

Ocean Tracking Network

Data Workflow and Data Model Detailed

December 2010

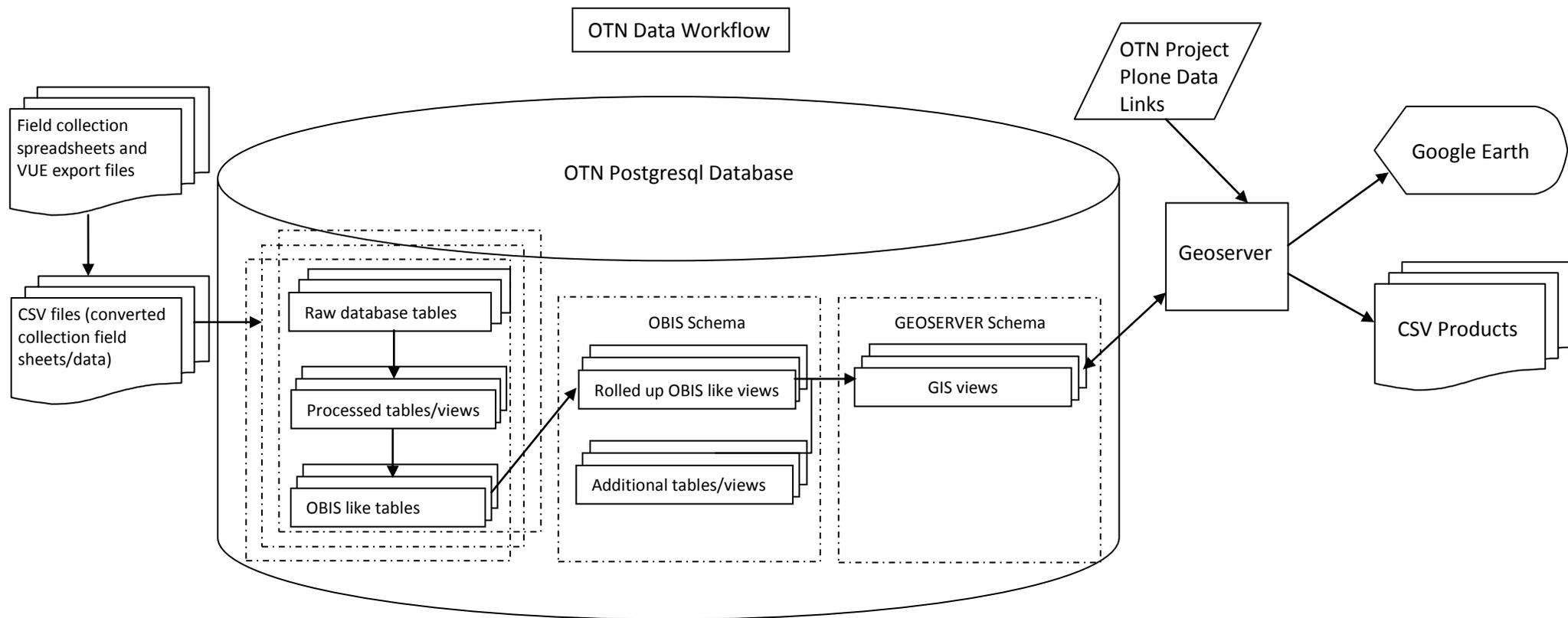
This document is an extension of the OTN Data Workflow [presentation](#) (Nov. 4, 2010), providing more detailed information for both OTN data workflow and database schema (via SQL scripts). As ERDs would be unwieldy they are not included.

As briefly outlined in the presentation, each OTN Collaborator's data is stored in a separate schema within the OTN Postgresql database. OTN Collaborators can be either Deployment Collaborator (involved in the deployment, maintenance, data collection of OTN CFI funded receivers and/or in some instances OTN Canada funded acoustic tags), Tracker Collaborator (involved in the tagging of animals with acoustic tags and providing the field collection metadata sheets to OTN via OTN Collection Sheets), or Data Collaborator (involved in deployment of acoustic receivers and/or tagging animal with acoustic/satellite tags and providing the field collection metadata and data to OTN, not via OTN Collection Sheets).

