Outgoing triggers examples

Attribute mapping triggers

When you are configuring an agent and defining the attribute mappings of connectors, depending on the connector type, it will be able to define BeanShell scripts that will be triggered when data is loaded into the target system, **outgoing triggers**.

Triggers can be used to validate or perform a specific action just before performing an operation or just after performing an operation into target objects.

The trigger result will be a boolean value, true to continue or false to stop.

Use case examples

Example 1

Update or insert a user only when the user is internal (PreInsert User or PreUpdate User)

```
name = source{"userName"};
user = serviceLocator.getUserService().findUserByUserName(name);
if (user != null) {
  if (user.userType.equals("I")) {
      return true;
  }
}
return false;
```

Example 2

Update or insert a user only when the company is Soffid. Be in mind that company attribute is a custom attribute.

```
name = source{"userName"};
company = source{"attributes"}{"company"};
```

```
user = serviceLocator.getUserService().findUserByUserName(name);

if (user != null) {
  if (company != null && company.toUpperCase().equals("SOFFID")) {
  return true;
  }
}

return false;
```

Example 3

Recover a response and process it.

```
if (response != null) {
  for (o : response.getObjects()) {
  if (o != null && o{"result"} != null) {
  //TO-DO
    }
  }
}
return true;
```

Example 4

Send a HTML mail with the response info.

```
.....
to = newObject{"identity"};
subject = "LinOTP QR";
body = o{"result"};
serviceLocator.getMailService().sendHtmlMailToActors(new String[] {to}, subject, body);
.......
```

Example 5

Update a user with the response info

```
.......
body = o{"result"};
ac = newObject{"user"};
```

```
data = new com.soffid.iam.api.UserData();
data.setAccountName(ac);
data.setSystemName("LinOTP");
data.setAttribute("token");
data.setBlobDataValue(body.getBytes());
serviceLocator.getAccountService().updateAccountAttribute(data);
......
```

Example 6

Call an API and process the response.

```
.......
userN = accontList.get(0).name;
result = dispatcherService.invoke(
"GET",
 "https://" + DOMINIO + "crmRestApi/resources/....?onlyData=true&q=Username=" + userN,
 null);
if (result != null && !result.isEmpty()) {
 //TO-DO
}
.......
```

Example 7

Pre Update: assign a value to an attribute depending on the value of a group attribute. The user can belong to many groups.

```
userName = source{"userName"};
attributes = serviceLocator.getUserService().findUserAttributes(userName);
dateChange = attributes.get("dateChange");

if (dateChange == null || dateChange == void ) {
return true;
}

userGroupList = serviceLocator.getGroupService().findUsersGroupByUserName(userName);
contFalse = 0;
contNull  = 0;
isFound = false;
for (ug : userGroupList) {
```

```
  group = serviceLocator.getGroupService().findGroupByGroupName(ug.group);
  att = group.attributes.get("indicator");
  if (att != void) {
     if (att == null) contNull++;
     else if (att.equals("0")) contFalse++;
     else if (att.equals("1")) {
       isFound  = true;
       break;
     }
  }
}

if (isFound) {
  newObject{"id-indicator"} = "1";
} else {
  if (contFalse > 0) {
    newObject{"id-indicator"} = "0";
  } else if (contNull > 0) {
    newObject{"id-indicator"} =  null;
  }
}

return true;
```

Revision #24
Created 10 August 2021 09:31:01
Updated 30 November 2022 12:31:53