \(+\)](#) icon to add a context.
6. In the [Rules Contexts](#) section, for each context, select [Yes](#) or [No](#) from the drop-down list.

Only after you add context to the rule, the default for all contexts is [Yes](#), which means that the rules would only be triggered in those screens.

If you change the setting to [No](#), that means that the rule is not processed in that context for the HRIS element.

7. Select [Done](#) to exit the pop-up.
8. Save your changes.

## 11.2.9 Example Employee Central Business Rules

Here are a few examples of how you can use business rules in your system.

### General

For all business rules, do the following:

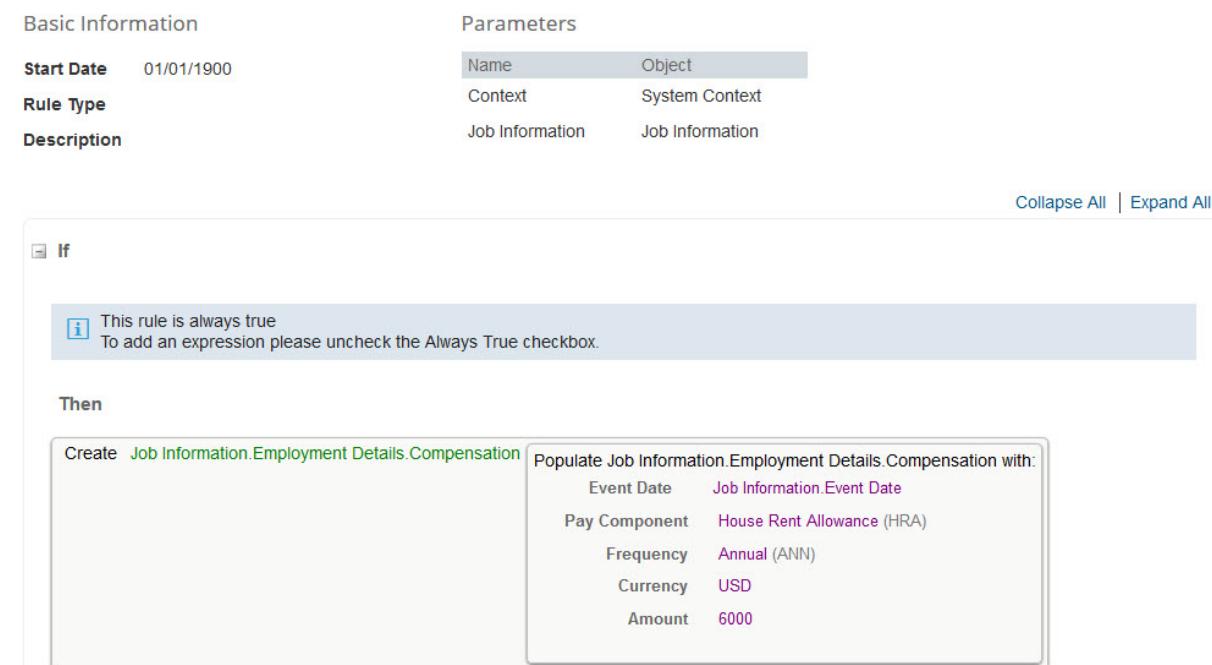
1. Go to  [Admin Center](#)  [Configure Business Rules](#).
2. Create a new rule.
3. Select a basic rule. Choose the correct base object.
4. Create a rule based on the examples shown.
5. Save your changes.

### Create Objects

You can create objects that have multiple instances for the same entity for a given user, for example, Phone Information, Email Information, National ID, Pay Component Recurring, and so on.

To create an entity using cross-block OnSave rules, use the "CREATE" function in the THEN condition, not "SET".

Here is an example of how to create a Pay Component Recurring object in the OnSave rule on Job Information.



The screenshot shows the configuration of a business rule for 'Job Information'. The 'Basic Information' section includes 'Start Date' (01/01/1900), 'Rule Type' (Context), and 'Description'. The 'Parameters' section shows 'Name' (Object) with two entries: 'Context' (System Context) and 'Job Information' (Job Information). A 'Parameters' table shows 'Name' (Object) with two entries: 'Context' (System Context) and 'Job Information' (Job Information). The 'If' condition is expanded, showing a note: 'This rule is always true. To add an expression please uncheck the Always True checkbox.' The 'Then' condition is expanded, showing a 'Create' action: 'Create Job Information.Employment Details.Compensation'. This action populates 'Job Information.Employment Details.Compensation' with the following values:

Event Date	Job Information.Event Date
Pay Component	House Rent Allowance (HRA)
Frequency	Annual (ANN)
Currency	USD
Amount	6000

If the rule tried to navigate to a datagrid object, for example, Email Information or Pay Component Recurring, enter the value for the filter. A best practice here is to filter by primary key. If you don't enter the value for the filter, it may not be clear which instance of the same object is updated.

## Create Custom MDF Objects

You can create data for a custom MDF objects when saving changes to Job Information. This means that instead of having to manually create the data for the custom MDF object in the [Manage Data](#) page, you can create a rule that does the same for you.

This only works for:

- Custom MDF objects
- Job Information and Job Information Model base objects
- onSave rules for Job Information HRIS element

Here's an example of how to create a custom MDF object for cust\_benefitPlanOverride when the employee subgroup changes:

EC\_CAN\_TESTSAP (EC\_CAN\_TESTSAP) Insert New Record

Basic Information

Start Date 01/01/1900

Rule Type

Description Job Information Model

Parameters

Name	Object
Context	System Context
Job Information Model	Job Information Model

If

and

Job Information Model.Employee Subgroup.Value is equal to CSRP (35774)

Job Information Model.Employee Subgroup.Previous Value is not equal to CSRP (35774)

Then

Create Job Information Model.Employment Details Model.cust\_benefitPlanOverride

Populate Job Information Model.Employment Details Model.cust\_benefitPlanOverride

effectiveStartDate	Job Information Model.Event Date.Value
externalCode	Job Information Model.User ID.Value
externalName	Salaried Employees (A)

Collapse All | Expand All

## Adding Additional Parameters

Do not add additional base objects as parameters in the rule to access other elements in the same rule. To do this, add it to the If condition using the default navigation provided in the Employment Details element that is available for each Employee Central base object.

**Basic Information**

Rule Name *	TestParams
Rule ID *	TestParams
Start Date *	01/01/1900
Rule Type	No Selection
Description	

**Parameters**

Name	Code	Object
Context	context	System Context
Job Information	jobInfo	Job Information

[+ Add Parameter](#)

[Collapse All](#) | [Expand All](#)

**If**

and

Job Information Department is equal to Client Service (SVCS)

Job Information.Employment Details.Compensation Information.AnnualizedSalary > 10000

Select Job Information.Employment Details.Compensation Information where...

**Then**

Raise Message

Message: No Selection

Severity: Select Severity

[Add Else If](#)   [Add Else](#)

[Cancel](#) [Save](#)

## Setting Field Properties

You can use OnInit rules to set field properties such as required=true or visibility=false based on some business conditions. This is generally used in new hire flow with Employee Information model as the base object.

If you want to set properties for a field using an OnInit rule, it should be assigned as an OnInit rule for that element only. If the requirement is to set field properties for fields from multiple elements, you need to create one OnInit rule for each element, for example, Job Information and Compensation Information. You can use fields from other elements in the If condition, that is valid, but you should not try to set properties for fields from other elements in the rule.

**OnInitRuleForJobInfo (OnInitRuleForJobInfo)**

Insert New Record

Basic Information		Parameters	
Start Date	01/01/1900	Name	Object
Rule Type	Context	System Context	
Description	Employee Information Model Employee Information Model		

Collapse All | Expand All

If

Employee Information Model.Job Information.Company.Value is equal to Ace USA (ACE\_USA)

Then

Set Employee Information Model.Job Information.Department.Required to be equal to true  
 Set Employee Information Model.Job Information.Division.Visibility to be equal to none  
 Set Employee Information Model.Job Information.FinanceTeam-GO Required to be equal to true

**OnInitRuleForComplInfo (OnInitRuleForComplInfo)**

Insert New Record

Basic Information		Parameters	
Start Date	01/01/1900	Name	Object
Rule Type	Context	System Context	
Description	Employee Information Model Employee Information Model		

Collapse All | Expand All

If

Employee Information Model.Job Information.Company.Value is equal to Ace USA (ACE\_USA)

Then

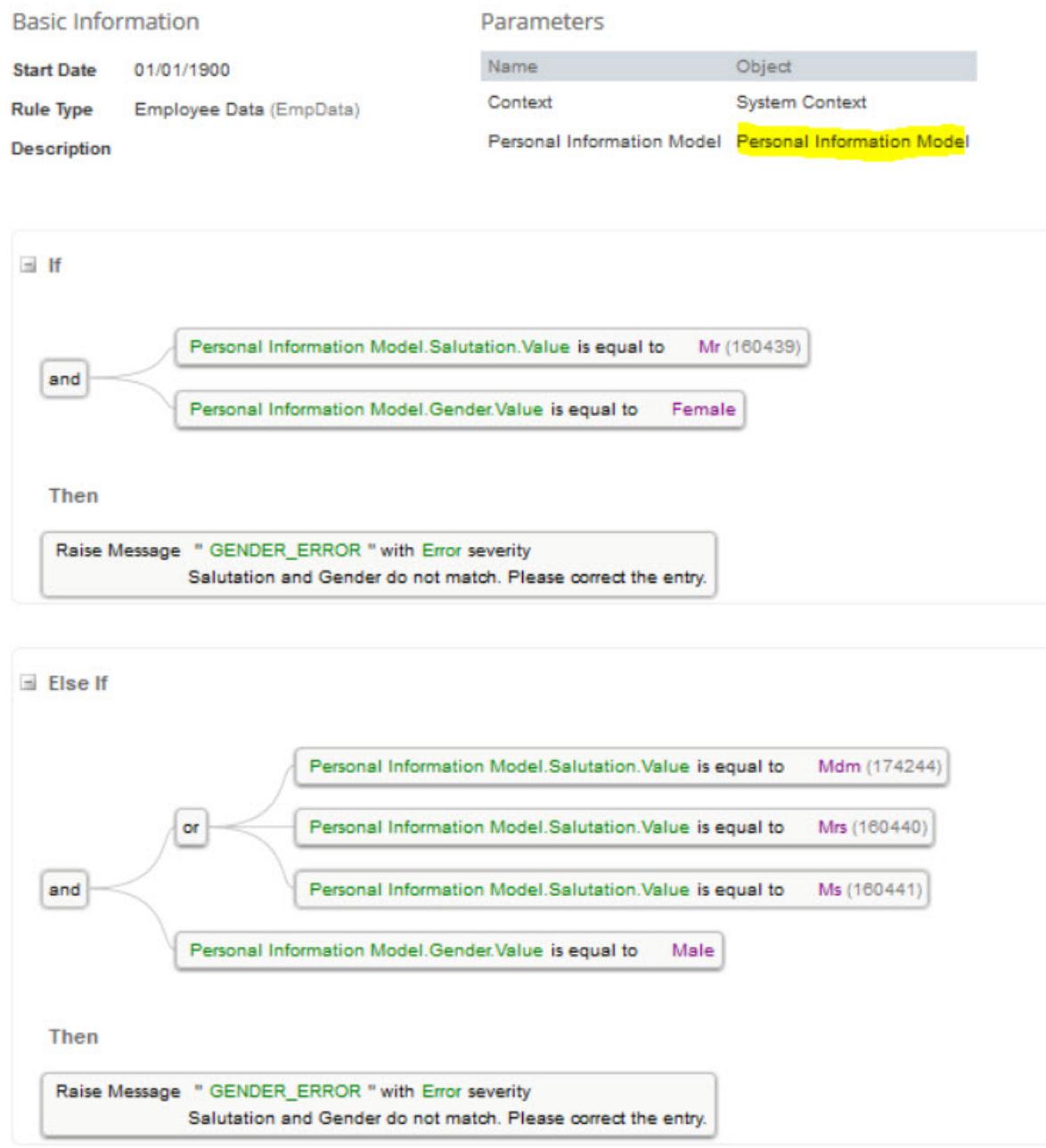
Set Employee Information Model.Compensation Info.Pay Grade.Required to be equal to true  
 Set Employee Information Model.Compensation Info.Pay Type.Required to be equal to true  
 Set Employee Information Model.Compensation Info.Pay Group.Visibility to be equal to none

## Check Correct Salutation and Gender

You can set up the system to ensure that an employee receives the correct salutation and gender settings.