Presented here are all of the field collection metadata and data sheets which would reflect the workflow and data for a collaborator involved in both receiver deployments and animal tagging (via OTN Collection Sheets). As such, some collaborators will only be involved with the subset related to their data involvement (e.g. Tracker Collaborator would only include Mission Reports and Tagging Collection Sheets).

The workflow and associate SQL provided in this document represent the template standard scripting and data model. Occasional modifications are performed where field collection metadata and/or data do not conform to the expected standard. Also, as this is still a work in progress, some information has not been provided, including but not limited to handling of BCT/VMT. Finally, in the interest of brevity this document does not include the information regarding corrections to VUE events and/or detections nor the mapping of Data Collaborator received data from collaborator's formats/data model to OTN formats/data model.

Detailed information including tasks excluded from this document will be included in the Developer's Guide to OTN Data Loading and Processing which is currently being completed. As the initial workflow and data model were developed using manual steps, the system is currently being automated and the Developer's Guide will also include these automation processes. The Developer's Guide will be included in the Complete OTN Workflow (from initial collaborator contact, to data acquisition, to data processing/loading, to data access and distribution) documentation which is also currently under development. When the future data access and availability have been implemented a separate document will also be developed and distributed.

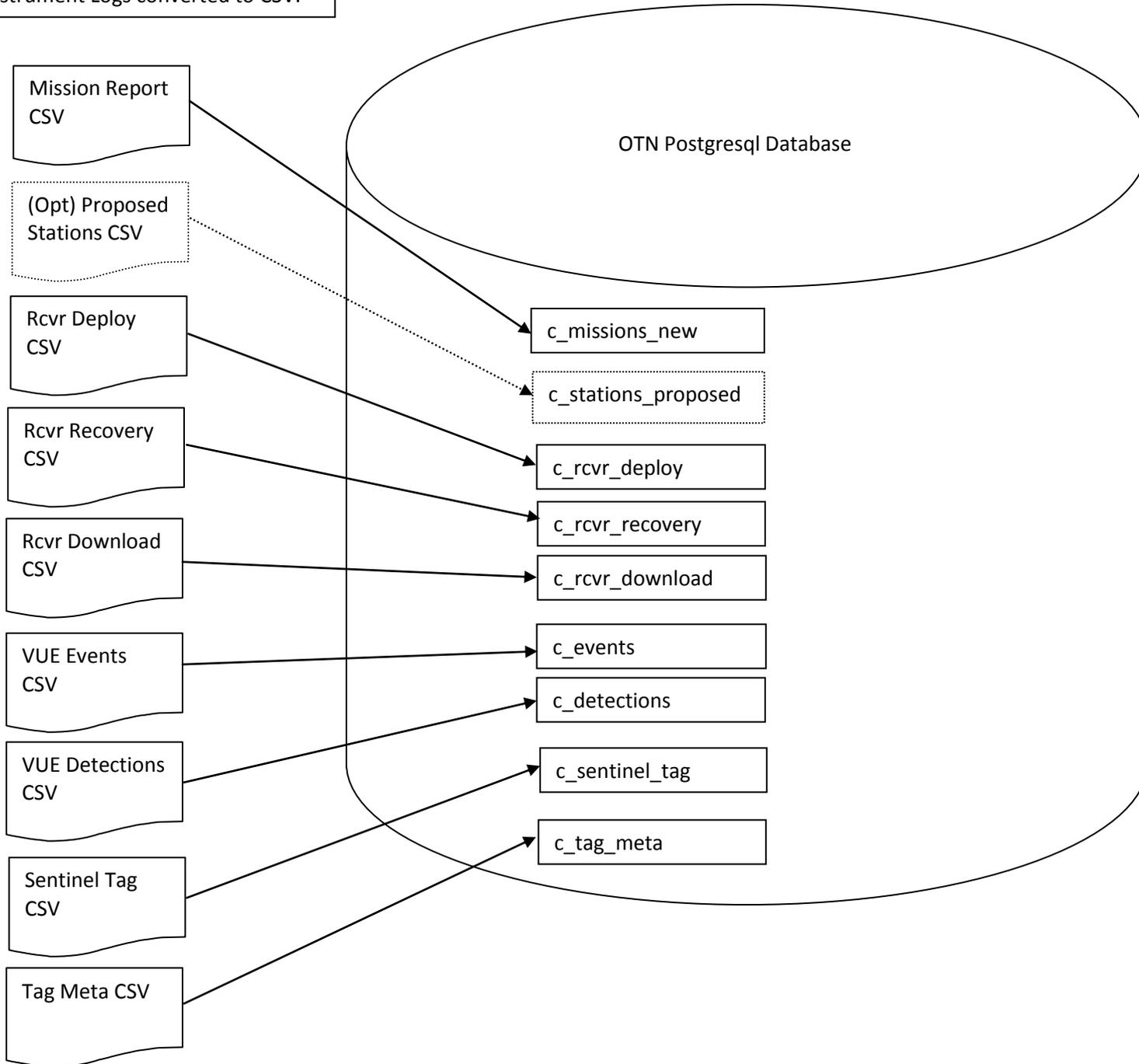


Individual collaborator's data are kept in their own password protected PostgreSQL schema or userid. Major processing steps include:

1. Field collection spreadsheets and VUE export files are converted to CSV files and loaded "as is" to raw database tables (for specific collaborator schema);
2. Raw database tables are loaded into processed (mostly type casting) tables and then views (for specific collaborator schema).
