SAP IBP Troubleshooting Analysis - Inbound Integration

Deep Singh and Parveen Kumar, SAP
June, 2020
Agenda

- Data Import Scenario
  - How does it work
  - Staging and reporting tables

- Best Practices
  - Avoiding long running jobs
  - Parallel jobs
  - Usage of Shared MDT
  - Attribute as KF
  - Downloading logs from reporting tables

- External Scheduling
SAP IBP Architecture Overview

System Landscape

Frontend

Cloud NW

REST APIs

IBP Applications

Applications
- SC Control Tower
- Collaboration
- SD&OP
- Demand Driven
- Response & Supply

Configuration

HANA Database

Calculations and Algorithms

Data Model

Backend

Web User Interface

IBP Fiori

IBP Add-In

MS Excel

OData Service

Backend

Data Integration for SDI

Job Scheduling

IAM

Data Integration for CPI-DS

Data Model

SAP CPI-DS

SAP SDI DP Agent

SAP CPI-DS Agent

SAP Integration AddOn

SAC

SAP Analytics Cloud

SAC

SAP Ariba

SAP CP-IAS

SAP CoPilot

SAP Cloud Platform

Customer Cloud Systems

SAP SDI

SAP IPS

SAP IAG

S&OP

Demand

Response & Supply

S/4 Hana or ECC

Other (BW, …)

© 2019 SAP SE or an SAP affiliate company. All rights reserved. | PUBLIC
Data Import Scenario
Sequence of the flow of data for Import jobs

- **Source Data:**
  - Data from any system or .csv files

- **Activation of the objects-Master Data/Planning area creates 3 tables**

- **Staging Table**
  - Intermediate table before loading to core table
  - Used for Data Validation i.e format, structure etc

- **Core Table**
  - Main table where you want to load the data
  - Example Timeseries table/ MDT tables

- **Reporting Tables**
  - Data loaded successfully/processed with errors are logged

---

**Diagram:**

1. **Data Files**
   - Step 1: This step is responsible for getting the data from CSV/any Source Systems.

2. **STAGING Tables**
   - Step 2: Insert/Update/Delete/Replace
   - Step 3: Update the status of records in reporting tables

3. **CORE Tables**
4. **Reporting Tables**
Staging and Rep Tables

- MDT Staging table – Acts as Target.
- Key Figure Staging table for each planning area and version
- Core and Rep Table - Only as a source
- Staging tables do not hold any data - Just used temporarily during the Data Integration Job
- Version Specific MDT share the Staging table - So you need to specify Version for data upload.
- Separate time series table for version key figure data
Best Practices
Avoiding Long Running Jobs

- Flows importing data should be optimized
- Splitting of process flows is recommended to avoid long running jobs
- More import jobs in process means longer times and more delta to be build up
- It also means more load on the system because of more updates on database tables
- Longer runs make system exposed to data locks in case simultaneous job also updates the same table and the same record
- Merging of KF task updating data at same planning level is advised
- The scheduling of jobs should be done in such a way so that overlaps between multiple jobs do not occur
- Understand the impact of shared MDT on job performance
Parallel Jobs

- JOB_THREAD_COUNT allows the parallel Processing.
- Requires perfect Job scheduling to be effective.
- Random Job failure due to database locks.
- Shall not be used for master data uploads.
- Master Data Share planning levels
  - Small key figure uploads at the different planning can be processed in parallel
  - Key figure uploads at same planning level

For e.g PRODUCT and LOCATION might sound as a separate master data, however the planning data also needs to be updated. So updating data at PERPRODLOC can cause database locks and subsequently job failures.
Shared Master Data

• One should review the performance impact of using shared MDT

• Shared MDT across too many planning areas can have severe performance impact

• Data needs to be updated in all the planning areas.

• Updating records in MDT1 will in turn lead to updates in reference MDT’s and also the planning object table in all the 5 planning areas

• This leads to an increase in run time of the jobs loading data to MDT1

• Activate all the planning areas for any configuration changes.

• For e.g. Attribute length changes require re-activation of all planning areas.