In the AND/OR condition, you can set the system to check the combination of Salutation & Gender. If there is any discrepancy, then an error message is displayed.

This rule can either be an onChange rule for the fields for Gender/Salutation or an onSave rule for employee personal information or both.



## Automatically Set Retirement Date

You can automatically set the employee retirement date based on the employee age & pay grade when the HR admin saves the new hire data.

You can use the Date Plus rule function to set the retirement date based on the date of birth, for example, set the retirement date when employee age is 50 years (600 months) old.

This is an onSave rule for job information.

The screenshot shows a rule configuration window with the following details:

**Basic Information**

- Start Date: 01/01/1900
- Rule Type: Employee Data (EmpData)
- Description: Set the Retirement Date of Employee

**Parameters**

Name	Object
Context	System Context
Job Information	Job Information

**Condition (If):** Job Information.Job Grade is equal to Test (Test)

**Action (Then):**

Set Job Information.Employment Details.Retirement Date to be equal to Date Plus()

Base Date: Job Information.Employment Details.Biographical Information.Date of Birth  
Number of Months: 600  
Number of Days: 0

**Buttons:** Collapse All | Expand All

## Default Phone Information

You can set a default Business Phone as primary phone contact information & set the default business phone number for all employees during creation of business phone data. Only the extension would need to be provided as an input for Business Phone during data maintenance.

This is an onChange rule for the `<phone type>` field in phone information model object.

The screenshot shows a rule configuration window with the following details:

**Basic Information**

- Start Date: 01/01/1900
- Rule Type: EmpData (EmpData)
- Description:

**Parameters**

Name	Object
Context	System Context
Phone Information Model	Phone Information Model

**Condition (If):** Phone Information Model.Phone Type.Value is equal to Business (1572)

**Action (Then):**

Set Phone Information Model.Is Primary.Value to be equal to Yes

Set Phone Information Model.Phone Number.Value to be equal to (1234)567-1234

**Buttons:** Add Else If | Add Else

## Set Phone Number Format

You can set up the system to format a phone number, so that numbers beginning with 0, are formatted to begin with (0).

The If condition uses the Matches() function to check whether the phone number entered matches the defined format. The format is defined as any number that begins with 0. The asterisk indicates zero or more occurrences of the preceding element. The Then condition firsts uses the Format() function to format the number using the template. The template is defined as (0)%s. This is a string starting with (0), where (%s = a string.)

The Then condition then makes use of the Substring() function to define the end of the string (%s) from the template. The phone number is the string that the substring is taken from. The substring starts at index 2. The length is optional and should be left 'Null' as there may be different length phone numbers. The index of a sting is the position of a character in the string. Strings start at index 1.

This is an onChange rule for the <phone-number> field in phone information.

The screenshot shows the 'Format Phone Number (Format\_Phone\_Number)' rule configuration. It includes sections for 'Basic Information', 'Parameters', 'If' condition, and 'Then' action.

**Basic Information:**

- Start Date: 01/01/1900
- Rule Type: System Rule
- Description: Rule to format phone number, so that a number beginning with 0, is formated to begin with (0).

**Parameters:**

Name	Object
Context	System Context
Phone Information	Phone Information

**If:**

Matches()  
String to be checked: Phone Information.Phone Number  
Regular expression: 0.\*

is equal to Yes

**Then:**

Set Phone Information.Phone Number to be equal to Format()  
Template: (0)%s  
Argument: Substring()  
Text: Phone Information.Phone Number  
Start: 2  
(Optional) Length: Null

## Set Job Title

You can automatically set the job title based on the job code selection.

In this rule, we are setting the job title value in employee job information from the value which is maintained at job classification object level.

This is an onChange rule for the <job classification> field in job information.

## SET\_JOB\_TITLE (SET\_JOB\_TITLE)

### Basic Information

Start Date 01/01/1900  
Rule Type EmpData (EmpData)  
Description

### Parameters

Name	Object
Context	System Context
Job Information	Job Information

#### If

Job Information.Job Classification is not equal to Null

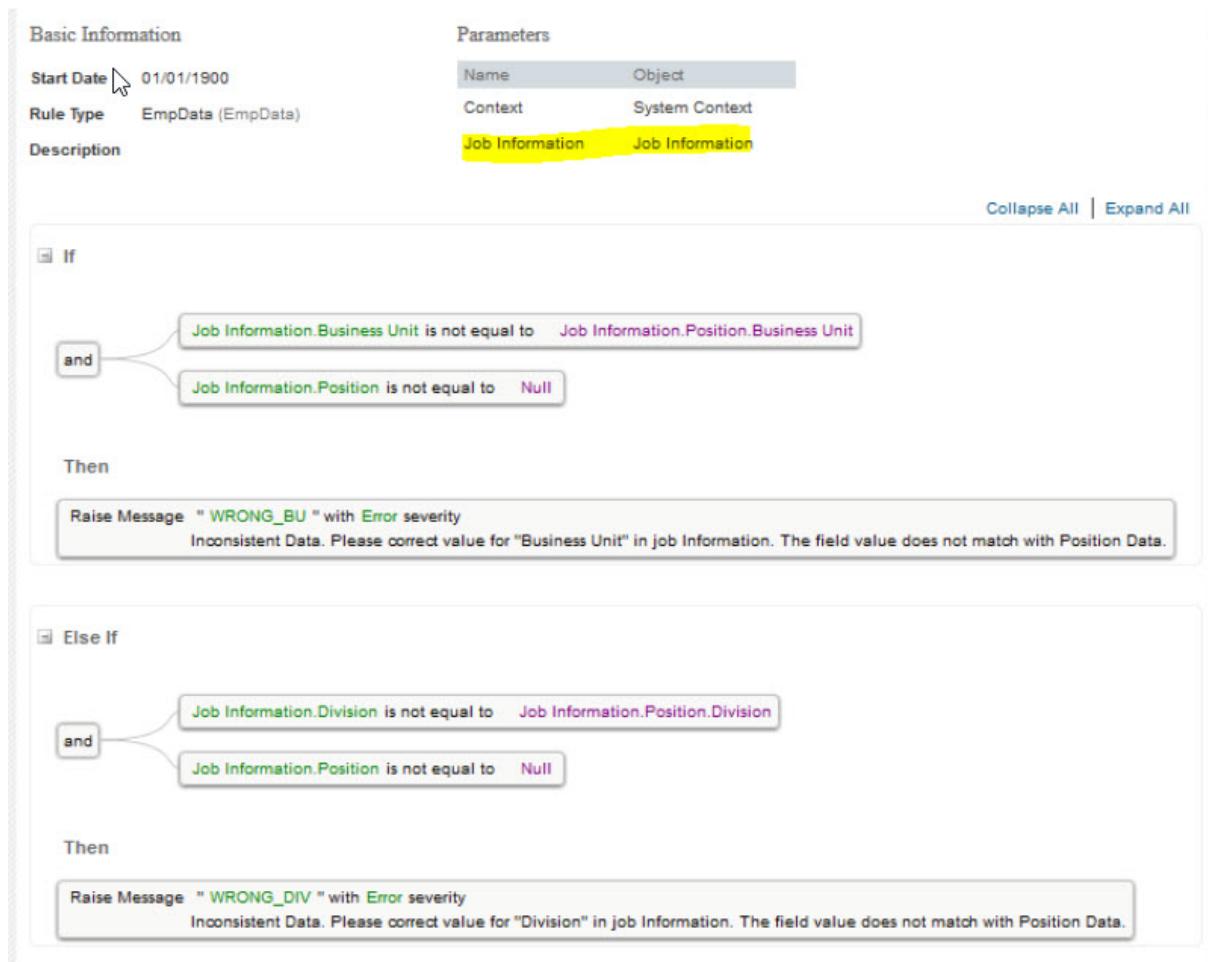
#### Then

Set Job Information.Job Title to be equal to Job Information.Job Classification.Job Title.Default Value

## Validate Data in Position and Job Information

You can set the system to validate the data integrity of data at position object level against what is there for an employee at job information level. If they are different, an error message is displayed.

This is an onSave rule for job information.



## Insert Record in Job Information for Every Change in Job Relationship

You can use a cross-block business rule to create a record in the job information block for every change in job relationships information for a user.

This is an OnSave rule for job relationship information.

**DCHNG\_JOBRELATION (DCHNG\_JOBRELATION)**

**Basic Information**

**Start Date** 01/01/1900  
**Rule Type**  
**Description**

**Parameters**

Name	Object
Context	System Context
Job Relationships Model Job Relationships Model	

**If**

**and**

Job Relationships Model. Name .Value is not equal to Job Relationships Model. Name .Previous Value  
Job Relationships Model.Event Date.Value is not equal to Job Relationships Model.Employment Details Model.Job Information.Event Date

**Then**

Set Job Relationships Model.Employment Details Model.Job Information.My Event Reason to be equal to Correction - Status (DATASTAT)

**Insert New Record**

**Collapse All | Expand All**

## End Deductions on Termination

You can use a cross-block business rule to end recurring deductions as of the termination date.

This is an OnSave rule for employment information.

**Terminate Deductions (Terminate\_Deductions)**

**Basic Information**

**Start Date** 01/01/1900  
**Rule Type**  
**Description**

**Parameters**

Name	Object
Context	System Context
Employment Details Employment Details	

**If**

Employment Details.Termination Date is not equal to Null

**Then**

Create Employment Details.RecurringDeduction

Populate Employment Details.RecurringDeduction with:

Effective Start Date	Date Plus()
User	Base Date: Employment Details.Termination Date
	Number of Months: 0
	Number of Days: 1
User	Context.Current User

**Insert New Record**

**Collapse All | Expand All**

## Set Field-Level View Permissions for Non-Effective Dated Blocks

Use OnView rules to control who can see which fields in the non-effective dated blocks and ensure data protection and privacy for your users, since you cannot use permissions to set field-level *View* permissions for such blocks.

This is an example of an OnView rule that controls the visibility of the <Date of Death> field in the Biographical Information for a permission group.

The screenshot shows the configuration of an OnView rule named 'bio\_Model'. It includes sections for 'Basic Information' (Start Date: 01/01/1900, Rule Type: Biographical Information Model), 'Parameters' (Name: Object, Context: System Context, Biographical Information Model), and 'Variables' (empty). The main configuration area is titled 'If' and contains a condition: 'Is User in Permission Group' (User ID: Login User(), Permission Group Name: Alice Poon Only). The 'Then' section contains an action: 'Set Biographical Information Model.Date of Death.Visibility to be equal to none'. Buttons for 'Add Else If' and 'Add Else' are also visible.

This is an example of an OnView rule that controls the visibility of the <Date of Death> field in the Biographical Information for a user.

## bio\_Model (bio\_Model)

### Basic Information

Start Date 01/01/1900

Rule Type

Description

### Parameters

Name	Object
Context	System Context 
Biographical Information Model	Biographical Information Model

### Variables

### If

 Login User()  is equal to  User Aanya Singh 

Identifies the logged in user.

### Then

Set Biographical Information Model.Date of Death.Visibility to be equal to none 

Add Else If

Add Else

# 12 Human Resource Information System (HRIS) Synchronization

Human Resource Information System (HRIS) synchronization is the sync of data from Employee Central (EC) to user data tables when you have Employee Central enabled.

Employee Central is the core HR system of record for SAP SuccessFactors. It's the repository of effectively dated data to manage employees through their lifecycle within an organization. However, some of our mature offerings including the User Directory continue to use data from the legacy data tables. The data in legacy tables is updated using the User Data File (UDF). HRIS Sync is run to make sure the data in Employee Central is also written to the legacy tables so that user data is consistent across all modules in SAP SuccessFactors HXM Suite.

