Document Attributes

| Attribute | Value |
| --- | --- |
| Application ID /  Application Name | MOTS ID:  14724 / GCP (Global Computing Platform) |
| Owner | GCP: Ming Ho |
| Owner Contact Information | mh6892, 732-420-7673 |
| Other Attribute | Project ID:  297100a |

## 

Revision History

The following table lists the revision history of this document:

| Author | Date | Version # | Revision Description |
| --- | --- | --- | --- |
| Ming Ho | 3/25/2016 | 0.01 | <289116.141314> Initial draft |
| Ming Ho | 4/5/2016 | 0.02 | Updates during the call with EM team:  Add enhancement to query getCktAssetDetailByCktId |
| Ming Ho | 12/21/2017 | 0.01 | Updates for PID 297100a to support AOTS TM: Adding Drivenet indicator to  API: InquireAssetSummary  Legacy transaction: getAssetDetails |
|  |  |  |  |
| Ming Ho | 2/22/2018 | 0.01 | <301033> US374866  Re-engineer EDF logic to use the A&AI data sourced from DMaaP instead of the A&AI batch feed.  Support Vyatta uCPE. |

Table of Contents

[Overview 3](#_Toc503259099)

[Problem Statement 3](#_Toc503259100)

[Project Description: 3](#_Toc503259101)

[Design Decisions 3](#_Toc503259102)

[getAssetDetails – 297100a 3](#_Toc503259103)

[Common logic – getEISContractType <284465h> 57](#_Toc503259104)

[Alternative Designs 58](#_Toc503259105)

[Assumptions/Risks 58](#_Toc503259106)

[Traceability Matrix 58](#_Toc503259107)

[Pre-Production Disaster Recovery Planning 58](#_Toc503259108)

[Other Plans and References 58](#_Toc503259109)

[Acceptance & Approvals 58](#_Toc503259110)

## Overview

The High Level Design (HLD) describes how an application will implement the architectural concept and meet the requirements allocated to that application. The High Level Design describes the functionality the application will provide, the overall design for providing that functionality and meeting the nonfunctional requirements, and the rationale for choosing that design. The High Level Design also describes how the application will cooperate and interface with other applications to provide an integrated solution that achieves the architectural concept.

The High Level Design covers these topics:

* Problem Statement
* Design Decisions
* Alternative Designs
* Assumptions/Risks
* Traceability Matrix
* Other Plans/References

## Problem Statement

### Project Description:

See SR document.

## Design Decisions

getAssetDetails – 297100a

|  |  |
| --- | --- |
| **Query Name** | getAssetDetails  As one of the transactions supported in  **API InquireAssetSummary** |
| **PIDs, CRs** | Asset Detail Query (P7392) & (PA307) & PA308, PA846, (PA987/P9370/PB353),PB293(R15), PB655(R16), PE915 (R17), PF360 (R18), PF413 (R19), 195547, 207810, 222248, 185108e, 218950, 240195-CR21, R26/231615, 231615, 232946 CR#12, 236171, 240520,240520a, 238833, 232213, 235050, 249601, 237570, 264700a, 15642ah, 255187– doc-only, Oct’2012), 237057c, 236941, 265609, 274399, 272078i, 285914, 282720, 285940.147558, 289116.141314, 284465h, 297100a |
| **Service Type** | Managed |
| **Data Source** | SiDBOR, ICORE, INSTAR, GPS, AAI, CSINX (EIS\_MCN\_LIST, EIS\_MCN\_INVTRY) |
| **WSDL File** |  |
| **Client App** | AOTS TM, EM |

getAssetDetails returns the details about an asset.

**Query Change Summary:**

|  |  |
| --- | --- |
| **Project ID / Ticket** | **Change Summary / Notes** |
| 207810 | * Add ATI Indicator to the main output structure – ICORE.Site.ati. * This field is populated only when the input is customer router (Name or IP) * This can only be retrieved when the input asset is associated to PVC. |
| 231615 | Add CustomerAssetAliasName to the main output structure |
| 236171 | GetAssetDetails   * Adding new field, reference IP Address, to the main output structure. * The data source is INSTAR.ipv4\_unnumbered.reference\_ip\_addr; the valid values are IPv4 Address * This can only be retrieved when the input asset is associated to PVC. |
| 240520 | * Support new assettag\_type – ‘ATS\_ROOM’, which is valid for FunctionalArea = ‘SD\_HOSTING’ only * Add new field to the output structure – ATS\_Room * Supporting CustomerAssetAlias for this new search type |
| 240520a/b | **<HLD-240520-GCP/AOTS-TM 110> map to <240520-DM-GCP-FR-100>**  Adding logic to support ATS Exchange equipment data retrieval.  Project 240520 (that was deployed during 2012-JUN) did not cover EXCHANGE equipment in a sufficient and correct way between AOTS and GCP, so when AOTS sends a data retrieval request to GCP with an EXCHANGE equipment id, GCP will not be able to return data to AOTS.  This project contains the addition of EXCHANGE equipment to the full ticketing life cycle  for ATS equipment between SAK Portal (as the original ATS data source), the GCP ATS Replica  (as the ATS data source for all down-stream systems), VizGems, AOTS and GCP.  In order to resolve this functionality “gap” this is the proposed solution approach:  When a request from AOTS for equipment data is received by GCP, GCP needs to determine if the equipment is EXCHANGE or “standard” equipment, by checking its “equipment type” value in the ATS room equipment table.  If the “equipment type” value starts with “EXCH-“, then it is EXCHANGE equipment, then the data to be returned to AOTS needs to pulled specifically for that piece of EXCHANGE equipment.  Assumptions:   * The hostname is a unique identifier for Exchange Equipment. SAK Portal will propagate hostname (or other unique identifier) into the Equipment ID field for exchange equipment. * The equipment type for Exchange equipment will always begin with “EXCH-“ (therefore GCP can determine exchange vs. standard equipment type). * <240520a-upd-1> For EXCHANGE equipment, we must not use the "ATS.ats\_room\_equipment\_active\_vw" view, but the "ATS.ats\_room\_equipment\_vw" view. This is because for exchange equipment, there is no order number data, and especially no one with a leading "INVENTORY" to indicate "active" equipment, |
| 255187 | **<HLD-255187-GCP/AOTS-TM 110> map to <255187-DM-GCP-FR-100>**  Adding logic to support ATS Exchange equipment data retrieval.  Project 240520 (that was deployed during 2012-JUN) did not cover EXCHANGE equipment in a sufficient and correct way between AOTS and GCP, so when AOTS sends a data retrieval request to GCP with an EXCHANGE equipment id, GCP will not be able to return data to AOTS.  This project contains the addition of EXCHANGE equipment to the full ticketing life cycle  for ATS equipment between SAK Portal (as the original ATS data source), the GCP ATS Replica  (as the ATS data source for all down-stream systems), VizGems, AOTS and GCP.  In order to resolve this functionality “gap” this is the proposed solution approach:  When a request from AOTS for equipment data is received by GCP, GCP needs to determine if the equipment is EXCHANGE or “standard” equipment, by checking its “equipment type” value in the ATS room equipment table.  If the “equipment type” value starts with “EXCH-“, then it is EXCHANGE equipment, then the data to be returned to AOTS needs to pulled specifically for that piece of EXCHANGE equipment.  Assumptions:   * The hostname is a unique identifier for Exchange Equipment. SAK Portal will propagate hostname (or other unique identifier) into the Equipment ID field for exchange equipment. * The equipment type for Exchange equipment will always begin with “EXCH-“ (therefore GCP can determine exchange vs. standard equipment type). * <240520a-upd-1> For EXCHANGE equipment, we must not use the "ATS.ats\_room\_equipment\_active\_vw" view, but the "ATS.ats\_room\_equipment\_vw" view. This is because for exchange equipment, there is no order number data, and especially no one with a leading "INVENTORY" to indicate "active" equipment, |
| 235050 | **<HLD-235050-GCP/AOTS-TM 110>** map to < FR-235050-GCP-AOTS-100/110>  GCP-AOTS will provide AOTS with the Cascaded Router indicator. Impacted query: GetAssetDetail.  Cascaded Router indicators will be inventoried in SIDBOR for use in SIEI webservice operations. Those are   * Cascaded Router * Cascaded Access   Ralph:  I would introduced four new fields in SIDBOR  1. EQUIPMENT.cascaded\_fg  2. CIRUIT.cascaded\_fg  3. CIRCUIT.connection\_type  4. PVC.cascaded\_fg  All (Y/N/null) CHAR(1) Nullable.  I guess you are just interested in EQUIPMENT.cascaded\_fg. |
| 249601 | <249601-AID-GCP/AOTS-TM #1> map to 249601:GCP-FR-1010  FR190 Enhance the existing **getAssetDetails web service** between AOTS and GCP to return the third party pseudo circuit ID stored in the GCP Custom Deals database schema, (instead of dummy circuit that is returned today). |
| 237570 | Adding logic to provide 2 additional parameters in the output response:  maintainerNm from SIDBOR.ASSET.maintainer\_nm  cpeMaintenanceOption from SIDBOR.EQUIPMENT.cpe\_maintenance\_option |
| 265609  Oct’2014 | <HLD-265609-GCP/AOTS-TM 140> map to <FR-265609-GCP/AOTS-TM-100/110/120/130>  Add the following fields to the output structure – populated ony for input associated to PVCs (e.g. Asset – router, or access circuit):  For main structure:   * Multi Hop Loopback Address * Internet VLAN Indicator * Internet CER IPv4/v6 Address   For new substructure:   * iVLAN LAN IP Address Customer Provided Indicator * LAN IP address with Subnet |
| 264700a  Oct’2014 | In support of CALNET3, added new input functionalArea value = ‘CALNET3’ and the following new output attributes: calnetIndicator, agencyId, agencyName |
| 15642ah  Oct’2014 | Added fields and data retrieval logic for Managed Care  Including fixing defect R3S1400007205, |
| 185108e | support CPE Resale |
| 274399  Feb'2015 | <HLD-274399-GCP/AOTS-TM 120> map to <FR-274399-GCP/AOTS-TM-100>  GCP shall provide AOTS with the ipv4/v6 Customer Ordered MTU, and Provider MTU.  <274399-upd2>  GCP Query getAssetDetails:   * Issue – PVC data (including c/p MTUs) is not returned to AOTS for Managed scenario. * Reason: The logic retrieving C/P MTU should take all PVC IDs retrieved from both SiDBOR and NC3, but it currently is executed on PVCs from SiDBOR only. * Fix: Move this retrieving C/P MTU logic to later steps, specify the PVC IDs coming to this step include data retrieved from both SiDBOR and NC3. |
| 272078i  Iteration 19 | <HLD-272078i-IT19-GCP/AOTS-TM 030> map to GCP User Story US486488   * Add VMS indicator to the output at main level * Change all the fields in the main output structure to Optional, and the substructures in the main structure, optional. * If VMS Indicator is Y, then only this field is populated and returned to AOTS, all the other fields are blank. * If VMS Indicator is N, then BAU logic for this query is used.   **Note:**  ~~This VMS indicator is only derived for input AssetTagType = Name (router), this will be the device PTNii.~~  ~~For other input types, the VMS indicator is default to ‘N’~~  <272078i-ite19-upd2>   * As the first step in the query, the VMS indicator is derived for input assetTag, assuming it’s Router PTNii. * This step is performed regardless the input assettagType – removing checking on the assettagType |
| 285914  Dec.2015 | <HLD-285914-GCP/AOTS-TM 010> map to GCP User Story US535993   * Add Cascaded uCPE Indicator at router level * Allower AOTS to call NODDeviceDetail API using the input router name (ptnii). |
| 282720 Feb 2016 | US542882 – GCP SA (AOTS-TM) Add two new fields from ICORE to getAssetDetails  Add two new fields, performance group and performance characteristics, from ICORE sit extension and performance data tables respectly |
| 285940.147558  Oct’2016 | US729996:  Enhance getAssetDetails query to include SAK portal data elements.  Populating the LocationID field with the site\_ID from SAK Portal/ATS Data. |
| 289116.141314  July’2016 | <HLD-289116.141314-GCP-EM 010>Map to US640793- CR141314 - US GCP-EM supporting blocking transport ticketing  For GMIS or other Managed Services such as AVPN-M, MRS, MLAN : getAssetDetails (input : functional area, assetTag, assetTagType), Add   * uCPE indicator – existing as isCascadedUCpe * managed by – added ucpeManagedBy   Note:   * These fields are for input is router name. * If uCPE indicator is Y, then the calling application can call INODDeviceDetails with the same router used to call this webservice. * So a UCPE hostname is not needed.   289116-upd01:  Update HLD for getAssetDetails to address test issue of “No managedBy returned when isVms=Y” |
| 284465h  Feb’2017 | Support AOTS TM with new field contract type, for EIS (Enterprise Infrastructure Solutions) |
| 297100a  April 2018 | <297100a> Added Drivenet/Arista indicator for input Router name (PTNii), to allow AOTS TM users to trigger the INOD APIs with the input Drivenet, Arista devices (PTNii)   * This field is applicable for input Router Names. * Set to True for input Drivenet/Arista device PTNii; Set to False for other device PTNii; Set to Null for non-PTNii as input. (ARISTA – ARISTA Networks, vendor); * For ARISTA devices, this API will not return device inventory. So when this field is set to True, the API will stop further processing, and return this flag to client application. This will allow AOTS TM users to trigger the INOD APIs with the input Drivenet, Arista devices (PTNii) to get its inventory; |
| 301033  June’2018 | US374866  Re-engineer EDF logic to use the A&AI data sourced from DMaaP instead of the A&AI batch feed.  Support Vyatta uCPE. |

The signature of the method is as follows:

com.att.aots.is.AssetDetails getAssetDetails(CommonHeader header, String functionalArea, String assetTag, String assetTagType, Sequence of RestrictionType) throws com.att.aots.is.AssetNotFoundException

The input parameters are defined as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| **DBOR**  **WSDL NAME** | **M/O** | **DBOR Field Name/Description** | AOTS Field Mapping  **GUI Label/** Size /DB Name |
| functionalArea | M | Functional\_area  GPS.Sw\_customer.atgemsscid  Equivalently (fed from GPS)  SIDBOR.asset. FUNCTIONAL\_AREA | **Functional Area**  varchar2(14)  CLIENT\_ID  ‘CPESVCS’ (185108e, AOTS TM send new value)  For 15642t, AOTS TM shall send a value in <*MCFAS list*>  ‘CALNET3’ (264700a) |
| assetTag | M | Asset\_tag  If assetTagType=’NAME’ then  SIDBOR.asset\_details.asset\_nm  Elseif assetTagType=’ID’ then  SIDBOR. asset\_details.asset\_id | **Asset ID**  varchar2(55)  ASSET\_ID  <240520> Note:   * The ATS Room/equipment inventory will be from ATS replica in GCP. * When Asset Type=ATS Room Name, the Asset ID will be a concatenation of the gemsOrgId and ATS Room\_Name * Searching from the Asset ID field must match exactly on the string entered * Room\_name will be an asset type for ATS and the room-level ID will be used as the asset ID when opening trouble tickets * Searching on ATS Equipment input (for VizGems tickets on ATS equipment) is supported * Room Level ID is Room ID = GEMS Org ID + Room Name   <249601> Note:   * When the assetTag\_type = ‘NAME, the value in this assetTag field can also be a 3rd Party Pseudo Circuit ID. * This Pseudo Circuit input applies mainly to FA=’GLOBAL AM’ and ,SHELL\_ITI’ but that could change in the future. So the search logic should apply to other Functional Areas, excluding SD\_HOSTING, ATT\_IT, .   Sample values for 3rd Party Pseudo Circuit, where ‘3’ means third party circuit; ‘G’ means GPS created pseudo circuit  DV3G116567  DH3G111947  **15642t Note:**  If Asset Tag Type is “NAME”, then:  (label=’Asset ID’ in AOTS GUI)  Varchar2(55)  ASSET\_ID  If Asset Tag Type is “ID”, then:  (hidden in AOTS GUI)  Number(15))  INTERNAL\_ASSET\_ID  --the tagtype of ‘Name’ is linked to the alphanumeric asset name identifier used in both SIDBOR and GPS DB;  --the tagtype of ‘ID’ is linked to the sequence /primary key of SIDBOR.asset table (only used in SIDBOR) |
| assetTagType | O | Asset\_tag\_type  185108e Valid values include:  NAME | Derived  <PF360>  Values used for PF360: ID, NAME, HOST\_IP (New)  </PF360  <240520>  Added value – ‘ATS\_ROOM’  Valid for FunctionalArea = ‘SD\_HOSTING’ only  </240520> |
| enableAssetClassExclusionFilters | O | Boolean | 231615:  ‘true’ enables the exclusion of assets having an assetClass defined in the EM maintained assetClass exclusion filter list (database table).  Defaults to not filter assets by assetClass.  Used by EM. |
| restrictionType | O | RestrictionType | *N/A*  Varchar2 (30)  *RestrictionType* |
| customerNumber | O |  | Note: New for 185108e  ABSCON Customer Number  varchar2(15)  ABSCON\_CUSTOMER\_NUMBER |