• Easy to use the share master data, but can be difficult to maintain.
Shared Master Data

<table>
<thead>
<tr>
<th>Master Data</th>
<th>Planning Object</th>
<th>Time Series Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>MDT</td>
<td>PA1</td>
<td>PA1 (50 Periods)</td>
</tr>
<tr>
<td></td>
<td>PA2</td>
<td>PA2 (100 Periods)</td>
</tr>
<tr>
<td></td>
<td>PA3</td>
<td>PA3 (200 Periods)</td>
</tr>
<tr>
<td>Record Count</td>
<td>10000</td>
<td>500,000</td>
</tr>
<tr>
<td></td>
<td>1,000,000</td>
<td>1,000,000</td>
</tr>
<tr>
<td></td>
<td>2,000,000</td>
<td>2,000,000</td>
</tr>
</tbody>
</table>

Total: 9.5 Million

Time Series Data (If Attribute as MDT is configured)
- All the Attribute as Key Figures needs to be updated.
- Record count X Number of time periods

Simple MDT

Planning Area
- PA1
- PA2
- PA3

Planning Level
- 15 Planning Levels
- 20 Planning Levels
- 30 Planning Levels.

ZPRODUCT

Planning Area: PA1, PA2, PA3
Planning Level: Simple MDT
Replace Job

If a task is executed in ‘REPLACE’ mode all data of the target table is deleted and replaced with the selected data.

So, based on the requirement you have to judge if using replace mode is the best option or a combination of Copy Operator and Insert_update should be used because of following benefits of Copy operator

✓ It only updates where the KF is not NULL
✓ No Need to clear the KF values at the same level multiple times.
✓ You can control the deletion of the key figure data.
✓ Multiple tasks can update the same IBP table
Attribute as Key Figure – Upcoming Changes

Specially Configured master data attribute. Used to create the planning data or key figure data in special use cases.

- **Master Data Integration Job:**
  - Insert/update the master data
  - Update existing planning data
  - Create the time series entries.
  - Repeat the above steps for all associated PAs.

  Because of the prolonged loading time, the master data load can interfere with other processes or jobs that run concurrently, resulting in an even longer runtime and higher memory consumption. If you want to avoid this, the most efficient way to fill the time series of a key figure with values is to load the values directly in a key figure data load.

- **Limit: 1 million records**
  - Time-dependent key figures
  - Normalized planning areas: Creation/update of key figure data is skipped
  - Not-normalized planning areas: A warning is issued
  - Plan to fill 20 periods- You can upload 1 million/20=50,000 master data records

Normalized Planning Areas

To prevent the system from running out of memory, the number of time series entries that can be created or updated is limited to 1 million per attribute as key figure in normalized planning areas. If this limit is exceeded, the system does not create or update the time series entries.
I'm having problems with the master data load – what can I do now?

- Does the key figure need to be time-dependent?
  - No
    - Define the key figure as a time-independent key figure
  - Yes
    - Can I use a time-reference attribute to provide the period?
      - Yes
        - Add a time reference attribute
      - No
        - Specify a time period range
          - Has a time period range been specified?
            - Yes
              - Split the master data load
            - No
              - Does the time period range need to stay as it is?
                - Yes
                  - Model the key figure as a time-independent key figure and use key figure calculations to bring in the time dimension
                - No
                  - Reduce the time period range
          - Does the time period range need to stay as it is?
            - Yes
              - Split the master data load
            - No
              - Can you split the master data load?
                - Yes
                  - Model the key figure as a time-independent key figure and use key figure calculations to bring in the time dimension
                - No
                  - Reduce the time period range
  - Do I need to use attributes as key figures for this master data type?
    - Yes
      - Delete the attribute as key figure definition for this master data type and load the key figure data directly
    - No
      - Can you split the master data load?
        - Yes
          - Model the key figure as a time-independent key figure and use key figure calculations to bring in the time dimension
        - No
          - Reduce the time period range
# Alternative Configurations for Attribute as Key Figure

Refer to the KBA [2922453 - Alternative Configurations for Attribute as Key Figure](http://example.com/2922453)

<table>
<thead>
<tr>
<th>Master Data Type ID</th>
<th>Attribute ID</th>
<th>Planning Area Attribute Description</th>
<th>From Period</th>
<th>To Period</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z96LOCATIONPRODUCT</td>
<td>KF1</td>
<td>KF1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Z96SMDT</td>
<td>ZAAKF</td>
<td>Attribute as KF</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### ZAAKF ZAAKF

**Description:** ZAAKF

**Base Planning Level:** DAYZID

**Type:**

- ☯ ☯

**Key Figures:** 1

**Status:** Active

### Calculations

- **ZAAKF@REQUEST** = SUM("ZAAKF@DAYZID")
- **ZAAKF@TWKZID** = SUM("ZAAKF@DAYZID")