HRIS Sync is achieved through a job that you set up in HRIS Sync is achieved through a job set up in provisioning.

You can monitor the HRIS Sync job using the *Scheduled Job Manager* in Admin Center.

The sync could be a full HRIS Sync or an incremental sync based on how the job is configured.

## → Remember

As a customer, you don't have access to Provisioning. To complete tasks in Provisioning, contact your implementation partner. If you're no longer working with an implementation partner, contact SAP Cloud Support.

## Related Information

[Monitoring Scheduled Jobs in Admin Center](#)

### 12.1 What Triggers HRIS Sync?

HRIS Sync can be triggered in a manner transparent to you due to certain actions taken on the UI, or it can be kicked off due to a scheduled job being triggered.

HRIS Sync can be triggered by any of the following ways:

### Real-Time Sync Due to UI Operation

If an Employee Central record being updated is effective at that time, then the sync occurs immediately and doesn't trigger the HRIS Sync job. EC UI operations (new hire processing, MSS job change, and personal info

change) have integrated with HRIS Sync. The data changes due to such operations from the UI to user data tables happens right away.

**i Note**

Future-dated records are also synced, but their status is inactive. The status becomes active as soon as the future date becomes current.

## Scheduled HRIS Sync

HRIS Sync job is set up in provisioning to either trigger HRIS Sync on a regular schedule or trigger a one time full sync.

**→ Remember**

As a customer, you don't have access to Provisioning. To complete tasks in Provisioning, contact your implementation partner. If you're no longer working with an implementation partner, contact SAP Cloud Support.

## EC Data Import or User Data Changes Through API

If there is at least one recurring HRIS Sync job configured in your company instance, an HRIS Element (Employee Central data) import triggers an HRIS Sync job run. This import of data can be from the UI or through API. Only one HRIS sync job can run at a time.

**i Note**

HRIS Sync only syncs data for valid Employee Central users that is, for those users that the `isECRecord` value is set to 1.

## 12.2 HRIS Sync Job

HRIS Sync can be incremental or full data push from Employee Central to user data tables.

The HRIS Sync job is set up in Provisioning.

You can monitor the HRIS Sync job using the [Scheduled Job Manager](#) in Admin Center.

**→ Remember**

As a customer, you don't have access to Provisioning. To complete tasks in Provisioning, contact your implementation partner. If you're no longer working with an implementation partner, contact SAP Cloud Support.

### Note

This is an SAP SuccessFactors Business Beyond Bias feature. Use it to support processes that detect, prevent, or eliminate the influence of bias, helping you achieve your diversity and inclusion goals.

### Caution

If using Employee Central, you should not be making user data changes by importing User Data File either from the UI or through a scheduled job. Doing so can overwrite data coming from Employee Central and cause data inconsistencies.

## Incremental vs. Full HRIS Sync Job

When configuring the HRIS Sync job, one of the parameters needed is *Modified Date Since*. There are two options:

- *Last successful HRIS sync job run date*: When the system runs an HRIS Sync job with this option, the records that have changed since the last successful HRIS Sync job and the records that are now current are synced to user directory tables. This partial sync is **incremental sync**. All EC data, whether effective dated or not, is synced incrementally with this option. If it's effective dated data but for a date in the future, it's synced when the record becomes current.
- *Specify a date*: You can use this option for **full sync** from a specific date for special needs.

### Note

Full sync can have negative impact on performance so it's recommended that you use this option only once and not recurringlly.

Consider running an HRIS Sync job when:

- You updated sync configuration in the data model and want the new configuration to be applied to all the data including the existing data.
- There are data inconsistencies between EC data and data in legacy tables. Data inconsistency could happen for several reasons including, in the past if basic import was used to upload data to the legacy tables.

## Future Hire User Sync

Future dated job info records that have events as H, GA, and SPP are considered as future-dated hires. All job info records of the future-dated hire are synced when the HRIS Sync job is run. If there are multiple records in a time slice, the last job info record in the time slice wins.

## Generic Object Sync

If generic objects (GO) are enabled in Provisioning, then the GO type field is synced as follows:

- If synced by HRIS Sync job
  - If GO name of job owner's locale isn't empty, sync to job owner's locale.
  - If GO name of job owner's locale is empty, and GO name of company locale isn't empty, then sync to company locale.
  - If GO name of job owner's locale and company locale are both empty, and GO name's default value isn't empty, then sync to default value.
  - If none of these values are populated, then sync "externalCode (externalCode)."
- If synced by UI change
  - If GO name of company locale isn't empty, sync to company locale.
  - If GO name of company locale is empty, and GO name's default value isn't empty, then sync to default value.
  - If both values are empty, then sync "externalCode."

### • Example

jobInfo, department/division/job-code fields are MDF type fields. When syncing these fields by hard-coded mapping, the generic object's name is taken out and translated by locales.

## Foundation Object Sync

If foundation objects are enabled in Provisioning, note the following:

If the EC HRIS field is a foundation object and the standard element field is NOT a picklist, the field is converted into FOName (FOExternalCode) and synced. If FO name is empty, FOExternalCode (FOExternalCode) is synced.

If the EC HRIS field is a foundation object and the standard element field is a picklist, the system assumes that the external code of the foundation object equals the external code of the picklist. The system then gets the option ID by the external code of the picklist and syncs the option ID back to user directory table. If the external code of the FO object can't be found in the external code of the picklist, the mapping value to standard element is set to null. Other fields of the record can still sync successfully.

## Subsequent Sync to Refresh Job Code Picklist

An HRIS Sync job triggers a [RefreshJobCodePicklistJob](#). The purpose of this subsequent job is to make sure that changes to job codes are also synced to the picklist table for job codes.

### i Note

The HRIS Sync job syncs entities in a particular order. There could be scenarios where an entity synced earlier could be overwritten by an entity synced later.

## Related Information

[Monitoring Scheduled Jobs in Admin Center](#)

### 12.2.1 Configuring an HRIS Sync Job

You can schedule a recurring or a one time sync job in the system to sync data from your Employee Central module to the legacy tables.

#### Prerequisites

- You have access to *Provisioning*.
- You've verified that the data model configuration is as desired.

#### → Recommendation

Avoid all data changes during an HRIS Sync job run. Import of user data, user data changes using OData APIs, or any other operations that change the user data at the same time as an HRIS Sync job run can lead to data discrepancies. If you aren't aware of the schedule for your HRIS Sync jobs, contact your Implementation Partner. If you're no longer working with an Implementation Partner, contact SAP Cloud Support.

#### → Remember

As a customer, you don't have access to Provisioning. To complete tasks in Provisioning, contact your implementation partner. If you're no longer working with an implementation partner, contact SAP Cloud Support.

You can monitor the HRIS Sync job using the *Scheduled Job Manager* in Admin Center.

#### i Note

This is an SAP SuccessFactors Business Beyond Bias feature. Use it to support processes that detect, prevent, or eliminate the influence of bias, helping you achieve your diversity and inclusion goals.

## Context

Employee Central pushes data to user data tables using the HRIS Sync job configured and scheduled in your instance.

## Procedure

1. Go to  [Provisioning](#)  [Manage Scheduled Jobs](#).
2. Select [Create New Job](#).
3. Enter the following required information in the [Create New Job](#) screen:
  - **Job Owner:** Choose a valid user in your company. This user should have permissions to all fields being synced and the target population includes all the users you're syncing the fields for.
  - **Job Name:** Choose a descriptive name of your liking.
  - **Job type:** HRIS Sync
4. Complete any optional settings including for [Job Parameters](#):
  - [Last successful HRIS sync job run date \(MM/dd/yyyy\)](#): Run incremental syncs.
  - [Specify a date](#): Run a full sync starting from the specified date.

### Note

Full sync can have negative impact on performance so it's recommended that you use this option only once and not recurringlly.

5. Select the appropriate options for [Automatic Manager Transfer](#).

Select...	If enabled...
<b>Automatic insertion of new manager as next document recipient if not already</b>	The new manager becomes a part of the review process and the old manager is removed from any further accountability.
<b>Automatic Inbox Document Transfer to New Manager</b>	All the documents are moved from the old manager's Inbox to the new manager's Inbox.
<b>Automatic En Route Document Transfer To New Manager</b>	All the documents are moved from the old manager's En Route folder to the new manager's En Route folder.
<b>Automatic Completed Document Copy to New Manager</b>	All the documents are moved from the old manager's Completed folder to the new manager's Completed folder.
<b>Automatic Process Owner Change To New Manager For In-Progress Documents When Old Manager is Process Owner (Only for 360)</b>	The process owner is automatically changed from the old manger to the new manager, when the in-progress forms are transferred to the manager.
<b>Automatic Process Owner Change To New Manager For Completed Documents When Old Manager is Process Owner (Only for 360)</b>	The process owner is automatically changed from the old manger to the new manager, when the completed forms are transferred to the manager.

6. Set up [Occurrence](#) of the job to be [Once](#) or on a [Recurring](#) schedule.

## Related Information

[Monitoring Scheduled Jobs in Admin Center](#)

### 12.2.2 Sync Order for an HRIS Sync Job

The HRIS Sync job syncs entities in a particular order. There could be scenarios where an entity synced earlier could be overwritten by an entity synced later.

Table 3: Syncing Order

HRIS Sync Element	hris-element-id
Phone	phoneInfo
Email	emailInfo
Business Address	corporateAddress
Salary Change	complInfo
Personal Info sync	personalInfo
Person Info sync	personInfo
Generic Entity	homeAddress nationalIdCard
Employment Info	employmentInfo
Status Change	jobInfo (emplStatus)
Manager Change	jobInfo (managerId)
Job Information Change	jobInfo
User Info Elements save for custom sync.	-

#### ❖ Example

addressLine1 field is set up to be synced from homeAddress.address1 in HRIS Sync mapping. addressLine1 is also hard-coded mapped from corporateAddress.address1. When you run the Full HRIS Sync job, which selects both "Generic Entities" entity and "Job Information," corporateAddress.address1 is always synced to the employee profile, meaning homeAddress.address1 is always overwritten.

## 12.3 HRIS Element and Fields Sync Logic

The system supports hard-coded HRIS sync. This means that the system will sync some HRIS elements and HRIS fields into user directory tables without any configuration based on hard-coded rules, such as syncing HRIS field: job-code in HRIS element: jobInfo to standard element: jobCode.

However, you also have a flexible configurable framework to sync HRIS fields back to user directory tables where you can specify HRIS fields to map to standard elements by configuring the Succession Data Model.

HRIS Entity	HRIS Element	Comments	Supported Sync Mapping Configuration	Included Hard-coded Sync Logic	Full sync/ Incremental sync
email	emailInfo	Business email has hard-coded sync logic; other type of email does not have hard-coded sync logic.	Yes	Partly	Full
phone	phoneInfo		Yes	Yes	Full
address	homeAddress		Yes	No	Incremental
corporate address	corporateAddress	Used in location address (HRIS element: corporateAddress itself is in corporate data model.)  1) When employee changes location, the corporateAddress will be synchronized together with other jobInfo change;  2) When location info is changed, the corporateAddress will be synchronized when HRIS sync job runs.	Yes	Yes	incremental

HRIS Entity	HRIS Element	Comments	Supported Sync Mapping Configuration	Included Hard-coded Sync Logic	Full sync/ Incremental sync
personal	personalInfo	Name related fields will be synchronized from personalInfo.	Yes	Yes	incremental
person	personInfo		Yes	Yes	Full
job	jobInfo	Direct manager change is synchronized when the job records are synchronized.	Yes	Yes	incremental
job relationships	jobRelationsInfo	An employee might have multiple job relationships, for example, three active matrix managers. In such cases, the job relationships are synchronized to user directory tables in the ascending order of RELATIONSHIP_TYPE (the OptionId of the JobRelationships picklist).	No	Yes	incremental
employee status	jobInfo		No	Yes	incremental
comp	complInfo		Yes	Yes	incremental
employment	employmentInfo		Yes	No	full
social account	imInfo		Yes	No	full
national ID Card	nationalIdCard		Yes	No	full
work eligibility	workEligibilityInfo		Yes	No	full

HRIS Entity	HRIS Element	Comments	Supported Sync Mapping Configuration	Included Hard-coded Sync Logic	Full sync/ Incremental sync in DM?
entities that support sync with limitations	more entities	The following entities theoretically support sync in a generic way, but sometimes the generic way is not suitable and need some special handling to suit the real needs for sync in future if needed: globalInfo; payComponentRecurring; payComponentNonRecurring; EmpDocumentInfoEO (workPermitInfo; visalInfo; citizenshipInfo )	Yes	No	
entities which don't support sync	entities which don't support sync	The entity below does not support sync:  EmpEmergency-ContactInfoEO	No	No	

Clients don't need to configure all these fields in the Succession Data Model. The system automatically syncs the related HRIS data into user directory tables based on the mapping shown above.