The method **getDetails** uses the method **findAssetID** to locate the DBOR asset\_id from the ASSET table based on the assetTag provided. The table below specifies how assetTag is used for identifying the assetID. The method findAssetID is utilized by other methods in the interface as required when assetTag and assetTagType is used. *assetTagType* of “ID” requires no further search, making it the most efficient call and clients are encouraged to use this type of call if they possess the asset\_id. The valid assetTagTypes are described below:

|  |  |
| --- | --- |
| **AssetTagType Attribute** | **NOTES** |
| ID | the tag is DBOR asset\_id, requiring no further lookup. Using this tagType would thus be the most efficient for retrieval of details. Apply Access Restriction rule by linking asset & client\_org. (P9654) |
| ALARM | The tag is Alarm (used primarily by auto-ticketing), thus a lookup is required in XLT table. If it is not found in XLT table, the Asset table is searched for a matching tag in the asset\_nm attribute and the corresponding asset\_id is used. Apply Access Restriction rule by linking asset & client\_org. (P9654) |
| NAME | the Asset table is searched for a matching tag in the asset\_nm attribute and the corresponding asset\_id is used. Apply Access Restriction rule by linking asset & client\_org. (P9654)  <249601> Note:  Under this assetTag\_type, the input value in assetTag field can be a 3rd Party Pseudo Circuit ID |
| INFO-RNID | Will be treated as assetID (“ID” above) |
| KIA | Will perform a keyItemsAffected search (same as findAssetByKeyItemAffected) and the corresponding assetID is used. If multiple assetIDs are found, only the first one found is used. Apply Access Restriction rule by linking asset & client\_org. (P9654) |
| APPL\_NAME | SIEI would search based on application name and use the first assetID found for the functionalArea. Apply Access Restriction rule by linking asset & client\_org. (P9654) |
| IP | Look into ip\_netwk\_addr field and identify the assetID to use. Apply Access Restriction rule by linking asset & client\_org. (P9654) |
| <PF360> HOST\_IP</PF360> | Look into the LEGACY\_ASSET.ip\_address or LEGACY\_ASSET.hostname for a matching assetTag value.  </PF360> |
| ATS\_ROOM | <240520>  The tag is an ATS room, and will be matched to the ATS schema. This type is valid only for Functional Area = ‘SD\_HOSTING’ |

