**Document Attributes**

| **Attribute** | **Value** |
| --- | --- |
| Application ID /  Application Name | 295674 - |
| Owner | Yun Wan |
| Owner Contact Information | 732 420 1772 yw3298 |
| Other Attribute |  |

**Revision History**

The following table lists the revision history of this document:

| **Author** | **Date** | **Version #** | **Revision Description** |
| --- | --- | --- | --- |
| Yun Wan | 06/16/17 | 0.01 | Initial draft for 295674 |
| Yun Wan | 06/28/17 | 0.02 | Initial draft for 295674 Cont’d |
| Yun Wan | 07/01/17 | 0.03 | Initial draft for 295674 Cont’d, added one time processing for correlated site, and customer sites |
| Yun Wan | 07/12/17 | 0.04 | 295674: added a GLID address sync-up requirement |
| Yun Wan | 08/02/17 | 0.05 | 295674: added UIS.site as a new source of address as per requested by Ralph and Fathima. All changes are tagged with <295674-1> for differentiation. |
| Tofael Khan | 02/06/18 | 0.06 | <Defect-421578> Use IDIS.GCP\_LOCATION.RELATEDNAME if not NULL else use NAME as usual for type = ‘STATE’ |
| Akarsh V/Mahesh MP | 06/11/18 | 0.07 | Defect 510193: Added IDIS.GCP\_SERVICE as the source for the raw address for Geoaddress validation. Changes tagged with <286475a CR181990> |
| Akarsh V/Mahesh MP | 06/19/18 | 0.08 | Changed IDIS.GCP\_SERVICE to IDIS.GCP\_SERVICE\_LOC\_VW2 under <286475a CR181990> |
| Yun Wan | 11/6/18 | 0.09 | Updated data mapping excel sheets to make requests to OVAL api and batch process (GDB address\_notation as input only) to use address\_line1 ONLY as input instead of the fallback approach based on Tier 2’s suggestion. |

**Table of Contents**

[**Overview** 4](#_Toc351987618)

[**Problem Statement** 4](#_Toc351987619)

[**Design Decisions** 4](#_Toc351987620)

[**Alternative Designs** 5](#_Toc351987621)

[**Assumptions/Risks** 5](#_Toc351987622)

[**Traceability Matrix** 12](#_Toc351987623)

[**Pre-Production Disaster Recovery Planning** 13](#_Toc351987624)

[**Other Plans and References** 13](#_Toc351987625)

[**Acceptance & Approvals** 15](#_Toc351987626)

[**Appendix A: JMS Requirements** 17](#_Toc351987627)

[**Public Queues:** 17](#_Toc351987628)

**Overview**

EDF GDB is using Envinsa third-party vendor as a data source for address normalization and geocoding. However Envinsa license is expired and it is determined that OVALS will be used instead afterwards.

**Problem Statement**

This HLD is about everything that is needed to switch to OVALS from Envinsa.

**Design Decisions**

### Database

#### HLD-295674-EDF-GDB-DBA-010 [ Primary Key ]

Ensure for all processing that the primary key creation is based on the following logic:



#### HLD-295674-EDF-GDB-DBA-020 [ Change System and Change User ]

Make sure the following data is added to the GCP GDB Data:

|  |  |
| --- | --- |
| **GDB.CHANGE\_SYSTEM** | |
| ID | *HLD-295674-EDF-GDB-DBA-010 [ Primary Key ]* |
| NAME | “GEOADDR\_TO\_GDB” |

|  |  |
| --- | --- |
| **GDB.CHANGE\_USER** | |
| ID | *HLD-295674-EDF-GDB-DBA-010 [ Primary Key ]* |
| NAME | “GEOADDR\_TO\_GDB” |

#### HLD-295674-EDF-GDB-DBA-030 [ Metadata ]

Make sure the following data is added to the GCP GDB Data:

|  |  |
| --- | --- |
| **METADATA.SOURCE\_PROCESS** | |
| ID | *HLD-295674-EDF-GDB-DBA-010 [ Primary Key ]* |
| NAME | “OVALS-GEOCODING” |

<295674-1>

|  |  |
| --- | --- |
| **METADATA.SOURCE\_PROCESS** | |
| ID | *HLD-295674-EDF-GDB-DBA-010 [ Primary Key ]* |
| NAME | “UIS.SITE.MAPINFO” |



</295674-1>

#### HLD-295674-EDF-GDB-DBA-040 [ ADDRESS\_NOTATION ]

Make sure the following column is added to the GDB.ADDRESS\_NOTATIOIN table

|  |  |  |  |
| --- | --- | --- | --- |
| **GDB.ADDRESS\_NOTATION** | | | |
| **Name** | **Data Type** | **Nullable** | **Comments** |
| ID\_GLOBAL\_LOCATION | String, A/N 10 | N | PK |

#### HLD-295674-EDF-GEOADDR-DBA-010 [ GEOGRAPHIC\_ADDRESS ]

Ensure the following new column is created in GEOADDR.GEOGRAPHIC\_ADDRESS table.

|  |  |  |  |
| --- | --- | --- | --- |
| **GEOADDR.GEOGRAPHIC\_ADDRESS** | | | |
| **Name** | **Data Type** | **Nullable** | **Comments** |
| OVALS\_GLID | String, A/N 10 | Y |  |

#### ~~HLD-295674-EDF-GEOADDR-DBA-020 [ OVALS\_GLID\_USE ]~~

~~Ensure the following new table is created in GEOADDR schema.~~

|  |  |  |  |
| --- | --- | --- | --- |
| **~~GEOADDR.OVALS\_GLID\_USE~~** | | | |
| **~~Name~~** | **~~Data Type~~** | **~~Nullable~~** | **~~Comments~~** |
| ~~OVALS\_GLID~~ | ~~String, A/N 10~~ | ~~N~~ | ~~PK~~ |
| ~~ID\_SOURCE\_KEY~~ | ~~Number, 20~~ | ~~N~~ | ~~Id\_source\_key for matched OVALS address (GDB’s normalized address~~ |
| ~~ID\_GEOGRAPHIC\_ADDRESS~~ | ~~Number, 10~~ | ~~N~~ |  |

### Initial Batch Data Load

#### HLD-295674-EDF-Initial-Batch-Data-010 [ Process ]

OVALS Batch File Interface only supports Domesitc addresses, EDF will sent all domestic addresses from GDB, and all sources USRP, UIS, and etc. in a file as specified in the interface document. OVALS will send back a file with matched addresses with match code.

#### HLD-295674-EDF-Initial-Batch-Data-020 [ ADDRESS\_NOTATION ]

All the records of GDB.ADDRESS\_NOTATION are qualified for initial batch processing if they satisfy the following conditions:

1. Reference to address\_notation\_type.type ‘INVENTORY\_ADDRESS\_REPRESENTATION’ or ‘INVENTORY\_VIRTUAL\_ADDRESS\_REPRESENTATION’
2. OVALS\_GLID is not populated

#### HLD-295674-EDF-Initial-Batch-Data-030 [ USRP ]

This requirement is an excerpt from HLD-188413a-GCP-GEOCODING-7.1 to retrieve USRP’s addresses, generate EKTs in metadata schema (source\_key, source\_key\_value, source\_process, source\_key\_digest, and etc.). This process could be reused for the initial batch data processing (BAU).

|  |  |
| --- | --- |
| **Meta Data element** | **Value** |
| source\_process.name | “USRP.CUSTOMER\_LOCATION.MapInfo” |
| meta\_system.name | “USRP” |
| meta\_table.name | “CUSTOMER\_LOCATION” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Column name** | **Used by** | **Instruction** |
| ISO\_COUNTRY | LOC\_COUNTRY | SQL join | Join with (1) |
| ISO\_COUNTRY\_CODE | Raw Address | ISO country code |
| CUSTOMER\_LOCATION | CUST\_LOCATION\_ID | Enterprise Key Translation | Source Key Value |
| LOC\_COUNTRY | SQL join | Join with (1) |
| LOC\_STATE | Raw Address | State |
| LOC\_ZIP | Raw Address | Postal code |
| LOC\_CITY | Raw Address | City |
| LOC\_STR1 | Raw Address | Address Candidate |
| LOC\_STR2 | Raw Address | Address Candidate |
| LOC\_STR3 | Raw Address | Address Candidate |

#### HLD-295674-EDF-Initial-Batch-Data-040 [ UIS ]

This requirement is an excerpt from HLD-188413a-GCP-GEOCODING-7.2 to retrieve UIS’s addresses, generate EKTs in metadata schema (source\_key, source\_key\_value, source\_process, source\_key\_digest, and etc.). This process could be reused for the initial batch data processing (BAU).

|  |  |
| --- | --- |
| **Meta Data element** | **Value** |
| source\_process.name | “UIS.UIS\_CKT\_LOC.MapInfo” |
| meta\_system.name | “UIS” |
| meta\_table.name | “UIS\_CKT\_LOC” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Column name** | **Used by** | **Instruction** |
| UIS\_CKT\_LOC | CIRCUIT\_ID | Enterprise Key Translation | Source Key Value |
| CIRCUIT\_SEGMENT\_NUMBER | Enterprise Key Translation | Source Key Value |
| CKL\_NO | Enterprise Key Translation | Source Key Value |
| ESERVICE\_COUNTRY\_CODE | Raw Address | ISO Country Code |
| ESERVICE\_STATE | Raw Address | State |
| ESERVICE\_ZIP\_CODE | Raw Address | Postal code |
| ESERVICE\_CITY | Raw Address | City |
| ESERVICE\_STREET | Raw Address | Address Candidate |

<295674-1>

|  |  |
| --- | --- |
| **Meta Data element** | **Value** |
| source\_process.name | “UIS.SITE.MapInfo” |
| meta\_system.name | “UIS” |
| meta\_table.name | “SITE” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Column name** | **Used by** | **Instruction** |
| SITE | SITE\_ID | Enterprise Key Translation | Source Key Value |
| PREM\_COUNTRY | Raw Address | ISO Country Code |
| PREM\_STATE | Raw Address | State |
| PREM\_ZIP | Raw Address | Postal code |
| PREM\_CITY | Raw Address | City |
| PREM\_ADDRESS | Raw Address | Address Candidate |

</295674-1>

#### HLD-295674-EDF-Initial-Batch-Data-050 [ BVoIP CSI ]

This requirement is an excerpt from HLD-188413a-GCP-GEOCODING-7.3.1, and HLD-188413a-GCP-GEOCODING-7.3.2 to retrieve BVoIP CSI’s addresses, generate EKTs in metadata schema (source\_key, source\_key\_value, source\_process, source\_key\_digest, and etc.). These processes could be reused for the initial batch data processing (BAU).

|  |  |
| --- | --- |
| **Meta Data element** | **Value** |
| source\_process.name | “CSI.CUSTOMER\_SITE.MapInfo” |
| meta\_system.name | “CSI” |
| meta\_table.name | “CUSTOMER\_SITE” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Column name** | **Used by** | **Instruction** |
| CUSTOMER\_SITE | CUSTOMER\_SITE\_ID | Enterprise Key Translation | Source Key Value |
| SITE\_COUNTRY | Raw Address | ISO Country Code |
| SITE\_STATE | Raw Address | State |
| SITE\_ZIP | Raw Address | Postal code |
| SITE\_CITY | Raw Address | City |
| SITE\_ADDRESS | Raw Address | Address Candidate |

|  |  |
| --- | --- |
| **Meta Data element** | **Value** |
| source\_process.name | “CSI.REMOTE\_SITE\_DETAIL.MapInfo” |
| meta\_system.name | “CSI” |
| meta\_table.name | “REMOTE\_SITE\_DETAIL” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Column name** | **Used by** | **Instruction** |
| REMOTE\_SITE\_DETAIL | RM\_SITE\_ID | Enterprise Key Translation | Source Key Value |
| RM\_SITE\_COUNTRY | Raw Address | ISO Country Code |
| RM\_SITE\_STATE | Raw Address | State |
| RM\_SITE\_ZIP | Raw Address | Postal code |
| RM\_SITE\_CITY | Raw Address | City |
| RM\_SITE\_ADDRESS | Raw Address | Address Candidate |