## 12.4 Hard-Coded Sync Mapping

Certain mappings are hard-coded into the system. Here is a list of those mappings.

Hard-coded fields (by default) are synchronized from Employee Central to basic user information fields in the SAP SuccessFactors platform, which means that you **must not** configure HRIS Sync mapping for these fields in the Succession Data Model. Configuring an HRIS Sync mapping in the data model for these hard-coded fields can cause undesired results.

The system prevents anyone from adding duplicate sync mappings to the sync-mappings section of the Succession Data Model. Duplicate sync mappings lead to data not syncing to Employee Profile correctly.

To guarantee seamless integration between Employee Central and other SAP SuccessFactors modules, Employee Profile is updated with information stored in Employee Central. Some Employee Profile attributes are

updated online when changes are recorded in Employee Central data. Other attributes are only updated when the HRIS synchronization process is run. Employee Central changes that are entered as of a future date are reflected in Employee Profile as soon as the HRIS synchronization process is run after the future date has been reached.

HRIS Element	HRIS Field	Standard Element	Overwriting Allowed?	Comment
phoneInfo	country-code	homePhone	Yes	Phone Info
	area-code	homePhone	Yes	Only sync phoneInfo records when the phone type is configured as a picklist and the option external code equals H or B or C or F.
	phone-number	homePhone	Yes	
	extension	homePhone	Yes	
	country-code	cellPhone	Yes	
	area-code	cellPhone	Yes	
	phone-number	cellPhone	Yes	
	extension	cellPhone	Yes	
	country-code	businessPhone	Yes	
	area-code	businessPhone	Yes	
	phone-number	businessPhone	Yes	
	extension	businessPhone	Yes	
	area-code	fax	Yes	
	area-code	fax	Yes	
	phone-number	fax	Yes	
	extension	fax	Yes	
emailInfo	email-address	email	Yes	Email Info
employmentInfo	start-date	hireDate	No	Employee Status
	end-date	companyExitDate	Yes	
jobInfo	emplStatus	status	No	
	manager-id	managerId	No	
jobRelationsInfo	rel-user-id (picklist external_code - hr manager)	hrId	No	Job Relationship

HRIS Element	HRIS Field	Standard Element	Overwriting Allowed?	Comment
	rel-user-id  (picklist external_code - matrix manager)	matrixManager	No	
	rel-user-id  (picklist external_code - second manager)	secondManager	No	
	rel-user-id  (picklist external_code - custom manager)	customManager	No	
jobInfo	job-code	jobCode	No	Job Information
	job-title	title	Yes	
	company	country	Yes	
	department	department	No	
	division	division	No	
	location	location	No	
	timezone	timezone	Yes	
	emplStatus	status	No	
	paygrade	paygrade	Yes	
	manager-id	manager-id	No	
corporateAddress	address1	addressLine1	Yes	Business Address
	address2	addressLine2	Yes	
	address3	addressLine3	Yes	
	city	city	Yes	
	state	state	Yes	
	zip-code	zipCode	Yes	
	country	country	Yes	
personallInfo	first-name	firstName	Yes	Personal Information
	last-name	lastName	Yes	
	middle-name	mi	Yes	

HRIS Element	HRIS Field	Standard Element	Overwriting Allowed?	Comment
	salutation	salutation	No	
	suffix	suffix	No	
	gender	gender	No	
	ethnic-group	ethnicity	Yes	
	martial-status	married	No	
	preferred-name	nickname	Yes	
	title	title	Yes	
	nationality	nationality	Yes	
	visible-minority	minority	No	
personInfo	date-of-birth	dateOfBirth	No	Biographical Information
personalInfo				
nationalIDInfo	national-id	national ID	No	National ID Information

#### i Note

Don't configure username as a standard-element-ref.

Do NOT map the national ID or minority fields to custom fields.

#### i Note

Never attempt to map the field Manager (hard-coded). Doing so causes issues in the organization structure.

## Overwrite Hard-Coded Sync Mapping

If required, it is possible to rewrite the hard-coded sync mapping. However, it is not recommended.

For example, in the hard-coded sync mapping, Employee Central data: jobInfo department to standard element: division. You can configure it as follows:

```
<hris-element-ref refid="jobInfo">
    <hris-mapping>
        <hris-field-ref refid="department"/>
        <standard-element-ref refid="division"/>
    </hris-mapping>
</hris-element-ref>>
```

## Related Information

[Adding Sync Mapping for Standard Field](#)

## 12.5 HRIS Sync Mapping (hris-sync-mappings) and Configuration Validation Rules

You can modify the Succession Data Model to include HRIS Sync mapping.

Using HRIS Sync mapping, you map Employee Central fields to the user data fields you wish to push data to. You would configure a sync mapping for a hard-coded sync mapped field only if you wish to override the system driven mapping. Here's some sample code showing how you would map a standard-element or a userinfo-element or a user-info-record-key to an hris-field:

```
...
    </edit-template>
</view-template>
<hris-sync-mappings>
    <hris-element-ref refid="phoneInfo">
        <hris-mapping entity-type="H" >
            <hris-field-ref refid="custom-long2"/>
            <standard-element-ref refid="custom02"/>
        </hris-mapping>
    </hris-element-ref>
    <hris-element-ref refid="jobInfo">
        <hris-mapping >
            <hris-field-ref refid="company"/>
            <user-info-record-key>user-company</user-info-record-key>
        </hris-mapping>
        <hris-mapping >
            <hris-field-ref refid="employee-class"/>
            <userinfo-element-ref refid="employeeClass"/>
        </hris-mapping>
        <hris-mapping >
            <hris-field-ref refid="timezone"/>
            <standard-element-ref refid="timeZone"/>
        </hris-mapping>
    </hris-element-ref>
    ...
<hris-sync-mappings>
```

### Configuration Validation Rules

- HRIS fields with visibility="none" aren't synced. The rule is applied to both hard-coded sync logic and configurable sync mapping.
- `<hris-sync-mappings>` must be put after `<view-template>` definition in the Succession Data Model (SDM).
- Zero or more `<hris-sync-mappings>` can be defined in a Succession Data Model.
- Under `<hris-sync-mappings>`, you can define one or more `<hris-element-ref>`.
- Under `<hris-element-ref>`, you can define one or more `<hris-mapping>`.
- Each `hris-field-ref` can only be mapped to one `standard-element-ref` or one `userinfo-element` or one `user-info-record-key`.

## i Note

Enable the *Edit* permission in role-based permissions (RBP) for the `userinfo-elements` that you've configured in `hris-sync-mapping`. The sync process only pushes data into `userinfo-elements` with the *Edit* permission.

The `user-info-record-key` is used by other modules that need additional information for integration and that are accessed through an application programming interface (API). The `user-info-record-key` that is stored in the user directory is consumed only through API; the key values aren't displayed on any user interface.

You can enter any arbitrary string value for the `user-info-record-key` in the Succession Data Model, so it isn't a `refid`. Whatever value you use here is used as a key in the user directory.

- `entity-type` attribute is used for address, e-mail, phone, and `globalInfo` to specify the type. For HRIS-mapping configuration of address, e-mail, phone, and `globalInfo`, `entity-type` is a mandatory attribute.
- You can use the XML attribute `date-format` to sync dates from Employee Central to the Employee Profile. The `date-format` allows you to also sync only parts of the date. For example, if you want to show only parts of the birthday information without showing the age information, you can sync only the day and month fields, but not the year.

## i Note

You can only use the following date formats, which are case-sensitive:

- Year in four digits: `yyyy`
- Month and year: `MMM-yyyy`
- Month: `MMM`
- Date and month: `dd/MMM`

- You can only sync date type HRIS field to a string type standard element.
- The existing hard-coded syncing of an employee's date of birth from Employee Central (HRIS field `date-of-birth`) to the Employee Profile (standard-element `dateOfBirth`) isn't affected by the `date-format` syncing. The system syncs the complete date.
- All `refid` attributes must be valid. The fields used for mapping are fields in the Succession Data Model or the Corporate Data Model. For example `corporateAddress` is defined in the Corporate Data Model while others in the code examples here are defined in the Succession Data Model.
- Duplicated HRIS-element-ref `refid` definition fails validation.
- If fields fail data type validation (such as mapping string to date), the Succession Data Model (SDM) can't be imported successfully (mapping anything to string is acceptable).
- If a standard element field being mapped is a picklist, the Employee Central HRIS field must be a picklist or a foundation object or a territory (country) object.
- If the HRIS field is a picklist, make sure the field it's mapped to has an identical picklist id.

## 12.6 Picklist Configuration for Employee Status and Job Relationship Type

For the sync to work properly, both the job relationship type and employee status picklists must have the correct external codes.

There are two picklists that are very important for sync:

- Job Relationships

For the jobRelationshipInfo HRIS element, the relationship-type HRIS field must be defined as an Employee Central picklist in the data model. In order to sync the known relationship types correctly into users' legacy tables, the dedicated external codes for widely known relationship types are defined. The sync logic will regard the external code for each known relationship type as a fixed value. The system will run the different sync logic (HR manager/matrix manager /custom managers/second manager) based on the external code. The table below shows the definition of external code for each known relationship type. It has fixed values in our system. So, if the client needs to support some or all of the known relationship types, they need to define the external code for the picklist option.

Existing customers can manually add the new job relationship types to their picklist and re-import it to have the new job relationships in the system.

For more information about Job Relations, see [Additional Information for Job Relationships](#) in the Employee Central Master guide on the SAP Help Portal.

Once the sync has run, you can update the permissions for the job relationships. For more information, see [Delegate Relationship Assignments](#) in the Using Role-Based Permissions guide on the SAP Help Portal.

- Relationship Types

External Code	Relationship Name
hr manager	HR Manager
second manager	Second Manager
matrix manager	Matrix Manager
additional manager	Additional Manager
custom manager	Custom Manager
delegate 1	Delegate A
delegate 2	Delegate B

### i Note

Do not configure multiple picklist labels that point to the same external code.

- Employment status

Employment status needs to be defined as a picklist. Default picklist ID is employee-status if clients do not explicitly define otherwise in the corporate data model. You can define the picklist ID for HRIS field "emplStatus" in HRIS element "eventReason" in the corporate data model. The external code of

employment status is important for deciding the employment status. Here's a table, showing some mapping between external code and employment status:

External Code	Employment Status in EC	users_sys_valid flag in Legacy Table
A	Active	t
U	Unpaid Leave	t
P	Paid Leave	t
S	Suspended	t
F	Furlough	f
R	Retired	f
T	Terminated	f

Any mapping other than the above will be regarded as inactive in legacy tables.

### i Posting Instructions

Once the HRIS sync is run, you can set up job relationships between employees in the system.

## 12.7 Special Handling for Syncing Fields

Here is some further information about the fields to be synced.

- **Logic about entity-type is Primary**

The validation rule is used when you are importing Succession Data Model: For HRIS-mapping, configuration of address, email or phone, *entity-type* is a mandatory attribute. If there are records with *isPrimary=true* for the specific entity type of email or phone for a specific person, the system will just sync the record; for example, for business email of user01, record 123: business email is primary, then just record 123 business email not other business email records for user01 will be synced.