**OUTPUT: AssetDetails New Field Addded to the OUTPUT Structure**

| **DBOR WSDL NAME** | | **DBOR Field Size** | AOTS Field Mapping  **GUI Label/** Size /DB Name |
| --- | --- | --- | --- |
| assetManagementType | Varchar2 (50) | Asset Management Type  Varchar2 (50)  Asset\_Management\_Type | Added for PA307 |
| isVms | Derived | Varchar2(1)  R | VMA Indicator  <272078i-ITE19> Valid values  Y, N (default) |
| isCascadedUCpe | Derived | Varchar2(1)  R | Cascaded Router Indicator -  <285914> Valid values  Y – The input router cascades to a uCPE device which has NO vRouter.  N - default  Note: A ‘Y’ value will direct AOTS TM to call GCP API for uCPE device details, using the input router name (ptnii). |
| ucpeManagedBy | ~~AAI\_device.managed\_by~~  AAI\_pserver.management\_option <301033> | Varchar2(50)  O | <289116.141314-US640793> Support EM:  To show the UCPE is managed by ATT or customer. |
| hasCustProcedureForLocation | HasCustProcedureForLocation - Indicates if Customer Procedures exist for theLocationID associated with the asset | Boolean | Added for PA308  Valid Values will be ‘Y’ or ‘N’ |
| hasCustProcedureForAssetNm | HasCustProcedureForAssetNm - Indicates if Customer Procedures exist for the Asset | Boolean | Added for PA308  Valid Values will be ‘Y’ or ‘N’ |
| performanceGroup | From ICORE site\_extension table | Varch2(40)  Optional | <282720> |
| performanceCharacteristics | From ICORE performance\_data table | Varch2(7500)  Optional | <282720> |
| CustomerProcedureFg | Customerprocedurefg – indicates if Cust Procedures exist for the OrgCd | Boolean | Existing Field  Valid Values are ‘Y’ or ‘N’, |
| assetName | String assetName | ASSET.asset\_nm/  LEGACY\_ASSET.asset\_nm |  |
| alarmedAssetName | String alarmedAssetName | ASSET.asset\_nm/ LEGACY\_ASSET.asset\_nm |  |
| assetType | Int assetType | ASSET.asset\_type\_cd/ LEGACY\_ASSET.asset\_type\_cd | 25641ah: Derived for Managed Care assets only (otherwise code is BAU) |
| assetProtocol | String assetProtocol | Varchar2 (40) | Asset Protocol  Varchar2 (40)   * ETHER * IPETHER * TRETHER   <240520> for ATS asset:   * ATS\_room\_equipment\_vw.sb\_protocol - for equipment records; * Null - for room records   **Note:** For ATS asset, this indicates the protocol for a Standards Based piece of equipment.  ~~Valid values are:~~  ~~1 = H.323~~  ~~2 = SIP~~ |
| assetClass | String assetClass  M-MC | ASSET.asset\_class\_cd,  LEGACY\_ASSET.asset\_class\_cd  Varchar2(50) | 195547 - WIFISVCS  Type  Type  <240520> for ATS asset, the values are:   * ‘ATS\_ROOM’ – for Room record * ATS\_room\_equipment\_vw.equipment\_type – for equipment record |
| isCascadedRouter | String | Varchar2(1)  O  SiDBOR.Equipment.Cascaded\_fg | <235050> Note:   * Added new field. This is for informational purpose only, will not be saved in the ticket (ARS). * This field is populated when Equipment table is access for the input Asset * This field is NOT applicable for Functional area of ATT\_IT and SD\_HOSTING   Valid values: Y, N, or null |
| srcAssetId | String Src\_asset\_id | Varchar2(15)  Legacy\_asset.src\_asset\_id | Added for PB353  Unique ID (CM)  Unique AssetID (TM) |
| assetObjectId | Long | Asset.id/Number(20) | <232213>gdbAssetObjectID  Number(20)  hidden  assetObjectId - GCP Object ID is a unique identifiers associated to GDB objects (Contact, Assets, Etc.). |
| service | String service | Varchar2(100)  ASSET.Service  retrieve from PA031 Mapping table – see PA031 rqmt below | Asset Service  Varchar2(100)  Reported Service Line  retrieve from PA031 Mapping table – see PA031 rqmt below  <240520> for ATS asset:  ‘ATS’ |
| prntOrgCd | String prntOrgCd | Varchar2(20)  Client\_org. org\_cd\_prnt | parent\_org\_group  <PB655> |
| gprntOrgCd | String gprntOrgCd | Varchar2(20)  Client\_org. org\_cd\_gprnt | Grandparent\_org\_group |
| atiIndicator | String atiIndicator | varchar2(6)  O  ICORE.Site.ati | ATI Indicator  <207810>  This field will be provided only if the input assetTagType is NAME, ID, HOST\_IP |
| assetPriority | Int assetPriority | ASSET.prty\_cd | Asset Priority  Number (15)  TT. Asset\_Priority |
| circuitCarrierID | String circuitCarrierID | ASSET.ckt\_carrier\_id |  |
| circuitMaxEndptCt | Int circuitMaxEndptCt | ASSET.ckt\_max\_endpt\_ct  ASSET.co\_nm  <240520> for ATS asset:  ARS\_room\_active\_vw.Customer\_name |  |
| companyName | String companyName | ASSET.co\_nm | <240520> for ATS asset:  ARS\_room\_vw.Customer\_name |
| dialBackUpFg | Boolean dialBackUpFg | ASSET.dial\_backup\_fg |  |
| equipMdgMux | Boolean equipMdgMux | ASSET.managed\_mux\_fg |  |
| ~~\_ircuit nat~~ |  | ~~ASSET.nat\_fg~~ |  |
| slaFg | String flaFg | ASSET.sla\_fg (if sla\_fg is ‘C’ or ‘S’ it is reset to “”) |  |
| clientPriorityCd | String clientPriorityCd | ORGANIZATION.client\_priority\_cd |  |
| prtyDescTx | String prtyDescTx | LST\_CO\_PRTY.client\_priority\_desc\_tx |  |
| ~~ipreSvcFg~~ |  | ~~ASSET.ipre\_svc\_fg~~ |  |
| ~~multiCastFg~~ |  | ~~ASSET.multicast\_fg~~ |  |
| ~~multiLinkAccessRedunFg~~ |  | ~~ASSET.multi\_link\_access\_redun\_fg~~ |  |
| ProductID | String productID  M-MC | ASSET.prod\_id  LEGACY\_ASSET.product | Device Model  Varchar2(50)  TT.product\_id  <240520> for ATS asset:   * Null - for ATS Rooms record * ATS\_room\_equipment\_vw.Model\_number – for equipment record |
| ~~Int nxt1SerialIntfcCt~~ |  | ~~ASSET.nxt1\_serial\_intfc\_ct~~ |  |
| repairIntervalMinutes | Int repairIntervalMinutes | ASSET.asset\_rpr\_intvl\_nb |  |
| Product | String product | ASSET.product  LEGACY\_ASSET.product | Reported\_Product  Varchar2 (23)  TT. Reported\_Product |
| netwk\_trans | String netwk\_trans | ASSET.netwk\_trans |  |
| co\_id\_prvdr | String co\_id\_prvdr | ASSET.co\_id\_prvdr |  |
| Agns\_appl\_nm | String Agns\_appl\_nm | ASSET.agns\_appl\_nm |  |
| ~~lig\_id~~ |  | ~~EQUIPMENT.lig\_id~~ |  |
| ~~Agns\_netwk\_nm~~ |  | ~~ASSET.agns\_netwk\_nm~~ |  |
| ~~suprec\_id~~ |  | ~~ASSET.suprec\_id~~ |  |
| Agns\_bus\_gp | String Agns\_bus\_gp | ASSET.agns\_bus\_gp |  |
| ncc\_monitor\_cd | String ncc\_monitor\_cd | ASSET.ncc\_monitor\_cd |  |
| admin\_access\_ph | String admin\_access\_ph | ASSET.admin\_access\_ph |  |
| notification\_group\_name | String notification\_group\_name | ASSET.notification\_group\_name |  |
| network\_node1 | String network\_node1 | ASSET.network\_node1,  Legacy\_ASSET.network\_node1 | Network Node 1  Varchar2(30)  (was 20) CR PF413.14- |
| network\_node2 | String network\_node2 | ASSET. Network\_node2 |  |
| managedby\_netid1 | String managedby\_netid1 | ASSET. Managedby\_netid1 |  |
| managedby\_netid2 | String managedby\_netid2 | ASSET. Managedby\_netid2 |  |
| hostname | String hostname | Legacy\_ASSET.hostname | Host Name  VARCHAR2(60) (was 30, CR PF413.14)  TT.Hostname  195547 – WIFISVCS  Hostname  Hostname  <240520> for ATS asset:   * ATS\_room\_equipment\_vw.host\_name – for equipment record * Null – for ATS room record |
| hostname2 | String hostname2 | ASSET.hostname2 | 195547 - WIFISVCS  Hostname2  Varchar2(30)  Host\_name\_2 |
| Tcpip\_stack\_name | String Tcpip\_stack\_name | ASSET. Tcpip\_stack\_name |  |
| call\_indicator | String call\_indicator | ASSET. Call\_indicator |  |
| Bons\_monitor\_cd | String Bons\_monitor\_cd | ASSET. Bons\_monitor\_cd |  |
| asset\_insvc\_dt | String asset\_insvc\_dt | ASSET. Asset\_insvc\_dt, Legacy\_ASSET. Asset\_insvc\_dt | Install Date  Varchar2(15)  TT.Install\_Date |
| asset\_outsvc\_dt | String asset\_outsvc\_dt | LEGACY\_ASSET.Asset\_outsvc\_dt | Planned Disconnect Date  Varchar2(15)  TT. Planned\_Disconnect\_Date |
| report\_id | String report\_id | ORGANIZATION.report\_id |  |
| asset\_src | String asset\_src | ASSET. Asset\_src,  LEGACY\_ASSET.asset\_src\_cd |  |
| asset\_rnid | String asset\_rnid | ASSET.asset\_rnid |  |
| RestrictionType | String RestrictionType | Client\_Org.access\_type <P9654> |  |
| LocId | String LocId | LEGACY\_ASSET.src\_loc\_id | Location ID  Varchar2 (11)  TT.Location\_Id |
| productType | String productType | Legacy\_product\_catalog.product\_type  varchar2(255) | Product Type  Varchar2 (255) TT.Product\_Type  <240520> For ATS asset:   * ATS\_room\_ active\_vw.room\_type – for room record * ATS\_room\_equipment\_vw.equipment\_type – for equipment record |
| assetGroup | String assetGroup | Legacy\_asset.asset\_group\_cd  Varchar2(38, was 20) | Asset Group  Varchar2 (38) (was 20, CR PF413.14)  TT.Asset\_Group |
| AssetStatusCd | String AssetStatusCd | Legacy\_asset.asset\_status\_cd Varchar2(255) | Asset Status  Varchar2(255)  TT.Asset\_Status |
| teamTag | String teamTag | Legacy\_asset.team\_tag Varchar2(45) | Other Asset Tag  Varchar2(45)  TT.Other\_Asset\_Tag |
| edsTag | String edsTag | Legacy\_asset.eds\_tag Varchar2(30, was 20) | EDS Tag  Varchar2(30) (was 20 CR PF413.14)  TT.EDS\_Tag |
| assetTag | String assetTag | Legacy\_asset.asset\_no Varchar2(~~20~~ 54) | Asset Tag  Varchar2(~~20~~ 54)  TT.Asset\_No  <240520> for ATS asset:   * ATS\_room\_ active\_vw.room\_id – for room record * ATS\_room\_equipment\_vw.equipment\_id – for equipment record |
| platform | String platform | Legacy\_asset.platform  Varchar2(30, was 12) | Platform Description  Varchar2(30) (was 12, CR PF413.14))  TT.Platform\_Desc |
| rootId | String rootId | Legacy\_asset.root\_item\_inst\_id | -  Varchar2 (38) (was Int, CR PF413.14) |
| mfgPartnbr | String mfgPartnbr | Legacy\_product\_catalog.mfg\_part\_nm  Varchar2(20) | Mfg Part Number  Varchar2(20)  TT.Mfg\_Part\_Nbr |
| acsPartnbr | String acsPartnbr | Legacy\_product\_catalog.prod\_nm  Varchar2(18) | ACS Part Number  Varchar2(18)  TT.ACS\_Part\_Nbr |
| Asset\_Id | String Asset\_Id | Legacy\_Asset.asset.Id  number(15)  This is used to store the GCP internal unique key for the asset.  (existing attribute, from SIDBOR.asset\_details.asset\_id) | Key field for all subsequent queries  hidden  number(15)  INTERNAL\_ASSET\_ID |
| AssetAlias | String AssetAlias | ASSET.asset\_alias\_name/ Legacy\_ASSET.asset\_alias\_name | Asset Name  varchar2(45)  ASSET\_NAME |
| customerAssetAliasName | String customerAssetAliasName  (231615) | Varchar2(~~45~~ 50)  O  GCP gets data from a new DEDICATED DB | Customer Asset Alias Name  Varchar2(~~45~~ 50)  Optional  <240520> for ATS asset:  Populate this field for Room ~~or Equipment~~ asset, if the alias is available  <240520.68197> This field is only populated for ATS Room record, not for ATS equipment, for this project |
| thirdPartyType  </AID-222248-GCP-AOTS #1> | String thirdPartyType | GPS.at\_circuit\_base.3rdpartytype  Varchar2(30)  Optional  Valid values:  LOA (Letter of Agency)  COR (Customer of Record)  None (Not 3rd Party)  NULL | Third Party Type  Varchar 2(30)  Third\_Party\_Type |
| thirdPartySupplierName  <AID-222248-GCP-AOTS #2> | String thirdPartySupplierName | GPS.at\_circuit\_base.atprovider  Varchar2(70)  Conditional  This field will only be populated if thirdPartyType values are either **LOA** or **COR**. | ThirdPartysuppliername  Varchar 2(70)  Third\_Party\_Supplier\_Name |
| customerPOFlag | String customerPOFlag | New for 185108e  Data source is ABSCON.  Valid values:  Y (Yes)  N (No) | Customer PO Flag  varchar2(1)  CUSTOMER\_PO\_FLAG |
| HandlingNotes | String HandlingNotes  Varchar2(32,000)  C | customer\_level\_notes + Site\_level\_notes + CPR\_level\_Notes | New for 240195 CR201.  Data source is ABSCON.  Each Note type shall be followed by a colon and a space, followed by the Note content, followed by a semi-colon (semi-colon shall be used as the delimiter), before the next Note type.  Here is an example:  Customer Notes: Always request a customer PO before creating a trouble ticket for this customer.; Site Notes: Please enter at the north side of the building.;CPR Notes: Always dispatch  Note - For each of the Note types that do not have any content, GCP shall return the text "No note exists."  Here is an example when there is no Customer Notes and no CPR Notes:  Customer Notes: No note exists.;Site Notes: Please enter at the north side of the building.;CPR Notes: No note exists.  Note - the new HandlingNotes field is mandatory for Functional Area = CPESVCS and when the asset is found.  When Asset is not found this field should not be sent. |
| hubInd | String hubInd | Varchar2(1), Valid values – H,S, blank | Hub/Spoke Ind |
| customerASN2 | String customerASN | Varchar2(11) | Customer ASN |
| orgObjectId <232213> | ORGANIZATION.ID /Number(20)  SE Note: Value is null if that asset type is not in scope of the GDB | gdbOrgObjectID  Number(20)  hidden |  |
| referenceIPAddress  - <236171> |  | Varchar2(15)  O  Data source:  INSTAR.ipv4\_unnumbered.reference\_ip\_addr  Valid values: IPv4 address | Reference IP Address  Reference\_ip\_address  Varchar2(15) |
| multiHopLoopbackIpAddr |  | Varchar2(39)  O  From INSTAR Web service, getLayer3Data | <265609>  Populate with the value of MultiHop.LoopbackAddress.IpAddress.  This is populated only when referenceIpAddress is populated |
| isInternetVlan |  | Varchar2(1)  O | <265609>  Derived from ICORE data.  Valid values: Y, N |
| InternetCerIpv4Address |  | Varchar2(15)  O | <265609>  This field is only populated for iVLAN, data from INSTAR. |
| InternetCerIpv6Address |  | Varchar2(39)  O | <265609>  This field is only populated for iVLAN, data from INSTAR. |
| serviceRestriction |  | M-MC  GPS.Sw\_customer.AtServiceRestriction | for Managed Care assets only.  Valid values:  Restrict, Disconnect, Hold, NULL |
| invoiceIndicator |  | Varchar2(1)  O-MC  SW\_SERVICE\_ORDER.atindividualinvoice  (SR level) | for Managed Care assets only.  Valid values include:   * ‘I’ (Individual Invoice) – this value will be passed to AOTS when GPS value is 1. * ‘C (Consolidated Invoice) – this value will be passed to AOTS when GPS value is null or 0 |
| customerPONumber |  | O-MC  sw\_service\_order.atcustponbr (SR level) | for Managed Care assets only. |
| AttHelpDeskNumber |  | Varchar2(30)  M-MC  GPS.SW\_PERSON.  swOfficephonecntry ||  swOfficephonearea ||  swofficephone ||  swOfficephoneext  (trim to 30)  xxx xxx xxxxxxx xxx  (spaces in between data fields) | for Managed Care assets only. |
| costCenter |  | Varchar2(50)  M-MC  GPS.sw\_inst\_product.AtClientCostCenter | for Managed Care assets only. |
| supportingNetwork |  | Varchar2(30)  O-MC  GPS.At\_access\_circuit.Atsupportingnetwork | for Managed Care assets only.  Valid values:  B, S, T  Note: this field will be retrieved from the access circuit record if available, otherwise from the circuit base record |
| isManagedCare |  | Varchar2(1)  C  GPS.Sw\_agreement. atmgdvoicecontracttype | for Managed Care assets only.  Managed Care Indicator  Valid Values: ’N’, ’Y’  N (if GPS column not equal to ’OS’, the indicator is N - not a managed care asset) – or if the record not exists in GPS  Y (if GPS column =’OS’, the indicator is Y – is a managed care asset.  This field is conditionally manadatory for all functional areas in MCFAS list EXCEPT ‘MGDVCESVCS’. |
| ATSRoom | String ATSRoom | Varchar2(~~30~~ 32)  O | ATS Room Name  Ats\_room\_name  <240520> for ATS asset:   * The field will be populated when the asset is an ATS room, and ATS equipment. * This is the room name without leading GEMS org. * ATS\_room\_ active\_vw.room\_name – applicable for both Room and equipment records |
| maintainerNm | string | Varchar2 (32)  Optional  ASSET.maintainer\_nm | maintainerNm |
| cpeMaintenanceOption | string | Varchar2 (32)  Optional  EQUIPMENT.cpe\_maintenance\_option  Values:  ‘Lite Support’  ‘Full Support’ | cpeMaintenanceOption |
| isInterworkingRequired | IP Protocol Translation. IP\_Protocol\_TranslationEQ | VARCHAR2(1)  M | <238833> Requires Interworking at Site Level  Value: Y/N |
| calnetIndicator | String CalnetIndicator | VARCHAR(2) | Optional  ‘C3’ shall be the value for 264700a |
| agencyId | String agencyId | VARCHAR(9) | Conditional and only populated when CalnetIndator is C3  Added for CALNET3 264700a |
| agencyName | String agencyName | VARCHAR(40) | Conditional and only populated when CalnetIndicator is C3  Added for CALNET3 264700a |
| contractType | Derived | VARCHAR(25)  O | <284465h> Support EIS, with values:  EIS, or null   * Add contractType for type of EIS, retrieved based on MCN, GRC, SOC, or asset IP. * This field is populated only when the input is customer router (Name or WAN IP), and only for EIS contract type. * This contractType field can only be populated when the input router name is associated to PVC, in order to retrieve MCN, GRC, or the input is router IP (CR IP, or AR IP, or WAN IP); |
| drivenetAristaDeviceFlag | Derived | cdm:TrueFalseInfo  O | <297100a> This field is applicable for input Router Names (ptnii);  Values:  True for input Drivenet/Arista device;  False for other device name;  Null for non-router name as input;  This will allow AOTS TM users to trigger the INOD APIs with the input Drivenet, Arista devices (PTNii); |
| **Sub-structures:** | | | |
| <pvcInfoList> | Occurance: 0:1 | Complex, Optional | <274399> List of PVC records |
| <pvcInfo> | Occurance: 1:N, N<=150 | Complex, Optional | <274399> List of PVC records |
|  | pvcId | Integer(4), R  ICORE.IPFR.pvc\_id |  |
| mtu\_v4 | Integer(4), O  ICORE.ipfr.v4\_mtu - for Ethernet orders <274399-upd1>  ICORE.SITE.v4\_mtu – for POS <274399-upd1> |  |
| mtu\_v6 | Integer(4), O  ICORE.ipfr.v6\_mtu - for Ethernet orders <274399-upd1>  ICORE.SITE.v6\_mtu – for POS <274399-upd1> |  |
| pmtu | Number(22), O  ICORE.ipfr.pmtu  Only populated for Ethernet, not POS. |  |
| </pvcInfo> |  |  |  |
| </pvcInfoList> |  |  |  |
|  |  |  |  |
| **AssetChangeActivity AssetChangeActivity** | activityId | Varchar2 (32) | Change Activity ID  Varchar2 (32)  Change\_Activity\_ID |
| activityType | Varchar2 (12) | Varchar2 (12) |
| activityDate | Varchar2 (26) | Change Activity Date  Varchar2 (26)  <240520> for ATS asset   * ATS\_room\_active\_vw.activity\_date - for room record * ATS\_room\_equipment\_vw.activity\_date - for room equipment record |
| activitySource | Varchar2 (32) | Change Activity Source  Varchar2 (32)  <240520> for ATS asset:  ‘ATS’ +   * ATS\_room\_active\_vw.activity\_comments - for room record * ATS\_room\_equipment\_vw.activity\_comments - for equipment record   Note: Only retrieve the first 32-char of the string |
| **ServiceOptionListType**  **ServiceOptionList** | serviceOption | Varchar2(100)  PVC.service\_option | Asset Service Indicators  Table  Valid value-VOIP for PB293 |
| **AssetClient** | String clientCompany | ASSET.co\_id (default = functionalArea)/ LEGECY\_ASSET.functionalArea  <240520> for ATS asset:  ATS\_customer\_directory\_vw.Functional\_area | (For PF360 =ATTIT) |
| String clientCompanyName | ASSET.co\_nm (default = functionalArea)/ LEGECY\_ASSET.functionalAre  <240520> for ATS asset:  ATS\_room\_active\_vw.Customer\_name | (For PF360 =ATTIT)  Client Name  varchar2(80)  CLIENT\_NAME |
| String clientCompanyOrg | ASSET.org\_cd/ LEGECY\_ASSET.org\_cd  <240520> for ATS asset:  ATS\_customer\_directory\_vw.gems\_organization\_id | (For PF360 =ATTIT) |
| **AssetContact AssetContact** | String contactId | ASSET.contact\_id when FA=ATT\_ID  else  contact.src\_contact\_id  <232213>SE Note: Value is null if contact information is returned from the GDB | End User ID +  Varchar2 (12)  TT.End\_User\_ID\_\_ |
| Long contactObjectId | Contact.id/Number(20)  SE Note: Value is null if that asset type is not in scope of the GDB | **<232213>**gdbContactObjectID  Number(20)  hidden |
| String contactFirstName | ASSET.cntct\_first\_nm  <232213>GDB CONTACT.FIRST\_NAME  varchar(100) | End User Name  Varchar2 (43)  TT. End\_User\_Name  <240520a> for ATS asset:  ~~ATS\_room\_active\_vw.room\_coordinator\_name~~  ATS.ats\_contact\_directory\_vw. Contact\_name |
| String contactMiddleInitial | ASSET.cntct\_midinit\_nm  <232213>GDB CONTACT.FIRST\_NAME  varchar(100) |
| String contactLastName | ASSET.cntct\_last\_nm  <232213>GDB CONTACT.LAST\_NAME  varchar(100)  <240520a> for ATS asset:  ATS.ats\_contact\_directory\_vw. Contact\_name |
| String contactBusPhone | ASSET.cntct\_bus\_ph  <240520a> for ATS asset:  ATS.ats\_contact\_directory\_vw. Phone\_number | End User Phone / Contact Phone  Varchar2(25)  - END\_USER\_PHONE |
| <232213>  String contactPrefPhone | GDB PHONE.PHONE\_NUMBER Where IS\_PREFERRED = ‘Y’  Note: Null will be returned if contact data source is not GDB | Asset Contact Phone  Callback\_Phone |
| **AssetLocation** | String locationID | ASSET.loc\_id  LEGACY\_ASSET.src\_loc\_id  ATS.ATS\_room\_active\_vw.site\_id | Location ID  varchar2(11)  LOCATION\_ID  <285940.147558> US729996:  For ATS asset, populate this field with the site\_id from SAK portal. |
| String locationName | Varchar2(60)  M-MC  LOCATION.loc\_nm  for Managed Care assets only:  Trim(AT\_ATTRIBUTE\_VALUES.ATTEXTVALUE,60);  If no data above is returned, use SW\_SITE.SWSITENAME | Location Name  Varchar2(60)  LOCATION\_NAME  <240520> for ATS asset:   * ATS\_room\_active\_vw.room\_location – for room record * ATS\_room\_equipment\_vw.equipment\_location – for equipment record |
| String locationTimeZone | LOCATION.loc\_tmzone\_cd  <240520> for ATS asset:  ATS\_room\_active\_vw.timezone  <240520a> for ATS Exchange equipment:  ATS\_exchange\_vw.timezone |  |
| String locationAccessBegTime | LOCATION.loc\_access\_beg\_tm |  |
| String locationAccessEndTime | LOCATION.loc\_access\_end\_tm |  |
| String locationBusBegTime | LOCATION.loc\_bus\_beg\_tm |  |
| String locationBusEndTime | LOCATION.loc\_bus\_end\_tm |  |
| String locationVendorSiteTx | LOCATION.loc\_vendor\_site\_tx |  |
| String locationNoteTx | LOCATION.loc\_note\_tx |  |
| **AssetLocation.AddressInfo** | funtionalArea | ADDRESS.functional\_area |  |
| String CountyNm | ADDRESS.county\_nm  This attribute already exists in GCP. For 185108e, it is new for AOTS. | County  varchar2(12)  LOCATION\_COUNTY |
| String Floor |  | Floor  varchar2(10)  LOCATION\_FLOOR |
| String LocRoomId |  | Room  varchar2(12)  LOCATION\_ROOM |
| String CityNm | ADDRESS.city\_nm  <240520> for ATS asset:  ATS\_room\_active\_vw.City  <240520a> for ATS Exchange equipment:  ATS\_exchange\_vw.city | City  varchar2(60)  LOCATION\_CITY |
| String StprvCd | ADDRESS.stprv\_cd  <240520> for ATS asset:  ATS\_room\_active\_vw.State  <240520a> for ATS Exchange equipment:  ATS\_exchange\_vw.state | State/Province  varchar2(20)  LOCATION\_STATE\_PROVINCE |
| String PostalCd | ADDRESS.postal\_cd  <240520> for ATS asset:  ATS\_room\_active\_vw.Zip\_postal\_cd  <240520a> for ATS Exchange equipment:  ATS\_exchange\_vw.zip\_postal\_code | Zip/Postal  varchar2(18)  LOCATION\_ZIPPOSTAL |
| String CountryCd | ADDRESS.country\_cd  <240520> for ATS asset:  ATS\_room\_active\_vw.Country\_code  <240520a> for ATS Exchange equipment:  ATS\_exchange\_vw.country\_code | Country Code  varchar2(5)  LOCATION\_COUNTRYCODE |
| String addr\_line\_tx | ADDRESS.addr\_line\_tx  <240520> for ATS asset:  ATS\_room\_active\_vw.Address\_line1  <240520a> for ATS Exchange equipment:  ATS\_exchange\_vw.address\_line1 | Address Line 1  varchar2(75)  LOCATION\_ADDRESS\_1 |
| String addr\_line2\_tx | ADDRESS.addr\_line2\_tx  <240520> for ATS asset:  ATS\_room\_active\_vw.Address\_line2  <240520a> for ATS Exchange equipment:  ATS\_exchange\_vw.address\_line2, | Address Line 2  varchar2(75)  LOCATION\_ADDRESS\_2 |
| Jack | varchar2(30)  M-MC  GPS.SW\_INST\_PRODUCT.ATJACK | for Managed Care assets only |
| cube | varchar2(30)  M-MC  GPS.SW\_INST\_PRODUCT.ATCUBE | for Managed Care assets only |
| building | varchar2(10)  M-MC GPS.SW\_INST\_PRODUCT.ATBUILDING | for Managed Care assets only |
| afterHoursAccessInstructions | varchar2(2000)  O-MC  GPS.sw\_site. Atafterhoursinstructions | for Managed Care assets only |
| attCircuitId | varchar2(44)  O-MC  GPS.AT\_CIRCUIT\_BASE.ATCIRCUITNAME | for Managed Care assets only  example: used in BellSouth |
| accessProviderCircuitId | varchar2(44)  O-MC  GPS.AT\_ACCESS\_CIRCUIT.ATCIRCUITNAME | for Managed Care assets only  example: classic T (ATT prior to merger, ATT did not have access to customer premise) used now when we do not want to give att circuit access to 3rd party vendor, so we give them this one.  Could also be BellSouth |
| **AssetOrgs assetOrgsType** | String billing | ASSET.billing\_org\_cd/ Legacy\_ASSET.billing\_org\_cd |  |
| String active | <PF360 Legacy\_>  ASSET.active\_org\_cd/ Legacy\_ASSET.active\_org\_cd | Active\_Org  Varchar2 (15)  TT. Active\_Org |
| String managing | <PF360  ASSET.managing\_org\_cd/ Legacy\_ASSET.managing\_org\_cd | Managing\_Org  Varchar2 (11)  TT. Managing\_Org |
| String owning | <PF360 ASSET.owning\_org\_cd/ Legacy\_ASSET.owning\_org\_cd | Owning\_Org  Varchar2 (20)  TT.Owning\_Org |
| **AssetOrgs.SharedOrgsList SharedOrgsList** | String orgCd | LOCATION.loc\_org\_cd  <240520> for ATS asset:  ATS\_customer\_directory\_vw.gems\_organization\_id |  |
| **arrayofServiceOptions** | serviceOption | PVC.service\_option |  |
| **OrgGroupList sharedOrgGroupList** |  | LOCATION.loc\_org\_cd |  |
| AssetChangeActivity AssetChangeActivity | activityId | Varchar2 (32) |  |
| activityType | Varchar2 (12) |  |
| activityDate | Varchar2 (26) | <240520> for ATS asset   * ATS\_room\_active\_vw.activity\_date - for room record   ATS\_room\_equipment\_vw.activity\_date - for equipment record |
| activitySource | Varchar2 (32) | <240520> for ATS asset ‘ATS’ +   * ATS\_room\_active\_vw.activity\_comments - for room record * ATS\_room\_equipment\_vw.activity\_comments - for equipment record   Note: Only retrieve the first 32-char of the string |
| arrayLanIpAddressInfo  Occurance: 0:N, N<=100  <265609>  This structure is only populated for internet VLAN | String lanIpAddress | INSTAR.assign\_ip.ip for IPv4  INSTAR.ipv6\_assign\_ip.ipv6\_ip for IPv6 | This is the Network IP Address with Subnet mask (for IPv4), populated only for Internet VLAN.  For IPv4, the subnet is part of the IP value. e.g. .27, .28, .23, .24, .25   |  | | --- | | **IP** | | 12.32.93.128/27 | | 12.32.93.160/28 | | 199.105.98.0/23 | | 199.105.100.0/24 | | 12.9.149.0/25 |   For IPv6, the subnet is in separate field. |
| String lanIpv6Subnet | INSTAR.ipv6\_assigned\_subnets.Subnet\_ip | This is the Network IPv6 Address Subnet.  Sample values:   |  | | --- | | SUBNET\_IP | | 2001:1890:1209:0400 | | 2620:0000:0F3E | | 2620:0000:0F7C | | 2001:1890:1200:0400 | | 2620:0000:0F4E | | 2620:0000:0F2C | | 2001:1890:1200:1000 | | 2001:1890:1200:1400 | | 2001:1890:1200:0000 | | 2001:1BE0:1101 | | 2001:1BE0:1102 | | 2001:1BE0:1100:0000 | | 2001:1890:1C00:0200 | | 2001:1BE0:1100:1F00 | |
| sCUstomerProvided |  | Derived from INSTAR LAN IPv4/v6 record.  Valid values: Y, N |

Please note: Where ever there is reference to LEGACY (eg: LEGACY\_ASSET), it’s for PF360

**<AOTS Rel.2- cVOIP>**

If the Asset Data Interface.getAssetDetails method is invoked with the input functionalArea = ‘‘CONSUMER\_SVC”, invoke cVOIPDataAccessObject.getAssetDetails(CommonHeader header, String assetTag, String assetTagType) throws com.att.aots.is.AssetNotFoundException. AOTS will always bind to the AssetDataInterface even for cVOIP inventory (Ideal model would be for AOTS to directly bind to cVOIP webservice. AssetDataInterface.getAssetDetails() is used as a AOTS adapter in this context)

**<AOTS Rel.2- cVOIP>**

**<AOTS Rel.3- EM NB-IPVPN Support>**

The local\_router field in FWToplogy table will be in lower case. The asset\_nm field in the Asset table for Router assets will be in upper case. To facilitate the correlation of Routers in the FWTopology table and corresponding record in the Asset table, the seach by assetTagType = ‘NAME’ will be performed as a case insensitive search.