#### HLD-295674-EDF-Initial-Batch-Data-060 [ BVoIP BPO ]

This requirement is an excerpt from HLD-188413a-GCP-GEOCODING-7.3.3, and HLD-188413a-GCP-GEOCODING-7.3.4 to retrieve BVoIP BPO’s addresses, generate EKTs in metadata schema (source\_key, source\_key\_value, source\_process, source\_key\_digest, and etc.). These processes could be reused for the initial batch data processing (BAU).

|  |  |
| --- | --- |
| **Meta Data element** | **Value** |
| source\_process.name | “BPO.CUSTOMER\_SITE.MapInfo” |
| meta\_system.name | “BPO” |
| meta\_table.name | “CUSTOMER\_SITE” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Column name** | **Used by** | **Instruction** |
| CUSTOMER\_SITE | CUSTOMER\_SITE\_ID | Enterprise Key Translation | Source Key Value |
| SITE\_COUNTRY | Raw Address | ISO Country Code |
| SITE\_STATE | Raw Address | State |
| SITE\_ZIP | Raw Address | Postal code |
| SITE\_CITY | Raw Address | City |
| SITE\_ADDRESS | Raw Address | Address Candidate |

|  |  |
| --- | --- |
| **Meta Data element** | **Value** |
| source\_process.name | “BPO.REMOTE\_SITE\_DETAIL.MapInfo” |
| meta\_system.name | “BPO” |
| meta\_table.name | “REMOTE\_SITE\_DETAIL” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Column name** | **Used by** | **Instruction** |
| REMOTE\_SITE\_DETAIL | RM\_SITE\_ID | Enterprise Key Translation | Source Key Value |
| RM\_SITE\_COUNTRY | Raw Address | ISO Country Code |
| RM\_SITE\_STATE | Raw Address | State |
| RM\_SITE\_ZIP | Raw Address | Postal code |
| RM\_SITE\_CITY | Raw Address | City |
| RM\_SITE\_ADDRESS | Raw Address | Address Candidate |

#### HLD-295674-EDF-Initial-Batch-Data-070 [ ICORE ]

This requirement is an excerpt from HLD-188413a-GCP-GEOCODING-7.4 to retrieve ICORE’s addresses, and generate EKTs in metadata schema (source\_key, source\_key\_value, source\_process, source\_key\_digest, and etc.). This process could be reused for the initial batch data processing (BAU).

|  |  |
| --- | --- |
| **Meta Data element** | **Value** |
| source\_process.name | “ICORE.PREMISE.MapInfo” |
| meta\_system.name | “ICORE” |
| meta\_table.name | “PREMISE” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Column name** | **Used by** | **Instruction** |
| COUNTRY | COUNTRY\_NAME | SQL join | Join with (1) |
| ISO\_2CHAR\_CODE | Raw Address | ISO country code |
| PREMISE | LOC\_ID | Enterprise Key Translation | Source Key Value |
| PREM\_COUNTRY | SQL join | Join with (1) |
| PREM\_STATE | Raw Address | State |
| PREM\_ZIP | Raw Address | Postal code |
| PREM\_CITY | Raw Address | City |
| PREM\_ADDRESS | Raw Address | Address candidate |

#### HLD-295674-EDF-Initial-Batch-Data-080 [ LPP-CPE (NC3) ]

This requirement is an excerpt from HLD-188413a-GCP-GEOCODING-7.5 to retrieve NC3’s addresses, and generate EKTs in metadata schema (source\_key, source\_key\_value, source\_process, source\_key\_digest, and etc.). This process could be reused for the initial batch data processing (BAU).

|  |  |
| --- | --- |
| **Meta Data element** | **Value** |
| source\_process.name | “LPP-CPE.SITE.MapInfo” |
| meta\_system.name | “LPP-CPE” |
| meta\_table.name | “SITE” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Column name** | **Used by** | **Instruction** |
| SITE | ODBID | Source Key Translation | Source key value |
| COUNTRY\_CODE | Raw Address | ISO country code |
| STATE | Raw Address | State |
| POSTAL\_CODE | Raw Address | Postal code |
| CITY | Raw Address | City |
| ADDRESS\_LINE1 | Raw Address | Address candidate |
| ADDRESS\_LINE2 | Raw Address | Address candidate |
| ADDRESS\_LINE3 | Raw Address | Address candidate |

#### HLD-295674-EDF-Initial-Batch-Data-090 [ INSTAR ]

This requirement is an excerpt from HLD-188413a-GCP-GEOCODING-7.6 to retrieve INSTAR’s addresses, and generate EKTs in metadata schema (source\_key, source\_key\_value, source\_process, source\_key\_digest, and etc.). This process could be reused for the initial batch data processing (BAU).

|  |  |
| --- | --- |
| **Meta Data element** | **Value** |
| source\_process.name | “INSTAR.PREMISE.MapInfo” |
| meta\_system.name | “INSTAR” |
| meta\_table.name | “PREMISE” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Column name** | **Used by** | **Instruction** |
| COUNTRY | COUNTRY\_NAME | SQL join | Join with (1) |
| COUNTRY\_ABBR | Raw Address | ISO country code |
| PREMISE | LOC\_ID | Enterprise Key Translation | Source Key Value |
| COUNTRY\_NAME | SQL join | Join with (1) |
| STATE\_ABBR | Raw Address | State |
| ZIP | Raw Address | Postal code |
| CITY | Raw Address | City |
| ADDRESS | Raw Address | Address candidate |
| ADDRESS2 | Raw Address | Address candidate |
| ADDRESS3 | Raw Address | Address candidate |

#### HLD-295674-EDF-Initial-Batch-Data-100 [ GPS ]

This requirement is an excerpt from HLD-188413a-GCP-GEOCODING-7.7 to retrieve GPS’s addresses, and generate EKTs in metadata schema (source\_key, source\_key\_value, source\_process, source\_key\_digest, and etc.). This process could be reused for the initial batch data processing (BAU).

|  |  |
| --- | --- |
| **Meta Data element** | **Value** |
| source\_process.name | “GPS.SW\_ADDRESS.MapInfo” |
| meta\_system.name | “GPS” |
| meta\_table.name | “SW\_ADDRESS” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Column name** | **Used by** | **Instruction** |
| AT\_COUNTRY | ATCOUNTRYID | SQL join | Join with (1) |
| ATCOUNTRYNAME | SQL join | Alternate join with (2) |
| ATISOCODE | Raw Address | ISO country code |
| SW\_ADDRESS | SWADDRESSID | Source Key Translation | Source Key Value |
| ATCOUNTRYID | SQL join | Join with (1) |
| SWCOUNTRY | SQL join | Alternate join with (2) |
| SWSTATE | Raw Address | State |
| SWZIP | Raw Address | Postal code |
| SWCITY | Raw Address | City |
| SWADDRESS1 | Raw Address | Address candidate |
| SWADDRESS2 | Raw Address | Address candidate |
| SWADDRESS3 | Raw Address | Address candidate |
| SWADDRESS4 | Raw Address | Address candidate |
| SWADDRESS5 | Raw Address | Address candidate |

#### HLD-295674-EDF-Initial-Batch-Data-110 [ SIDBOR ]

This requirement is an excerpt from HLD-188413a-GCP-GEOCODING-7.8 to retrieve SIDBOR’s addresses, and generate EKTs in metadata schema (source\_key, source\_key\_value, source\_process, source\_key\_digest, and etc.). This process could be reused for the initial batch data processing (BAU).

|  |  |
| --- | --- |
| **Meta Data element** | **Value** |
| source\_process.name | “SIDBOR.ADDRESS.MapInfo” |
| meta\_system.name | “SIDBOR” |
| meta\_table.name | “ADDRESS” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Column name** | **Used by** | **Instruction** |
| LST\_COUNTRY | COUNTY\_CD | SQL join | Join with (1) |
| COUNTRY\_CD2 | Raw Address | ISO country code |
| ADDRESS | ADDR\_ID | Source Key Translation | Source Key Value |
| COUNTRY\_CD | SQL join | Join with (1) |
| STPRV\_CD | Raw Address | State |
| POSTAL\_CD | Raw Address | Postal code |
| CITY\_NM | Raw Address | City |
| ADDR\_LINE\_TX | Raw Address | Address Candidate |
| ADDR\_LINE2\_TX | Raw Address | Address Candidate |
| ADDR\_LINE3\_TX | Raw Address | Address Candidate |
| ADDR\_LINE4\_TX | Raw Address | Address Candidate |
| ADDR\_LINE5\_TX | Raw Address | Address Candidate |

#### HLD-295674-EDF-Initial-Batch-Data-120 [ EMAP05 (ECDB) ]

This requirement is an excerpt from HLD-188413a-GCP-GEOCODING-7.9 to retrieve EMAP05’s addresses, and generate EKTs in metadata schema (source\_key, source\_key\_value, source\_process, source\_key\_digest, and etc.). This process could be reused for the initial batch data processing (BAU).

|  |  |
| --- | --- |
| **Meta Data element** | **Value** |
| source\_process.name | “EMAP05.MAI\_ADDR.MapInfo” |
| meta\_system.name | “EMAP05” |
| meta\_table.name | “MAI\_ADDR” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Column name** | **Used by** | **Instruction** |
| COUNTRY\_CODE | CLC | SQL join | Join with (1) |
| COUNTRY\_NAME | SQL join | Alternate join with (2) |
| ISO3 | Raw Address | ISO country code |
| MAI\_ADDR | SITE\_ID | Source Key Translation | Source Key Value, process only when NOT NULL |
| SRC\_COUNTRY\_NAME | SQL join | Alternate join with (2) |
| SRC\_COUNTRY | SQL join | Join with (1) |
| SRC\_STATE | Raw Address | State |
| SRC\_ZIP | Raw Address | Postal code |
| SRC\_CITY | Raw Address | City |
| SRC\_STREET | Raw Address | Address candidate |

#### HLD-295674-EDF-Initial-Batch-Data-130 [ DSM ]

This requirement is an excerpt from HLD-188413a-GCP-GEOCODING-7.10 to retrieve DSM’s addresses, and generate EKTs in metadata schema (source\_key, source\_key\_value, source\_process, source\_key\_digest, and etc.). This process could be reused for the initial batch data processing (BAU).

|  |  |
| --- | --- |
| **Meta Data element** | **Value** |
| source\_process.name | “INR.DSM\_ADDRESS.MapInfo” |
| meta\_system.name | “INR” |
| meta\_table.name | “DSM\_ADDRESS” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Column name** | **Used by** | **Instruction** |
| DSM\_ADDRESS | ID | Source Key Translation | Source Key Value |
| COUNTRY | Raw Address | Country name  Map to Geographic Address Collection COUNTRY.ISO\_ALPHA2\_CODE via COUNTRY.NAME  Map to ‘US’ where NULL |
| STATE | Raw Address | State |
| ZIP | Raw Address | Postal code |
| CITY | Raw Address | City |
| ADDR1 | Raw Address | Address candidate |
| ADDR2 | Raw Address | Address candidate |