If no records exist with *isPrimary=true* for the specific entity type of email or phone for a specific person, the system will sync all the records of the entity type, the last record will win; for example, for 3 business email records of user01, if none of the three business emails is labeled as primary, then the three business email records for user01 will be synced. *isPrimary* field is valid for email, phone, and nationalIdCard.

If there is EmpNationalIdCardEO record with *isPrimary=true* for a specific person, the system just syncs the record with *isPrimary=true*.

If there is NO EmpNationalIdCardEO record with *isPrimary=true* for a specific person, the system just syncs all the records.

- **Country**

If HRIS field is country field, sync country name to user directory tables.

- **Phone number**

If users configure phone-number in phoneInfo for <HRIS-sync-mappings>, 4 fields countryCode, areaCode, phoneNumber, extension will be merged into one value: (countryCode) areaCode

phoneNumber'x'extension and sync to the mapped standard element, for example, (086) 021 21345501x0619.

- **Picklist**

If the EC HRIS field is a picklist but the standard element field isn't a picklist, the picklist label will be synced to user directory tables.

If the EC HRIS fields is a picklist and the standard element field is also a picklist, the option ID will be synced to standard element field. (option ID > option ID)

If standard element is a picklist and HRIS field is not a picklist, synchronization is not supported.

- **Foundation objects**

If foundation objects are enabled in Provisioning, the FOs will be converted into FOName (FOExternalCode) and synced to user directory tables, for example, engineer dept (eng).

- **Others**

- If standard element is gender and HRIS field is null, sync '' into standard element: gender.
- If HRIS field is Gender, convert gender into its legacy value.
- If HRIS field is MaritalStatus, convert marital status into its legacy value.
- Otherwise if HRIS field is null, null value will be synced back to user directory tables.

 **Caution**

To avoid future data inconsistency between Employee Central tables and legacy tables, do not use basic import to directly update data in the legacy tables.

## 12.8 Syncing the Termination Date Between Employee Central and Standard User Fields

Set up HRIS sync mapping between Employee Central and the standard user field `<companyExitDate>` so that you can use the DRTM data purge function to purge inactive users from the system.

### Prerequisites

You are an administrator with access to the Business Configuration UI.

The standard element `<companyExitDate>` is already enabled in your data model.

### Context

HRIS sync mapping for the termination date is not hard-coded, so you have to map the relevant fields between Employee Central and the SAP SuccessFactors platform. If this sync is not set up correctly, the data purge function cannot work correctly.

If the standard element `<companyExitDate>` is not present in your Employee Export file, it is not enabled in your system and you cannot complete this task. You need to add this field to your system first.

If you do not have access to the Business Configuration UI in your system, you can also submit a request to SAP Cloud Support to have the following XML added to your data model in the Provisioning application:

#### ↳ Sample Code

```
<hris-element-ref refid="employmentInfo">
    <hris-mapping>
        <hris-field-ref refid="end-date"/>
        <standard-element-ref refid="companyExitDate"/>
    </hris-mapping>
</hris-element-ref>
```

## Procedure

1. Go to  [Admin Center](#)  [Manage Business Configuration](#).
2. Go to  [Employee Central](#)  [HRIS Elements](#)  [employmentInfo](#) in the navigation pane.
3. Under *HRIS Fields*, find the row with `<end-date>` in the *Identifier* column.
4. In the row for `<end-date>`, click *Details* and scroll to the *HRIS Sync Mapping* section in the dialog window.
5. Use the *Standard Field* search box to find and select `<companyExitDate>`.  
If you do not see `<companyExitDate>` in the search box, it is not enabled in your system. You need to add it before you can complete this task.
6. Leave the *Entity Type* field blank.
7. Select *Done* and then save your changes.

## Results

The effective-dated end date of an employment in Employee Central is now mapped to the user's company exit date in the SAP SuccessFactors platform. This ensures the employment end date in Employee Central is used to calculate data retention times.

## Next Steps

After the sync mapping is added, make sure that the user (userId) used for HRIS Sync is granted View and Edit permissions for this field.

# 13 Block-Specific Information

## 13.1 Additional Information for Addresses

Here is a little more information about some of the features and functions in Employee Central.

### Simple Address Format

In systems enabled with People Profile, you can define a compact address format for the Overview page. This is called the Simple Address Format.

The default address format will be displayed for all countries. If you want other fields to be displayed, then you can create the Simple Address Format for each country/region to display only specific fields.

1. Navigate to the  [Admin Center](#) .
2. In the [Create New](#) field, select *SimpleAddressFormat*.
3. Enter the required information:
  - externalCode
  - effectiveStatus
  - Country
4. Then enter how the address should be displayed for the standard format and the alternates. Enter the field name from the data model with an @ (at) symbol in front, where each field is separated by commas.  
For example: @address1, @address2, @city, @state, @zip-code

#### SimpleAddressFormat: Aus Test (AusTest)

```
externalName Aus Test
* externalCode AusTest
* effectiveStatus Active
* country Australia (AUS) 
standardDisplayFormat @address1, @address2, @city, @state, @zip-code
standardDisplayFormatAlt1 @address1, @address2, @city, @state, @zip-code
standardDisplayFormatAlt2 @address1, @address2, @city, @state, @zip-code
```

5. Save your changes.

For more information, refer to [Creating the Simple Address Format](#)

## Time Gap in Addresses

The address is an effective-dated entity, meaning that as of the rehire date, you are making a change, and deleting the address record as of that date. So the address block will capture that change as effective from rehire date, which is why you see a blank record in address info as of rehire date.

In cases where an employee changes or deletes any address information, or has no address for a specific amount of time, then this can be reflected in the *History* panel.

For example, User A was hired on 01/01/2014 and most likely starting on that same date, so there will be a home address record that starts at same date as the hire date. Now, the employee moves from the current home address from 01/01/2016. Then employee edits for the address with an effective date of 01/01/2016 and deletes the home address. (The employee address block has no other address type). Now, there will be no address shown in the UI - it will display only empty block information.

In Addresses History, employees can undo the changes they've made by deleting the corresponding change history entry. However, the deletion action can't be undone, therefore you can't restore a deleted address record.

## Adding Attachments

To allow employees to upload attachments, you need to enable the HRIS field in the Business Configuration UI.

1. Navigate to the [Admin Center](#)  [Manage Business Configuration](#) .
2. Select [Employee Central](#)  [homeAddress](#) .
3. Set the attachment-id field to *Enabled*.
4. Save your settings.

### i Note

Since the *Dependents* block also has a *Address* section in its *Details* pop-up, the attachment field will be also shown there. To disable the *Address* attachment in the *Dependents* pop-up, change the settings in the [Manage Business Configuration](#) page.

## Validations

A new (optional) validation prevents the deletion of the home-address (or any other subtype of addresses that are used for replication in a payroll system). This is controlled by a configuration in the [Company System and Logo Settings](#)  [Protect address from deletion](#)  that allows the Address subtype to be protected against deletion. If entered, this subtype is protected from deletion. All other subtypes work as before.

## Rules for Addresses

**i Note**

onInit and onView rules do not work with homeAddress.

## 13.2 Additional Information for Biographical Information

Here is a little more information about some of the features and functions in Employee Central.

### Adding Attachments

To allow employees to upload attachments, you need to enable the HRIS field in the Business Configuration UI:

1. Navigate to the [Admin Center](#)  [Manage Business Configuration](#) .
2. Select [Employee Central](#)  [personInfo](#) .
3. Set the attachment-id field to *Enabled*.
4. Save your settings.

**i Note**

Since the *Dependents* block also has a *Biographical Info* section in its *Details* pop-up, the attachment field will be also shown there. To disable the *Biographical Info* attachment in the *Dependents* pop-up, change the settings in the [Manage Business Configuration](#) page.

## 13.3 Additional Information for Employment Information

Here is a little more information about some of the features and functions in Employee Central.

### General

The data displays a user's employment information, including the company and start date. Employment Information is effective dated, meaning it shows the start and end date for the information.

## Person Type

You can create a data model for the Employee person type for Employment Information. They are then taken into account for the Profile, Take Action, Workflow, and History.

For more information about the supported person types and the overall concept, refer to the [Setting Up and Using Business Configuration UI \(BCUI\)](#) guide on the SAP Help Portal.

## 13.4 Additional Information for Job Information

Here is a little more information about some of the features and functions in Employee Central.

### General

As part of Employment Information, Job Information stores data related to an employee's function within the company. It is defined during the hiring process. It is an effective-dated entity and no gaps are allowed, meaning that an employee must always have a current Job Information record. All changes to records are available in the history. Multiple changes for each day using sequence numbers are allowed. Changes to an employee's data should be done using the [Take Action](#) menu options rather than from the [History](#) UI to ensure that all follow-on processes are triggered and prevent data inconsistencies. The employment status of the user controls which actions can be done, for example, a user with the status can't be terminated or book time off.

It is partially configurable in the Business Configuration UI (BCUI). It can be defined either globally, country specific, or person type specific.

It is available in the Manager Self-Service scenario.

Rule contexts, events, and field-level permissions are supported.

### Expected Return Date

For employees on a leave of absence (LOA), you can define an expected return date. This field can be enabled and made visible in either the Succession Data Model or in the Business Configuration UI. Once enabled, you can see the [Expected Return Date](#) field in [Job History](#) for records with the Paid Leave and Unpaid Leave event (these event reasons that start a leave of absence).

To see the field in Advanced Reporting, you must set the visibility to [Both](#).

Permissions must also be enabled for this field in [Permission Settings](#) [User Permissions](#) [Employee Central Effective Dated Entities](#) .

## Event Reason

Event Reason is a system hard-coded field and therefore is not enabled or configured in the data model. However, if you need to trigger onChange business rules from the *Event Reason* field, you must enable the `<event-reason>` field in either the Succession Data Model or in the Business Configuration UI.

## Job Code Propagation

You can propagate job code values to the Job Information block from the Work Schedule to allow admins to choose custom codes for the company.

Update the Job Code object definition so that the custom string has the following settings:

- Data Type: Generic Object
- Valid Values Source: Work Schedule

This ensures that the selectable values for the Work Schedule are then identical in the job code instances as well as in the Job Information block.

You must then update the Work Schedule values for the different job codes in the  [Manage Data](#)  [Job Classification](#) screen.

## Position to Job Information Sync

You can sync field values from position to Job Information using business rules.

For more information, refer to [Define Synchronization Position to JobInformation](#).

## 13.5 Additional Information for Job Relationships

Here is a little more information about some of the features and functions in Employee Central.

### General

Job relationships can show hierarchical relationships, meaning there is a reporting line between the granted user and the target user. These are job relationships between employees and their managers as well as employees and their second managers or alternate managers. However, job relationships can also show non-hierarchical relationships, which are single-level relationships. These include the relationship of an employee to the HR manager, the matrix manager, additional manager, and custom manager.

The job relationships are either entered into the system during the new hire process or during an import. The standard relationships can be used by the system to route workflows. This means that customer-defined job relationships are not supported for workflow routing.

Job relationship records are effective-dated records to cover the employment history from hire to termination, although, gaps are allowed. Making multiple changes to the records each day is not supported.

It is partially configurable in the Business Configuration UI (BCUI). It must be defined globally, since a country-specific is not supported.

Here is the list of possible relationships:

- HR Manager
- Second Manager
- Matrix Manager
- Additional Manager
- Custom Manager
- Delegate 1  
Someone who can act on behalf of the manager against all of their direct reports excluding the manager.
- Delegate 2  
Someone who can act on behalf of the manager against all of their direct reports excluding the manager.

Existing customers can manually add the new job relationship types to their picklist and re-import it to have the new job relationships in the system.

Job relationship entries must be synced between Employee Central and the Employee Profile. For more information, refer to [Picklist Configuration for Employee Status and Job Relationship Type](#)

You can update a job relationship from the employee's profile by going to  [Take Action](#)  [Change Job and Compensation Info](#). Then under [Change Job and Compensation Info](#), select [Job Relationships](#). Select the new relationship and save your changes.

Relationships between positions can also be defined in the position org chart. These relationships can be synced automatically into job relationships for position incumbents as well. For more information, refer to [Define Synchronization Position to JobInformation](#).