**<AOTS Rel.3- EM NB-IPVPN Support>**

**<AOTS Rel.3.1 – DoJ Support>**

If the value of input parameter functionalArea = ‘DOJ\_JUTN’, apply the following logic in addition to the common function of this query:

|  |  |  |
| --- | --- | --- |
| **Table Name** | **Column Name(s)** | **Condition for selection** |
| **SiDBOR:**  location l,  asset a | distinct l.loc\_org\_cd | a.functional\_area = input ‘functionalArea’  a.src\_asset\_id = input ‘assetTag’  l.functional\_area = a.functional\_area  l.hosting\_src\_loc\_id = a.src\_loc\_id  l.loc\_type\_cd = ‘HGD’ |

**<AOTS Rel.3.1 – DoJ Support>**

**<AOTS Rel7 – P7392 Change Correlation>**

Using Asset\_Id,

Select Chg\_Activity\_ID, Chg\_Activity\_Type, Chg\_Activity\_Date, chg\_Activity\_Src

From SIDBOR.Asset\_Chgact

Where Asset\_ID=Asset\_ID input from AOTS

**< AOTS Rel7 – P7392 Change Correlation>**

**<AOTS R7.1 – PA031 Service Line Mapping>**

**Table layout for new table SvcLineMapping is in Appendix 16.2**

Retrieve asset.service

Select FaultSvcLine

From SvcLineMapping

Where ProvSvcLine=Asset.Service

If found,

Return FaultSvcLine to AOTS in the output field service

Else

Return Asset.Service in the output field service

**<AOTS R7.1 – PA031 Service Line Mapping>**

**<AOTS R9 – P9654>**

Put addition logic to filter via INPUT RestrictionType field. If the input has RestrictionType populated, match the input RestrictionType to client\_org.access\_type where asset.org\_cd = client\_org.org\_cd and Asset.funcational\_area = client\_org.functional\_area and client\_org.access\_type = INPUT restrictiontype. If INPUT does not match, then return NO MATCH FOUND. See Appendix “Access Restriction Matrix” for output case scenarios.

**<AOTS R9 – P9654>**

<PB293>

Data Access Logic:

**<249601> Support Third Party Pseudo Circuit ID as input**

<249601-upd2> Remove conditions on FunctionalArea for 3rd Party Pseudo Circuit logic,

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Data source- SiDBOR** | **Condition for selection** |
| Asset\_nm | SiDBOR.Asset  SiDBOR.Circuit | Asset.Asset\_nm = <input assetTag>  OR  (  Asset.Asset\_id = Circuit.Asset\_id  And Circuit.Lec\_ckt1 = <input assetTag> (3rd party circuit)  ) |

Following existing logic using the retrieved Asset\_ID or Asset\_nm, whichever applys.

**</249601>**

**<185108e> Notes:**

For 185108e, functional area is equal to ‘CPESVCS’. One new field is added to input structure. It is customerNumber and is an optional field. One new field is added to output structure. It is customerPOFlag. All other output fields are business as usual.

If customerNumber is present in input, use the following logic to retrieve data from ABSCON.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step | ABSCON Table Name | Attribute Name | Condition for selection | Note |
| 1 | CUSTOMER | Select customer\_po\_flag | ABSCON\_Customer\_Num = ‘From Input” | This will retrieve data for customerPOFlag. |

If customerNumber is not present in input, use the following logic to retrieve ABSCON Customer Number from functionalArea and assetTag in GPS database.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step | GPS Table Name | Attribute Name | Condition for selection | Note |
| 1 | SW\_Inst\_Product S,  SW\_Customer SC | SC.ATGEMSCOID | S.ATspecassetname = ‘Input assetTag’  and SC.ATGEMSSCID = ‘CPESVCS’  and S.SWCUSTOMERID = SC.SWCUSTOMERID | This will retrieve ABSCON Customer Number.  Use the result to populate customerNumber field. |
| 2 | Use the ABSCON Customer Number to retrieve ABSCON data from the above table | | | |
| 3 | Data retrieval logic for HandlingNotes added for 240195 CR201 –  Select Concatenate ( ‘Customer Notes: ’ a.Customer\_level\_notes ‘;’, ‘Site Notes: ’ b.Site\_Level\_Notes‘;’, ‘CPR Notes: ’ c.CPR\_level\_notes ‘;’) as HandlingNotes  From Customer\_Note a, Site\_Note b, CPR\_Note c  Where   1. ABSCON\_customer\_num = CPR. ABSCON\_customer\_num   And CPR.CPR\_Number = C.cpr\_number  And c. ABSCON\_site\_num = b. ABSCON\_site\_num  And IF  a.Customer\_level\_notes and/or b.Site\_Level\_Notes and/or c.CPR\_level\_notes = null ( i.e it has no content)  Then populate the value “No Note Exists.”  Example 1 when all notes have value -  Customer Notes: Always request a customer PO before creating a trouble ticket for this customer.; Site Notes: Please enter at the north side of the building.;CPR Notes: Always dispatch  Example2 - When a note type that do not have any content, GCP shall return the text "No note exists."    when there is no Customer Notes and no CPR Notes:  Customer Notes: No note exists.; Site Notes: Please enter at the north side of the building.;CPR Notes: No note exists  Note – If no content exist for all the three fields follow the same format. For example –  Customer Notes: No note exists.; Site Notes: No Note exists.; CPR Notes: No note exists | | | |

**</185108e>**

**<207810> Notes:**

**<207810 –GCP-AOTS Req. #140>**

* Add ATI Indicator to the main output structure – ICORE.Site.ati.
* This field is populated only when the input is customer router (Name or IP) or ID (GCP Asset ID)
* This can only be retrieved when the input asset is associated to PVC.
* In the case of customer router (Name or IP) as input, there can be multi ports on the router and so lead to multi accesses. So the ATI is actually at the service option/Port level.
* This field is informational, and AOTS team wants to have it in the main output structure of this query. So the solution is:

1. Retrieve the AIT field at service option level,
2. If any one of the ATI retrieved is “PVOT”, then the ATI indicator in the output is set to “PVOT”.
3. If none of the ATI retrieved is “PVOT”, then the ATI indicator is set to the first value from the ATI list.

**</207810>**

**<231615> Note on access logic change:**

When FunctionalArea is not ‘ATT\_IT’

* Retrieve Asset.Asset\_ID field for the input asset – add this attribute in existing logic
* Determine Asset.Asset\_src\_cd – the Source DB code for the Asset\_ID
* Use the Source DB code and the asset\_ID to determine the CustomerAssetAliasName

When FunctionalArea is ‘ATT\_IT’

* This is for ATT internal equipments for IT organization. The Source DB code will not be added for these inventory and so the customerAssetAlias can not be determined for these equipments.
* Leave the CustomerAssetAliasName blank.

**</231615>**

**<236171> Notes:**

* Add reference IP Address, to the main output structure - INSTAR.ipv4\_unnumbered.reference\_ip\_addr
* This field is populated only when the input is customer router (Name or IP) or ID (GCP Asset ID)
* This can only be retrieved when the input asset is associated to PVC.

**</236171>**

**<240520> Note on access logic change:**

When FunctionalArea is ‘SD\_HOSTING’ and assetTagType = ‘ATS\_ROOM’

* Data source change to ATS
* Retrieve Room record matching the input from the ATS DB – step ATS-1

When FunctionalArea is ‘SD\_HOSTING’ and assetTagType is not ‘ATS\_ROOM’

* If existing logic didn’t retrieve data for the input, then treat it as ATS equipment and try to retrieve ATS
* Retrieve the equipment record matching the input from the ATS DB – step ATS-2

**</240520>**

**<272078i-ite19>**

Added VMS indicator.

* ~~This VMS indicator is only derived for input AssetTagType = Name (router) – this is the router PTNii.~~
* ~~For other input types, the VMS indicator is default to ‘N’~~
* <272078i-ite19-upd2> As the first step in the query, the VMS indicator is derived for input assetTag, assuming it’s Router PTNii.
* <272078i-ite19-upd2> This step is performed regardless the input assettagType – removing checking on the assettagType

**</272078i-ite19>**

**<285914> Determine Cascated Indicator (using A&AI schema)**

* isCascatedUCpe field is default to ‘N’
* If isCascatedUCpe field is set to ‘Y’, AOTS can call the NoD DeviceDetails API to retrieve uCPE and vFW info.

|  |  |
| --- | --- |
| **Steps** | **Logic and Description** |
| <272078i>  isVms | **STEP NOD-100-1**: Deriving VMS indicator  **<272078i-ite19> Determine VMS Indicator (using A&AI schema)**   * If VMS Indicator is Y, then only this field is populated and returned to AOTS, all the other fields are left blank. * If VMS Indicator is N, then BAU logic for this query is used.   <301033> Note:  Replace A&AI tables from feed file by the ones from DMaap.  The input device can be the PTNii or hostname of uCPE, real router, or VNF.  Set isVms = ‘N’  Select hosename as deviceName  From aai\_pserver  Where Upper(hostname) = Upper (<input-assetTag>)  Or Upper(pserver\_name2) = Upper (<input-assetTag>)  Union  Select pnf\_name as deviceName  From aai\_pnf  Where Upper(pnf\_name) = Upper (<input-assetTag>)  Or Upper(pnf\_name2) = Upper (<input-assetTag>)  Union  Select vnf\_name as deviceName  From aai\_generic\_vnf  Where Upper(vnf\_name) = Upper (<input-assetTag>)  Or Upper(vnf\_name2) = Upper (<input-assetTag>)  ~~Select count(\*)~~  ~~From INSTAR.INSTAR\_AAI\_DEVICE~~  ~~Where ptnii\_name = <input-assetTag> assuming it’s router-ptnii <272078i-ite19-upd2>~~  289116-upd01: Address test issue of “No managedBy returned when isVms=Y”  If the above SQL returns data, then  set isVms=Y  Continue to STEP NOD-100-2 retrieving uCPE info  ~~Return the isVms field back to calling application.~~  ~~Else~~  ~~Follow BAU data retrieval logic.~~  End If |
| <285914>  isCascadedUCpe | **STEP NOD-100-2**: Deriving Cascated UCPE indicator  Derive isCascatedUcpe field – Can be the first step in BAU logic  <289116.141314-US640793> Note:  Retrieve the managed indicator for UCPE, for input real router PTNii, need to relate UCPE to the real router – UCPE ‘depdent on’ Real router  <301033> Note:  This is to support cascaded uCPE (with vFW, but without vRouter - RT) behind Real Router, LAN side internet, as the diagram shown below. This is uCPE without routing functionality, relying on real router to provide the routing functionality   * This setup has a Real-Router (aka physical , non-virtualized, traditional) in front of the uCPE device. This Router can be AT&T managed or Customer managed. * If the router is third party, and access is third party then AT&T will not monitor the 3rd party access. Customer will be responsible for 3rd party transport. * In this configuration, the routing function is done by the real router. The uCPE will have vFW.   <301033> Note:  New table created in AAI schema containing equipment types for UCPE (including Juniper and Vyatta UCPEs), so no hardcode needed.  ucpe\_equip\_type\_set.ucpe\_equip\_type    <301033>  STEP NOD-100-2-A: Determine whether this is a uCPE behind a Real Router  Search path: pnf > p\_interface > physical\_link > p\_interface > pserver  SELECT Unique pserver.hostname  FROM  aai\_pserver pserver,  aai\_pnf pnf,  Aai\_p\_interface pinterface1,  Aai\_p\_interface pinterface2,  Aai\_p\_interface\_physical\_link pip\_link1,  Aai\_p\_interface\_physical\_link pip\_link2  WHERE 1 =1  And pnf.pnf\_name = <input-assetTag>  And pinterface2.pnf\_name = pnf.pnf\_name  And pinterface2.interface\_name = pip\_link2.interface\_name  And pinterface2.pnf\_name = pip\_link2.pnf\_name  And pip\_link1.link\_name = pip\_link2.link\_name  And pinterface1.interface\_name = pip\_link1.interface\_name  And pinterface1.hostname = pip\_link1.hostname  And pserver.hostname = pinterface1.hostname  And pserver.equip\_type IN  (ucpe\_equip\_type\_set.ucpe\_equip\_type)  ~~('JUNIPER UCPE',~~ ~~' SILICOM UCPE')~~  If the above returned data, then continue to the below step:  STEP NOD-100-2-B: Determine this uCPE has firewall (FW) but no vRouter (RT)  Search path: pserver > vserver\_pserver> vserver> generic\_vnf\_vserver> generic\_vnf  Select vnf.vnf\_id, vnf.vnf\_type  From  Aai\_generic\_vnf vnf,  Aai\_generic\_vnf\_vserver vnfVserver,  Aai\_vserver\_pserver vPserver  Where 1=1  And vPserver.hostname = <hostname> --retrieved from STEP NOD-100-2-A  And vPserver.vserver\_id = vnfVserver.vserver\_id  And vnfVserver.vnf\_id = vnf.vnf\_id  And vnf.vnf\_type IN ('RT', 'FW')  If the above step returns only vnf\_type=’FW’, then  set isCascadedUCpe =Y  Else (no data returned, or RT is returned as well)  set isCascadedUCpe =N  End If  <301033> Remove below logic  ~~Select Unique ptnii\_name, equip\_type <285914>~~  ~~From INSTAR.instar\_aai\_device~~  ~~Where ptnii\_name = <input-assetTag> assuming it’s router-ptnii~~  ~~And equip\_type IN (‘MRS’, ‘AVPN-M’)~~  ~~Select Unique~~  ~~dev.ptnii\_name AS ucpeHostName,~~  ~~dev.managed\_by AS ucpeManagedBy~~  ~~From~~  ~~Instar\_aai\_dualEnded dend,~~  ~~Instar\_aai\_device dev~~  ~~Where 1=1~~  ~~And dend.z\_ptnii\_name = <input-assetTag> assuming it’s real router-ptnii~~  ~~And dend.a\_ptnii\_name = dev.ptnii\_name – UCPE~~  ~~And dev.equip\_type = 'JUNIPER UCPE'~~  </301033> |
| <289116>  ucpeManagedBy | <301033>  **STEP NOD-100-3**: Determine ucpeManagedBy value  Select NVL (management\_option, ‘ATT’) AS ucpeManagedBy  From aai\_pserver  Where Upper(hostname) = Upper (<input-assetTag>)  Or Upper(pserver\_names) = Upper (<input-assetTag>)  Or Upper(hostname) = Upper (<hostname>) -- from STEP NOD-100-2-A  )  ~~set isCascadedUCpe =’N’~~  ~~If the above SQL returns data, then~~  ~~set isCascadedUCpe =Y~~  ~~Set ucpeManagedBy = managed\_by retrieved above <289116.141314-US640793>~~  ~~End If~~ |
|  | 289116-upd01: Address test issue of “No managedBy returned when isVms=Y”  If isVms=Y, then  Return the isVms, isCascadedUCpe, and ucpeManagedBy fields back to calling application.  Else  Follow BAU data retrieval logic.  End If  ~~Follow BAU data retrieval logic.~~ |

**<297100a> Determine rivenetAristaDeviceFlag for the input Router name**

* This field is applicable for input Router Names.
* Set to True for input Drivenet/Arista device PTNii; Set to False for other device PTNii; Set to Null for non-PTNii as input. (ARISTA – ARISTA Networks, vendor);
* For ARISTA devices, this API will not return device inventory. So when this field is set to True, the API will stop further processing, and return this flag to client application. This will allow AOTS TM users to trigger the INOD APIs with the input Drivenet, Arista devices (PTNii) to get its inventory;

|  |  |
| --- | --- |
| **Steps** | **Logic and Description** |
| drivenetAristaDeviceFlag | <297100a>  STEP DN-100-1: Deriving drivenetAristaDeviceFlag, for input Routers  If input assetTagType=NAME  Check the last 3-characters of the input Router name (PTNii)  If the 3-chat string is ‘mee’, or ‘sw8’, or ‘sw9’, or ‘swb’, ~~or ‘swa’~~, Then  Set drivenetAristaDeviceFlag = True  Else  Set drivenetAristaDeviceFlag = False  End If  Else  Set drivenetAristaDeviceFlag = Null  End If  If drivenetAristaDeviceFlag = True, then  Stop further processing,  Return the drivenetAristaDeviceFlag and the input PTNii fields back to calling application.  Else (for drivenetAristaDeviceFlag = False or Null)  Follow BAU data retrieval logic.  End If  **Device types based on the last 3-character of the PTNii:**  mee – new DriveNets MIS/AVPN edge router in US and MOW  sw8 – new Arista Leaf Switch in US and MOW  sw9 – new Arista Spine Switch in US and MOW  swb – new Arista Switch (TOR) in US  ICORE.service.service\_name (partial list)   |  |  | | --- | --- | | **(count)** | **serv\_name** | | 2 | ATM | | 1 | MLPPP | | 1 | GATEWAY | | 1 | MULTICAST | | 1 | BROADCAST DOMAIN | | 1 | OPT-E-MAN | | 1 | MLFR | | 1 | VAFA | | 1 | MPLS VPN | | 1 | UBB | | 1 | PRIORITY FR | | 1 | L3 EGS Access | | 1 | Cloud | | 1 | Multi-Homing | | 1 | BICI/NNI | | 1 | DIVERSITY | | 1 | VPN | | 1 | MPLS PNT | | 1 | FBS | | 1 | Managed Srvc | | 1 | EVPN | | 1 | GMIS | | 1 | INCS | | 1 | IPFR | | 1 | CES | | 1 | OPT-E-WAN | |