#### HLD-295674-EDF-Initial-Batch-Data-140 [ IDIS ]

This requirement is an excerpt from HLD-188413a-GCP-GEOCODING-7.11 to retrieve IDIS’ addresses, and generate EKTs in metadata schema (source\_key, source\_key\_value, source\_process, source\_key\_digest, and etc.). This process could be reused for the initial batch data processing (BAU).

|  |  |
| --- | --- |
| **Meta Data element** | **Value** |
| source\_process.name | “IDIS.GCP\_LOCATION.MapInfo” |
| meta\_system.name | “IDIS” |
| meta\_table.name | “GCP\_LOCATION” |

NM55 device’s addresses:

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Column name** | **Used by** | **Instruction** |
| GCP\_LOCATION | LOCATIONID of the immediate row. | Source Key Translation | Source Key Value |
| ‘US’ | Raw Address | Country code |
| PROVINCE | Raw Address | State |
| ZIP | Raw Address | Postal code |
| TOWNCITY | Raw Address | City |
| ADDRESS | Raw Address | Address candidate |
| ADDRESS1 | Raw Address | Address candidate |
| ADDRESS2 | Raw Address | Address candidate |
| ADDRESS3 | Raw Address | Address candidate |

For MoW records, the above hierarchical structure is not present in Canopi. Hence the data will be loaded from the ‘EQUIPMENT LOCATION’ type record directly. MoW records can be determined by checking the ADDRESS3 field where it is NOT NULL and not ‘USA’. The fields should be loaded from:

* City from TOWNCITY
* State from PROVINCE
* Country from ADDRESS3 (note, this will be ISO-3 country code, ‘GBR’, ‘DEU’, ‘SGP’ etc)

Domestic Addresses, and non-NM55 addresses:

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Column name** | **Used by** | **Instruction** |
| GCP\_LOCATION | LOCATIONID of the immediate row. | Source Key Translation | Source Key Value |
| NAME, where TYPE=’COUNTRY’, lookup via the parent hierarchy. | Raw Address | Country code  Use a default of ‘US’ where not existing for the processed row |
| <Defect-421578> RELATEDNAME if not NULL or Empty else  NAME, where TYPE=’STATE’, lookup via the parent hierarchy. | Raw Address | State |
| ZIP of the immediate row. | Raw Address | Postal code |
| NAME, where TYPE=’CITY’, lookup via the parent hierarchy.  <Defect 47478>  Or  TOWNCITY, if ‘CITY’ is not present in hierarchy | Raw Address | City |
| ADDRESS of the immediate row if not empty. | Raw Address | Address candidate |
| ADDRESS1 of the immediate row if not empty. | Raw Address | Address candidate |
| ADDRESS2 of the immediate row if not empty. | Raw Address | Address candidate |
| ADDRESS3 of the immediate row if not empty. | Raw Address | Address candidate |

Note: Refer to HLD-188413a-GCP-GEOCODING-7.11 for the details regarding the difference in MoW, NM55 devices, and IDIS location hierarchy structure.

<286475a CR181990>

|  |  |
| --- | --- |
| **Meta Data element** | **Value** |
| source\_process.name | “~~IDIS.GCP\_SERVICE.MapInfo~~”  “IDIS.GCP\_SERVICE\_LOC\_VW2.MapInfo” |
| meta\_system.name | “IDIS” |
| meta\_table.name | “GCP\_SERVICE\_LOC\_VW2” |

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Column name** | **Used by** | **Instruction** |
| ~~GCP\_SERVICE~~  GCP\_SERVICE\_LOC\_VW2 | SERVICEID of the immediate row. | Source Key Translation | Source Key Value |
|  | Raw Address | Country code  Use a default of ‘USA’ where not existing for the processed row |
| STATE | Raw Address | State |
|  | Raw Address | Postal code  Need to be identified during normalization process. |
| CITY | Raw Address | City |
| ADDRESS1 of the immediate row if not empty. | Raw Address | Address candidate |
| ADDRESS2 of the immediate row if not empty. | Raw Address | Address candidate |

</286475a CR181990>

#### HLD-295674-EDF-Initial-Batch-Data-200 [ OVALS Batch Request File ]

All the address records and corresponding EKT or address\_notation.id from the following sections should be dumped into a file based on the format defined in interface document and sent to OVALS GIS for address matches.

*HLD-295674-EDF-Initial-Batch-Data-020 [ ADDRESS\_NOTATION ]*

*HLD-295674-EDF-Initial-Batch-Data-030 [ USRP ]*

*HLD-295674-EDF-Initial-Batch-Data-040 [ UIS ]*

*HLD-295674-EDF-Initial-Batch-Data-050 [ BVoIP CSI ]*

*HLD-295674-EDF-Initial-Batch-Data-060 [ BVoIP BPO ]*

*HLD-295674-EDF-Initial-Batch-Data-070 [ ICORE ]*

*HLD-295674-EDF-Initial-Batch-Data-080 [ LPP-CPE (NC3) ]*

*HLD-295674-EDF-Initial-Batch-Data-090 [ INSTAR ]*

*HLD-295674-EDF-Initial-Batch-Data-100 [ GPS ]*

*HLD-295674-EDF-Initial-Batch-Data-110 [ SIDBOR ]*

*HLD-295674-EDF-Initial-Batch-Data-120 [ EMAP05 (ECDB) ]*

*HLD-295674-EDF-Initial-Batch-Data-130 [ DSM ]*

*HLD-295674-EDF-Initial-Batch-Data-140 [ IDIS ]*

Since OVALS Batch interface only supports domestic addresses, all the addresses from above are not domestic in any format should be excluded. For example, country are in these values (case insensitive and not limited to): ‘USA’, ‘US’, ‘U.S.A’, ‘UNITED STATES’, ‘UNITED STA’, and ‘UNITED STATES OF AMERICA’.

In order to avoid from sending duplicated addresses to OVALS GIS, any EKTs from sources (USRP, UIS, BVoIP CSI, and etc.) should be checked with address\_notation.id from *HLD-295674-EDF-Initial-Batch-Data-020 [ ADDRESS\_NOTATION ]*, any EKTs are already in the address\_notation.id from *HLD-295674-EDF-Initial-Batch-Data-020 [ ADDRESS\_NOTATION ]* should be removed.

In addition, any EKTs from above are existing in GEOADDR.GEOGRAPHIC\_ADDRESS\_USE, and OVALS\_GLID is populated in GEOADDR.GEOGRAPHIC\_ADDRESS should be removed too. If the initial batch process failed in the middle, and need to be kicked off again, this condition will avoid from sending the address processed previously.

All the records selected from above should be dumped into a file (pipe-delimited) based on OVALS batch file interface and the following data mapping:



The dump file should be sent to OVALS based on the batch file interface via Connect Direct, and once the file is sent, a new record should be sent into OVAL GIS table GLID\_BATCH\_RUN to specify the file name, app\_name, and etc. as below:

|  |  |  |
| --- | --- | --- |
| **Column** | **Data type** | **Remarks** |
| ID | number | primary key, GLIDBATCH\_ID\_SEQ.nextval |
| APP\_NAME | varchar2(15) | name of the app which is sending file, ‘edf’ |
| EMAIL\_IDS | varchar2(100) | comma separted email IDs, OVALS GIS will send completion notification |
| STATUS | varchar2(1) | N - indicates new file (set by client app) P1 - indicates EGM in progress(set by ovals gis) P2 - indicates supplemental standadizer in progress(set by ovals gis) P3 - indicates existing GLID check in progress(set by ovals gis) P4 - indicates new GLID creation in progress(set by ovals gis) C - indicates processing complete(set by ovals gis) E - indicates processing failed/couldn’t process/file corrupt(set by ovals gis) X - indicates client app has picked up the record |
| FILE\_NAME | varchar2(50) | name of the file stored in the folder |
| INPUT\_TS | timestamp | time when file was sent to ovals gis |
| START\_TS | timestamp | time when processing started |
| END\_TS | timestamp | time when processing completed/interim times |
| REMARKS | varchar2(150) | error reason - updated in case of status E |

OVALS Batch File has limit on number of records in a single file, which is 2 million. EDF will send multiple files if needed with 2 millions of records in each file.

#### HLD-295674-EDF-Initial-Batch-Data-300 [ OVALS Batch Response File ]

Once OVAL GIS sent the response file to the specified folder in GLID\_BATCH\_RUN record (sent in request), the GLID\_BATCH\_RUN.status will be updated to ‘C’ for the request record. In case that EDF gets status ‘E’, EDF will try to send the file again.

Each record in response file should be processed into GEOADDR schema and/or GDB schema based on the following sections:

*OVALS Data Loading*

*GDB Data Loading*

All the addresses retrieved from OVALS will be referred as Normalized Address in the document.

### Initial International Address Load

#### HLD-295674-EDF-Initial-International-Address-010 [ Process ]

OVALS Batch File Interface only support Domesitc addresses, EDF’s and it sources’s international addresses can’t be sent to OVALS via the Batch File Interface. Instead, EDF will send international addresses to API ProcessLocationAttributes (PLA) and wait for the synchronous response. In order not to overwhelm the API server’s system load, EDF should control the number of requests, for example, one or few at a time.

#### HLD-295674-EDF-Initial-International-Address-020 [ PLA Request ]

All the international addresses from the following sections should be sent to API PLA:

*HLD-295674-EDF-Initial-Batch-Data-020 [ ADDRESS\_NOTATION ]*

*HLD-295674-EDF-Initial-Batch-Data-030 [ USRP ]*

*HLD-295674-EDF-Initial-Batch-Data-040 [ UIS ]*

*HLD-295674-EDF-Initial-Batch-Data-050 [ BVoIP CSI ]*

*HLD-295674-EDF-Initial-Batch-Data-060 [ BVoIP BPO ]*

*HLD-295674-EDF-Initial-Batch-Data-070 [ ICORE ]*

*HLD-295674-EDF-Initial-Batch-Data-080 [ LPP-CPE (NC3) ]*

*HLD-295674-EDF-Initial-Batch-Data-090 [ INSTAR ]*

*HLD-295674-EDF-Initial-Batch-Data-100 [ GPS ]*

*HLD-295674-EDF-Initial-Batch-Data-110 [ SIDBOR ]*

*HLD-295674-EDF-Initial-Batch-Data-120 [ EMAP05 (ECDB) ]*

*HLD-295674-EDF-Initial-Batch-Data-130 [ DSM ]*

*HLD-295674-EDF-Initial-Batch-Data-140 [ IDIS ]*

In order to avoid from sending duplicated addresses to OVALS GIS, any EKTs from sources (USRP, UIS, BVoIP CSI, and etc.) should be checked with address\_notation.id from *HLD-295674-EDF-Initial-Batch-Data-020 [ ADDRESS\_NOTATION ]*, any EKTs are already in the address\_notation.id from *HLD-295674-EDF-Initial-Batch-Data-020 [ ADDRESS\_NOTATION ]* should be removed.

In addition, any EKTs from above are existing in GEOADDR.GEOGRAPHIC\_ADDRESS\_USE, and OVALS\_GLID is populated in GEOADDR.GEOGRAPHIC\_ADDRESS should be removed too. If the initial batch process failed in the middle, and need to be kicked off again, this condition will avoid from sending the address processed previously.

International Addresses should be sent based on the following data mapping:



All the addresses retrieved from above will be referred as Raw Address in the document.

#### HLD-295674-EDF-Initial-International-Address-030 [ PLA Response ]

API ProcessLocationAttributes’ response should be processed into GDB schema, metadata schema, and GEOADDR schema in a similar way as *HLD-295674-EDF-Initial-Batch-Data-300 [ OVALS Batch Response File ]*.

PLA API Response could have multiple ProcessLocationAttributesResponse/InternationalLocationAttributes, and in this case, process only the one with highest standardizedScore value.