## Target Groups for Workers

Fields of the type [Worker](#) (for example, supervisor in Job Information or HR/matrix manager in Job Relationship, and so on) now respect target groups defined in permissions. This means that, if configured, users can only add managers that are included in the target group defined in the permissions.

To enable this feature, please go to  [Admin Center](#)  [Company System and Logo Settings](#) and select the feature [Enable target group based filtering for Worker fields](#). If checked, Worker type fields value dropdown list will based on the target group settings in role based permission. If not checked, all users will be available in the dropdown list.

## 13.6 Additional Information for National ID

Here is a little more information about some of the features and functions in Employee Central.

### Temporary IDs

Admins can provide an employee's temporary national ID in the National ID Information if that employee does not have a valid national ID during the hire process.

For more information, refer to [Using Temporary National ID While Hiring an Employee](#) in the Employee Central Imports guide on the SAP Help Portal.

### Adding Attachments

To allow employees to upload attachments, you need to enable the HRIS field in the Business Configuration UI:

1. Navigate to the [Admin Center](#)  [Manage Business Configuration](#) .
2. Select [Employee Central](#)  [nationalIDcard](#) .
3. Set the attachment-id field to *Enabled*.
4. Save your settings.

#### i Note

Since the *Dependents* block also has a *National ID* section in its *Details* pop-up, the attachment field will be also shown there. To disable the *National ID* attachment in the *Dependents* pop-up, change the settings in the [Manage Business Configuration](#) page.

## 13.7 Additional Information for Personal Information

Here is a little more information about some of the features and functions in Employee Central.

### Special Characters in Picklists

Special characters are not supported in picklist IDs for Personal Information.

## Validations for Telephone Numbers

For telephone numbers in Personal Information, we have now introduced validations to prevent employees from adding non-printable characters at the beginning or end. Empty spaces before or after the number are now automatically removed when the number is saved.

These validations are not run against existing telephone numbers in the system. You do not have to update all existing telephone numbers.

## Behavior with BCUI

Personal Information can be configured in both the Succession Data Model (SDM) and the Business Configuration UI (BCUI). The configurations should be the same.

However, an admin can create a customized version in the BCUI, for example, `personallInfo_employee`. If `personallInfo_employee` exists in BCUI, then Personal Information block on the employee profile will refer to the configuration of `personallInfo_employee` in BCUI only rather than that of `personallInfo` in SDM.

## Adding Attachments

To allow employees to upload attachments, you need to enable the HRIS field in the Business Configuration UI:

1. Navigate to the [Admin Center](#)  [Manage Business Configuration](#) .
2. Select [Employee Central](#)  [personallInfo](#) .
3. Set the attachment-id field to *Enabled*.
4. Save your settings.

### i Note

Since the `Dependents` block also has a `Personal Info` section in its `Details` pop-up, the attachment field will be also shown there. To disable the `Personal Info` attachment in the `Dependents` pop-up, change the settings in the [Manage Business Configuration](#) page.

## Global Information

If the `GlobalInfo` hris-element is configured in the Succession Data Model (SDM), it is part of the Personal Information block. However, there are no hris-fields in the SDM for `GlobalInfo`. For each country/region-specific field needed, fields for the `globalInfo` hris-element can be added in the Country/Region-Specific Data Model.

OnInit rules can be configured under both SDM or CSF-SDM. However, with respect to execution, only one rule for each country/region is executed. You can have different rules for different countries in the CSF-SDM.

However, if there is more than one rule for the same country/region, then only one rule is executed. If multiple rules are needed, the logic must be set within the same rule.

## Removing Flags

In the Global Information section in the Personal Information block, the flag of the country/region is displayed next to the country/region name by default.

For more information, refer to [Hiding Country/Region Flags](#).

## 13.8 Additional Information for Primary Emergency Contact

Here is a little more information about some of the features and functions in Employee Central.

### Copy Address from Employee

For the [Copy Address from Employee](#) field, the address is copied from the employee's home address, even if there are multiple address types listed for the employee.

### Mandatory Field Validation

You can set up the system to validate all mandatory fields for emergency contacts.

By default, mandatory field validation for emergency contacts covers no more than the following fields, depending on your field configuration:

- Relationship
- Name
- Phone
- Primary
- eMail

However, you can extend the validation to cover mandatory fields available in the details section of the [Emergency Contact](#) block.

To extend the validation, you can:

1. Navigate to the  [Admin Center](#)  [Company System and Logo Settings](#).
2. Select [Enable validations for mandatory fields in the Emergency Contact's details section](#).
3. Save your changes.

## Duplicate Emergency Contacts

Avoid creating duplicate records for emergency contacts. You would expect errors if adding emergency contacts with the same name and relationship for an employee.

### i Note

You can add alternate fields, such as the *Alt Phone* field, for an emergency contact in one record instead of duplicating the contact. For more information about the fields for the *Emergency Contact* block, refer to [Emergency Contact](#).

## 13.9 Additional Information for Work Permits

Here is a little more information about some of the features and functions in Employee Central.

### Editing Work Permits

When you edit a work permit in [People Profile](#), be sure to comply with the following requirements:

- First, go to [Manage Business Configuration](#) to enable Country, Document Type, and Document Number in order to view and edit these fields.
- These three fields on the UI are mandatory and must not be left empty.
- Business keys are generated by concatenating these three fields and User ID. A business key must be unique. Do not enter a record with duplicate Country, Document Type, and Document Number values (at least one of them should be unique).

### Adding Attachments

To allow employees to upload attachments, you need to enable the HRIS field in the Business Configuration UI:

1. Navigate to the [Admin Center](#) [Manage Business Configuration](#).
2. Select [Employee Central](#) [workPermitInfo](#).
3. Set the attachment-id field to [Enabled](#).
4. Save your settings.

## **Behavior with Global Assignments**

When Global Assignments are active in the system, the workPermitInfo block moves from Employment Information to Personal Information.

# 14 Forward Propagation in Employee Central

Forward propagation means that a change in the value of a field in an object is also made ("propagated") to future records for the same object. The forward propagation of this field change stops as soon as one of the future records has a field value maintained that is different than the original field value.

Forward propagation is supported in the following:

- MDF Objects

## i Note

Forward propagation is NOT supported for custom MDF objects.

- Job Information
- Job Relationship
- Compensation Information

## i Note

Rules are not triggered for propagated time slices.

## i Note

The default language is propagated to fields such as job classification. This means that the logon language is not taken into account.

## Example

There is a future change for an employee where they have a promotion consisting of a grade change already entered into the system. They transfer into a new department 1 month before the promotion takes effect. The change to the department should be made both before and after the date of the promotion.

## Example

There is a future change for an employee where they have a promotion due to a transfer to a new division and department including a grade change already entered into the system. Their location stays the same. They are then moved into a different department and their location changed as a part of an office reorganization 1 month before the transfer/promotion takes effect. The change to the department and location should be made before the promotion/transfer and the department change will stop correctly for the promotion/transfer however the location change will continue to be propagated.

## Example

There is a future dated change where an employee is changing location and departments. The location is changed 1 month before the department change takes effect. The change to location should be made both before and after the transfer of the department.

## Job Information

Forward propagation for Job Information is on by default in the system and cannot be switched off.

You must also enable forward propagation for imports. For more information, refer to the [Forward Propagation in Employee Central Imports](#) topic.

Changes to job information should be propagated when:

- Changes are made in the MSS screen
- Changes are made using [Insert New Record](#) on the [History of Job Information](#) screen

Changes to job information are **NOT** propagated when:

- Corrections are done in the [History of Job Information](#) screen
- Job history records are deleted in the [History of Job Information](#) screen

## Job Relationship

Forward propagation for Job Relationship is on by default in the system and cannot be switched off.

Changes to job relationship information should be propagated when:

- Changes are made in the MSS screen
- Changes are made using [Insert New Record](#) on the [History of Job Relationships](#) screen
- Changes are made to name or relationship type

Changes to job relationship information are **NOT** propagated when:

- Corrections are done in the [History of Job Relationships](#) screen
- Job relationship records are deleted in the [History of Job Relationships](#) screen

## Compensation Information

Forward propagation for Compensation Information can be turned on in the [Admin Center](#) [Company System and Logo Settings](#).

### i Note

Propagation stops for all fields for recurring pay components as soon as a future record is found in which one field does not match the original value.

### i Note

Business rules are applied **only** once and the changes resulting from these rules are propagated.

Changes to Compensation Information (HRIS element: comlInfo) are propagated individually, whereas recurring pay components (HRIS element: payComponentRecurring) are propagated as a group. This is due to the fact that pay components consist primarily of a group of field whose values make sense only as a group (amount + currency + frequency or number + currency + frequency + unit + custom fields) so that it is reasonable to stop propagation entirely when one field changes.

When a change to a recurring pay component is propagated, the time stamp and user of the last change to the Compensation Information record is also updated. In case recurring pay components are propagated, the field 'calculated amount' is recalculated for the future recurring pay components instead of being propagated.

Changes to compensation information should be propagated when:

- Changes are made to the compensation data using MSS
- Changes are made using *Insert New Record* on the *History of Compensation Information* screen
- Changes are made to the compensation data by cross-block rules

### i Note

These changes are propagated as if they were made using the compensation UI; that is, the rules are applied once and the changes are propagated independently of the end date of the original changes to job info.

- Changes are made to the compensation data in Benefits
- Changes are made to the compensation data using Pay Scale Pay Increase  
Pay Scale Pay Increase reads and processes future compensation record independently of the settings made in the ► *Admin Center* ► *Company System and Logo Settings* ▶. Benefits processing reads future records of compensation information and changes them accordingly independently of the settings made in the ► *Admin Center* ► *Company System and Logo Settings* ▶.

## Example

A company may have 2 pay components, one for full-time employee and one for part-time employees. The employees have both pay components in their compensation information data. The amount for the part-time pay is calculated using a business rule based on the full-time pay amount and the employee's FTE.

In this example, an employee has a future change from full-time to part-time (80%) on April 1st at a 20% salary reduction.

Pay Component/Month	JAN - FEB	MAR	APR-JUN
FT	5000	5000	5000
PT	5000	5000	4000

The employee receives a pay increase on March 01. The full-time pay component is changed manually from 5000 to 5050 and the part-time pay component is adjusted using a business rule from 5000 to 5050 (as the employee is still working full time in March).

The system tries to propagate the following changes:

1. Pay component FT changes from 5000 to 5050

Since the record for April for this pay component has the same original value (5000), the pay component can be changed.

2. Pay component PT changes from 5000 to 5050

The record for April for this pay component has the value 4000, which differs from the original March value. The change cannot be propagated.

After the change and propagation, the employee has the following pay components:

Pay Component/Month	JAN - FEB	MAR	APR-JUN
FT	5000	5050	5050
PT	5000	5050	4000

# 15 Deep Links in Employee Central

This section lists the deep links available for Employee Central.

A deep link is a direct link to a page, in which the URL contains all the information needed to go that page rather than having to navigate to the page from the Home screen.

For more information, refer to the [SuccessFactors Deep Links](#) guide on the SAP Help Portal.

Link	Description	Parameters
/sf/employmentinfo	Takes the user to the Employment Info page	selected_user(optional) = user sys id.
/sf/employeeupdate	Takes the user to the Update Employee Records page	selected_user(optional) = user sys id.
/sf/orgchart?type=position	Takes user to the Position Org Chart	selected_position=<position external code>  (selected_user will contain the person id of a user)
/sf/personallInfo	Takes the user to the Personal Info page	selected_user(optional) = user sys id.
/sf/employeeterminate?selectquestion=essMssTerminateActionController&selected_user=<username>	Takes the user to the Terminate/Retire page	selected_user(optional) = user sys id.
/xi/ui/apprenticemanagement/pages/apprenticemanagement.xhtml	Takes apprentice supervisor to Apprentice Management	N/A
/xi/ui/apprenticemanagement/pages/apprenticemanagementdepartment.xhtml	Takes on-site supervisor to Apprentice Management	N/A
sf/timeoff	Takes user to Time Off employee self service page	N/A
sf/timeoffworkbench	Takes user to Time Off Workbench (PP3) or Administer Time(v12)	selected_user can be entered as a parameter. If it is not, the Workbench or Administer Time is opened for the logon user.
sf/timeoffcalendars	Takes user to Time Off change calendars	N/A

You can use the parameter to change the user ID in the URL to go directly to the page for that user.