1. Based on the retrieval of the existing asset detail record, retrieve the array of service options from SIDBOR using the below logic.

|  |  |  |
| --- | --- | --- |
| **Table Name** | **Attribute Name** | **Condition for selection** |
| ASSET,  PVC | distinct  PVC.service\_option | ASSET.service = ‘NB-IPVPN’ and  (upper(PVC.pvc\_src\_asset\_nm) = upper(input asset\_Nm ) or upper(PVC.pvc\_dest\_asset\_nm) = upper(input asset\_Nm))  and PVC.service\_option is not Null |
| SIDBOR.ASSET,  SIDBOR.PVC | **Not for output:**  Unique  ASSET.asset\_nm as icore\_pvc\_id  <231615>  ASSET.asset\_id | **<207810> When input Functional Area is NOT “ATT\_IT”**  **Access SiDBOR: retrieve icore\_pvc\_id**  **Step A1: When input AssetTagType is Name (router)** (upper(PVC.pvc\_src\_asset\_nm) = <input\_asset> OR upper(PVC.pvc\_dest\_asset\_nm) = <input\_asset>) AND PVC.service\_option = 'VOIP' AND ASSET.asset\_id = PVC.asset\_id  **Step A2: When input AssetTagType is IP**  (PVC.pvc\_src\_ip\_addr = <input\_IP> OR PVC.pvc\_dest\_asset\_nm = <input\_IP>) AND PVC.service\_option = 'VOIP' AND ASSET.asset\_id = PVC.asset\_id  **Step A2: When input AssetTagType is ID (Asset\_ID)** ASSET.asset\_id = <input ID>  AND PVC.service\_option = 'VOIP' AND ASSET.asset\_id = PVC.asset\_id |
| SIDBOR.LEGACY\_ASSET,  SIDBOR.PVC | **Not for output:**  Unique  LEGACY\_ASSET.asset\_nm as icore\_pvc\_id | **<207810> When input Functional Area is “ATT\_IT”**  **Access SiDBOR: retrieve icore\_pvc\_id**  **Step B1: When input AssetTagType is Name (router)** (upper(PVC.pvc\_src\_asset\_nm) = <input\_asset> OR upper(PVC.pvc\_dest\_asset\_nm) = <input\_asset>) AND PVC.service\_option = 'VOIP' AND LEGACY\_ASSET.asset\_id = PVC.asset\_id  **Step B2: When input AssetTagType is HOST\_IP**  (PVC.pvc\_src\_ip\_addr = <input\_IP> OR PVC.pvc\_dest\_asset\_nm = <input\_IP>) AND PVC.service\_option = 'VOIP' AND LEGACY\_ASSET.asset\_id = PVC.asset\_id  **Step B3: When input AssetTagType is ID (Asset\_ID)** LEGACY\_ASSET.asset\_id = <input ID>  AND PVC.service\_option = 'VOIP' AND LEGACY\_ASSET.asset\_id = PVC.asset\_id |
| ICORE.Site  ICORE.pvc | Not for output:  Unique Site.ATI  Site.site\_id | **Step ~~B~~C: For all the icore\_pvc\_id retrieved above (Step A, B), access ICORE to get ATI:**  pvc.pvc\_id = <icore\_pvc\_id> from the step above  And site.site\_id = pvc.pvc\_lsite\_id |
| ICORE.site\_extension | Populate performanceGroup from site\_extension.performance\_group | <282720>  Step D1: for the first site.site\_id received from step C:  Select performance\_group  From site\_extension  site  Where site.site\_id = site\_extension.site\_id  and site.site\_id = <site\_id from step C>  Stop  This process only chose the first site\_id retrieved to populate access level performance group information. |
| ICORE.performance\_data | Populate performanceCharacteristics  from performance\_data.performance\_characteristicsperf\_charstc | <282720>  Step D2: for the first site.site\_id received from step C:  Select performance\_characteristics perf\_charstc  From site  performance\_data  Where site.site\_id = performance\_data.site\_id  and site.site\_id = <site\_id from step C>  Stop  This process only chose the first site\_id retrieved to populate access level performance performanceCharacteristics information. |
| **Step C: For all the ATIs retrieved from Step ~~B~~C:**  If any one of them has value of “PVOT”  Then set AtiIndicator in the Main structure to “PVOT”  Else, set AtiIndicator in the Main structure to the value of the first ATI retrieved | | |
| INSTAR  ipv4\_unnumbered  ip\_serv\_acc\_pt | **Unique**  reference\_ip\_addr | **Step D:** <236171>  For all the icore\_pvc\_id retrieved above (Step A, B), access INSTAR:  ip\_serv\_acc\_pt.icore\_pvc\_id = **<icore\_pvc\_id>** from the step above  And  ipv4\_unnumbered.serv\_acc\_pt\_id = ip\_serv\_acc\_pt.serv\_acc\_pt\_id |
| **Step E: For all the** reference\_ip\_addr **retrieved from Step D:**  If any one of them is not NULL  Then set reference\_ip\_addr in the Main structure to this reference\_ip\_addr  Else, set reference\_ip\_addr in the Main structure to Null | | |
|  | | |

</PB293>

**<PB655>**

1. For the input assetTag and FunctionalArea, get the org\_cd and FunctionalArea from the Asset table.

|  |  |  |
| --- | --- | --- |
| **Table Name** | **Attribute Name** | **Condition for selection** |
| Asset | One occurrence of  (org\_cd,Functional\_area) | Asset.AssetID = input assetTag  And  Asset.functional\_area = <input functional\_area> |

1. Retrieve the org\_cd\_prnt and org\_cd\_gprnt from the client\_org table for the input org\_cd and functional\_area derived from step 1

|  |  |  |
| --- | --- | --- |
| **Table Name** | **Attribute Name** | **Condition for selection** |
| Client\_org | One occurrence of  (org\_cd\_prnt,org\_cd\_gprnt) | Client\_org.org\_cd = org\_cd derived from step 1 and client\_org.functional\_area = functional\_area derived from step 1 |

**Note:** For the input assetTag and functional\_area, if there is no corresponding value in the Asset table, return “Data not found”

</PB655>

**Data Access Logic:**

1. If the Input functional\_area = ‘ATT\_IT’ and the assetTagType = ‘Name’ retrieve the data using the below logic.

|  |  |  |
| --- | --- | --- |
| **SIDBOR Table Name** | **Attribute Name** | **Condition for selection** |
| LEGACY\_ASSET,  LEGACY\_EQUIPMENT,  ORG,  CONTACT,  LOCATION,  CLIENT\_ORG,  ADDRESS,  LEGACY\_PRODUCT\_CATALOG | LEGACY\_ASSET.asset\_id,  LEGACY\_ASSET.functional\_area,  LEGACY\_ASSET.asset\_class\_cd,  LEGACT\_ASSET.asset\_nm,  LEGACY\_ASSET.asset\_alias\_name,  LEGACY\_ASSET.asset\_type\_cd,  LEGACY\_ASSET.asset\_src\_cd,  LEGACY\_ASSET.src\_asset\_id,  LEGACY\_ASSET.src\_root\_item\_inst\_id,  LEGACY\_ASSET.src\_parent\_item\_inst\_id,  LEGACY\_ASSET.team\_tag,  LEGACY\_ASSET. Org\_cd,  LEGACY\_ASSET.asset\_insvc\_dt,  LEGACY\_ASSET.src\_loc\_id,  LEGACY\_ASSET.loc\_detl\_id,  LEGACY\_ASSET.asset\_outsvc\_dt,  LEGACY\_ASSET.product,  LEGACY\_ASSET.contact\_id,  LEGACY\_ASSET.active\_org\_cd,  LEGACY\_ASSET.managing\_org\_cd,  LEGACY\_ASSET.billing\_org\_cd,  LEGACY\_ASSET.owning\_org\_cd,  LEGACY\_ASSET.asset\_status\_cd,  LEGACY\_ASSET.asset\_group\_cd,  ~~LEGACY\_ASSET.service\_id,~~  LEGACY\_ASSET.src\_org\_id,  LEGACY\_ASSET.platform,  LEGACY\_ASSET.mac\_address,  LEGACY\_ASSET.node\_name,  LEGACY\_ASSET.hostname,  LEGACY\_ASSET.asset\_no,  LEGACY\_ASSET.eds\_tag,  LEGACY\_ASSET.serial\_num,  LEGACY\_ASSET.src\_root\_item\_inst\_id,  LEGACY\_EQUIPMENT.asset\_id,  CONTACT.cntct\_first\_nm,  CONTACT.cntct\_midinit\_nm,  CONTACT.cntct\_last\_nm,  CONTACT.cntct\_bus\_ph,  LOCATION.loc\_nm,  LOCATION.loc\_tmzone\_cd,  LOCATION.loc\_access\_beg\_tm ,  LOCATION.loc\_access\_end\_tm ,  LOCATION.loc\_bus\_beg\_tm ,  LOCATION.loc\_bus\_end\_tm ,  LOCATION.loc\_vendor\_site\_tx ,  LOCATION.loc\_note\_tx ,  ADDRESS.city\_nm,  ADDRESS.country\_cd ,  ADDRESS.stprv\_cd,  ADDRESS.postal\_cd,  ADDRESS.addr\_line\_tx,  ADDRESS.addr\_line2\_tx,  ADDRESS.county\_nm,  ADDRESS.functional\_area,  ADDRESS.src\_address\_id,  ORG.org\_nm,  ORG.client\_priority\_cd,  LEGACY\_EQUIPMENT. equip\_netwk\_ipaddr\_id,  LEGACY\_PRODUCT\_CATALOG.mfg\_part\_nm(mfg part nb)  LEGACY\_PRODUCT\_CATALOG.prod\_nm (acs part nb),  LEGACY\_PRODUCT\_CATALOG.product\_type | LEGACY\_ASSET.asset\_nm = input assetTag and LEGACY\_ASSET.functional\_area = input functionalArea and LEGACY\_ASSET.org\_cd = ORG. src\_org\_cd and LEGACY\_ASSET.functional\_area = ORG.functonal\_area and LEGACY\_ASSET.org\_cd = CLIENT\_ORG.org\_cd and LEGACY\_ASSET.functional\_area = CLIENT\_ORG.functional\_area And LEGACY\_ASSET.contact\_id = CONTACT.contact\_id(+) and LEGACY\_ASSET.functional\_area = LOCATION.functional\_area (+) and LEGACY\_ASSET.src\_loc\_id = LOCATION.src\_loc\_id(+) and LOCATION.addr\_id = ADDRESS.addr\_id and LEGACY\_ASSET.asset\_id = LEGACY\_EQUIPMENT.asset\_id (+)  And LEGACY\_ASSET.product = LEGACY\_PRODUCT\_CATALOG.prod\_id |

1. If the Input functional\_area = ‘ATT\_IT’ and the assetTagType = ‘ID’ retrieve the data using the below logic.

|  |  |  |
| --- | --- | --- |
| **SIDBOR Table Name** | **Attribute Name** | **Condition for selection** |
| LEGACY\_ASSET,  LEGACY\_EQUIPMENT,  ORG,  CONTACT,  LOCATION,  CLIENT\_ORG,  ADDRESS,  LEGACY\_PRODUCT\_CATALOG | LEGACY\_ASSET.asset\_id,  LEGACY\_ASSET.functional\_area,  LEGACY\_ASSET.asset\_class\_cd,  LEGACT\_ASSET.asset\_nm,  LEGACY\_ASSET.asset\_alias\_name,  LEGACY\_ASSET.asset\_type\_cd,  LEGACY\_ASSET.asset\_src\_cd,  LEGACY\_ASSET.src\_asset\_id,  LEGACY\_ASSET.src\_root\_item\_inst\_id,  LEGACY\_ASSET.src\_parent\_item\_inst\_id,  LEGACY\_ASSET.team\_tag,  LEGACY\_ASSET. Org\_cd,  LEGACY\_ASSET.asset\_insvc\_dt,  LEGACY\_ASSET.src\_loc\_id,  LEGACY\_ASSET.loc\_detl\_id,  LEGACY\_ASSET.asset\_outsvc\_dt,  LEGACY\_ASSET.product,  LEGACY\_ASSET.contact\_id,  LEGACY\_ASSET.active\_org\_cd,  LEGACY\_ASSET.managing\_org\_cd,  LEGACY\_ASSET.billing\_org\_cd,  LEGACY\_ASSET.owning\_org\_cd,  LEGACY\_ASSET.asset\_status\_cd,  LEGACY\_ASSET.asset\_group\_cd,  ~~LEGACY\_ASSET.service\_id,~~  LEGACY\_ASSET.src\_org\_id,  LEGACY\_ASSET.platform,  LEGACY\_ASSET.mac\_address,  LEGACY\_ASSET.node\_name,  LEGACY\_ASSET.hostname,  LEGACY\_ASSET.asset\_no,  LEGACY\_ASSET.eds\_tag,  LEGACY\_ASSET.serial\_num,  LEGACY\_ASSET.src\_root\_item\_inst\_id,  LEGACY\_EQUIPMENT.asset\_id,  CONTACT.cntct\_first\_nm,  CONTACT.cntct\_midinit\_nm,  CONTACT.cntct\_last\_nm,  CONTACT.cntct\_bus\_ph,  LOCATION.loc\_nm,  LOCATION.loc\_tmzone\_cd ,  LOCATION.loc\_access\_beg\_tm ,  LOCATION.loc\_access\_end\_tm ,  LOCATION.loc\_bus\_beg\_tm ,  LOCATION.loc\_bus\_end\_tm ,  LOCATION.loc\_vendor\_site\_tx ,  LOCATION.loc\_note\_tx ,  ADDRESS.city\_nm,  ADDRESS.country\_cd ,  ADDRESS.stprv\_cd,  ADDRESS.postal\_cd,  ADDRESS.addr\_line\_tx,  ADDRESS.addr\_line2\_tx,  ADDRESS.county\_nm,  ADDRESS.functional\_area,  ADDRESS.src\_address\_id,  ORG.org\_nm,  ORG.client\_priority\_cd,  LEGACY\_EQUIPMENT. equip\_netwk\_ipaddr\_id,  LEGACY\_PRODUCT\_CATALOG.mfg\_part\_nm(mfg part nb)  LEGACY\_PRODUCT\_CATALOG.prod\_nm (acs part nb),  LEGACY\_PRODUCT\_CATALOG.product\_type | LEGACY\_ASSET.asset\_id = input assetTag and LEGACY\_ASSET.functional\_area = input functionalArea and LEGACY\_ASSET.org\_cd =ORG. src\_org\_cd and LEGACY\_ASSET.functional\_area = ORG.functonal\_area and LEGACY\_ASSET.org\_cd = CLIENT\_ORG.org\_cd and Legacy\_ASSET.functional\_area = CLIENT\_ORG.functional\_area and LEGACY\_ASSET.contact\_id = CONTACT.contact\_id(+) and LEGACY\_ASSET.functional\_area = LOCATION.functional\_area (+) and LEGACY\_ASSET.src\_loc\_id = LOCATION.src\_loc\_id(+) and LOCATION.addr\_id = ADDRESS.addr\_id and LEGACY\_ASSET.asset\_id = LEGACY\_EQUIPMENT.asset\_id (+)    And LEGACY\_ASSET.product = LEGACY\_PRODUCT\_CATALOG.prod\_id |