The address in response should be processed into GEOADDR schema and/or GDB schema based on the following sections:

*OVALS Data Loading*

*GDB Data Loading*

All the addresses retrieved from OVALS will be referred as Normalized Address in the document.

### Delta Daily Address Load

#### HLD-295674-EDF-Delta-Address-010 [ Process ]

EDF will utilize API ProcessLocationAttributes (PLA) for the delta changes in GDB.ADDRESS\_NOTATION and other sources defined in the following sections:

*HLD-295674-EDF-Initial-Batch-Data-020 [ ADDRESS\_NOTATION ]*

*HLD-295674-EDF-Initial-Batch-Data-030 [ USRP ]*

*HLD-295674-EDF-Initial-Batch-Data-040 [ UIS ]*

*HLD-295674-EDF-Initial-Batch-Data-050 [ BVoIP CSI ]*

*HLD-295674-EDF-Initial-Batch-Data-060 [ BVoIP BPO ]*

*HLD-295674-EDF-Initial-Batch-Data-070 [ ICORE ]*

*HLD-295674-EDF-Initial-Batch-Data-080 [ LPP-CPE (NC3) ]*

*HLD-295674-EDF-Initial-Batch-Data-090 [ INSTAR ]*

*HLD-295674-EDF-Initial-Batch-Data-100 [ GPS ]*

*HLD-295674-EDF-Initial-Batch-Data-110 [ SIDBOR ]*

*HLD-295674-EDF-Initial-Batch-Data-120 [ EMAP05 (ECDB) ]*

*HLD-295674-EDF-Initial-Batch-Data-130 [ DSM ]*

*HLD-295674-EDF-Initial-Batch-Data-140 [ IDIS ]*

All the addresses selected above needs to go through the following filtering to be the candidates of delta data loading.

1. any EKTs from sources (USRP, UIS, BVoIP CSI, and etc.) should be checked with address\_notation.id from *HLD-295674-EDF-Initial-Batch-Data-020 [ ADDRESS\_NOTATION ]*, any EKTs are already in the address\_notation.id from *HLD-295674-EDF-Initial-Batch-Data-020 [ ADDRESS\_NOTATION ]* should be removed.
2. Any EKT and address’s digest value should be calculated and compare with metadata.source\_key\_digest.source\_digest for the same EKT. If the calculated digest value is the same as metadata.source\_key\_digest.source\_digest value, the address should be removed from the candidate list.

Each International Address from the candidate list could follow the process defined in *HLD-295674-EDF-Initial-International-Address-020 [ PLA Request ]* to make a PLA request, and handle the response as described in *HLD-295674-EDF-Initial-International-Address-030 [ PLA Response ]*

Each Domestic Address from the candidate list could follow the following sections to make a PLA request and handle the corresponding response.

All the addresses retrieved from above will be referred as Raw Address in the document.

#### HLD-295674-EDF-Delta-Domestic-Address-020 [ PLA Request ]

Domestic Address should be sent to PLA API base on the following data mapping:



#### HLD-295674-EDF-Delta-Domestic-Address-030 [ PLA Response ]

The address in response should be processed into GEOADDR schema and/or GDB schema based on the following sections:

*OVALS Data Loading*

*GDB Data Loading*

All the addresses retrieved from OVALS will be referred as Normalized Address in the document.

### Monthly Address Load

#### HLD-295674-EDF-Monthly-Address-010 [ Process ]

EDF will utilize Batch File Interface (Domesitc) and API ProcessLocationAttributes (PLA, for international addresses) for the monthly data loading of addresses in GDB.ADDRESS\_NOTATION and other sources defined in the following sections:

*HLD-295674-EDF-Initial-Batch-Data-020 [ ADDRESS\_NOTATION ]*

*HLD-295674-EDF-Initial-Batch-Data-030 [ USRP ]*

*HLD-295674-EDF-Initial-Batch-Data-040 [ UIS ]*

*HLD-295674-EDF-Initial-Batch-Data-050 [ BVoIP CSI ]*

*HLD-295674-EDF-Initial-Batch-Data-060 [ BVoIP BPO ]*

*HLD-295674-EDF-Initial-Batch-Data-070 [ ICORE ]*

*HLD-295674-EDF-Initial-Batch-Data-080 [ LPP-CPE (NC3) ]*

*HLD-295674-EDF-Initial-Batch-Data-090 [ INSTAR ]*

*HLD-295674-EDF-Initial-Batch-Data-100 [ GPS ]*

*HLD-295674-EDF-Initial-Batch-Data-110 [ SIDBOR ]*

*HLD-295674-EDF-Initial-Batch-Data-120 [ EMAP05 (ECDB) ]*

*HLD-295674-EDF-Initial-Batch-Data-130 [ DSM ]*

*HLD-295674-EDF-Initial-Batch-Data-140 [ IDIS ]*

All the addresses selected above needs to go through the following filtering to be the candidates of delta data loading.

1. any EKTs from sources (USRP, UIS, BVoIP CSI, and etc.) should be checked with address\_notation.id from *HLD-295674-EDF-Initial-Batch-Data-020 [ ADDRESS\_NOTATION ]*, any EKTs are already in the address\_notation.id from *HLD-295674-EDF-Initial-Batch-Data-020 [ ADDRESS\_NOTATION ]* should be removed.
2. Any EKT and its corresponding GEOADDR.GEOGRAPHIC\_ADDRESS\_USE and GEOADDR.GEOGRAPHIC\_ADDRESS referencing to a GEOADDR.GEOCODE\_QUALITY with match\_type not like ‘S5%’, ‘S6%’, ‘S7%’, or ‘S8%’ .

All the addresses retrieved from above will be referred as Raw Address in the document.

Follow the process described in to *HLD-295674-EDF-Initial-International-Address-030 [ PLA Response ]* (International) or *HLD-295674-EDF-Delta-Domestic-Address-030 [ PLA Response ]* (Domestic) process the response.

### GLID Address Sync-Up Load

This requirement described the process to sync up address data for the GLIDs with updates in OVALS GIS.

#### HLD-295674-EDF-GLID-Address-Sync-up-010 [ Process ]

OVALS GIS could update the address data for some GLIDs, or retire some GLIDs if same addresse for different GLIDs are detected. The updates will be stored in a table GIS\_EGM\_RFRSH, and retired GLIDs will be stored in a table GLID\_MASTER\_RETD. These records in these two tables could be retained for a month before removed. Each record in these two tables has a timestamp (OVAL GIS’s time) associated.

These two tables’ structures are defined the AID, and listed as below:

|  |  |  |
| --- | --- | --- |
| **GIS\_EGM\_RFRSH** | | |
| **Column Name** | **Data Type** | **Description** |
| GLID | VARCHAR2(10) | GIS GLID |
| HSE\_NBR | VARCHAR2(20) | House number from EGM. |
| STR\_DIR | VARCHAR2(20) | Standardized pre directional from EGM. |
| STR\_NM | VARCHAR2(120) | Standardized street name from EGM. |
| STR\_THRGHFR | VARCHAR2(20) | Standardized thoroughfare from EGM. |
| STR\_NM\_SUFX | VARCHAR2(20) | Standardized post directional from EGM. |
| ADDR | VARCHAR2(200) | Standardized street name including house number, directionals and thoroughfare from EGM. |
| STRUCTURE\_TYPE | VARCHAR2(20) | Standardized structure type. |
| STRUCTURE\_VALUE | VARCHAR2(10) | Structure value. |
| LEVEL\_TYPE | VARCHAR2(20) | Standardized level type. |
| LEVEL\_VALUE | VARCHAR2(10) | Level value. |
| UNIT\_TYPE | VARCHAR2(20) | Standardized unit type. |
| UNIT\_VALUE | VARCHAR2(10) | Unit value. |
| CITY | VARCHAR2(50) | Standardized city from EGM. |
| ST | VARCHAR2(20) | Standardized state from EGM. |
| ZIP\_CD | VARCHAR2(5) | Zip5 from EGM. |
| ZIP4 | VARCHAR2(5) | Zip4 from EGM. |
| LAT | VARCHAR2(20) | Latitude from EGM. |
| LONGTD | VARCHAR2(20) | Longitude from EGM. |
| MTCH\_CD | VARCHAR2(5) | EGM Match Code: Indicates the address portions that matched or did not match to the database. See attached Location\_Match\_Codes.pdf for details. |
| LOC\_CD | VARCHAR2(5) | EGM Location Code: Indicates the accuracy of the assigned geocode. See attached Location\_Match\_Codes.pdf for details. |
| CBSA | VARCHAR2(50) | The CBSA name. |
| CBSAID | VARCHAR2(11) | The CBSA ID. |
| CHG\_CD | VARCHAR2(10) | Change code indicating what fields have changed. Each position in the change code represents a field. The value “1” in that position represents a change whereas the value “0” represents no change.   |  |  | | --- | --- | | **Position** | **Changed Field** | | 1 | MTCH\_CD Change | | 2 | LOC\_CD Change | | 3 | LAT Change | | 4 | LONGTD Change | | 5 | CBSA Change | | 6 | CBSAID Change | | 7 | Reserved for future use | | 8 | Reserved for future use | | 9 | Reserved for future use | | 10 | Reserved for future use |   Example: “1000100000” – The MTCH\_CD and CBSA fields changed, all other fields remained the same. |
| ADDR\_CHG\_CD | VARCHAR2(5) | Address change code indicating what address components have changed. The value “1” in that position represents a change whereas the value “0” represents no change.   |  |  | | --- | --- | | **Position** | **Changed Field** | | 1 | HSE\_NBR Change | | 2 | STR\_NM Change | | 3 | CITY Change | | 4 | ZIP Change | | 5 | STATE Change |   Example: “01010” – The STR\_NM and ZIP fields changed, all other fields remained the same. |
| LAST\_UPDT\_DT | TIMESTAMP | Timestamp of the last update. |

|  |  |  |
| --- | --- | --- |
| **GLID\_MASTER\_RETD** | | |
| **Column Name** | **Data Type** | **Description** |
| RETD\_GLID | VARCHAR2(10) | GLID that was retired. |
| ACTV\_GLID | VARCHAR2(10) | GLID that replaces the retired GLID. |
| RETD\_DT | TIMESTAMP | Date when the GLID was retired. |

In order to avoid from missing any updated or retired GLIDs, this sync-up process should run more frequently than the frequency of the table clean-up in OVALS GIS. Hence, it is recommended to run this process weekly.

Every execution of this sync-up process needs to identify the records from above two tables based on the timestamp, LAST\_UPDT\_DT and/or RETD\_DT. Initially, before running the process described in Initial Batch Data Load, EDF should query OVALS GIS database’s sysdate as the Initial/Start timestamp for this process. Any execution of this process should query OVALS GIS database’s sysdate as End timestamp. Any subsequential execution of this process should take last End timestamp minus 1 hour (grace period) as new Start timestamp.

The reason for the 1-hour grace period is that OVALS GIS inserts records into those two tables and commit for every 1000 records, and 1 hour should be good enough to insert 1000 records and commit.

Follow the following sections to sync up the updated/retired GLIDs into GEOADDR schema and GDB schema.

#### HLD-295674-EDF-GLID-Address-Sync-up-020 [ GIS\_EGM\_RFRSH ]

Select candidate records from GIS\_EGM\_RFRSH based on the following criteria:

1. LAST\_UPDT\_DT is between Start timestamp and End timestamp
2. CHG\_CD value’s position 1, 3, 4 has any value 1.

For each candidate record identified above, search GEOADDR.GEOGRAPHIC\_ADDRESS corresponding record with the same GLID (OVALS\_GLID). If there is no match, ignore the candidate record, and continue with next candidate.

