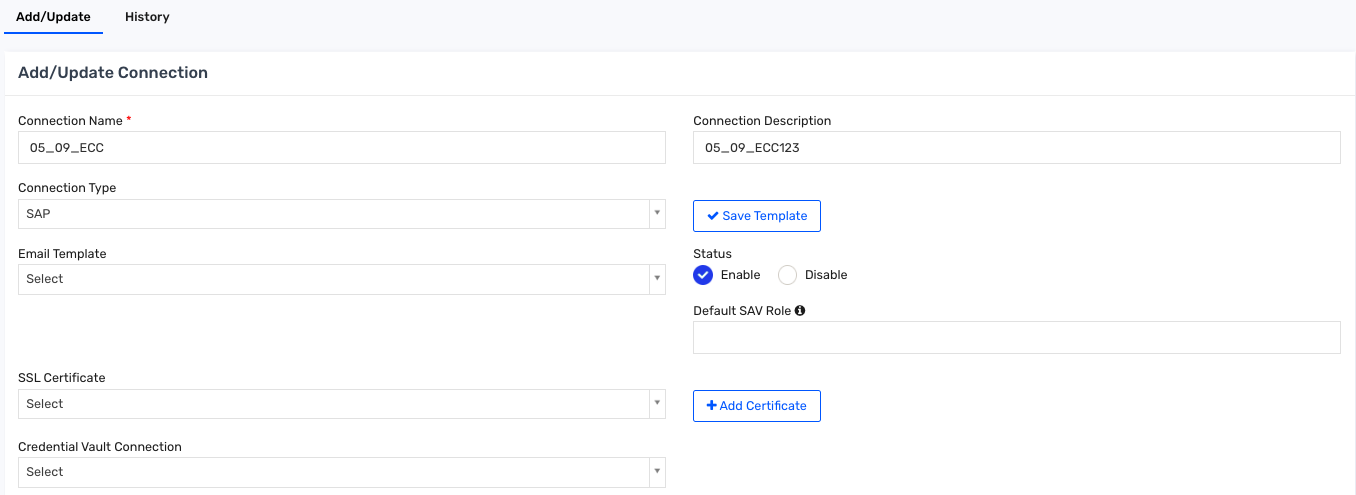
Sample SAP connector

Introduction:  
Saviynt EIC offers OOB capability to integrate SAP applications with Saviynt with the help of SAP connectors. Saviynt Provides OOB templates that can be configured accordingly.  
  
  
How it works:  
  
Saviynt offers a connector module which can be configured to make a connection with the application.  
  
  
Prerequisites:   
  
Network connectivity should be established before we can begin operations like recon and provisioning. The network needs to be whitelisted, and firewall should allow the network to flow from Saviynt to EIC and vice versa.  
  
Integration Steps:  
  
Create an SAP connection. Let’s look at the integration steps.



1. Create the connection name
2. Provide the Connection Type as SAP and the template will load .

A screenshot of a computer

Description automatically generated

1. MESSAGESERVER: set it to TRUE if the Message Server is going to be used  
   for connecting to SAP, Usually its False
2. JCO\_AHOST: This is the hostname for connection for Import.
3. JCO\_SYSNR: syste, number for the SAP instance for import
4. JCO\_Client: Cliemt Number for the SAP instance for Import
5. JCO\_USER: Username to connect to SAP using JCO for Import
6. Password: provide the password to connect.
7. JCO\_LANG: Language for the SAP instance for Import.

A screenshot of a computer

Description automatically generated

1. SNC: TRUE/FALSE, set it to TRUE if Secure Network Connection (SNC) is setup  
   between ABAP and ASHOST
2. JCo\_SNC\_MODE: Enable/Disable Secure Network Connection (SNC) Mode. 0 - Disabled, 1 -  
   Enabled
3. Tables : USR02, USR04, USER\_ADDR, USRACL, USR06, TSTC, TSTCT, AGR\_AGRS, AGR\_TEXTS, AGR\_DEFINE, AGR\_USERS, USORG, AGR\_1252, AGR\_1251  
   The above should be the order in which table order should be mentioned.
4. SYSTEMNAME: ECC (System name to be used for import) Default is EC6
5. TERMINATED\_USER\_ROLE\_ACTION: DELETEACCOUNT (Action to take for the user roles when the user gets terminated oir roles are removed)