1. If the Input functional\_area = ‘ATT\_IT’ and the assetTagType = ‘HOST\_IP’ retrieve the data using the below logic.

|  |  |  |
| --- | --- | --- |
| **SIDBOR Table Name** | **Attribute Name** | **Condition for selection** |
| LEGACY\_ASSET,  LEGACY\_EQUIPMENT,  ORG,  CONTACT,  LOCATION,  CLIENT\_ORG,  ADDRESS | LEGACY\_ASSET.asset\_id,  LEGACY\_ASSET.functional\_area,  LEGACY\_ASSET.asset\_class\_cd,  LEGACT\_ASSET.asset\_nm,  LEGACY\_ASSET.asset\_alias\_name,  LEGACY\_ASSET.asset\_type\_cd,  LEGACY\_ASSET.asset\_src\_cd,  LEGACY\_ASSET.src\_asset\_id,  LEGACY\_ASSET.src\_root\_item\_inst\_id,  LEGACY\_ASSET.src\_parent\_item\_inst\_id,  LEGACY\_ASSET.team\_tag,  LEGACY\_ASSET. Org\_cd,  LEGACY\_ASSET.asset\_insvc\_dt,  LEGACY\_ASSET.src\_loc\_id,  LEGACY\_ASSET.loc\_detl\_id,  LEGACY\_ASSET.asset\_outsvc\_dt,  LEGACY\_ASSET.product,  LEGACY\_ASSET.contact\_id,  LEGACY\_ASSET.active\_org\_cd,  LEGACY\_ASSET.managing\_org\_cd,  LEGACY\_ASSET.billing\_org\_cd,  LEGACY\_ASSET.owning\_org\_cd,  LEGACY\_ASSET.asset\_status\_cd,  LEGACY\_ASSET.asset\_group\_cd,  ~~LEGACY\_ASSET.service\_id,~~  LEGACY\_ASSET.src\_org\_id,  LEGACY\_ASSET.platform,  LEGACY\_ASSET.mac\_address,  LEGACY\_ASSET.node\_name,  LEGACY\_ASSET.hostname,  LEGACY\_ASSET.asset\_no,  LEGACY\_ASSET.eds\_tag,  LEGACY\_ASSET.serial\_num,  LEGACY\_ASSET.src\_root\_item\_inst\_id, LEGACY\_EQUIPMENT.asset\_id,  CONTACT.cntct\_first\_nm,  CONTACT.cntct\_midinit\_nm,  CONTACT.cntct\_last\_nm,  CONTACT.cntct\_bus\_ph,  LOCATION.loc\_nm,  LOCATION.loc\_tmzone\_cd ,  LOCATION.loc\_access\_beg\_tm ,  LOCATION.loc\_access\_end\_tm ,  LOCATION.loc\_bus\_beg\_tm ,  LOCATION.loc\_bus\_end\_tm ,  LOCATION.loc\_vendor\_site\_tx ,  LOCATION.loc\_note\_tx ,  ADDRESS.city\_nm,  ADDRESS.country\_cd ,  ADDRESS.stprv\_cd,  ADDRESS.postal\_cd,  ADDRESS.addr\_line\_tx,  ADDRESS.addr\_line2\_tx,  ADDRESS.county\_nm,  ADDRESS.functional\_area,  ADDRESS.src\_address\_id,  ORG.org\_nm,  ORG.client\_priority\_cd,  LEGACY\_EQUIPMENT. equip\_netwk\_ipaddr\_id,  LEGACY\_PRODUCT\_CATALOG.mfg\_part\_nm(mfg part nb)  LEGACY\_PRODUCT\_CATALOG.prod\_nm (acs part nb),  LEGACY\_PRODUCT\_CATALOG.product\_type | LEGACY\_EQUIPMENT.equip\_netwk\_ipaddr\_id = input assetTag or  LEGACY\_ASSET.hostname = = input assetTag and LEGACY\_ASSET.functional\_area = input functionalArea and LEGACY\_ASSET.org\_cd =ORG. src\_org\_cd and LEGACY\_ASSET.functional\_area = ORG.functonal\_area and LEGACY\_ASSET.org\_cd = CLIENT\_ORG.org\_cd and Legacy\_ASSET.functional\_area = CLIENT\_ORG.functional\_area and LEGACY\_ASSET.contact\_id = CONTACT.contact\_id(+) and LEGACY\_ASSET.functional\_area = LOCATION.functional\_area (+) and LEGACY\_ASSET.src\_loc\_id = LOCATION.src\_loc\_id(+) and LOCATION.addr\_id = ADDRESS.addr\_id and LEGACY\_ASSET.asset\_id = LEGACY\_EQUIPMENT.asset\_id (+)  And LEGACY\_ASSET.product = LEGACY\_PRODUCT\_CATALOG.product\_id |

**<235050> Access logic for retrieving isCascadedRouter**

When SiDBOR.Equipment table is accessed for base structure data retrieval, pull EQUIPMENT.cascaded\_fg field (adding EQUIPMENT.cascaded\_fg to the Select clause), and populate it to field isCascadedRouter in the base structure.

**</235050>**

**<237570>** Access logic for retrieving maintainerNm and cpeMaintenanceOption

##### **HLD\_237570\_GCP\_AOTS\_100 - Update** **getAssetDetails**

**Reference: FR-237570-GCP-SA-100**

DO when functional\_area = GLOBAL\_AM

When **SIDBOR.ASSET** Table is accessed for base structure data retrieval, pull **ASSET.maintainer\_nm** (adding ASSET.maintainer\_nm to the Select clause) and populate it to field ‘**maintainerNm**’ in the base structure output response

When **SIDBOR.Equipment** table is accessed for base structure data retrieval, pull EQUIPMENT.cpe\_maintenance\_option field (adding EQUIPMENT.cpe\_maintenance\_option to the Select clause), and populate it to field cpeMaintenanceOption in the base structure output response.

ENDO

##### **End of HLD\_237570\_GCP\_AOTS\_100 - Update** **getAssetDetails**

**</237570>**

**<264700a>** Access logic for retrieving agencyId, agencyName and calnetIndicator

##### 

DO when **input** functional\_area = ‘CALNET3’

<ticket # R3S1400007660>

IF input assettag\_type = ID

SELECT ASSET\_NM

FROM SIDBOR.ASSET A

WHERE input ID = A.ASSET\_ID

SELECT AGENCY\_ID, AGENCY\_NAME

FROM CSI\_NX.C3\_ASSET\_XREF C

WHERE A.ASSET\_NM retrieved above = C.Asset\_NM

</ticket # R3S1400007660>

ELSE

SELECT AGENCY\_ID, AGENCY\_NAME

FROM CSI\_NX.C3\_ASSET\_XREF C

WHERE input assettag = C.Asset\_NM

IF a match is found,

Populate the base structure output response with the following fields:

Set agencyId = C.AGENCY\_ID

Set agencyName = C.AGENCY\_NAME

Set calnetIndicator = ‘C3’

ELSE

ENDIF

ENDO

DO when input functional\_area is not ‘CALNET3’

IF input assettag\_type = ID

SELECT ASSET\_NM

FROM SIDBOR.ASSET A

WHERE input ID = A.ASSET\_ID

SELECT AGENCY\_ID, AGENCY\_NAME

FROM CSI\_NX.C3\_ASSET\_XREF C

WHERE A.ASSET\_NM retrieved above = C.Asset\_NM

</ticket # R3S1400007660>

ELSE

SELECT AGENCY\_ID, AGENCY\_NAME

FROM CSI\_NX.C3\_ASSET\_XREF C

WHERE input assettag = C.Asset\_NM

IF a match is found,

Populate the base structure output response with the following fields:

Set agencyId = C.AGENCY\_ID

Set agencyName = C.AGENCY\_NAME

Set calnetIndicator = ‘C3’

ELSE

ENDIF

ENDIF

ENDO

**</264700a>**

**<222248> Access logic:**

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Data source- SiDBOR** | **Condition for selection** |
| thirdPartyType | Circuit. thirdPartyType | **Determine thirdPartyType**  Circuit. thirdPartyType <222248.012>  **<HLD-222248-GCP-AOTS # 1>**  **Case**  **When AssetTagType=ID**  Access SIDBOR.Circuit where <input AssetTag=Circuit.ASSET\_ID>  If row found  Retrieve <222248.012>thirdPartyType</222248.012>  Else (row not found)  Access SIDBOR.Equipment where <input AssetTag=ASSET\_ID>  If row found  If OWNSHP\_TYPE\_CD=’P’  Using EQUIPMENT.ASSET\_ID, access ASSET where EQUIPMENT.ASSET\_ID= ASSET.ASSET\_ID  Retrieve ASSET.ASSET\_NM  Using ASSET.ASSET\_NM, access CIRCUIT where ASSET.ASSET\_NM = CIRCUIT.A\_ASSET\_NM  If row found  Retrieve <222248.012>thirdPartyType</222248.012>  Else (row not found)  Using ASSET.ASSET\_NM, access CIRCUIT where ASSET.ASSET\_NM = CIRCUIT.Z\_ASSET\_NM  If row found  Retrieve <222248.012>thirdPartyType</222248.012>  Else (row not found)  Endif  Endif  Else (OWNSHP\_TYPE\_CD is not ‘P’)  <222248.CR035>  Access SIDBOR.PVC where  <input  AssetTag=ASSET\_ID>  If row found  Using PVC.ASSET\_ID, access ASSET where PVC.ASSET\_ID= ASSET.ASSET\_ID  Retrieve ASSET.ASSET\_NM  Using ASSET.ASSET\_NM, access PVC where ASSET.ASSET\_NM= PVC.PVC\_SRC\_ASSET\_NM  If row found  Retrieve <222248.012>thirdPartyType</222248.012>  Else (row not found)  Using ASSET.ASSET\_NM, access PVC where ASSET.ASSET\_NM=PVC.PVC\_DEST\_ASSET\_NM  If row found  Retrieve <222248.012>thirdPartyType</222248.012>  Else (row not found)  Endif  Endif  Else (no row found)</222248.CR035>  Endif  Else (no row found)  Endif  Endif  **When AssetTagType=Name**  Retrieve ASSET\_ID from SIDBOR.ASSET  where (<input assetTag>=ASSET.ASSET\_NM  OR <249601>  Retrieve ASSET\_ID from SIDBOR.CIRCUIT  where <input assetTag>=Circuit.Lec\_ckt1 (3rd party circuit)  )  Access SIDBOR.Circuit where  ASSET.ASSET\_NM retrieved above=Circuit.ASSET\_ID>  If row found  Retrieve <222248.012>thirdPartyType<222248.012> value  Else (row not found)  Access SIDBOR.Equipment where <input AssetTag=EQUIP\_DEVICE\_NM>  If row found  If OWNSHP\_TYPE\_CD=’P’  Using EQUIPMENT.ASSET\_ID, access ASSET where EQUIPMENT.ASSET\_ID=ASSET.ASSET\_ID  Retrieve ASSET.ASSET\_NM  Using ASSET.ASSET\_NM, access CIRCUIT where ASSET.ASSET\_NM=CIRCUIT.A\_ASSET\_NM  If row found  Retrieve <222248.012>thirdPartyType</222248.012>  Else (row not found)  Using ASSET.ASSET\_NM, access CIRCUIT where ASSET.ASSET\_NM= CIRCUIT.Z\_ASSET\_NM  If row found  Retrieve <222248.012>thirdPartyType</222248.012>  Else (row not found)  Endif  Endif  Else (OWNSHP\_TYPE\_CD is not ‘P’)  <222248.CR035>  Access SIDBOR.PVC where <input AssetTag=PVC\_SRC\_ASSET\_NM>  If row found  Retrieve <222248.012>thirdPartyType</222248.012>  Else (row not found)  Access SIDBOR.PVC where <input= AssetTag=PVC\_DEST\_ASSET\_NM>  If row found  Retrieve <222248.012>thirdPartyType</222248.012>  Else (row not found)  Endif  </222248.CR035>  Endif  Else (no row found)  Endif  Endcase  **</HLD-222248-GCP-AOTS # 1>** |
| thirdPartySupplierName | Circuit. thirdPartyType | **Determine thirdPartySupplierName**  Asset.vendor\_id  Varchar2(70)  **<HLD-222248-GCP-AOTS # 2>**  If <222248.012>thirdPartyType</222248.012> value is ‘LOA’ or ‘COR’  If asset is found in SIDBOR.Equipment  Using ASSET\_ID value Access ASSET where EQUIPMENT.ASSET\_ID=ASSET.ASSET\_ID  Retrieve VENDOR\_ID and  populate in <222248.012>thirdPartySupplierName</222248.012>  Else (asset is found in SIDBOR.Circuit)    Using CIRCUIT.A\_ASSET\_NM, access ASSET where CIRCUIT.A\_ASSET\_NM=ASSET.ASSET\_NM  If row found  Retrieve ASSET.VENDOR\_ID and  Populate in <222248.012>thirdPartySupplierName</222248.012>  Else (row not found)  <222248.035>  Else (asset is found in SIDBOR.PVC)  Using ASSET\_ID value, Access ASSET where PVC.ASSET\_ID= ASSET.ASSET\_ID  Retrieve VENDOR\_ID and  populate in thirdPartySupplierName  </222248.0035>  **Endif**  **</HLD-222248-GCP-AOTS # 2>** |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **Table Name** | **Attribute Name** | **Condition for selection** |
| ICORE  ipfr | **Unique**  pvc\_id,  v4\_mtu,  v6\_mtu,  pmtu | **Step F:** <274399>  <274399-upd2> For all the icore\_pvc\_id retrieved from SiDBOR, and the PVC IDs retrieved from NC3 in earlier steps, access ICORE DB:  **Input: <icore\_pvc\_id>** --this includes PVC IDs retrieved from both SiDBOR and NC3.  Ipfr.pvc\_id = **<icore\_pvc\_id>**  Populate these fields to the pvcInfo structure in pvcInfoList. |
|  |  | <274399-upd1>  **If no data returned from the above step, do the following:** |
| ICORE:  Pvc,  site | **Unique**  Pvc.pvc\_id,  site.v4\_mtu,  site.v6\_mtu,  ‘’ pmtu | **Step F:** <274399-upd1>  **Get attributes for the Remote and local site**  For all the icore\_pvc\_id retrieved (for the asset) above, access ICORE:  pvc.pvc\_id = **<icore\_pvc\_id>**  Site.site\_id = pvc.pvc\_rsite\_id    **UNION**  pvc.pvc\_id = **<icore\_pvc\_id>**  Site.site\_id = pvc.pvc\_lsite\_id  Populate these fields to the pvcInfo structure in pvcInfoList. |

**<284465h> Notes: Determine contractType of EIS:**

* Add contractType for type of EIS, retrieved based on MCN, GRC, SOC, or asset IP.
* This field is populated only when the input is customer router (Name or WAN IP), and only for EIS contract type.
* This contractType field can only be populated when the input router name is associated to PVC, in order to retrieve MCN, GRC, or the input is router IP (CR IP, or AR IP, or WAN IP);
* This field is not applicable for assetTagType = ‘ATS\_ROOM’

|  |  |  |
| --- | --- | --- |
| **~~DB Table Name~~** | **~~Column Name(s)~~** | **~~Condition for selection~~** |
|  |  | **~~STEP-200 <284465h>~~**  **~~For input Router IP (CR IP, or AR IP, or WAN IP), retrieve MCN, GRC, SOC~~** |
| ~~INSTAR:~~  ~~Serial\_ip\_addr,~~  ~~Ip\_port\_asgmt,~~  ~~Ip\_serv\_acc\_pt,~~  ~~Facility\_account~~ | ~~Facility\_account.mcn,~~  ~~Facility\_account.grc,~~  ~~Facility\_account.soc,~~ | **~~STEP-200-A: For input Router IPv4~~**  ~~Input: <IPv4>~~  ~~Serial\_ip\_addr.ip\_address = <IPv4>~~  ~~AND (Serial\_ip\_addr.serial\_ip\_addr\_id = Ip\_port\_asgmt.wan\_addr\_id~~  ~~Or Serial\_ip\_addr.serial\_ip\_addr\_id = Ip\_port\_asgmt.ar\_addr\_id~~  ~~Or Serial\_ip\_addr.serial\_ip\_addr\_id = Ip\_port\_asgmt.cr\_addr\_id~~  ~~)~~  ~~AND Ip\_port\_asgmt.assgn\_status = 'PRINCIPAL'~~  ~~AND Ip\_port\_asgmt.serv\_acc\_pt\_id = Ip\_serv\_acc\_pt.serv\_acc\_pt\_id~~  ~~AND Ip\_serv\_acc\_pt.cust\_id = Facility\_account.cust\_id~~ |
| ~~INSTAR:~~  ~~Ipv6\_assigned\_link\_ips,~~  ~~Ipv6\_port\_asgmt\_map,~~  ~~Ip\_port\_asgmt,~~  ~~Ip\_serv\_acc\_pt,~~  ~~Facility\_account~~ | ~~Facility\_account.mcn,~~  ~~Facility\_account.grc,~~  ~~Facility\_account.soc,~~ | **~~STEP-200-B: For input Router IPv6 – If the above step did not return data, perform the following logic:~~**  ~~Input: <IPv6>~~  ~~(Ipv6\_assigned\_link\_ips.ipv6\_address\_compress = <IPv6>~~  ~~OR~~  ~~Ipv6\_assigned\_link\_ips.ipv6\_address=~~~~<IPv6>~~  ~~)~~  ~~And Ipv6\_assigned\_link\_ips.ipv6\_link\_ip\_id=~~  ~~Ipv6\_port\_asgmt\_map. ipv6\_link\_ip\_id~~  ~~And Ipv6\_port\_asgmt\_map.sdid = Ip\_port\_asgmt.sdid~~  ~~AND Ip\_port\_asgmt.serv\_acc\_pt\_id = Ip\_serv\_acc\_pt.serv\_acc\_pt\_id~~  ~~AND Ip\_serv\_acc\_pt.cust\_id = Facility\_account.cust\_id~~ |
| ~~EDF:~~ | ~~contractType~~ | **~~STEP-200-C: Retrieve ContractType~~**  **~~Input:~~** ~~<input-mcn>, <input-grc>, <input-soc>~~  ~~If the above steps returned data (MCN, GRC, SOC), then~~  ~~Execute Comon logic getEISContractType (<input-mcn>, <input-grc>, <input-soc>)~~  ~~Returns contractType~~  ~~If the common logic returned data, then~~  ~~Override contractType field with the value returned.~~  ~~Else,~~  ~~Leave contractType field unpopulated.~~  ~~End If~~ |

|  |  |  |
| --- | --- | --- |
| Step 210 |  | **STEP-210 <284465h>**  **For input Asset ~~Router Name~~, assuming icoer\_pvc\_id has been retrieved in earlier steps (see steps for PID 207810 and Step F).**  **For all the icore\_pvc\_id retrieved from SiDBOR, and the PVC IDs retrieved from NC3 in earlier steps, access ICORE DB:** |
| ICORE:  Customer,  Site,  Pvc,  Port\_asgmt,  Cust\_access | Unique  Customer.cust\_mcn,  Site.grc | STEP-210-A: For any one PVC\_ID from the asset record,  Input: <icore\_pvc\_id> --This can be a list PVC IDs  pvc.pvc\_id IN (<icore\_pvc\_id>)  And site.site\_id = pvc.pvc\_lsite\_id  And site.site\_id = Port\_asgmt.site\_id  And Port\_asgmt.site\_id = cust\_access.site\_id  And cust\_access.cust\_id = customer.cust\_id  And customer.cust\_id !=395  Union  pvc.pvc\_id IN (<icore\_pvc\_id>)  And site.site\_id = pvc.pvc\_rsite\_id  And site.site\_id = Port\_asgmt.site\_id  And Port\_asgmt.site\_id = cust\_access.site\_id  And cust\_access.cust\_id = customer.cust\_id  And customer.cust\_id !=395 |
| EDF: | contractType | STEP-210-B: Retrieve ContractType for each pair of MCN and GRC retrieved above.  Input: <input-mcn>, <input-grc>  If the above steps returned data (MCN, GRC), then for each pair of MCN and GRC:  Execute Common logic getEISContractType (<input-mcn>, <input-grc>)  Returns contractType  If any execution of the common logic returned data, then  Override contractType field with the value returned.  Else,  Leave contractType field unpopulated.  End If |