For matched GEOGRAPHIC\_ADDRESS record, update the address data as below based on CHG\_CD and ADDR\_CHG\_CD values, and the following data mapping:

|  |  |  |
| --- | --- | --- |
| **Address Fields** | **GIS\_EGM\_RFRSH** | **Comments** |
| HOUSE\_NUM | HSE\_NBR | ADDR\_CHG\_CD =1 and CHG\_CD = 1 |
| STREET\_DIR | STR\_DIR | ADDR\_CHG\_CD =2 and CHG\_CD = 1 |
| STREET\_NM | STR\_NM | ADDR\_CHG\_CD =2 and CHG\_CD = 1 |
| STREET\_THOROUGHFARE | STR\_THRGHFR | ADDR\_CHG\_CD =2 and CHG\_CD = 1 |
| STREET\_NAME\_SUFFIX | STR\_NM\_SUFX | ADDR\_CHG\_CD =2 and CHG\_CD = 1 |
| BASE\_ADDRESS | ADDR | ADDR\_CHG\_CD =1, and/or 2 and CHG\_CD = 1 |
| CITY | CITY | ADDR\_CHG\_CD =3 and CHG\_CD = 1 |
| STATE | ST | ADDR\_CHG\_CD =5 and CHG\_CD = 1 |
| ZIP5 | ZIP\_CD | ADDR\_CHG\_CD =4 and CHG\_CD = 1 |
| ZIP4 | ZIP4 | ADDR\_CHG\_CD =4 and CHG\_CD = 1 |
| GIS\_GLID | GLID | Match key |
| LAT | LAT | CHG\_CD = 3 |
| LONG | LONGTD | CHG\_CD = 4 |

CHG\_CD = 3 and/or 4, follow *HLD-295674-EDF-GEOADDR-from-OVALS-20 [ Data Loading ]* to reprocess GEOADDR’s GEOCODE record, and GEOGRAPHIC\_ADDRESS.id\_geocode.

CHG\_CD = 1, and ADDR\_CHG\_CD is 3, 5, and/or 4, follow *HLD-295674-EDF-GEOADDR-from-OVALS-20 [ Data Loading ]* to reprocess GEOADDR’s MUNICIPALITY, SUBDIVISION, and/or POSTAL\_CODE records correspondingly, and GEOGRAPHIC\_ADDRESS’ id\_municipality, id\_subdivision, and/or id\_postal\_code.

CHG\_CD = 1, and ADDR\_CHG\_CD is 1, and/or 2, follow *HLD-295674-EDF-GEOADDR-from-OVALS-20 [ Data Loading ]* to reprocess GEOADDR’s STREET and BUILDING records, and GEOGRAPHIC\_ADDRESS’s id\_building, id\_street.

All the changes from above (GEOGRAPHIC\_ADDRESS) should be propagated to corresponding GDB.ADDRESS\_NOTATION record [ via EKT ].

Find all the records of GEOGRAPHIC\_ADDRESS\_USE whose id\_geographic\_address is referencing to the GEOGRAPHIC\_ADDRESS record got updated from above, and update corresponding metadata.source\_key\_digest.source\_digest value to 0 so that next *Delta Daily Address Load* will pick it up to reprocess.

A special handling is needed when any changes on id\_geocode, id\_ municipality, id\_subdivision, id\_postal\_code, id\_building, and/or id\_street cause the violation of unique index on GEOGRAPHIC\_ADDRESS table. Basically, there is another conflicted record in the table with the same combination on the columns in the unique index. In this case, find the conflicted record in GEOGRAPHIC\_ADDRESS, and process both as below:

1. If the conflicted GEOGRAPHIC\_ADDRESS record has no populated OVALS\_GLID, populate it with GIS\_EGM\_RFRSH’s GLID. Populate corresponding GDB.ADDRESS\_NOTATION’s id\_global\_location too.
2. Find all the records of GEOGRAPHIC\_ADDRESS\_USE whose id\_geographic\_address is referencing to the changed GEOGRAPHIC\_ADDRESS record, and for each id\_source\_key, update corresponding metadata.source\_key\_digest.source\_digest value to 0 so that next *Delta Daily Address Load* will pick it up to reprocess.

#### HLD-295674-EDF-GLID-Address-Sync-up-030 [ GLID\_MASTER\_RETD ]

Select candidate records from GLID\_MASTER\_RETD based on the following criteria:

1. RETD\_DT is between Start timestamp and End timestamp

For each GLID\_MASTER\_RETD record, RETD\_GLID is the retired GLID, and ACTV\_GLID is the new GLID to replace the RETD\_GLID.

Process the RETD\_GLID and ACTV\_GLID as belows for each candidate record from GLID\_MASTER\_RETD:

1. Find the record in GEOADDR.GEOGRAPHIC\_ADDRESS whose OVALS\_GLID matches with RETD\_GLID, if no record is found, ignore the current candidate record, and continue with next one.
2. Find the record in GEOADDR.GEOGRAPHIC\_ADDRESS whose OVALS\_GLID matches with ACTV\_GLID. If no record is found:
3. Ensure the OVALS\_GLID of the record found from step 1 is updated to ACTV\_GLID
4. Ensure the ID\_GLOBAL\_LOCATIONs of GDB.ADDRESS\_NOTATION record corresponding to GEOADDR.GEOGRAPHIC\_ADDRESS record [ via EKT ] from step 1 is updated to ACTV\_GLID too.
5. Continue with step 3.

Otherwise, continue with the following steps.

1. Find all the records of GEOGRAPHIC\_ADDRESS\_USE whose id\_geographic\_address is referencing to the GEOGRAPHIC\_ADDRESS record from step 1. For each id\_source\_key, update corresponding metadata.source\_key\_digest.source\_digest value to 0 so that next *Delta Daily Address Load* will pick it up to reprocess.
2. ~~Ensure to find all GDB.ADDRESS\_NOTATION records whose id\_address\_notation\_unified is the corresponding GDB.ADDRESS\_NOTATION record of GEOADDR.GEOGRAPHIC\_ADDRESS from step 1 (RETD\_GLID) [ via EKT ] , and update the id\_address\_notation\_unified to the GDB.ADDRESS\_NOTATION record of GEOADDR.GEOGRAPHIC\_ADDRESS from step 2 (ACTV\_GLID) [ via EKT ].~~
3. ~~Follow~~ *~~HLD-295674-EDF-GDB-from-GEOADDR-20 [ Correlated Site ]~~* ~~to ensure creating only a single Correlated SITE to ORGANIZATION ‘CONTRACTED\_BY’ ASSOCIATION record per SITE.id/ORGANIZATION.id pair.~~
4. ~~Follow~~ *~~HLD-295674-EDF-GDB-from-GEOADDR-20 [ Customer Site ]~~* ~~to ensure creating only a single Customer SITE to ORGANIZATION (account) ‘CONTRACTED\_BY’ ASSOCIATION record per ASSET.id/ORGANIZATION.id (account) pair.~~

### OVALS Match Code and Standardized Score

#### HLD-295674-EDF-OVALS-MatchCode-010 [ Data Mapping]

EDF was using Envinsa’s match codes and take addresses with match code S5, S6, S7 and S8. OVALS’s match code is different from Envinsa’s. In order to reduce the impacts on EDF Geocoding process, OVALS’s match code/location code should be mapped to Envinsa’s values.

GEOADDR schema has a table GEOCODE\_QUALITY which is used to store the match code from Envinsa. Same table could be used too for OVALS’ data.

#### HLD-295674-EDF-OVALS-Batch-File-MatchCode-020 [ Data Mapping]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **GEOADDR.GEOCODE\_QUALITY** | | **OVALS Batch File** | |  |
| **MATCH\_TYPE** | **ACCURACY** | **MATCHCODE** | **LOCATIONCODE** | **COMMENT** |
| S5-MATCHCODE-LOCATIONCODE | 1 | S?? | A?? | Prefix S5, concatenates with matchcode, locationcode from OVALS |
| S5-MATCHCODE-LOCATIONCODE | 1 | A?? | A?? | Prefix S5, concatenates with matchcode, locationcode from OVALS |
| S5-MATCHCODE-LOCATIONCODE | 1 | T?? | A?? | Prefix S5, concatenates with matchcode, locationcode from OVALS |
| S5-MATCHCODE-LOCATIONCODE | 1 | Q?? | A?? | Prefix S5, concatenates with matchcode, locationcode from OVALS |
| S5-MATCHCODE-LOCATIONCODE | 1 | Y?? | A?? | Prefix S5, concatenates with matchcode, locationcode from OVALS |
| S5-MATCHCODE-LOCATIONCODE | 1 | X?? | A?? | Prefix S5, concatenates with matchcode, locationcode from OVALS |
| S5-MATCHCODE-LOCATIONCODE | 1 | R?? | A?? | Prefix S5, concatenates with matchcode, locationcode from OVALS |
| Z1-MATCHCODE-LOCATIONCODE | 0 | <ANY other combinations> | <ANY other combinations> | Prefix Z1, concatenates with matchcode, locationcode from OVALS |

Note:

In case, if a raw address gets a matchcode starting with ‘E’ from Batch File Interface, update the raw address’s source\_digest value to 0 so that *Delta Daily Address Load* will pick it up to process it again via PLA API.

#### HLD-295674-EDF-OVALS-PLA-International-Address-StarndardizedScore-030 [ Data Mapping]

|  |  |  |  |
| --- | --- | --- | --- |
| **GEOADDR.GEOCODE\_QUALITY** | | **PLA** |  |
| **MATCH\_TYPE** | **ACCURACY** | **standardizedScore** | **COMMENT** |
| S5-standardizedScore | 1 | 700-799 | Prefix S5, concatenates with standardizedScore from PLA |
| S6-standardizedScore | 1 | 800-899 | Prefix S6, concatenates with standardizedScore from PLA |
| S7-standardizedScore | 1 | >899 | Prefix S7, concatenates with standardizedScore from PLA |
| Z1-standardizedScore | 0 | <700 | Prefix Z1, concatenates with standardizedScore from PLA |

#### HLD-295674-EDF-OVALS-PLA-Domestic-Address-MatchCode-040 [ Data Mapping]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **GEOADDR.GEOCODE\_QUALITY** | | **PLA** | |  |
| **MATCH\_TYPE** | **ACCURACY** | **gisMatchCode** | **gisLocationCode** | **COMMENT** |
| S5-MATCHCODE-LOCATIONCODE | 1 | S?? | A?? | Prefix S5, concatenates with matchcode, locationcode from OVALS |
| S5-MATCHCODE-LOCATIONCODE | 1 | A?? | A?? | Prefix S5, concatenates with matchcode, locationcode from OVALS |
| S5-MATCHCODE-LOCATIONCODE | 1 | T?? | A?? | Prefix S5, concatenates with matchcode, locationcode from OVALS |
| S5-MATCHCODE-LOCATIONCODE | 1 | Q?? | A?? | Prefix S5, concatenates with matchcode, locationcode from OVALS |
| S5-MATCHCODE-LOCATIONCODE | 1 | Y?? | A?? | Prefix S5, concatenates with matchcode, locationcode from OVALS |
| S5-MATCHCODE-LOCATIONCODE | 1 | X?? | A?? | Prefix S5, concatenates with matchcode, locationcode from OVALS |
| S5-MATCHCODE-LOCATIONCODE | 1 | R?? | A?? | Prefix S5, concatenates with matchcode, locationcode from OVALS |
| Z1-MATCHCODE-LOCATIONCODE | 0 | <ANY other combinations> | <ANY other combinations> | Prefix Z1, concatenates with matchcode, locationcode from OVALS |

### OVALS Data Loading

This requirement describe the process loading an address from OVALS ( Batch File or PLA API) into EDF’s GEOADDR schema

#### HLD-295674-EDF-GEOADDR-from-OVALS-10 [ Data Mapping ]

All fields in an address used for data loading are defined as below, which will be referred in later section.

