

Data Flow 2.2 proGres to PRIMERO Referral Decision UNHCR

Solution Requirements

Requirements for a recurring process to retrieve Referral decisions from proGres. This integration will use PING and OpenFN as intermediary platforms, replacing the previous DTP-based workflow.

Step 1: Retrieval of Referral decisions from proGres

High-Level Workflow

1. A proGres Case Worker makes a decision ("Acknowledged" or "Rejected") on a Referral that originated from PRIMERO.
2. proGres requires information about the case based on the Case Worker decision and populates relevant fields such as: Individual ID, Interoperability Referral Status, Rejection Reason, Review Decision, Record State.
3. OpenFN executes a recurring job every 'M' minutes to check for new decision records in proGres **Interoperability Referrals**.
4. Retrieve Decision Records from PING:
 - a. OpenFN authenticates with the PING Retrieval API using a Bearer Token.
 - b. OpenFN calls a PING endpoint that triggers Shipping Process **SHP-2795** to query the `progres_interoperabilityreferral` entity for new decision records. The query contains a single filter that retrieves records with a status of 'Acknowledged' or 'Rejected'.
5. Retrieve Full Referral Data from PING
 - a. For each decision record OpenFN could initiate a **second call** to the PING API.
 - b. The second call triggers Shipping Process **SHP-2796** to fetch the complete Referral data from the `'progres_interoperabilityreferral'` entity.

Functional Requirements:

For the OpenFN Team:

- Implement a recurring process to poll the PING Retrieval API.
- Implement the functionality to authenticate and connect to PING.
- Develop the logic to perform the two-step retrieval:
 - o Call the PING API (using **SHP-2795**) to get the list of new decision records from the `progres_interoperabilityreferral` entity.
 - o If needed: for each decision record, call the PING API again (using **SHP-2796**) to fetch the full Referral details from `progres_interoperabilityreferral` and other linked entities.

For the PING Team:

- Define and provide the specific filter conditions OpenFN should use to query the `progres_interoperabilityreferral` entity via **SHP-2795** to identify Referral decision records.
- Configure the Shipping Processes **SHP-2795** and **SHP-2796** in the BAU-UAT environment to allow data retrieval from the Interoperability Referrals entity.
- The necessary payload examples for OpenFN to call these retrieval processes are included in the Appendix.

Technical considerations:

For the PING Team:

- The record retrieved from `progres_interoperabilityreferral` entity contains the identifier for the full Referral. OpenFN will need to use the value on field **`progres_interoperabilityreferralid`** as the key to fetch the corresponding Referral in the subsequent API call.
- The Shipping Process **SHP-2796** in PING, which retrieves the Referral details, should be configured to use the FetchXML query named `GetInteropReferralDecision`. The latest version of this query is available in the appendix.
- Two distinct Shipping Processes will be used:
 - a. **SHP-2795**: To retrieve decision records with a minimal set of fields to reduce processing overhead.
 - b. **SHP-2796**: To retrieve the full Referral from the `progres_interoperabilityreferral` entity.

- All initial development and testing will be conducted in the BAU-UAT environment.

Step 2: Update progres_interoperabilityreferral Record Status

After successfully retrieving the Referral details from proGres, OpenFN must update the status of the original record in the Interoperability Referrals entity to prevent it from being processed again.

1. OpenFN will call the PING Ingestion API to update the status of the DTP_deliveries record from **Acknowledged** (value: 125080001) or **Rejected** (value: 125080002) to **Synced with PRIMERO** (value: 125080003).
2. PING Configuration: This update will be handled by the Shipping Process SHP-2284. An example of a payload is included in the Appendix.
3. **Note:** Shipping Process SHP-2284 is ready in PING, but we need to wait for the next proGres release on 14 January 2026 before we can start testing.

Step 3: Send Decision to PRIMERO

Once OpenFN has successfully retrieved the Referral and its corresponding decision from proGres (via PING), the next step is to transmit this information to PRIMERO.

Functional Requirements:

For the OpenFN Team:

- Transform the data retrieved from PING into the required format to update the Referral record in PRIMERO.
- Call the appropriate PRIMERO web service endpoint to send the decision data.
- Handle PRIMERO's Response:
 - o **On Success:** OpenFN will define the logic for handling a successful response from PRIMERO, as no further action is required on the proGres app.
 - o **On Failure:** Implement logic to handle any errors returned by the PRIMERO web service.