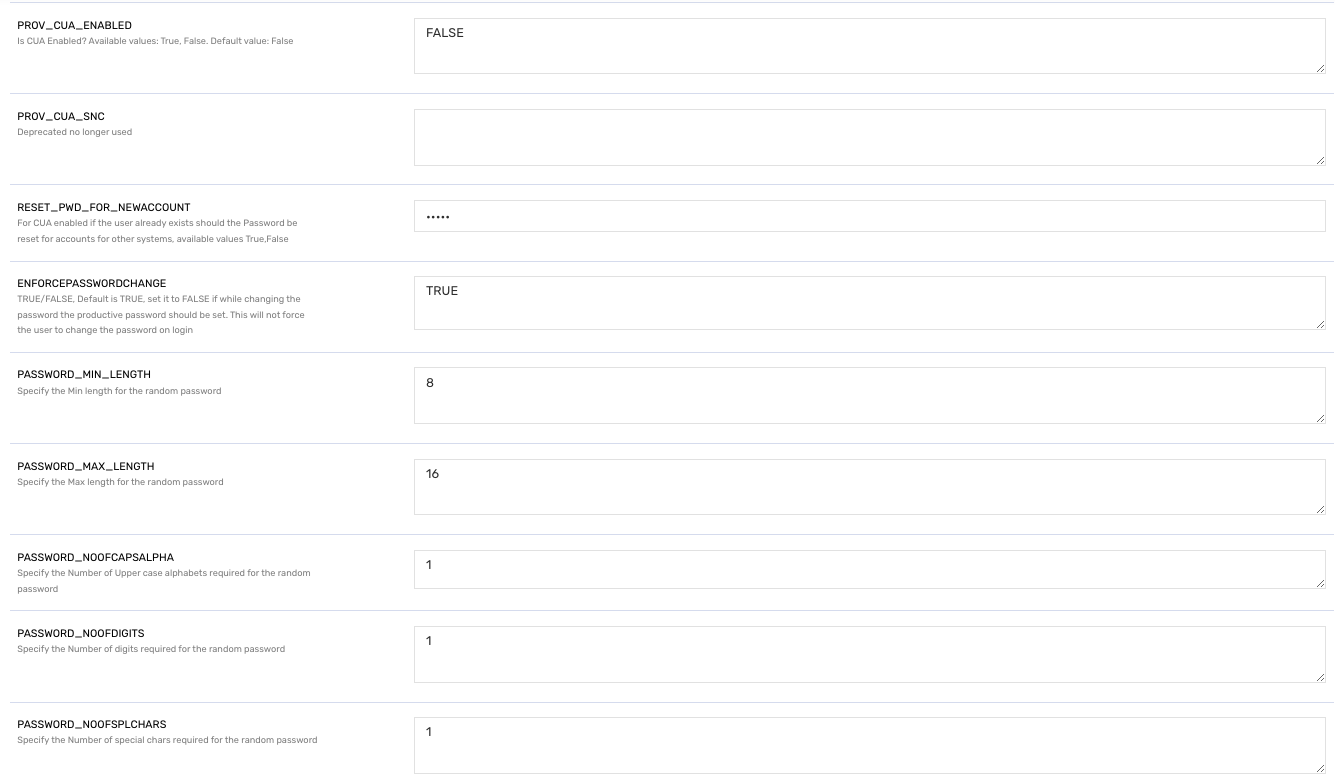
Provisioning Json

A screenshot of a computer

Description automatically generated

1. Create account json: Json to specify to create the new account

* {"ADDRESS":{"LASTNAME":"${user.lastname}","FIRSTNAME":"${user.firstname}","E\_MAIL":"${user.email}"},"DEFAULTS":{"DCPFM":"X","DATFM":"2","SPDA":"D"},"SNC":{"GUIFLAG":"U","PNAME":"p:${accountName}@CORP.INT"},"LOGONDATA":{"USTYP":"A","GLTGB":"99991231"},"PASSWORD":{"BAPIPWD":"${randomPassword}"}}  
    
    
  **ADDRESS**:
  + Stores the user’s primary identity details.
  + **LASTNAME**: Maps to user.lastname, capturing the user’s last name.
  + **FIRSTNAME**: Maps to user.firstname, capturing the user’s first name.
  + **E\_MAIL**: Maps to user.email, storing the user’s email address.
* **DEFAULTS**:
  + Contains system defaults and preferences for date and formatting.
  + **DCPFM**: Set to "X", possibly indicating a specific data or code format.
  + **DATFM**: "2" might specify a preferred date format or standard.
  + **SPDA**: "D" likely refers to a language or region setting, such as display language or localization format.
* **SNC**:
  + Defines Security Network Communication (SNC) settings.
  + **GUIFLAG**: "U" might specify GUI-related preferences or flags.
  + **PNAME**: "p:${accountName}@CORP.INT" represents a Principal Name, derived from ${accountName}, for secure authentication within the CORP.INT domain.
* **LOGONDATA**:
  + Provides user login-related data.
  + **USTYP**: "A" could represent a user type or profile setting, possibly indicating "Administrator" or similar.
  + **GLTGB**: "99991231" serves as an end-date for the user account (formatted as YYYYMMDD), often used to set accounts as "active indefinitely."
* **PASSWORD**:
  + Defines initial or reset password details.
  + **BAPIPWD**: Maps to ${randomPassword}, generating a randomized password for the user during creation or onboarding.
* This configuration is well-suited for initializing user details, login settings, and secure communication preferences within a corporate or enterprise environment.