|  |  |  |
| --- | --- | --- |
| Step 220 |  | **STEP-220 <284465h>**  **For input Asset ~~Router Name~~, assuming GPS data is retrieved (in existing logic retrieving GPS data)** |
| GPS Table:  Sw\_customer | Unique  contractType (derived) | STEP-210-A:  For existing logic retrieving data from GPS.SW\_Customer table, add below logic in SELECT clause to derive value for contractType field:  Sw\_customer.swAcctType field is checked for the values below, which presents EIS type:  "MRS-EIS"  "Outsourcing-EIS"  As an example, this line can be used in the SELECT clause:  Decode ( Upper(sw\_customer.swAcctType),  ‘MRS-EIS’, ‘EIS’,  OUTSOURCING-EIS’, ‘EIS’,  ‘’ ) AS contractType |
|  |  | Existing Code Accessing GPS:  SELECT  sip.swstatus ServiceStatus,  sip.swdateinstalled DateInstalled,  sc.atgemscoid CustomerBAN,  sip.atspecassetname ServiceBAN,  v.swname Vendor, spr.swname Product,  cr.atassetrole ContactRole,  ct.swextemailaddress ContactEmailAddress, sip.swnote specialHandling,  swl.swnote supportInfo,  sc.swname companyName,  sa.swcountry Country  FROM  sw\_customer\_gps sc,  sw\_prod\_release\_gps spr,  sw\_inst\_product\_gps sip,  sw\_customer\_gps v,  at\_contact\_asset\_role\_gps cr,  sw\_person\_gps ct,  sw\_work\_log\_gps swl, sw\_address\_gps sa  WHERE sip.swcustomerid = sc.swcustomerid  AND sip.swprodreleaseid = spr.swprodreleaseid  AND spr.swcustomerid = v.swcustomerid  AND sip.swinstprodid = cr.atinstproductid(+)  AND cr.atcontactid = ct.swpersonid(+)  AND cr.atassetrole = 'Outage'  AND cr.swobjecttype(+) = 'INSTALLED'  AND sip.swstatus = 'Installed'  AND sip.atspecassetname = ?  AND sip.swinstprodid = swl.swobjectid  AND upper(swl.SWSUBJECT(+)) = 'ADDITIONAL SUPPORT INFORMATION'  AND sc.swcustomerid = sa.swobjectid  AND sa.swobjecttype = 'CUSTOMER'  AND sa.swtype = 'Physical Location' ;    SELECT SC.ATGEMSCOID  FROM  SW\_Inst\_Product\_gps S,  SW\_Customer\_gps SC  Where S.ATspecassetname = ?  AND S.SWCUSTOMERID = SC.SWCUSTOMERID ; |

**</284465h>**

**<231615> Access logic – Determine CustomerAssetAliasName:**

**When FunctionalArea is not ‘ATT\_IT’, do the following**

|  |  |  |
| --- | --- | --- |
| **Table (in SiDBOR)** | **Attribute** | **Condition for selection** |
| asset | asset\_id,  asset\_src\_cd,  asset\_src\_primarykey | upper(asset\_nm) = upper( **<asset\_name>)** -- the value populated to the assetName in the output structure <231615.10>  **<231615.10>** Follow the access logic descriped “231615-GCP-EM HLD”, section for “Alias retrieval processing rules for SIDBOR.ASSET records” |

**</231615.10>**

**Alias retrieval processing rules for SIDBOR.ASSET records**

Extend the existing processing logic and lookup an optionally existing customer defined asset ticket alias value for each SIDBOR.ASSET row being the basis for the response:

1. Retrieve SIDBOR.ASSET.asset\_src\_cd, SIDBOR.ASSET.asset\_src\_primarykey and SIDBOR.ASSET.src\_asset\_id.
2. Use the following database constraints depending on SIDBOR.ASSET.asset\_src\_cd to lookup the alias value as described in **Error! Reference source not found.**.

Database constraints for SIDBOR.ASSET.asset\_src\_cd=’GRDB’:

|  |  |  |
| --- | --- | --- |
| **Table name** | **Column name** | **Containt(s)** |
| META\_SYSTEM | NAME | ‘LPP-CPE’ |
| SOURCE\_KEY\_VALUE | SOURCE\_VALUE | SIDBOR.ASSET.asset\_src\_primarykey |

*Note: The missing meta information in SIDBOR.ASSET does not support the identification of table name or column name of the source system.*

Database constraints for SIDBOR.ASSET.asset\_src\_cd=’GPS’:

|  |  |  |
| --- | --- | --- |
| **Table name** | **Column name** | **Containt(s)** |
| META\_SYSTEM | NAME | ‘GPS’ |
| SOURCE\_KEY\_VALUE | SOURCE\_VALUE | SIDBOR.ASSET.asset\_src\_primarykey |

*Note: The missing meta information in SIDBOR.ASSET does not support the identification of table name or column name of the source system.*

Database constraints for other SIDBOR.ASSET.asset\_src\_cd:

|  |  |  |
| --- | --- | --- |
| **Table name** | **Column name** | **Containt(s)** |
| META\_SYSTEM | NAME | ‘SIDBOR\_SOURCE’ |
| META\_TABLE | NAME | ‘ASSET’ |
| META\_COLUMN | NAME | ‘ASSET\_SRC\_CD’ |
| ‘SRC\_ASSET\_ID’ |
| SOURCE\_KEY\_VALUE | SOURCE\_VALUE | SIDBOR.ASSET.asset\_src\_cd |
| SIDBOR.ASSET.src\_asset\_id |

**</231615.10>**

**<240520> Note on access logic to retrieve data from ATS:**

When FunctionalArea is ‘SD\_HOSTING’ and assetTagType = ‘ATS\_ROOM’, then access ATS schema to

* Retrieve Room record matching the input from the ATS DB – step ATS-1

When FunctionalArea is ‘SD\_HOSTING’ and assetTagType is not ‘ATS\_ROOM’, If existing logic didn’t retrieve data for the input, then treat it as ATS equipment and access ATS schema

* Retrieve the equipment record matching the input from the ATS DB – step ATS-2

**Note:**

* It must be made sure that only data is taken into account where so-called “in-service” equipment exists (here: “active” rooms that have an “order\_number” value with a trailing “INVENTORY”). Only rooms, but not the “sub-equipment” data will be taken into account.
* Screening for active room is done in view ATS\_room\_active\_vw, so this view is used here in the access logic, removing the need for checking trailing ‘INVENTORY’ in the Order\_number field.

**</240520>**

**<~~240520a~~ 255187> Note on access logic on pulling Exchange Equipment data from ATS:**

When a request from AOTS for equipment data is received by GCP, GCP needs to determine if the equipment is EXCHANGE or “standard” equipment, by checking its “equipment type” value in the ATS room equipment table.

If the “equipment type” value starts with “EXCH-“, then it is EXCHANGE equipment, then the data to be returned to AOTS needs to pulled specifically for that piece of EXCHANGE equipment.

</240520a>

Assumptions:

* The hostname is a unique identifier for Exchange Equipment. SAK Portal will propagate hostname (or other unique identifier) into the Equipment ID field for exchange equipment.
* The equipment type for Exchange equipment will always begin with “EXCH-“ (therefore GCP can determine exchange vs. standard equipment type).
* <240520a-upd-1> For EXCHANGE equipment, we must not use the "ATS.ats\_room\_equipment\_active\_vw" view, but the "ATS.ats\_room\_equipment\_vw" view. This is because for exchange equipment, there is no order number data, and especially no one with a leading "INVENTORY" to indicate "active" equipment,

**Step ATS-1:** Retrieve the ATS Room record

|  |  |  |  |
| --- | --- | --- | --- |
| **Output Fields** | **ATS Table name** | **Column name** | **Conditions** |
| assetClass,  companyName,  Service,  productType,  assetTag,  ATSRoom  activityDate  activitySource,  clientCompany,  clientCompanyName,  clientCompanyOrg,  contactLastName,  contactBusPhone,  locationName,  locationTimeZone  CityNm,  StprvCd,  PostalCd,  CountryCd,  addr\_line\_tx,  addr\_line2\_tx,  orgCd  locationId <285940.147558> | ~~ATS\_room\_vw ar~~  ATS\_room\_active\_vw ar  ATS\_customer\_directory\_vw acd  ATS.ats\_contact\_directory\_vw acy,  ATS.ats\_order\_contact \_vw aoc, | ‘ATS ROOM’ AS assetClass,  ar.customer\_name,  ‘ATS’ AS Service,  ar.room\_type,  ar.room\_name,  ar.room\_name,  ar.activity\_date,  ‘ATS’ + ar.activity\_comments (take first 32-char)  ar.functional\_area,  ar.customer\_name,  acd.gems\_organization\_id,  ~~ar.room\_coordinator\_name~~  ~~ar.room\_coordinator\_number~~  acy.Contact\_name,  acy.Phone\_number,  ar.room\_location,  ar.timezone,  ar.city,  ar.state,  ar.zip\_postal\_code,  ar.country\_code,  ar.address\_line1,  ar.address\_line2,  acd.gems\_organization\_id  ar.~~sak\_internal\_~~site\_id  as locationId <285940.147558> US729996 | ar.room\_id = **<input-assetTag>**  And  acd.functional\_area = **<input-functionalArea>**  And  acd.customer\_name= ar.customer\_name  And  ~~INSTR(TRIM(ar.order\_number),'INVENTORY') <> 0~~  <240520a> Retrieve primary contact info:  And  ar.Order\_number = aoc.Order\_number  And  aoc.Customer\_name = acy.Customer\_name  And  aoc.Contact\_name = acy.Contact\_name  And  aoc.Role\_type = ‘1’ (for Primary contact)  **Note:**  If multiple Room records are returned for the input, the first record is used for the output |
| customerAssetAliasName |  |  | See Step ATS-3 |

**Step ATS-2:** Retrieve the Equipment records

|  |  |  |  |
| --- | --- | --- | --- |
| **Output Fields** | **ATS Table name** | **Column name** | **Conditions** |
| <255187>  <240520b> | ATS\_room\_equipment\_vw ae <255187-upd-1> | ae.Customer\_name,  Substr (ae.equipment\_type, 1, 5) ATS exchange\_flag,  INSTR(TRIM(ae.order\_number),'INVENTORY') active\_flag | **<ATS-2-A>**  <255187><240520b> Determine Equipment Type: Exchange or Standard equipment  ae.equipment\_id = **<input-assetTag>**  If ATS exchange\_flag = ‘EXCH-‘  Follow <ATS-2-B> - Retrieve Exchange Equipment  Else If active\_glag != 0 (equip active)  Follow <ATS-2-C> - Retrieve Standard Equipment (existing logic)  Else  The input standard equipment is in-active  Return “No data found”  End if |
| assetProtocol,  assetClass,  companyName,  productID,  Service,  Hostname,  productType,  assetTag,  ATSRoom,  activityDate  activitySource,  clientCompanyOrg,  contactLastName,  contactBusPhone,  locationName,  locationTimeZone  CityNm,  StprvCd,  PostalCd,  CountryCd,  addr\_line\_tx,  addr\_line2\_tx,  orgCd <255187-uat-01> | ATS\_room\_equipment\_vw ae, <255187-upd-1>  ATS\_exchange\_vw aex,  ATS\_customer\_directory\_vw acd <255187-uat-01> | DECODE(ae.sb\_protocol)  For 1, assetProtocol= H.323  For 2, assetProtocol=SIP  ‘EXCHANGE’ AS assetClass,  ae.customer\_name,  ae.Model\_number,  ‘ATS’ AS Service,  ae.host\_name,  ae.equipment\_type,  ae.equipment\_id,  ae.room\_name,  ae.activity\_date,  ‘ATS’ + ae.activity\_comments (take first 32-char) AS activitySource,  acd.gems\_organization\_id AS clientCompanyOrg <255187-uat-02>  aex.Primary\_contact\_name,  aex.Primary\_contact\_number,  ae.equipment\_location,  aex.timezone,  aex.city,  aex.state,  aex.zip\_postal\_code,  aex.country\_code,  aex.address\_line1,  aex.address\_line2,  Substr (ae.equipment\_type, 1, 5) ATS exchange\_flag (not for output)  acd.gems\_organization\_id AS orgCd | **<ATS-2-B>**  **Retrieve Exchange Equipment:**    ae.equipment\_id = **<input-assetTag>**  And  Upper (aex.Exchange\_name) = upper (ae.room\_name)  And  ae.Custome\_name = 'ATS-EXCHANGE'  And  exchange\_flag = ‘EXCH-‘  <255187-uat-01>  And  acd.Functional\_area = <input-functionalArea>  And  acd.Customer\_name (+) = ae.Customer\_name |
| assetProtocol,  assetClass,  companyName,  productID,  Service,  Hostname,  productType,  assetTag,  ATSRoom,  activityDate  activitySource,  clientCompany,  clientCompanyName,  clientCompanyOrg,  contactLastName,  contactBusPhone,  locationName,  locationTimeZone  CityNm,  StprvCd,  PostalCd,  CountryCd,  addr\_line\_tx,  addr\_line2\_tx,  orgCd | ATS\_room\_active\_vw ar  ATS\_room\_equipment\_active\_vw ae  ATS\_customer\_directory\_vw acd  ATS\_contact\_directory\_vw acy,  ATS\_order\_contact \_vw aoc, | DECODE(ae.sb\_protocol)  For 1, assetProtocol= H.323  For 2, assetProtocol=SIP  ae.equipment\_type AS assetClass,  ar.customer\_name,  ae.Model\_number,  ‘ATS’ AS Service,  ae.host\_name,  ae.equipment\_type,  ae.equipment\_id,  ar.room\_name,  ae.activity\_date,  ‘ATS’ + ae.activity\_comments (take first 32-char)  ar.functional\_area,  ar.customer\_name,  acd.gems\_organization\_id,  ~~ar.room\_coordinator\_name~~  ~~ar.room\_coordinator\_number~~  acy.Contact\_name,  acy.Phone\_number,  ae.equipment\_location,  ar.timezone,  ar.city,  ar.state,  ar.zip\_postal\_code,  ar.country\_code,  ar.address\_line1,  ar.address\_line2,  acd.gems\_organization\_id | **<ATS-2-C>**  **Retrieve Standard Equipment:**  (  ~~ar.room\_id =~~ **~~<input-assetTag>~~**  ~~OR~~  ae.equipment\_id = **<input-assetTag>**  )  And  ae.Room\_name = ar.Room\_name  And  ae.Customer\_name = ar.Customer\_name  And  acd.Functional\_area = <input-functionalArea>  And  acd.Customer\_name= ar.Customer\_name  And  Ae.Order\_number = ar.Order\_number  ~~And~~  ~~INSTR(TRIM(ar.Order\_number),'INVENTORY') <> 0~~  <240520a> Retrieve primary contact info:  And  ae.Order\_number = aoc.Order\_number  And  aoc.Customer\_name = acy.Customer\_name  And  aoc.Contact\_name = acy.Contact\_name  And  aoc.Role\_type = 1 (for primary contact)  **Note:**  This step is used for either  ~~When the input is ATS Room, to retrieve the equipments associated to the input ATS room record~~  ~~Or~~  When the input is ATS equipment, to retrieve the equipment record matching the input |
| customerAssetAliasName |  |  | See Step ATS-3 |

**Step ATS-3:** Retrieve customerAssetAliasName for the ATS room or the Room Equipment records retrieved above

**Customer Asset Alias retrieval processing rules for ATS.ATS\_ROOM\_ACTIVE\_VW, ATS.ATS\_ROOM\_EQUIPMENT\_VW records**

**Existing GCP logic implemented in project 231615.**

Extend the existing processing logic and lookup an optionally existing customer defined asset ticket alias value (as developed in project 231615) for each ATS.ATS\_ROOM\_ACTIVE\_VW or ATS.ATS\_ROOM\_EQUIPMENT\_VW record being the basis for the response:

1. Retrieve ATS.ATS\_ROOM\_ACTIVE\_VW.room\_id for room alias. <240520.68197> This field is only populated for ATS Room record, not for ATS equipment, for this project
2. Use the following database constraints to lookup the alias value

**Retrieve Customer Asset Alias for ATS Room record:**

|  |  |  |
| --- | --- | --- |
| **Table name** | **Column name** | **Containt(s)** |
| META\_SYSTEM | NAME | ‘ATS’ |
| META\_TABLE | NAME | 'ATS\_ROOM\_ACTIVE\_VW' |
| META\_COLUMN | NAME | 'ROOM\_ID' – for room alias name |
| SOURCE\_KEY\_VALUE | SOURCE\_VALUE | ATS\_ROOM\_ACTIVE\_VW.room\_id |

**Retrieve alias for ATS Room Equipment record:**

<240520.68197> This CustomerAssetAliasName field is only populated for ATS Room record, not for ATS equipment, for this project

|  |  |  |
| --- | --- | --- |
| **Table name** | **Column name** | **Containt(s)** |
| META\_SYSTEM | NAME | ‘ATS’ |
| META\_TABLE | NAME | 'ATS\_ROOM\_EQUIPMENT\_VW' |
| META\_COLUMN | NAME | ‘EQUIPMENT\_ID’ – for equipment alias name |
| SOURCE\_KEY\_VALUE | SOURCE\_VALUE | ATS\_ROOM\_EQUIPMENT\_VW.equipment\_id |