These fields could be from OVALS Batch File interface or API PLA.

The following table defines the data mapping among the fields in an address, Batch File, and PLA API (International and Domestic).

All the xpaths under PLA (International) are under ProcessLocationAttributesResponse/InternationalLocationAttributes

All the xpaths under PLA (Domestic) are under ProcessLocationAttributesResponse/GISLocationAttributes

|  |  |  |  |
| --- | --- | --- | --- |
| **Address Fields** | **OVAL Batch File Field** | **PLA (International)** | **PLA (Domestic)** |
| ID | ID | messageHeader/TrackingMessageHeader/messageId | messageHeader/TrackingMessageHeader/messageId |
| HOUSE\_NUM\_PREFIX | HOUSE\_NUM\_PREFIX |  |  |
| HOUSE\_NUM | HOUSE\_NUM |  | FieldedAddress/houseNumber |
| HOUSE\_NUM\_SUFFIX | HOUSE\_NUM\_SUFFIX |  |  |
| STREET\_DIR | STREET\_DIR |  | FieldedAddress/streetDirection |
| STREET\_NM | STREET\_NM |  | FieldedAddress/streetName |
| STREET\_THOROUGHFARE | STREET\_THOROUGHFARE |  | FieldedAddress/streetThoroughfare |
| STREET\_NAME\_SUFFIX | STREET\_NAME\_SUFFIX |  | FieldedAddress/streetNameSuffix |
| BASE\_ADDRESS | BASE\_ADDRESS | AddressAttributes/streetAddress | FieldedAddress/singleLineStandardizedAddress  First component delimited by comma |
| UNIT\_TYPE | UNIT\_TYPE |  | ~~FieldedAddress/unitType~~ |
| UNIT\_VALUE | UNIT\_VALUE |  | ~~FieldedAddress/unitValue~~ |
| LEVEL\_TYPE | LEVEL\_TYPE |  |  |
| LEVEL\_VALUE | LEVEL\_VALUE |  |  |
| STRUCTURE\_TYPE | STRUCTURE\_TYPE |  |  |
| STRUCTURE\_VALUE | STRUCTURE\_VALUE |  |  |
| CITY | CITY | AddressAttributes/city | FieldedAddress/city |
| STATE | STATE | <TBD> | FieldedAddress/state |
| ZIP5 | ZIP5 | AddressAttributes/postalCode | FieldedAddress/postalCode |
| ZIP4 | ZIP4 |  | FieldedAddress/postalCodePlus4 |
| GIS\_GLID | GIS\_GLID | globalLocationId | globalLocationId |
| STATUSCODE | STATUSCODE |  |  |
| STATUSDESC | STATUSDESC |  |  |
| LAT | LAT | Coordinates/latitude | LocationProperties/Coordinates/latitude |
| LONG | LONG | Coordinates/longitude | LocationProperties/Coordinates/longitude |
| MATCHCODE | MATCHCODE |  | LocationProperties/gisMatchCode |
| LOCATIONCODE | LOCATIONCODE |  | LocationProperties/gisLocationCode |
| COUNTRY | Default it as ‘USA’ | AddressAttributes/country | FieldedAddress/country, if not present, default it as ‘USA’ |

#### HLD-295674-EDF-GEOADDR-from-OVALS-20 [ Data Loading ]

Process all the fields in an address (from OVALS) into GEOADDR schema as below.

Each table has a primary key (id) and a unique index as specified in each table below. For each table, take the columns of the unique index, and compare the values with the address from OVALS. A new record should be inserted into the table if it is not in the table yet. If it is in the table already, and the table has any columns other than the primary key and columns of the unique index, update those columns only. Otherwise, no update should be done. Retrieve the primary key (id) from the table for future reference.

Note: Comparison of two NULLs should be considered as matched, even though Oracle doesn’t.

|  |  |  |  |
| --- | --- | --- | --- |
| **GEOADDR.COUNTRY** | | | |
| **Name** | **Nullable** | **Data Mapping (Address Fields)** | **Comments** |
| ID | N | - | PK, database sequence  Unique indexed on ISO\_ALPHA2\_CODE, ISO\_ALPHA3\_CODE, ISO\_NUMERIC\_CODE |
| ISO\_ALPHA2\_CODE |  | - | ISO 2-character code for country name |
| ISO\_ALPHA3\_CODE |  | - | ISO 3-character code for country name |
| ISO\_NUMERIC\_CODE |  | - | ISO numeric code for the country name |
| FIPS\_10\_CODE |  | - | <TBD> |
| TOP\_LEVEL\_DOMAIN |  | ~~-~~ | <TBD> |
| NAME |  | ‘United States’  COUNTRY | Since OVAL supports only domestic, all the addresses in the batch file will be for US  Domestic: default ‘United States’  International: COUNTRY |
| COMMENTS |  | - | NULL |

|  |  |  |  |
| --- | --- | --- | --- |
| **GEOADDR.SUBDIVISION** | | | |
| **Name** | **Nullable** | **Data Mapping (Address Fields)** | **Comments** |
| ID | N | - | PK, database sequence  Unique indexed on NAME, ID\_COUNTRY |
| ID\_COUNTRY | N | - | COUNTRY.id |
| NAME |  | STATE |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **GEOADDR.SECONDARY\_SUBDIVISION** | | | |
| **Name** | **Nullable** | **Data Mapping (Address Fields)** | **Comments** |
| ID | N | - | PK, database sequence  Unique indexed on NAME, ID\_SUBDIVISION |
| ID\_SUBDIVISION | N | - | SUBDIVISION.id |
| NAME |  | - | NULL |

|  |  |  |  |
| --- | --- | --- | --- |
| **GEOADDR.MUNICIPALITY** | | | |
| **Name** | **Nullable** | **Data Mapping (Address Fields)** | **Comments** |
| ID | N | - | PK, database sequence  Unique indexed on NAME, ID\_SECONDARY\_SUBDIVISION |
| ID\_SECONDARY\_SUBDIVISION | N | - | SECONDARY\_SUBDIVISION.id |
| NAME |  | CITY |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **GEOADDR.SUBMUNICIPALITY** | | | |
| **Name** | **Nullable** | **Data Mapping (Address Fields)** | **Comments** |
| ID | N | - | PK, database sequence  Unique indexed on NAME, ID\_MUNICIPALITY |
| ID\_MUNICIPALITY | N | - | MUNICIPALITY.id |
| NAME |  | - | NULL |

|  |  |  |  |
| --- | --- | --- | --- |
| **GEOADDR.STREET** | | | |
| **Name** | **Nullable** | **Data Mapping (Address Fields)** | **Comments** |
| ID | N | - | PK, database sequence  Unique indexed on NAME, ID\_SUBMUNICIPALITY, DIRECTION\_PREFIX, DIRECTION\_SUFFIX, TYPE\_PREFIX, TYPE\_SUFFIX |
| ID\_SUBMUNICIPALITY | N | - | SUBMUNICIPALITY.id |
| NAME |  | STREET\_NM |  |
| DIRECTION\_PREFIX |  | STREET\_DIR |  |
| DIRECTION\_SUFFIX |  | STREET\_NAME\_SUFFIX |  |
| TYPE\_PREFIX |  |  | NULL |
| TYPE\_SUFFIX |  | STREET\_THOROUGHFARE |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **GEOADDR.POSTAL\_CODE** | | | |
| **Name** | **Nullable** | **Data Mapping (Address Fields)** | **Comments** |
| ID | N | - | PK, database sequence  Unique indexed on POSTAL\_CODE, ID\_COUNTRY |
| ID\_COUNTRY | N | - | COUNTRY.id |
| POSTAL\_CODE |  | ZIP5||ZIP4 |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **GEOADDR.BUILDING** | | | |
| **Name** | **Nullable** | **Data Mapping (Address Fields)** | **Comments** |
| ID | N | - | PK, database sequence  Unique indexed on ID\_STREET, ID\_POSTAL\_CODE, BUILDING\_NUMBER, NAME, ADDRESS\_LINE |
| ID\_STREET | N | - | STREET.id |
| ID\_POSTAL\_CODE | N | - | POSTAL\_CODE.id |
| NAME |  |  | NULL |
| BUILDING\_NUMBER |  | HOUSE\_NUM |  |
| ADDRESS\_LINE |  | BASE\_ADDRESS |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **GEOADDR.BUILDING\_SUBDIVISION** | | | |
| **Name** | **Nullable** | **Data Mapping (Address Fields)** | **Comments** |
| ID | N | - | PK, database sequence  Unique indexed on ID\_BUILDING, NAME |
| ID\_BUILDING | N | - | BUILDING.id |
| NAME |  |  | NULL |

|  |  |  |  |
| --- | --- | --- | --- |
| **GEOADDR.GEOCODE\_CURRENT\_ENGINE** | | | |
| **Name** | **Nullable** | **Data Mapping (Address Fields)** | **Comments** |
| ID | N | - | PK, database sequence  Unique indexed on ID\_COUNTRY, NAME, ENGINE\_VERSION |
| ID\_COUNTRY | N | - | COUNTRY.id |
| NAME | N | - | ‘ProcessLocationAttributes’ |
| ENGINE\_VERSION |  | - | NULL |

|  |  |  |  |
| --- | --- | --- | --- |
| **GEOADDR.GEOCODE\_ENGINE** | | | |
| **Name** | **Nullable** | **Data Mapping (Address Fields)** | **Comments** |
| ID | N | - | PK, database sequence  Unique indexed on ID\_COUNTRY, NAME, ENGINE\_VERSION |
| ID\_COUNTRY | N | - | COUNTRY.id |
| NAME | N | - | ‘ProcessLocationAttributes’ |
| ENGINE\_VERSION |  | - | NULL |

|  |  |  |  |
| --- | --- | --- | --- |
| **GEOADDR.GEOCODE\_QUALITY** | | | |
| **Name** | **Nullable** | **Data Mapping (Address Fields)** | **Comments** |
| ID | N | - | PK, database sequence  Unique indexed on ACCURACY, MATCH\_TYPE, ID\_GEOCODE\_ENGINE |
| ID\_GEOCODE\_ENGINE | N | - | GEOCODE\_ENGINE.id |
| ACCURACY | N | - | *OVALS Match Code and Standardized Score* |
| MATCH\_TYPE |  | - | *OVALS Match Code and Standardized Score* |

|  |  |  |  |
| --- | --- | --- | --- |
| **GEOADDR.GEOCODE** | | | |
| **Name** | **Nullable** | **Data Mapping (Address Fields)** | **Comments** |
| ID | N | - | PK, database sequence  Unique indexed on LONGITUDE, LATITUDE, ID\_GEOCODE\_QUALITY |
| ID\_GEOCODE\_QUALITY | N | - | GEOCODE\_QUALITY.id |
| LONGITUDE | N | LONG |  |
| LATITUDE | N | LAT |  |
| ELEVATION | N |  | 0 |
| ID\_GEOCODE\_SPATIAL | N |  | 1 |