Appendix

Example payload for the shipping process SHP-2795

```
{
  "ShippingProcessId": "SHP-2795", // Check for new decision records
  "PartnerId": "PNR-1363",
  "InteropID": "INT-1436",
  "APIVersion": "V3",
  "QueryParameters": []
}
```

Example payload for the shipping process SHP-2796

```
{
  "ShippingProcessId": "SHP-2796",
  "PartnerId": "PNR-1363",
  "InteropID": "INT-1436",
  "APIVersion": "V3",
  "QueryParameters": [
    {
      "ParameterName": "{record_ids}",
      "Value": "d423cb74-86f1-f011-8406-002248896d99"
    }
  ]
}
```

Example payload for the shipping process SHP-2284

Endpoint for PING API (BAU-UAT environment):

<https://msawe-primes-ping-bau-uat-apimgmt.azure-api.net/api/ingestion/v2/PNR-1363/data>

```
{
  "ShippingProcessId": "SHP-2284",
  "InteropId": "INT-1436",
  "Data": [
    {
      "ExternalId": "12345",
      "Attachments": [],
      "Fields": [
        {
          "Name": "progres_interoperabilityreferralid",
          // Key field to identify the referral
          "Value": "72396a29-44f0-f011-8406-002248896d99"
        },
        {
          "Name": "progres_interoperabilityreferralstatus",
          // the field we are updating
          "Value": "125080003"
        },
        {
          "Name": "progres_comments",
          // Comment stored directly in the same record
          "Value": "Test: Calling SHP-2284 to update the
Referral Status in proGres"
        }
      ]
    }
  ]
}
```

```

    }
  ]
}

```

FetchXML query: GetInteropReferralDecision

```

<fetch distinct="true">
  <entity name="progres_interoperabilityreferral">
    <attribute name="progres_individualid" />
    <attribute name="progres_interoperabilityreferralid" />
    <attribute name="progres_interoperabilityreferralnumber" />
    <attribute name="progres_interoperabilityreferralrejectionomment" />
    <attribute name="progres_interoperabilityreferralstatus" />
    <attribute name="progres_interoperabilityreferralrejectionreason" />
    <attribute name="progres_organizationfrom" />
    <attribute name="progres_orgreferredby" />
    <attribute name="progres_reasonforreferral" />
    <attribute name="progres_reviewby" />
    <attribute name="progres_reviewdate" />
    <attribute name="progres_reviewdecision" />
    <attribute name="progres_unhcrd" />
    <attribute name="statecode" />
    <attribute name="statuscode" />
    <attribute name="ownerid" />
    <attribute name="progres_businessunit" />
    <attribute name="progres_comments" />
    <attribute name="progres_interoperabilityreferralrouteserviceunito" />
    <attribute name="progres_interoperabilityreferralrouteto" />
    <attribute name="progres_interventiontype" />
    <attribute name="progres_requestedservice" />
  <filter>
    <condition attribute="progres_interoperabilityreferralstatus" operator="in">
      <value>125080001</value>
      <value>125080002</value>
    </condition>
    <condition attribute="progres_interoperabilityreferralid" operator="eq"
value="{record_ids}" />
  </filter>
  <order attribute="progres_reviewdate" descending="true" />
  <link-entity name="progres_individual" from="progres_individualid"
to="progres_individualid" link-type="outer" alias="individuals">
    <attribute name="progres_coalocationlevel1" />
    <attribute name="progres_coalocationlevel2" />
    <attribute name="progres_coalocationlevel3" />
    <attribute name="progres_coalocationlevel4" />
    <attribute name="progres_coalocationlevel5" />
    <attribute name="progres_coalocationlevel6" />
    <attribute name="progres_commonyusedname" />
    <attribute name="progres_dateofbirth" />
    <attribute name="progres_familyname" />
    <attribute name="progres_givename" />
    <attribute name="progres_id" />
    <attribute name="progres_middlename" />
    <attribute name="progres_primaryphonenumber" />
    <attribute name="progres_registrationgroupid" />
    <attribute name="progres_sex" />
  </filter type="and">

```

```
<condition attribute="statecode" operator="eq" value="0" />
<filter type="or">
  <condition attribute="statuscode" operator="eq" value="1" />
  <condition attribute="statuscode" operator="eq" value="125080000" />
</filter>
</filter>
</link-entity>
</entity>
</fetch>
```