## **Example**

The link to a page to terminate a specific user may look like the following:

/sf/employeeterminate?selectquestion=essMssTerminateActionController&selected\_user=

# 16 Mobile Employee Central

## 16.1 Setting Up Mobile

This section describes how to set up mobile use for Employee Central users.

### Context

You can access certain Employee Central features on your mobile device. Since HR data is private and personal, the following features help ensure the security of the data:

- Users can only activate the device from a valid account
- You can wipe lost or stolen devices from your computer and erase all sensitive data
- You can add an on-device mobile password to create an extra layer of security
- There is Mobile Device Management Support (MDM)

For a list of all features available, refer to [SAP SuccessFactors Mobile Features](#) the guide on the SAP Help Portal.

#### i Note

There are pre-defined links that direct users straight to specific screens inside the mobile app, for example, for approvals or access to SAP Jam.

#### i Note

To set up mobile devices for Time Off, see the [Implementing Employee Central Time Off](#) guide on the SAP Help Portal.

### Procedure

1. Select which mobile functionality should be made available. In your instance, go to the  [Admin Center](#) .
2. Select who will be allowed to use the mobile features and then grant them permission to do so. For more information, refer to the [Using Role-Based Permissions](#) guide available on the SAP Help Portal.
3. Notify those users with permission about the available features using the Notification e-mail. Inform them how to install and use on their device.

For more information about mobile set-up, refer to the [Mobile Deployment Guide](#) on the SAP Help Portal.

## 16.2 Mobile To-Do Items in Employee Central

This section describes how to set up to-do items in Employee Central for your mobile device.

### Overview

To-Do items are a way of notifying users that there are tasks waiting that they need to complete. For example, if you are a manager, one of your To-Do items might include approving a job change or one-time bonus for one of your direct reports.

### Prerequisites

- Remember to register and activate your mobile device.
- In addition to the configuration steps described in [Enabling the To-Do List \[page 34\]](#), you must activate this feature for mobile use. In your instance, navigate to the *Admin Center*. In the *Tools* search field, type *Mobile Settings*. Ensure that Theming, SF Notifications, and On Device Support are enabled.
- In addition, you can select further Employee Central areas to be accessible on your mobile, for example, Time Off, Benefits, or Pay Summary.

### Features

Once you have performed all the registration, activation, and configuration steps, any Employee Central To Do items requiring your attention appear in the *Open To-Dos* screen on your mobile device.

### Supported Workflows

The following types of workflows are supported for your mobile device. This means that you can view the activities related to them, as well as approved or declined.

- Changes to job details
- Change to job relations
- Changes to recurring pay components
- Spot bonuses
- Leaves of absence

# 17 Retired Functionality

Some features and functionality included in earlier releases are no longer needed or are no longer supported in the new release. The following sections describe features that are retired and the support for which will be discontinued.

## 17.1 Direct Deposit or Old Payment Information block

For new customers, the old HRIS-based Direct Deposit or Payment Information (handled by the HRIS elements `directDeposit` and `paymentInfo`) is no longer supported. You must use the new MDF-based Payment Information.

For more information about the new MDF-based Payment Information, refer to the  [SAP SuccessFactors Employee Central](#)  [Administration](#)  [Implementing and Configuring Payment Information in Employee Central](#)  guide on the [SAP Help Portal](#)

## 17.2 HRIS Propagation Configuration Template

As of the Q1 2019 release, the HRIS Propagation Configuration XML template is deprecated in your system.

This means that you can no longer use XML-based propagation rules in the Succession Data Model to have the system automatically fill in fields in employment data from foundation objects. From now on, business rules must be used for this purpose. Existing rules in XML will be translated into `onChange` business rules and automatically migrated for you. For each object, one business rule will be created. All the migrated rules will start with the prefix `migratedRule` to make it easier to review and/or edit the rule. If there were multiple XML rules, then the business rule will contain multiple `THEN (Set)` conditions for non-country/region-specific rules or multiple `THEN` and `ELSE IF` conditions for country/region-specific rules. The new business rules will be automatically assigned to the respective HRIS objects in the system and can be found both in the data model and the Business Configuration UI.

From now on, any changes to HRIS propagation behavior have to be managed by business rules (either by changing the migrated objects or creating new rules).

SAP SuccessFactors is making this change to give admins more flexibility in the system. Propagation rules are now available in the admin tools for you to adjust as needed. You no longer need to submit support tickets to adjust the xml.

We recommend that customers perform spot testing in all instances to validate the migration. If any issues are found, please contact SAP Cloud Support. In case the migration is not successful, the existing propagations will continue to work until SAP SuccessFactors successfully migrates them. The Provisioning option to upload the XML is available only to fix any migration issues and this will be removed in a future release.

For more information about business rules, see the [Implementing Business Rules in SAP SuccessFactors](#) guide on the SAP Help Portal.

## Example

Here is an example of a migrated propagation rule for Location.

Propagate LOCATION to JOB\_INFO (migratedRule\_\_LOCATION\_JOB\_INFO\_1539238469183) Insert New Record

**Basic Information**

Start Date 01/01/1900  
Rule Type  
Description This is a migrated rule created from migration of legacy Propagation. This rule is auto generated

**Parameters**

Name	Object
Context	System Context
Job Information	Job Information SDM 1

**If**

This rule is always true.  
To add an expression please uncheck the Always True checkbox.

**Then**

Set Job Information.Timezone to be equal to Job Information.Location.Timezone

[Collapse All](#) | [Expand All](#)

## Example

Here is an example of a migrated propagation rule for Division.

Propagate DIVISION to JOB\_INFO (migratedRule\_\_DIVISION\_JOB\_INFO\_1539238469196) Insert New Record

**Basic Information**

Start Date 01/01/1900  
Rule Type  
Description This is a migrated rule created from migration of legacy Propagation. This rule is auto generated

**Parameters**

Name	Object
Context	System Context
Job Information	Job Information SDM 1

**If**

This rule is always true.  
To add an expression please uncheck the Always True checkbox.

**Then**

Set Job Information.Custom String 18 to be equal to Job Information.Division.Name.Default Value

[Collapse All](#) | [Expand All](#)

## Example

Here is an example of a migrated propagation rule for Legal Entity.

Propagate LEGAL\_ENTITY to JOB\_INFO (migratedRule\_LEGAL\_ENTITY\_JOB\_INFO\_1539238469182) Insert New Record

Basic Information		Parameters	
Name	Object	Name	Object
Start Date	01/01/1900	Context	System Context
Rule Type		Job Information	Job Information SDM 1
Description	This is a migrated rule created from migration of legacy Propagation. This rule is auto generated	<a href="#">Collapse All</a>   <a href="#">Expand All</a>	

If This rule is always true.  
To add an expression please uncheck the Always True checkbox.

Then  
Set `Job Information.Location` to be equal to `Job Information.Company.Default Location`

## Example

Here are some examples of migrated propagation rules for Job Classification.

Propagate JOB\_CLASSIFICATION to JOB\_INFO  
(migratedRule\_JOB\_CLASSIFICATION\_cust\_string\_JOB\_INFO\_custom-string25\_1538556428858)

 Insert New Record

Basic Information

Start Date 01/01/1900

Rule Type

Description This is a migrated rule created from migration of legacy Propagation. This rule is auto generated

Parameters

Name	Object
Context	System Context 
Job Information	Job Information SDM 1

[Collapse All](#) | [Expand All](#)

 If

Job Information.Company.countryOfRegistration is equal to USA (USA)

Then

Set Job Information.(USA) Custom String 25 to be equal to Job Information.Job Classification.string

 Else If

Job Information.Company.countryOfRegistration is equal to GBR

Then

Set Job Information.(GBR) Custom String 25 to be equal to Job Information.Job Classification.string

Propagate JOB\_CLASSIFICATION to JOB\_INFO  
(migratedRule\_JOB\_CLASSIFICATION\_JOB\_INFO\_1539238469183)

 Insert New Record

Basic Information

Start Date 01/01/1900

Rule Type

Description This is a migrated rule created from migration of legacy Propagation. This rule is auto generated

Parameters

Name	Object
Context	System Context 
Job Information	Job Information SDM 1

[Collapse All](#) | [Expand All](#)

 If

 This rule is always true.  
To add an expression please uncheck the Always True checkbox.

Then

Set Job Information.Job Title En US to be equal to Job Information.Job Classification.Job Title.Default Value

Set Job Information.Custom String 11 to be equal to Job Information.Job Classification.Employee Class.External Code

Set Job Information.Pay Grade to be equal to Job Information.Job Classification.Pay Grade

Set Job Information.(USA) EEO Category 4 to be equal to Job Information.Job Classification.Job Classification Countries.Job Classification Countries where...  
Select Job Information.Job Classification.Job Classification Countries where...  
Country is equal to USA (USA)

## Example

Here is an example of a migrated propagation rule for Recurring Pay Components.

Propagate PAY\_COMPONENT to PAY\_COMPONENT\_RECURRING  
(migratedRule\_PAY\_COMPONENT\_PAY\_COMPONENT\_RECURRING\_1539238469193)

 Insert New Record

Basic Information

Start Date 01/01/1900

Rule Type

Description This is a migrated rule created from migration of legacy Propagation. This rule is auto generated

Parameters

Name	Object
Context	System Context 

[Collapse All](#) | [Expand All](#)

 If

 This rule is always true.  
To add an expression please uncheck the Always True checkbox.

Then

Set Pay Component Recurring.Amount to be equal to Pay Component Recurring.Pay Component.Pay Component Value

Set Pay Component Recurring.Frequency to be equal to Pay Component Recurring.Pay Component.Frequency

Propagate to PAY\_COMPONENT\_RECURRING currency  
(migratedRule\_PAY\_COMPONENT\_RECURRING\_currency-code\_1538556428862)

 Insert New Record

Basic Information

Start Date 01/01/1900

Rule Type

Description This is a migrated rule created from migration of legacy Propagation. This rule is auto generated

Parameters

Name	Object
Context	System Context 

Pay Component Recurring Compensation

[Collapse All](#) | [Expand All](#)

 If

Pay Component Recurring.Pay Component.Currency is not equal to Null

Then

Set Pay Component Recurring.Currency to be equal to Pay Component Recurring.Pay Component.Currency

 Else If

Pay Component Recurring.Employment Details.Job Information SDM 1.Company.Currency.Currency Code is not equal to Null

Then

Set Pay Component Recurring.Currency to be equal to Pay Component Recurring.Employment Details.Job Information SDM 1.Company.C

## Example

Here is an example of a migrated propagation rule for Non-Recurring Pay Components.

Propagate PAY\_COMPONENT to PAY\_COMPONENT\_NON\_RECURRING  
(migratedRule\_PAY\_COMPONENT\_PAY\_COMPONENT\_NON\_RECURRING\_1539238469194)

 Insert New Record

Basic Information

Start Date 01/01/1900

Rule Type

Description This is a migrated rule created from migration of legacy Propagation. This rule is auto generated

Parameters

Name	Object
Context	System Context 

Pay Component Non Recurring Spot Bonus

[Collapse All](#) | [Expand All](#)

 If

 This rule is always true.  
To add an expression please uncheck the Always True checkbox.