|  |  |  |  |
| --- | --- | --- | --- |
| **GEOADDR.GEOGRAPHIC\_ADDRESS** | | | |
| **Name** | **Nullable** | **Data Mapping (Address Fields)** | **Comments** |
| ID | N | - | PK, existing database sequence  Unique indexed on ID\_BUILDING, ID\_POSTAL\_CODE, ID\_STREET, ID\_MUNICIPALITY, ID\_SUBMUNICIPALITY, ID\_SUBDIVISION, ID\_SECONDARY\_SUBDIVISION, ID\_COUNTRY |
| ID\_COUNTRY | N | - | GEOCODE\_QUALITY.id |
| ID\_SUBDIVISION | N | - | SUBDIVISION.id |
| ID\_SECONDARY\_SUBDIVISION | N | - | SECONDARY\_SUBDIVISION.id |
| ID\_MUNICIPALITY | N | - | MUNICIPALITY.id |
| ID\_SUBMUNICIPALITY | N | - | SUBMUNICIPALITY.id |
| ID\_STREET | N | - | STREET.id |
| ID\_POSTAL\_CODE | N | - | POSTAL\_COD.id |
| ID\_BUILDING | N | - | BUILDING.id |
| ID\_GEOCODE | N | - | GEOCODE.id |
| ID\_GEOCODE\_QUALITY | N | - | GEOCODE\_QUALITY.id |
| ID\_GEOCODE\_ENGINE | N | - | GEOCODE\_ENGINE.id |
| LAST\_UPDATE | N | - | sysdate |
| ID\_GEOCODE\_SPATIAL | N | - | 1 |
| OVALS\_GLID | Y | GIS\_GLID | Once populated, don’t update it any more. |

|  |  |  |  |
| --- | --- | --- | --- |
| **GEOADDR.GEOGRAPHIC\_ADDRESS\_USE** | | | |
| **Name** | **Nullable** | **Data Mapping (Address Fields)** | **Comments** |
| ID\_SOURCE\_KEY | N | ID | PK, id\_source\_key of Raw Address  Unique indexed on ID\_SOURCE\_KEY, ID\_GEOGRAPHIC\_ADDRESS |
| ID\_GEOGRAPHIC\_ADDRESS | N | - | GEOGRAPHIC\_ADDRESS.id |
| LAST\_UPDATE | N | - | sysdate |
| ID\_BUILDING\_SUBDIVISION |  | - | BUILDING\_SUBDIVISION.id |

Note: For Raw Addresses’ source\_key\_digest.source\_digest value, if the value in the table is different from the newly calculated digest value, new value should be populated back into source\_key\_digest.source\_digest. Since digest value is calculated in Java code only, and if initial batch file process is implemented by ETL team, ETL may not be able to calculate the new digest value. In this case, ETL process shouldn’t populate/update/overwrite the digest values.

|  |  |  |  |
| --- | --- | --- | --- |
| **~~GEOADDR.OVALS\_GLID\_USE~~** | | | |
| **~~Name~~** | **~~Nullable~~** | **~~Data Mapping (Address Fields)~~** | **~~Comments~~** |
| ~~OVALS\_GLID~~ | ~~N~~ | ~~GIS\_GLID~~ | ~~PK~~ |
| ~~ID\_SOURCE\_KEY~~ | ~~N~~ | ~~Via EKT~~ | ~~See below~~ |
| ~~ID\_GEOGRAPHIC\_ADDRESS~~ | ~~N~~ | ~~-~~ | ~~GEOGRAPHIC\_ADDRESS.id~~ |

### GDB Data Loading

This requirement describes the process to update GDB schema and/or metadata schema for each Raw Address and corresponding address returned from OVALS (Normalized Address), which is loaded into GEOADDR schema described in *OVALS Data Loading*.

#### HLD-295674-EDF-GDB-from-GEOADDR-10 [ Address\_Notation ]

This data loading is only for the Raw Address and corresponding Normalized Address with match\_type starting with ‘S5’, ‘S6’, S7’, or ‘S8’ defined in *OVALS Match Code and Standardized Score*, and Raw Address’s EKT (id\_source\_key) exists in GDB.ADDRESS\_NOTATION.id.

Follow *GEOADDR.GEOGRAPHIC\_ADDRESS (188413a) duplication into GDB ADDRESS\_NOTATION* (in family HLD) to create the ADDRESS\_NOTATION record for the Normalized Address.

Follow *GEOADDR.GEOGRAPHIC\_ADDRESS\_USE (188413a) reference update into GDB ADDRESS\_NOTATION* (in family HLD) to unify the Raw address to the Normalized Address from above.

#### HLD-295674-EDF-GDB-from-GEOADDR-20 [ Correlated Site ]

This process is only needed when updating Raw Address’s id\_address\_notation\_unified to an OVALS Normalized Address, and orginal id\_address\_notation\_unified value is populated with a different value already. In this case, the address\_notation record referred by the original id\_address\_notation\_unfied could have some correlated sites, which should be updated to the new Normalized Address from OVALS.

Take the original id\_address\_notation\_unfied match on GDB.SITE’s id\_address\_notation, and SITE\_TYPE is ‘CORRELATED\_SITE\_REPRESENTATION’, for all the correlated sites found, update the id\_address\_notation to the new ADDRESS\_NOTATION.id for the Normalized Address from OVALS.

Follow the requirement (in family HLD) described in *Maintain SITE (correlated) to ORGANIZATION ASSOCIATION records* to ensure creating only a single Correlated SITE to ORGANIZATION ‘CONTRACTED\_BY’ ASSOCIATION record per SITE.id/ORGANIZATION.id pair.

#### HLD-295674-EDF-GDB-from-GEOADDR-20 [ Customer Site ]

This process is only needed when updating Raw Address’s id\_address\_notation\_unified to an OVALS Normalized Address, and orginal id\_address\_notation\_unified value is populated with a different value already. In this case, the address\_notation record referred by the original id\_address\_notation\_unfied could have some customer sites, which should be updated to the new Normalized Address from OVALS.

Take the original id\_address\_notation\_unfied match on GDB.SITE’s id\_address\_notation, and SITE\_TYPE is ‘CUSTOMER\_LOCATION’, for all the customer sites found, update the id\_address\_notation to the new ADDRESS\_NOTATION.id for the Normalized Address from OVALS.

Follow the requirement (in family HLD) described in <Account CustLoc> SITE correlation within account ORGANIZATION to ensure creating only a single Customer SITE to ORGANIZATION (account) ‘CONTRACTED\_BY’ ASSOCIATION record per ASSET.id/ORGANIZATION.id (account) pair.

**Alternative Designs**

*Describes alternative designs, evaluation criteria, risks and issues that were considered in choosing the design. If a Solution Approach was developed for this project and the design deviates in its approach from that recommended in the Solution Approach, the HLD should explain the difference in approach and the rationale for the change.*

None Identified.

**Assumptions/Risks**

*Identifies design assumptions, issues, constraints, and risks and mitigation strategies for each.*

*Refer to the Project Workflow Module for Risks/Issues associated with this Project.*

Assumptions

Risk/ Constraints

1. Any scope changes that impact the initial work efforts must be communicated in a timely manner or it may affect meeting project deliverables milestones.

**Solution Design**

*SRT REQPRO NOTE: In order to utilize the auto-tagging feature within SRT ReqPro and enable easier tagging of design elements, please document the design elements in the following table:*

*Note: Req. ID should be used to identify the Design Element ID.*

(Examples in red text)

|  |  |  |
| --- | --- | --- |
| **Req. ID** | **Requirement Description** | **Trace-To** |
| HLD.#.PID. Application Impacted  Example:  HLD.1. 179864.CSI-NM | **InquireCablePairTerminalDetails – New**   1. Component – M2E 2. Description of change:CSI shall create a new InquireCablePairTerminalDetails SPM to provide a list of all the terminals that a particular cable pair appears in, the terminal type, status of the pair in each of the terminals, its binding post/color in each of the terminals, and loop related characteristics of the pair.   The minimal required input is wirecenter, employee identification, cable and pair.  Priority is optional. If it isn’t input by the client it will be populated with “I” for immediate.  Refer to the Application Interface Design for the all request (input data)/response (output data) schemas for the service. The response will have information about the cable pair, and information about the cable pair in each of the terminals in which it appears.  CSI will call the INQ APP (Multiple Appearance Inquiry) transaction in LFACS. | *SR Req. ID* |
| HLD.#.PID. Application Impacted | **InquireFacilityAssignmentDetails – New**   1. Component – M2E 2. Description of change:CSI shall create a new InquireFacilityAssignementDetails SPM to provide the current data for facilities and/or loops and is a good source for obtaining pending service order data associated with a facility. This information may be used for record verification and for obtaining service order data associated with the facility. The Outside Plant Facility Assignment inquiry provides read only access to LFACS inventory data. No updating functionality is provided.   There is different ways to call this transaction to require different information;   * Facility address call provides a current representation of a facility address with its associated loop data.   What is minimal required; Wirecenter, employee identification, house number, street name   * Cable and Pair call provides a current representation of a cable pair with its’ associated loop data.   What is minimal required; Wirecenter, employee identification, cable, pair   * Circuit ID call provides a current representation of a circuit id with its' associated loop data.   What is minimal required; Wirecenter, employee identification, circuit identification.  Refer to the AID for the all request (input data)/response (output data) schemas for the service. The response will have all associated data of the facility as it currently exists in the database.  CSI will be calling LFACS INQ FASG transaction. | *SR Req. ID* |
| HLD.#.PID. Application Impacted | **InquireAffiliateAccountProfile – New**   1. Component - M2E 2. Description of change – The purpose of InquireAffiliateAccountProfile is to provide customer account information from BCAM to CSI-SPM, which will include DTV account association.    * + CSI-Interface      + The input for this new service will be BTN or BAN. CSI-SPM will call BCAM and request the data. Response from BCAM to CSI-SPM will include account data for BTN, BAN, or Telco Pending Order as well as DTV data.      + The communication between Client 🡪 CSI-SPM 🡪 BCAM will all be in the standard schema format.      + If an error is generated from BCAM then CSI-SPM will receive a failed message from the BCAM and return that failed message to the client in standard schema format. Specific details will be apart of the Application Interface design and schemas      + Data Elements/tags/field names      + Request and response elements will be detailed as part of Application Interface design and schemas   Business Rules -   1. This new service InquireAffiliateAccountProfile will call the InquireWirelineRegion service to retrieve the wireline region details. 2. This new service InquireAffiliateAccountProfile will return the Account Information and DTV account data to client. | *SR Req. ID* |
| HLD.#.PID. Application Impacted | **InquireCombinedBillingEligibility – Enhanced**   1. Component – M2E 2. Description of change – InquireCombinedBillingEligibility will verify if the ATT account is eligible for combined billing with DTV via the iCBT (integrated Combined Billing Tool) API based on the customer’s Wireline Telco 13 digit BTN (Billing Telephone Number) and MODE.  * Add a new value for “D” (DirecTV, used to set DTVIndicator) for the MODE. MODE value of “D” is used to determine DTV Indicator. Will pass the DTV Indicator to iCBT, and iCBT will use the indicator in the determination if the account is eligible for combined billing. * WirelineCombinedBilling **-** This structure allows the client to indicate the AT&T Billing Account Number + Customer Code (13 digits). * Mode -A mode for combined bill to denote type of combined bill requests.   Wireless/Wireline, or Both or Wireline only or DTV  For DirecTV the MODE of “D” is required.  If the MODE is equal to a value of “D”, set the DTV Indicator to true.  Otherwise, set the DTV Indicator to false.  Request and response data elements will be detailed as part of Application Interface Design and Schemas.  Business Rules   1. The existing service uses the iCBT API 2. This existing service InquireCombinedBillingEligibliity will return if combined billing eligible to the client |  |
| HLD.#.PID. Application Impacted | **InquireUnifiedCustomerProfileByServiceLocation - Enhanced**     1. Component - M2E 2. Description of change:CSI shall enhance an existing service, *InquireUnifiedCustomerProfileByServiceLocation*, is called by client to retrieve the customer and service profiles based upon a wireless subscriber number or a wireline telephone number or a wireline account number and now for this project, based upon a U-verse BAN as well. This interface will continue to retrieve customer and service profiles from CCR via VRI. 3. Response elements will be detailed as part of Application Interface design and schemas. | *SR Req. ID* |
| HLD.#.PID. Application Impacted  Example:  HLD.1. 179864. CSFOBPM | **InquireUnifiedCustomerServiceProfile - Enhanced**   1. Component - BPM 2. Description of change:CSI shall enhance an existing service, *InquireUnifiedCustomerServiceProfile*, that returns a Customer Profile and Customer Service Profile for a subscriber number or a wireless telephone number or a wireline account number device and now for this project, based upon a U-verse BAN as well. If response has U-verse account only, we will send DTV eligibility as not eligible. 3. Request and response elements will be detailed as part of Application Interface design and schemas. | *SR Req. ID* |
| HLD.#.PID. Application Impacted. AdapterName  Example:  HLD.1. 179864.CSI-Adapter.ACIS | **Adapter Requirements**  The following is an example of how we would capture Adapter requirements.  **ACIS adapter – Enhanced or New**   1. Component – Adapter 2. **Method(s) – Adapter method Name fetchCSRForMidwest The adapter method name is the name of the API, interface, etc that the adapter will use to obtain data from the source system. In the case of multiple handlers in a single adapter, each handler’s information impacted would be listed in the corresponding section.**   This information can be found on existing adapters at the following link  <https://operations.web.att.com/sites/CSD_DC/CSD%20Project%20Tracking%20Tool/Lists/CSD%20Adapters/Default%20View.aspx>  **C.** Description of change:  Description of the change should be inserted here.  The fetchCSRForMidwest method will be enhanced to include credit information.   1. **Connection Details**  * We would work with development in identifying this information. Examples would be EMBUS, MQSERIES or WebService.  1. **Downstream Interface(s)**   XYTACP01  **Processing Rules:**  Any unique processing that the adapter will perform will need to be documented especially where a new adapter is being created which could include a new source system. Again requirements will be working with development for this information. | *SR Req. ID* |
| HLD.#.PID. Application Impacted  Example:  HLD.1. 179864.CSI | **Version Proxy**   * 1. Standard Version Proxy map will be provided for the existing interfaces. | *SR Req. ID* |
| HLD.#.PID. Application Impacted  Example:  HLD.1. 179864. PartnerProfile | **Partner Profile**  *If there are any service policies or partner profile changes necessary to fulfill the business logic for this project, they must be documented in this section. The need to update the Partner Profile, in most cases, comes as a result of needing to define a new Service Policy that would allow dynamic rules to be executed differently for different partners/clients.*  The following is an example of a partner profile change:   1. Component – SPM Product & Offer Management – InquireWirelineRegion 2. Description of change: A new service policy will be created. The service policy will be called ‘IWR-SKIP-BCAM-ZIPCODE-VALIDATION’. This service policy will control the call to skip the AAV/BCAM call when zip code validation is requested and a zip code is not returned from CCR. With this flag set to true, CSI will bypass call to AAV and return the response from CSI. This will be considered as a zip code match. | *SR Req. ID* |
| HLD.#.PID. Application Impacted. AdapterName | **Mapping Upgrades**  If there are request or response schema changes to CSI services that impact the CDM file (for common/shared data elements), the developer assigned to the project needs to run an impact analyzer script (a.k.a. Schema Analyzer tool) to determine the impacted services sharing the common data elements.  This step should happen on or before the CSI HLD internal review with the Design Assurance team.  Any impacted services found, should be added in the HLD as being impacted for mapping upgrades. The owners of the identified impacted services should be informed and engaged by the PM so they can upgrade their data maps.  **InquireAffiliateAccountProfile – Enhanced**   1. Component - SPM 2. Description of change – The **InquireAffiliateAccountProfile** schema request and/or response is impacted by data map upgrades due to shared structure/element definitions in the Cingular Data Model (CDM) file. The common data structure/element being updated is the **StructureName or DataElementFieldName**. The change is being made due to:  * The **StructureName** or **DataElementFieldName** is being changed from Required to Optional (or vice versa) or * The **DataElementFieldName** size is being increased/decreased or * The **DataElementFieldName**’s data type is being changed from Numeric to Alphanumeric , String to Numeric, etc * An optional/required structure/Data element being introduced impacting the private schema invoked by **InquireAffiliateAccountProfile** which needs to be absorbed by the service or * Any other reason for the data map change reason | *SR Req. ID* |