1. PROV\_JCO\_AHOST: Hostname for connection for provisioning. Example: sap2.remoteides.com
2. PROV\_JCO\_SYSNR: System Number for the SAP instance for Provisioning. Example: 45
3. PROV\_JCO\_Client: Client Number for the SAP instance for Provisioning. Example: 800
4. PROV\_JCO\_USERL Username to connect t SAP .example: ECC4518
5. Prov\_password: Password for connection of provisioning
6. PROV\_JCO\_LANG: en  
     
     
   

22) PROV\_CUA\_ENABLED: IS PROV cua enabled, Default value is False

1. RESET\_PWD\_FOR\_NEWACCOUNT: For CUA enabled if the user already exists should the Password be  
   reset for accounts for other systems, available values True,False

23)Password fields are self explanatory



24) ENABLEACCOUNTJSON : json to mention changes made at the user level after enabling the account  
  
example: {"LOGONDATA":{"GLTGV":"${new java.text.SimpleDateFormat('yyyyMMdd').format(user.startdate)}","GLTGB":"${user.enddate==null?'99991231': new java.text.SimpleDateFormat('yyyyMMdd').format(user.enddate)}"}}  
  
**GLTGV** (Start Date):

* This parameter captures the user's **start date** for account validity.
* It uses java.text.SimpleDateFormat('yyyyMMdd') to format the user.startdate in YYYYMMDD format.
* For example, if user.startdate is January 1, 2024, GLTGV will store it as "20240101".
* **GLTGB** (End Date):
* This parameter captures the **end date** of the account validity.
* If user.enddate is defined, it formats it to YYYYMMDD.
* If user.enddate is null, the system defaults to "99991231", often used to indicate that the account is valid indefinitely.
* For example, if user.enddate is December 31, 2030, GLTGB will store it as "20301231"; if there is no user.enddate, GLTGB will be "99991231".

25) UPDATEACCOUNTJSON: JSON Similar to Create Account to mention the changes that needs to be  
made at the user level after updating the account  
  
example: {"ADDRESS":{"FIRSTNAME":"${user.firstname}","LASTNAME":"${user.lastname}"},"LOGONDATA":{"CLASS":"TEST"},"DEFAULTS":{"LANGU":"EN","TIMEFM":"0","DATFM":"1","SPLD":"LOCL"}}  
  
  
**ADDRESS**:

* + Defines basic personal information for the user.
  + **FIRSTNAME**: Takes the user’s first name from user.firstname.
  + **LASTNAME**: Takes the user’s last name from user.lastname.
* **LOGONDATA**:
  + Specifies login data settings.
  + **CLASS**: Here, the class is set to "TEST", potentially denoting a specific user access classification or group for testing purposes.
* **DEFAULTS**:
  + Defines default settings for language, time format, date format, and print output.
  + **LANGU**: Set to "EN", establishing English as the default language.
  + **TIMEFM**: Set to "0", indicating a particular time format (likely 24-hour format or AM/PM based on system standards).
  + **DATFM**: Set to "1", specifying a date format (e.g., MM/DD/YYYY or DD/MM/YYYY, depending on the system convention).
  + **SPLD**: Set to "LOCL", indicating the default local printer for output.
* This configuration helps set up the essential profile details, access classification, and preferences for language and output.

26) STATUS\_THRESHOLD\_CONFIG: {"statusAndThresholdConfig":{"statusColumn":"userlock","activeStatus":["0","128"],"deleteLinks":true,"accountThresholdValue":1000,"correlateInactiveAccounts":true,"inactivateAccountsNotInFile":false,"deleteAccEntForActiveAccounts":false,"lockedStatusColumn":"userlock","lockedStatusMapping":{"Locked":["32","64"],"Unlocked":["0","128"]}}}  
  
  
**statusAndThresholdConfig**:

* **statusColumn**: Refers to "userlock", the column that tracks the account’s active or locked status.
* **activeStatus**: Lists values ["0", "128"] as indicators of an active account.
* **deleteLinks**: Set to true, enabling deletion of links between accounts and associated entities when specific conditions are met.
* **accountThresholdValue**: Set to 1000, representing a threshold limit, likely for managing accounts more efficiently or triggering actions when the limit is reached.
* **correlateInactiveAccounts**: Set to true, allowing the system to correlate data for inactive accounts across different sources.
* **inactivateAccountsNotInFile**: Set to false, meaning accounts not present in the input file won’t be inactivated automatically.
* **deleteAccEntForActiveAccounts**: Set to false, which means account entitlements won't be deleted for accounts marked as active.
* **lockedStatusColumn**: Repeats "userlock" as the column used for account locking status.
* **lockedStatusMapping**:
* **Locked**: Lists values ["32", "64"], indicating when an account is considered "Locked."
* **Unlocked**: Lists values ["0", "128"], meaning these values designate an "Unlocked" account.

**TROUBLESHOOTING:**  
Common error messages shown while configuring a connection are:  
com.sap.conn.jco.JCoException: (106) JCO\_ERROR\_RESOURCE: Destination 264416 does not exist

* A DestinationDataProvider can only deregister itself
* java.lang.IllegalStateException: A DestinationDataProvider can only deregister itself
* com.sap.conn.jco.JCoException: (180) JCO\_ERROR\_DESTINATION\_DATA\_INVALID: Destination 479215 was removed. Check the destination configuration  
    
    
  RESOLUTION:  
    
  Add this config to externalconfig.properties when you see below SAP connection failures
* **sap.usefilebasedapproach=true**