Then

Set Pay Component Non Recurring.Value to be equal to Pay Component Non Recurring.Type.Pay Component Value

Propagate to PAY\_COMPONENT\_NON\_RECURRING currency  
(migratedRule\_PAY\_COMPONENT\_NON\_RECURRING\_currency-code\_1539238469195)

 Insert New Record

Basic Information

Start Date 01/01/1900

Rule Type

Description This is a migrated rule created from migration of legacy Propagation. This rule is auto generated

Parameters

Name	Object
Context	System Context 
Pay Component Non Recurring	Spot Bonus

[Collapse All](#) | [Expand All](#)

If

Pay Component Non Recurring.Type.Currency is not equal to Null

Then

Set Pay Component Non Recurring.Currency Code to be equal to Pay Component Non Recurring.Type.Currency

Else If

Pay Component Non Recurring.Employment Details.Job Information SDM 1.Company.Currency.Currency is not equal to Null

Then

Set Pay Component Non Recurring.Currency Code to be equal to Pay Component Non Recurring.Employment Details.Job Information SD

## Example

Here is an example of a migrated propagation rule for Standard Hours / FTE

The rule will be triggered when the standard hour field in Job Information changes. It is also added as onInit rule to Job Information to calculate FTE in new hires.

The screenshot shows the 'Basic Information' section with fields: Start Date (01/01/1900), Rule Type (Context), and Description (This is a migrated rule created from migration of legacy Propagation. This rule is auto generated). The 'Parameters' section shows Context (System Context) and Job Information (Job Information SDM 1). A note indicates the rule is always true. The 'Then' section defines a business rule: Set Job Information.FTE to be equal to Calculate FTE based on Standard Hours(). The rule uses Weekly Standard Hours (Job Information.Standard Weekly Hours), Standard Hours of Position (Null), Standard Hours of Job Code (Job Information.Job Classification.Standard Weekly Hours), Standard Hours of Location (Job Information.Location.Standard Weekly Hours), and Standard Hours of Legal Entity (Job Information.Company.Standard Weekly Hours).

## 17.3 Event Reason Derivation from XML

As of 2H 2020 release, the XML-based approach to derive event reason for Job Information and Compensation Information is deprecated from your system.

The existing process of using XML-based rules in the Succession Data Model to derive event reasons for changes to Job Information and Compensation Information of employees is no longer supported. You can now employ business rules to achieve the same purpose.

We've added a new rule scenario to support onSave rules in Job Information and Compensation Information for event reason derivation. We've introduced this feature to give you, as an administrator, more flexibility in the system. As a result, you no longer need to submit support tickets to adjust the XML.

Existing rules in XML will be translated into onSave business rules and automatically migrated for you. For each object, one business rule will be created. All the migrated rules will start with the postfix 'migrated\_rule' to make it easier to review and edit the rule.

### Example

complInfoModel\_ERD\_migrated\_rule\_1582621769177

The new business rules will be automatically assigned to the respective HRIS objects in the system and can be found both in the data model and the Business Configuration UI.

We've also made some relative changes in Provisioning that include:

- The removal of the legacy *Enable youCalc rules engine for HRIS* switch.
- Decoupling of two switches, *Enable Business Rules for Workflow Derivation* and *Enable Business Rules for Event Reason Derivation*. The behavior of these two switches are as follows:
  - If *Enable Business Rules for Workflow Derivation* is enabled, workflows will be derived using business rules. Otherwise, workflow derivation happens from the XML model.

- If [Enable Business Rules for Event Reason Derivation](#) is enabled, event reasons will be derived using business rules. Otherwise, event reason derivation will be disabled.

### → Remember

As a customer, you don't have access to Provisioning. To complete tasks in Provisioning, contact your implementation partner. If you're no longer working with an implementation partner, contact SAP Cloud Support.

We recommend that you perform spot testing in all instances to validate the migration. If you face any issues, please contact SAP Cloud Support.

For more information about business rules, refer to the **Related Information** section.

## Use Cases

Consider the following table chart for reference:

Use Case No.	XML Rules	Job Information Event Reason	Compensation Information Event Reason
1	J1	J1	J1
2	C2	C2	C2
3	R3	R3	R3
4	JR4	JR4	JR4
5	CR5	CR5	CR5
6	JC6	JC6	JC6
7	D7	D7	D7

### Legend:

- J – Job Information
- C- Compensation Information
- R- Job Relationship
- JR – Job Information and Job Relationship
- CR – Compensation Information and Job Relationship
- JC – Job Information and Compensation Information
- D – Default or Catch all

### i Note

In all the use cases, ensure that there is the primary condition to check if the existing event reason value is null or blank.

### • Example

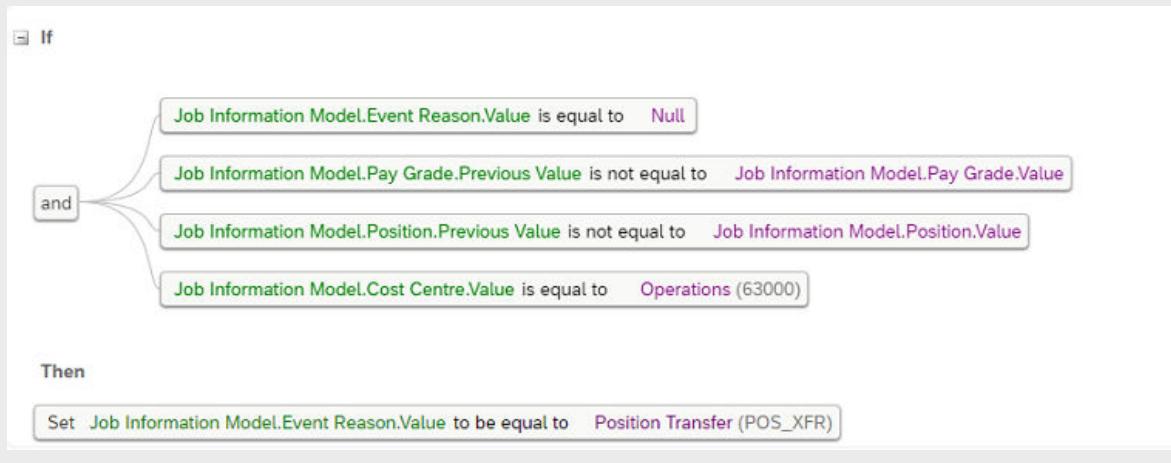
Use Case 1: Deriving the event reason based on Job Information.

XML Rule:

## ↳ Sample Code

```
<rule id="rule-PT1">
    <trueoutput>POS_XFR</trueoutput>
    <conditions>
        <and>
            <equal id="jobInfo.pay-grade" inverse="true"/>
            <equal id="jobInfo.position" value="" inverse="true"/>
            <equal id="jobInfo.cost-center" value="63000" />
        </and>
    </conditions>
</rule>
```

Migrated business rule:



## ❖ Example

Use Case 2: Deriving the event reason based on Compensation Information.

XML Rule:

## ↳ Sample Code

```
<rule id="rule-7">
    <trueoutput>PAYBEN</trueoutput>
    <conditions>
        <and>
            <equal id="compInfo.benefits-rate" inverse = "true"/>
        </and>
    </conditions>
</rule>
```

Migrated business rule:



## • Example

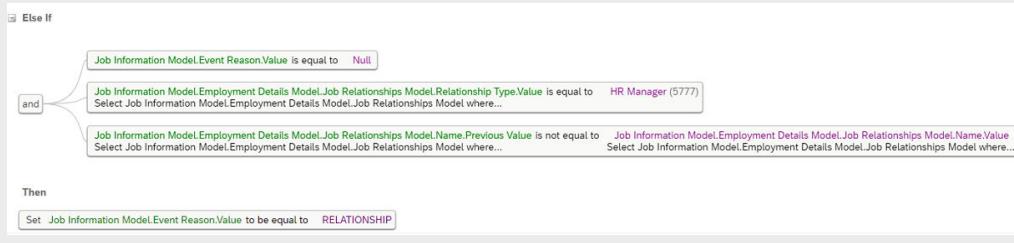
Use Case 3: Deriving the event reason based on the Job Relationship information of an employee.

XML Rule:

### ↳ Sample Code

```
<rule id="rule-090">
    <trueoutput>RELATIONSHIP</trueoutput>
    <conditions>
        <or>
            <equal id="jobRelationsInfo.relationship-type.hr manager"
inverse = "true"/>
        </or>
    </conditions>
</rule>
```

Migrated business rule:



## • Example

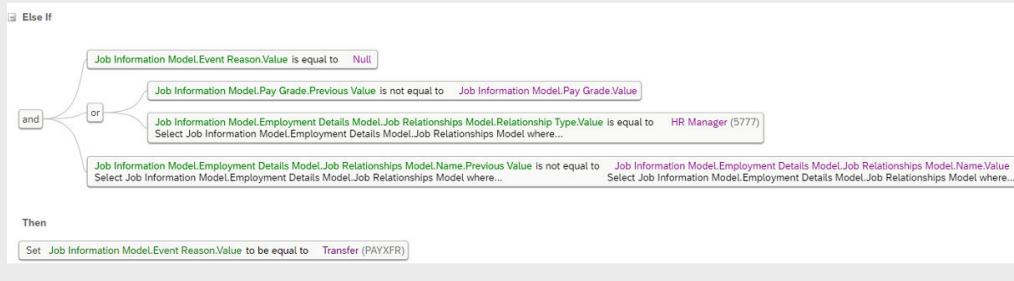
Use Case 4: Deriving the event reason based on the Job Information and Job Relationship of an employee.

XML Rule:

### ↳ Sample Code

```
<rule id="rule-090">
    <trueoutput>PAYXFR</trueoutput>
    <conditions>
        <or>
            <equal id="jobInfo.pay-grade" inverse="true"/>
            <equal id="jobRelationsInfo.relationship-type.hr manager"
inverse = "true"/>
        </or>
    </conditions>
</rule>
```

Migrated business rule:



## ❖ Example

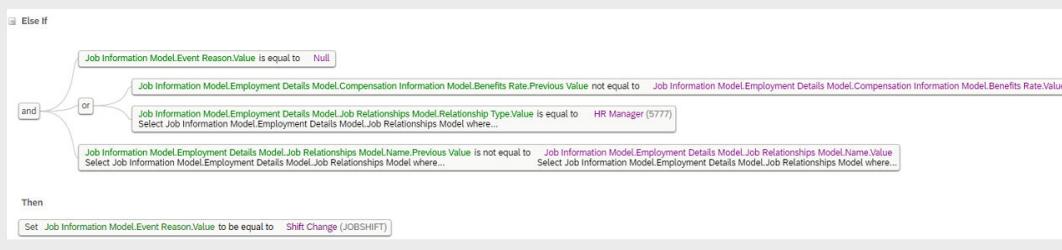
Use Case 5: Deriving the event reason based on the Compensation Information and Job Relationship of an employee.

XML Rule:

### ↳ Sample Code

```
<rule id="rule-090">
    <trueoutput>JOBSHIFT</trueoutput>
    <conditions>
        <or>
            <equal id="compInfo.benefits-rate" inverse = "true"/>
            <equal id="jobRelationsInfo.relationship-type.hr manager" inverse = "true"/>
        </or>
    </conditions>
</rule>
```

Migrated business rule:



## ❖ Example

Use Case 6: Deriving the event reason based on the Job Information and Compensation Information of an employee.

XML Rule:

### ↳ Sample Code

```
<rule id="rule-6">
    <trueoutput>PROPWP</trueoutput>
    <conditions>
        <and>
            <greater id="payComponentGroup.AnnualizedSalary" />
    </and>
    </conditions>
</rule>
```

```

        id="jobInfo.pay-grade.paygradeLevel"/>
                <greater
            </and>
        </conditions>
    </rule>

```

Migrated business rule:



## • Example

Use Case 7: Deriving the event reason based on any data change.

XML Rule:

### ↳ Sample Code

```

<rule id="rule-23">
        <!-- Catch all-->
        <trueoutput>DATACHG</trueoutput>
        <conditions>
            <or>
                </or>
        </conditions>
    </rule>

```

Migrated business rule:

Set **Job Information Model.Event Reason.Value** to be equal to **Data Change (DATACHG)**

## Related Information

[Implementing Business Rules in SAP SuccessFactors](#)

# Important Disclaimers and Legal Information

## Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.

About the icons:

- Links with the icon  : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:
  - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
  - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.
- Links with the icon  : You are leaving the documentation for that particular SAP product or service and are entering a SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

## Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

## Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.

The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

## Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

## Gender-Related Language

We try not to use gender-specific word forms and formulations. As appropriate for context and readability, SAP may use masculine word forms to refer to all genders.