*The* [SRT\_RM\_ReqPro\_Auto\_Tagging](https://cps.web.att.com/CPSWorkplace/getContent?id=current&vsId=%7B67070CE6-CDED-4932-8F1D-F6073BA40DF8%7D&objectStoreName=IT-Architecture.__.Planning.__.and.__.Integration&objectType=document&guestId=servicesguest) *Reference Document provides information on how to use auto-tagging within SRT ReqPro.*

*If the table is not used to capture requirements, please remove it from document.*

**System Agreements <CSI>**

**Note:** Although an attempt was made to capture all changes resulting from this design, unanticipated design changes may occur after this document has been turned over that may affect the impact on systems or applications.

This lists clients that have been identified as having impact.

*Instructions:  Only list the impacted clients within this project that are requesting these enhancements.*

|  |  |  |
| --- | --- | --- |
| **Client** | **Client of which System** | **Interface Name** |
| eElections IVR | CS FOBPM | ProcessSelfServiceCPNIElection |
| OPSS | CS FOBPM | ProcessSelfServiceCPNIElection |
|  |  |  |
| OPUS | CSI-Customer Care | InquireCPNIDetails |
| OPUS-Lite | CSI-Customer Care | InquireCPNIDetails |
| PDC | CSI-Customer Care | InquireCPNIDetails |
| PDC-Lite | CSI-Customer Care | InquireCPNIDetails |
| System Xi | CSI-Customer Care | InquireCPNIDetails |

**Traceability Matrix**

*Insert link or lists each requirement and references how it is addressed by the design. This will provide visibility on how well the design meets requirements. In any cases where some requirements are not addressed in the design, they should be called out in this section to enable the audience to better understand the design's level of traceability.*

*NOTE: The design element identifiers must be transferred over to the Requirements Traceability Matrix in order to complete the High Level Design.*

|  |  |
| --- | --- |
| **Requirement ID** | **Design Element Identifier** |
| *FR-1.1* | *HLD - 1* |

**Pre-Production Disaster Recovery Planning**

*As a reminder, for each new or modified application refer to* [ITSC Integrated Policies and Standards](http://itup.it.att.com/ittools/itmap/resources/cfm/itup/1_ProcessElement.cfm?xPEName=ITSC%20Integrated%20Policies%20and%20Standards). *Review disaster recovery requirements for DR Plans, related documentation, and exercise requirements and ensure all DR requirements are met and reflected in MOTS before an application is entered into production. Refer to* [IT Service Continuity Capability Guidance](http://itsc.web.att.com/itup.htm) *for more detailed instructions. If pre-production Disaster Recovery is not needed for this project, place an NA in this section with a business reason (i.e., OS version upgrade or Database version upgrade).*

*If not applicable for this project:*

N/A – This project does not impact the current Application Impact Analysis or Disaster Recovery Plan for the impacted Common Services (CSI) applications. A Pre-Production Disaster Recovery Exercise is not required for this project.

**Other Plans and References**

*Provide links to other plans at your discretion.*

|  |
| --- |
|  |
| **Reference** | | **Location** |
| Project Plan | | Place project document link here |
| Business Requirements Specifications | | Place project document link here |
| System Requirements | | Place project document link here |
| Application Interface Design | | Place project document link here |
| Uses Case document | | Place document link here |
| Requirements Traceability Guidelines | | <http://itup.it.att.com/ittools/itmap/itup/Method/mth_Requirements%20Traceability%20Guidelines.doc?CFID=190824&CFTOKEN=b5469ca5b19769e3-1AF3D333-D526-555C-731247FCB455064E> |
| Requirements Traceability Matrix | | Place project document link here |
| Any other documents that you see useful... (Example backends HLD and AID, Solution Approach, etc…) | | Place projects document link here |

**Acceptance & Approvals**

**Overview**

*Use this section to capture approvals in the event that electronic approvals via the PRISM Project Workflow Module will not be used.*

The Approvers of this work product agree that this document is acceptable and complete to the best of their knowledge and will be used by the project team as an official deliverable for the project. It is further agreed that this document can now be baselined and any changes to these sections from this point forward must follow the Managing Change in the IT UP.

Embed evidence of approval in the review table below, or use the PRISM Approval Functionality in the Project Workflow Module Workflow Template View.

**Approvers**

**PLEASE NOTE: This does not replace the Review Record. This is simply to gather the email approvals. The original email must be included and include the exact document name, version reviewed and being approved.**

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTUID and Name** | **Role** | **Group/Application** | **Version Approved, Approval Date and Approval Evidence** |
| *attuid – name* | Solution Architect | IT Architecture and Engineering | **Embed the email approval and make sure the exact doc name, version reviewed and approved.** |
| *attuid – name* | Architect | ACSI |  |
| *attuid – name*  *(for each impacted ACSI application)* | Developer | *Group/Application* |  |
| *attuid – name*  *(for Data Layer)* | Development Team Lead | *Group/Application* |  |
| *attuid – name*  *(for each impacted non-ACSI application)* | Developer | *Group/Application* |  |
| *Send to the TA\_TestDesign distribution list.*  *Remove if no CST-Walton application impacts.* | Test Architect | CST - Walton  (Test team for CSI – Customer Care, CSI - Order & Subscription Management, CSI – Product & Offer Management, CAM, ATLAS, SWOT BE, VRI, DITREX (GIS gateway), EAI, Jackcache and Data Layer) |  |
| *Send to the CST-Agarwal Test Architect assigned to the project.*  *Remove if no CST-Agarwal application impacts.* | Test Architect | CST - Agarwal  (Test team for CSI-Network Management, CSI-Workforce Management, CSI-Credit & Validation, CSI-Trouble Management, EBTA, CS FOBPM, CS BOBPM and Remedy Fallout Manager) |  |
| RF9578 - Bob Farmer  *Remove if no CSI or FOBPM impacts.* | Production Support Manager | Production Support | *Bob is only a reviewer (not approver).* |
| BK5747 - Brian Knop  *Remove if no CSI or FOBPM impacts.* | Production Support Manager | Production Support | *Brian is only a reviewer (not approver).* |

**Appendix A: JMS Requirements**

**Public Queues:**

| Unique ID | 1 |
| --- | --- |
| Queue Name | pub.m2e.inquiretelcoloopdetails.request |
| Status (CRUD) | Create |
| Message Size (KB) | 50Kb\* |
| Message Rate (x per y) | 1 Transaction Per Second (TPS)  22,500/day (7,500 x 3 contracts) |
| Message Expiration (sec) | 120 Seconds |
| Requirement Text | Public request queue |

| Unique ID | 2 |
| --- | --- |
| Queue Name | pub.m2e.inquiretelcoloopdetails.response |
| Status (CRUD) | Create |
| Message Size (KB) | 50Kb\* |
| Message Rate (x per y) | 1 Transaction Per Second (TPS)  22,500/day (7,500 x 3 contracts) |
| Message Expiration (sec) | 120 seconds |
| Requirement Text | Public response queue |

*\*The Message Size can be calculated as follows:*

1. *Open the schema in Altova XMLSpy.*
2. *On the menu, select DTD/Schema.*
3. *In the drop-down list, select Generate Sample XML File…*
4. *Select your desired options and select OK.*
5. *Right-click on the tab with the generated XML and select Save As. Save the generated XML file to your C: drive.*
6. *On your C: drive, look at the file properties of the generated XML file. The file size is documented in the properties.*