**</240520>**

<265609>

|  |  |
| --- | --- |
| **Steps** | **Description** |
| 100.1 | <265609> Retrieve Multi-Hop Loopback IP Address from INSTAR Web Service  **Input:** <PVC\_ID>retrieved in earlier step  **From ICORE DB, get INSTAR port assignment ID:**  ICORE: pvc\_ID -> ICORE.map\_instar\_port.Instar\_port\_asgmt AS <ip\_port\_asgmt\_id>  Select m.instar\_port\_asgmt  AS **<ip\_port\_asgmt\_id>**  From pvc p, map\_instar\_port m  Where p.pvc\_id = <pvc\_id>  And pvc\_stat = 'IN-SERVICE'  And m.pvc\_id = P.pvc\_id   1. NSTAR WSDL/XSD links   WSDL:  <http://int-lpp.oss.att.com:9096/lppbck-service/ws/serviceQueryLogicalConnectionsV1?wsdl>  XSD:  <http://int-lpp.oss.att.com:9096/lppbck-service/ws/serviceQueryLogicalConnectionsV1?xsd=LPP_ServiceQueryLogicalConnections_Messages.xsd>   1. Calling getLayer3DataRequest with:   ConnectionId = <ip\_port\_asgmt\_id>  ConnectionType = “PortAssignmentID”   1. Populate with the value of MultiHop.LoopbackAddress.IpAddress |
|  | **<265609>**  **Derive isInternetVlan field**  Retrieve CERIpAddress (IPv4/6) for AVPN and Internet VLAN |
| 100.3 | **<265609> Derive isInternetVlan field:**  **Input:** <PVC-ID> retrieved in earlier steps  **From ICORE DB** (two new tables added for Internet VLAN feature):  ICORE: pvc\_ID -> Conn\_param -> Service\_param.param\_name  Set isInternetVlan = ‘N’  Select sp.Param\_name  From Service\_param sp,  Conn\_param cp  Where cp.Pvc\_id = **<PVC-ID>**  And sp.service\_param\_id = cp.service\_param\_id  And sp.param\_name = ‘Internet VLAN’  If record retrieved, then  Set isInternetVlan = ‘Y’  End if  Populating isInternetVlan to the output field |
| 100.2 | **<265609> Retrieve Internet CER IP Address (IPv4/6) field – for ~~VLAN or~~ Internet VLAN:**  **Input:** <PVC-ID> retrieved in earlier steps  <ip\_port\_asgmt\_id> from step 100.1  **If isInternetVlan = Y: get CER IP Address (IPv4/6) fields from INSTAR DB:**  **From INSTAR:**  <ip\_port\_asgmt\_id> -> ip\_port\_asgmt.cr\_addr\_id -> serial\_ip\_addr.ip\_address – CER IPv4  ip\_port\_asgmt.sdid -> ipv6\_port\_asgmt\_map -> ipv6\_assigned\_link\_ips.ipv6\_address\_compressed  **From INSTAR: Retrieve CER IPv4:**  Select sia.ip\_address AS cerIpv4Address  From ip\_port\_asgmt ipa,  serial\_ip\_addr sia  Where ipa.ip\_port\_asgmt\_id = **<ip\_port\_asgmt\_id>**  And ipa.cr\_addr\_id = sia.serial\_ip\_addr\_id (+)  **From INSTAR: Retrieve CER IPv6:**  Select ialp.ipv6\_address\_compredd AS cerIpv6Address  From ip\_port\_asgmt ipa,  Ipv6\_port\_asgmt\_map ipam,  Ipv6\_assigned\_link\_ips ialp  Where ipa.ip\_port\_asgmt\_id = **<ip\_port\_asgmt\_id>**  And ipa.sdid = ipam.sdid  Andipam.ipv6\_link\_ip\_id = ialp.ipv6\_link\_ip\_id  And ipam.ipv6\_address\_type\_id = ‘10’  **Populate internetCerIpv4/v6Address fields.** |
| 100.3 | **<265609> To populate lanIpAddressInfo, retrieve LAN IP Address (IPv4/6) fields – for Internet VLAN:**  **Input:** isInternetVlan = Y  <ip\_port\_asgmt\_id>retrieved in earlier step – 100.2  **Data Source:** INSTAR  **Getting LAN IP – IPv4**  The IP values here included subnet, as shown below ().   |  | | --- | | **IP** | | 12.32.93.128.27 | | 12.32.93.160.28 | | 199.105.98.0.23 | | 199.105.100.0.24 | | 12.9.149.0.25 |   GCP needs to replace the last '.' by '/', so the IP field value will look as this:   |  | | --- | | **IP** | | 12.32.93.128/27 | | 12.32.93.160/28 | | 199.105.98.0/23 | | 199.105.100.0/24 | | 12.9.149.0/25 |   The logic is to get the position X of the fourth occurance of '.' in the IP value, the take the subset from the first character to the Xth, and concatenated '/' and the value from Slash field.  Select Unique  Substr (a.ip, 1, Instr(a.ip, '.',1,4)-1) || '/' || a.slash AS lanIpAddress,  '' AS lanIpv6Subnet,  Decode (assign\_ip.portable,  'A', 'N',  'S', 'N',  'Y') isCustomerProvided  From  assign\_ip,  connection,  rc\_acl  Where (1=1)  and rc\_acl.portassignment = <instar\_port\_asgmt>  and rc\_acl.conn\_id = connection.conn\_id  and connection.network\_id = assign\_ip.network\_id  **Getting LAN IP – IPv6**  Select Unique  ip.ipv6\_ip AS lanIpAddress,  subnet.subnet\_ip AS lanIpv6Subnet,  Decode (Instr(at.ipv6\_address\_type, 'CUST'),  0, 'N',  'Y') isCustomerProvided  from  ipv6\_lan\_port\_map pm,  ipv6\_address\_type at,  ipv6\_assign\_ip ip,  ip\_port\_asgmt pa,  ipv6\_assigned\_subnets subnet  where (1=1)  and pa.ip\_port\_asgmt\_id = <ip\_port\_asgmt\_id>  and pm.address\_type\_id = at.ipv6\_ address\_type\_id  and pm.ipv6\_ip\_id = ip.ipv6\_ip\_id  and pa.sdid = pm.sdid  and ip.subnet\_id = subnet.subnet\_id (+)  **Populate the lanIpAddressInfo structure with the data retrieved.** |

**Note: If no data found, will return “Data Not Found” in the response.**

**15642ah:**

**Data Access Logic**

BAU fields can be retrieved first (note: input key logic may need update based on assettagtype). If isManagedCare is derived

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **GCP WS tag name**  **Or intermediate step instruction** | **GPS or SIDBOR Table Name** | **GPS or SIDBOR Attribute Name** | **Condition for selection/instruction** | **Note** |
| 1 | Note the given  <assettagtype> |  |  |  | This input field will indicate what kind of assettag is sent by AOTS; valid values include: NAME, ID |
| 2 | AssetName (also known as ASSET\_ID in AOTS; appears on AOTS GUI) | SIDBOR.asset\_details | asset\_nm | If input value from assettagtype = ‘NAME’  then  AssetName = <assettag from input>  Else if assettagtype =’ID’ then  select AssetName = sidbor.asset\_details.asset\_name  where asset\_id = <ssettag from input> | Asset names are stored in the SIDBOR.asset.asset\_nm field regadless of what kind of asset it is. In GPS, assets are stored in different tables, depending on asset type. |
| 3 | Asset\_id  (also known as INTERNAL\_ASSET\_ID in AOTS; is hidden on AOTS GUI) | SIDBOR.asset\_details | asset\_id | If input value from assettagtype = ‘ID’ then Asset\_id = <ssettag from input>  Else if assettagtype =’NAME’ then  Select asset\_id = SIDBOR. Asset\_details.asset\_id where asset\_nm = <assettag from input> |  |
| 3.1 | ***CR 80126:*** | SIDBOR.asset | ASSET\_SRC\_CD, FUNCTIONAL\_AREA | IF FUNCTIONAL\_AREA in <*MCFAS list*> and ASSET\_SRC\_CD = ‘GPS’ THEN: continue with GPS field retrieval; ELSE: set isManagedCare = ‘N’ (if ‘N’,  do not need to evaluate further managed care functionality) |  |
| 3.2 | ***Identify link from SIDBOR to GPS***  ***-- source table in GPS is dependant on the asset type: circuit source is at\_access\_circuit, other type is sw\_inst\_product*** | SIDBOR.asset | ASSET\_CLASS\_CD | If SIDBOR.asset.ASSET\_CLASS\_CD = ‘CIRCUIT’ then the link SIDBOR 🡪 GPS is to the at\_access\_circuit. The field SIDBOR.asset.SRC\_ASSET\_ID stores the <at\_asset\_circuit. ATACCESSCIRCUITID || ‘A’> (note the ‘A’ for at\_**A**sset\_circuit.) Trim the ‘A’ off the SRC\_ASSET\_ID field, and link to at\_asset\_circuit. ATACCESSCIRCUITID.  Else (telephone, equipment) the link SIDBOR🡪GPS is SIDBOR.asset.asset\_nm=GPS.SW\_INST\_PRODUCT.atspecassetname | In GPS, the circuits are stored in at\_access\_circuit.  In GPS, all equipment and telephone assets are stored on the sw\_inst\_product table  Example: src\_asset\_id = ‘227603A’, trim off A, and primary key of at\_access\_circuit=227603  **Note: TBD – may be enough to link asset.asset\_nm=gps.at\_access\_circuit.atcircuitname** |
| 3.3 | ***Identifying the sw\_inst\_product record associated to the input asset*** | Sw\_inst\_product | swInstProdID | For a record in the AT\_ACCESS\_CIRCUIT, the link to the sw\_inst\_product table is: sw\_inst\_product.swinstprodid = AT\_CHANNEL\_PORT.atassetid\_a and   AT\_CHANNEL\_PORT.ataccesscircuitid = AT\_ACCESS\_CIRCUIT.ataccesscircuitid  For all other assets, use swinstprodid for the atspecassetname | Note 1: the link to the sw\_inst\_product tables from the circuit tables is the device which the circuit connects to.  Note 2: all circuits actually attach to two endpoints. The first endpoint is atassetid\_a, usually the device on the ATT side, the second endpoint is atassetid\_z, usually the device on the client side. AOTS has stated that GCP should return the client side device. |
| 3.4 | isManagedCare | Sw\_agreement A,  Sw\_inst\_product B, | If(atmgdvoicecontracttype)=’OS’ Then set isManagedCare=’Y’, else setisManagedCarey=’N’ | A.atnetworkid=B.atnetworkid where A.atspecassetname=asset\_nm from step 2  If isManagedCare = ‘N’ then can exit/ignore further managed care logic (applies only to isManaged = ‘Y’) |  |
| 3.5 | assettype | derived |  | If sidbor.asset.asset\_class\_cd in ('CIRCUIT'~~, 'PVC'~~) then assettype = 1 (see note at right)  Else  *//we need to derive from GPS*  If sw\_prod\_release.swsubtype <> 'Voice Adjunct Equipment'  And sw\_prod\_release.atassetsubclass <> 'Station'  Where  Sw\_inst\_product.atspecassetname=<assetName>  and  Sw\_inst\_poduct.swprodreleaseid =sw\_prod\_release.swprodreleaseid  *//then we have equipment*  Assettype=0 (see note at right)  Else  *//we need to derive from GPS*  If sw\_prod\_release.swsubtype = 'Voice Adjunct Equipment'  And sw\_prod\_release. atassetsubclass = 'Station'  Where  Sw\_inst\_poduct.atspecassetname=<assetName>  and  Sw\_inst\_poduct.swprodreleaseid =sw\_prod\_release.swprodreleaseid  *//then we have telephone*  Assettype=4 (see note at right) | From Jennifer Eng (AOTS T3):  Karen,   Just an FYI - Per AOTS Dev SME  (Venkat Gonna), below is the current BAU AOTS logic in regard to the output Asset Type in getAssetDetails:  If  GCP sends a value of 0,  then AOTS changes it to 1 (Equipment).  If  GCP sends a value of 1, then AOTS changes it to 0 (Circuit).  If GCP sends any other value, then AOTS will store as is.  So in the case of Telephone asset, GCP will send a value of 4, and AOTS will store as 4 (Telephone). |
| 4 | Floor,  Room,  Cube,  Jack | Sw\_inst\_product A | A.atfloor,  A.atroom,  A.atcube  A.atjack |  |  |
| 5 | ServiceRestriction | SW\_CUSTOMER C | ATSERVICERESTRICTION, | A.SWCUSTOMERID = C.SWCUSTOMERID |  |
| 5.1 | attHelpDeskNumber | SW\_PERSON | SW\_PERSON. swOfficephonecntry|| ‘ ‘ || swOfficephonearea || ‘ ‘ swofficephone || ‘ ’ || swOfficephoneext | SW\_inst\_prod.swsiteid=sw\_site.swsiteid  AND  Sw\_person.swpersonid = at\_contact\_loc\_role.atcontactid  AND at\_contact\_loc\_role.atsiteid = sw\_site.swsiteid | Changes for R3S1400007205 |
| 6 | SupportingNetwork | At\_access\_circuit | Atsupportingnetwork | AT\_ACCESS\_CIRCUIT.ATSUPPORTINGNETWORK | Supporting Network  NEW  Valid values:  B, S, T  Note: this field will be retrieved from two different locations based on circuit type |
| 7 | AccessProviderCircuitId, | At\_access\_circuit AC | ATSUPPORTINGNETWORK, ATCIRCUIT\_NAME | A.swinstprodid = AT\_channel\_port.atassetid\_a  and  AT\_channel\_port.ataccesscircuitid = AC.ataccesscircuitid | A Channel Port is associated to an Access Circuit |
| 7.1 | attCircuitId | At\_circuit\_base B | ATCIRCUIT\_NAME | A.SWINSTPRODID = B.ATASSETID\_A |  |
| 8 | LocationName | At\_attribute\_value | ATTEXTVALUE | SELECT DISTINCT  atv.attextvalue, atv.swobjectid, atv.SWOBJECTTYPE, ata.atattributecode FROM at\_attribute\_values atv, sw\_site site, sw\_inst\_product prod, at\_attributes ata, sw\_customer swcustomer WHERE prod.swsiteid = site.swsiteid AND site.swsiteid = atv.swobjectid AND atv.swobjecttype = 'SITE' AND ata.atattributesid IN (SELECT ata.atattributesid FROM at\_attributes ata WHERE ata.atattributecode = 'MVS\_SITE\_LOCATION\_NAME') AND prod.swcustomerid = swcustomer.swcustomerid AND swcustomer.atgemsscid = 'MGDVCESVCS' AND prod.atspecassetname IN <Asset ID from input or derived from input> AND ata.ATATTRIBUTESID =atv.ATATTRIBUTESID;    If no data above is returned, use  SW\_SITE.SWSITENAME  where  sw\_inst\_product.SWSITEID = sw\_site.SWSITEID and sw\_inst\_product.atspecassetname = <input asset or derived from input asset> | Trim the LocationName to 60 characters |
| 9 | CustomerPONumber,  IndividualInvoice | SW\_SERVICE\_ORDER SR,  Sw\_inst\_product PROD | ATCUSTPONBR,  atindividualinvoice | select  SR.atcustponbr,  SR.atinvoiceindicator  from  sw\_service\_order SR,  sw\_inst\_product PROD where PROD.atspecassetname = <input asset or derived from input asset> and nvl(PROD.atmodifyingserviceorderid, PROD.atcreatingserviceorderid) = OL.swserviceorderid |  |

Common logic – getEISContractType <284465h>

**Input:** <input-mcn>, <input-grc>, <input-soc>, <input-routerName>

**Output:** contractType

**Data Source:** EDF: eis\_mcn\_list, eis\_mcn\_invtry

|  |  |
| --- | --- |
| **Attribute Name** | **Comments** |
| contractType | Input: <input-mcn>, <input-grc>, <input-soc>, <input-routerName>  **Determine EIS as contractType:**  For given MCN and GRC, retrieve from these two tables.  If data is retrieved, then set contractType = EIC;  Else (no data returned), keep the existing contractType’s value  Select Unique ‘EIS’ AS contractType  From EIS\_mcn\_list  Where 1=1  And mcn = <input-mcn>  And grc = <input-grc> --If provided  And so = <input-soc> --If provided  And Rownum < 2  Union  Select Unique ‘EIS’ AS contractType  From EIS\_mcn\_invtry  Where 1=1  And ((mcn = <input-mcn> --If provided  And grc = <input-grc> --If provided  And so = <input-soc> --If provided  )  Or bvoip\_router\_id = <input-routerName> --If provided  )  And Rownum < 2 |

## Alternative Designs

N/A.

No alternative designs will be considered beyond the design already in production.

## Assumptions/Risks

N/A – Nothing changed here

## Traceability Matrix

|  |  |
| --- | --- |
| Requirement ID | Design Element Identifier |
|  |  |

## Pre-Production Disaster Recovery Planning

N/A – Nothing changed here

## Other Plans and References

N/A – Nothing changed here

|  |
| --- |
|  |

## 

## Acceptance & Approvals

Overview

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

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTUID and Name** | **Role** | **Group/Application** | **Version Approved, Approval Date and Approval Evidence** |
| mh6892 – Ming Ho | Artifact Owner - Application Architect | GCP | see PRISM for evidence of approval |
|  | Artifact Contributor | GCP  Note: xyz is approving for both the GCP Dev Mgr and GCP Test Mgr | see PRISM for evidence of approval |
|  | Lead PM |  | see PRISM for evidence of approval |
|  |  |  |  |

Appendix