3. Processed tables/views are loaded into OBIS like tables (for specific collaborator schema).
 - o Then collaborator schemas are rolled up into OBIS Schema views.
 - o Additional OBIS schema tables/views for metadata and internal use.
4. OBIS schema views are then used to create GEOSERVER schema views for use in creating Geoserver Layers.
5. The Geoserver Layers (Store otn) are then created using the OTN Postgresql database GEOSERVER Schema views.
 - o Only two of the Geoserver Layers (resources_points, moorings) may be accessed without a valid Geoserver username/password.
6. OTN Project Plone Data folder includes publically accessible specially constructed KML/KMZ and OTN Site Members only access to URLs under folder CSV product links for some of the OTN Geoserver Layers (only otn:resource_points and otn:moorings are accessible without a Geoserver username/password).

1. CSV files load "as is" to raw database tables (for specific collaborator schema).

1. Collection Metadata and Instrument Logs converted to CSV.



1. Loading Collaborator Field Metadata and Data to OTN Postgres Database

The “raw” data, i.e. field collection spreadsheets and VUE export files are loaded to postgres database otn (all are converted to CSV and loaded into postgresql database “as is” using pSQL's /copy command), raw data tables, each collaborator’s data being kept in a separate schema. Corrections are made if there are any time related issues associated with the VUE event records and/or issues with VUE detection records.

All tables with the prefix “c_” (raw data tables) will hold data without enforcing any sort of strict type casting. Most “c_” tables have columns with the type “character varying”. Character varying columns are chosen because they incur the least amount of type conflicts when importing data.

Mission Report collection sheet (<http://www.marinebiodiversity.ca/OTN/data/data-collection/data-sheet-templates/otn-mission-report.xls/view>) loaded “as is” to [c_missions_new](#).

(The optional) Proposed Station Locations collection sheet (<http://www.marinebiodiversity.ca/OTN/data/data-collection/data-sheet-templates/station-locations-v2.0.xls/view>) is loaded “as is” to table [c_stations_proposed](#).