**KF1** KF1

**Calculations**

- **KF1@REQUEST** = SUM("KF1@DAYPRODLOC")
- **KF1@DAYPRODLOC** = SUM("KF1@DAYPRODLOCZID")
- **KF1@DAYPRODLOCZID** = "KF1@PRODLOCZID"
  - Additional Inputs: ZAAKF@DAYZID

**KF1@PRODLOCZID** = ☯
$G\_LOAD\_DATE$

- Submit the data in UTC time zone only.
- So, $G\_LOAD\_DATE$ should always be set to SYSUTCDATE.
- All dates in IBP are stored in UTC.
- To sync between data upload timestamp and completed timestamp.
- This value is as shown to you completed on (converting in your local time when you view in Web-UI).
- You can encounter scenarios (where time-zone is ahead of UTC) for example.
- You might see start date is greater than end date because of the value passed in start date column is SYSDATE().
- Jobs can also fail with “Submitted data has older timestamp than what is already available in system.”
WEB UI uses Excel to handle the downloaded report,

It does not support to download large report due to the technical limitation of Excel itself.

There is no exact limit of specific number of records to be downloaded.

Its dependent on various factors such as the number of columns the file contains or to be specific the complexity of the configuration.

KBA: [CPI task to download data integration report from IBP](#)
- **MSGID** << this is the rejection code/name
- **MSGV1** << parameter &1 provides additional context for rejection
- **MSGV2** << parameter &2 provides additional context for rejection
- **MSGV3** << parameter &3 provides additional context for rejection
- **MSGV4** << parameter &4 provides additional context for rejection

**Input**

<table>
<thead>
<tr>
<th>Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>SOPMD_STAG_DLOCATION_REP</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BATCH</td>
<td>varchar (100)</td>
<td></td>
</tr>
<tr>
<td>FILENAME</td>
<td>varchar (100)</td>
<td></td>
</tr>
<tr>
<td>LASTMODIFIEDEXT</td>
<td>datetime</td>
<td></td>
</tr>
<tr>
<td>MSGID</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td>MSGV1</td>
<td>varchar (256)</td>
<td></td>
</tr>
<tr>
<td>MSGV2</td>
<td>varchar (256)</td>
<td></td>
</tr>
<tr>
<td>MSGV3</td>
<td>varchar (256)</td>
<td></td>
</tr>
</tbody>
</table>

**Output (SOPMD_STAG_DLOCATION...)**

<table>
<thead>
<tr>
<th>Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BATCH</td>
<td>varchar (100)</td>
<td></td>
</tr>
<tr>
<td>FILENAME</td>
<td>varchar (100)</td>
<td></td>
</tr>
<tr>
<td>LASTMODIFIEDEXT</td>
<td>datetime</td>
<td></td>
</tr>
<tr>
<td>MSGID</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td>MSGV1</td>
<td>varchar (256)</td>
<td></td>
</tr>
<tr>
<td>MSGV2</td>
<td>varchar (256)</td>
<td></td>
</tr>
<tr>
<td>MSGV3</td>
<td>varchar (256)</td>
<td></td>
</tr>
<tr>
<td>MSGV4</td>
<td>varchar (256)</td>
<td></td>
</tr>
</tbody>
</table>
External Scheduler
External Scheduler - Trigger Jobs in different customer (Onpremise, Cloud)

Setup

- Define the communication system
- Create communication user
- Configure the communication arrangement (SAP_COM_0064)
Frequent Issues

Connection Issues

Incorrect server - One source of issues in the past has been that the incorrect server was used

- Solution: Check the Communication Arrangement, and test the URL from the arrangement-
  https://myXXXXXX-api.scmibp.ondemand.com/sap/opu/odata/sap/BC_EXT_APPJOB_MANAGEMENT;v=002/JobTemplateSet

Error 401- Unauthorized – The communication user might have been locked

- Solution: You should unlock the communication user and use the correct password

CSRF Token Issues

CSRF Token handling

- Every POST request needs to pass a CSRF Token and a cookie
- Call to the HEAD_REQUEST fetches the csrf-token and the cookie.
- This should be used subsequent POST request.
- CSRF Token missing or might not validate whenever not handled properly.
Next Webinar

SAP IBP Troubleshooting Analysis - Outbound Integration

Registration link:

Thank you.

Deep Singh
Parveen Kumar
IBP Customer Office

Deep.singh02@sap.com
Parveen.kumar01@sap.com