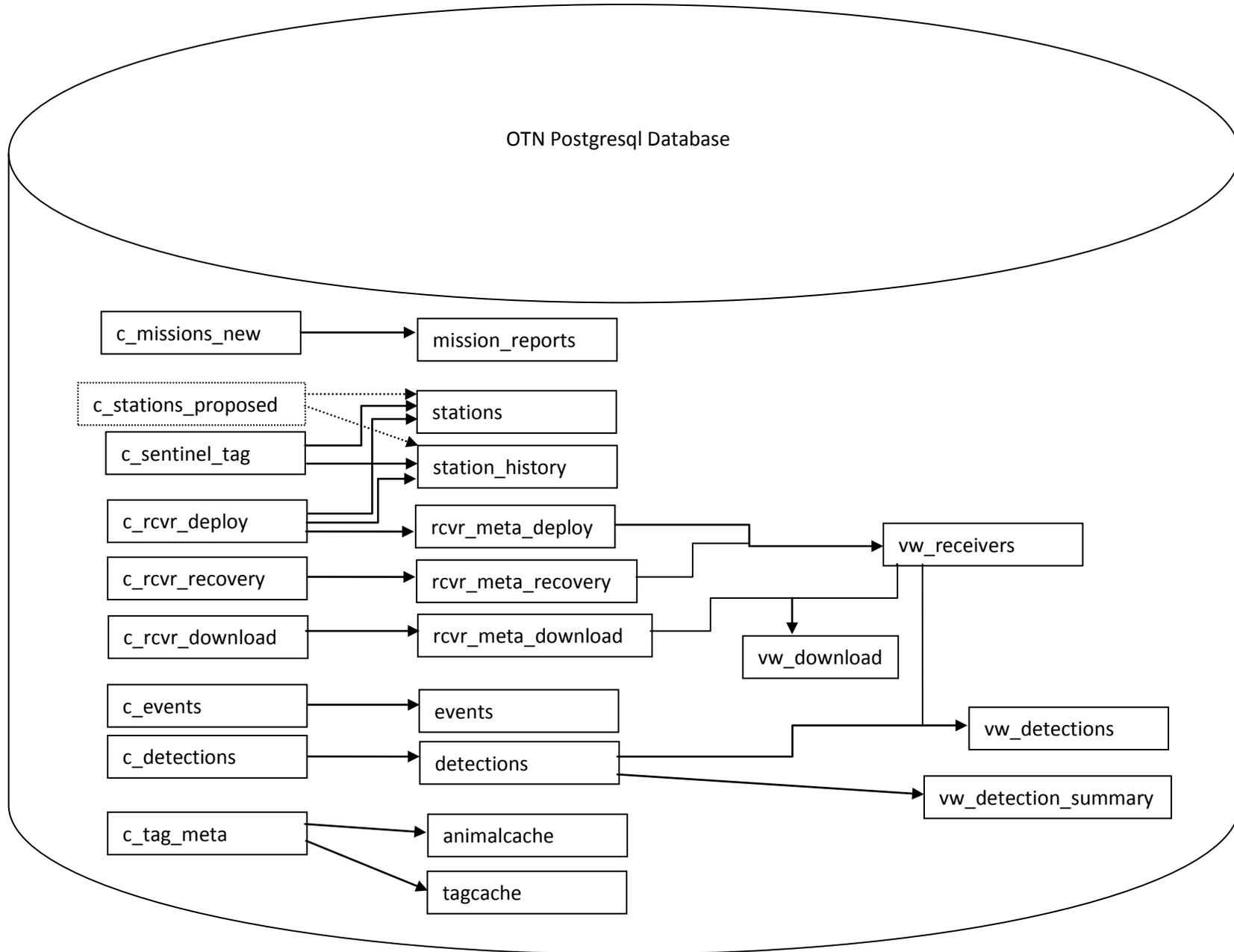
Receiver Collection Sheet (<http://www.marinebiodiversity.ca/OTN/data/data-collection/data-sheet-templates/otn-metadata-receiver.xls/view>) three worksheets are loaded “as is” to tables [c_rcvr_deploy](#), [c_rcvr_recovery](#), and [c_rcvr_download](#).

VUE export files for events and detections are loaded “as is” to tables [c_events](#) and [c_detections](#).

Sentinel Tag collection sheet (<http://www.marinebiodiversity.ca/OTN/data/data-collection/data-sheet-templates/stag-logsheet/view>) is loaded “as is” to [c_sentinel_tag](#).

Tagging Collection Sheet (http://www.marinebiodiversity.ca/OTN/data/data-collection/data-sheet-templates/otn_metadata_tagging.xls/view) is loaded “as is” to [c_tag_meta](#).

2. Raw database tables are loaded into processed (mostly type casting) tables and then views (for specific collaborator schema).



2. The raw database tables are then loaded into processed (mostly type casting) tables and from these views are created.

TABLES

Table c_missions_new is loaded into table [mission_reports](#).

(Optional) Table c_stations_proposed is loaded into tables [stations and station_history](#).

Tables c_rcvr_deploy, c_rcvr_recovery and c_rcvr_download are loaded into tables [rcvr_meta_deploy, rcvr_meta_recovery and rcvr_meta_download](#). Also, distinct stations are loaded from c_rcvr_deploy to table stations and distinct stations, intendedlat, intendedlon are loaded from c_rcvr_deploy to table station_history.

Table c_events is loaded into table [events](#). Also subset extracted to table [pc_time](#).

Table c_detections is loaded into table [detections](#), adding field detection_key (sequence generated).

Distinct (sentinel) stations are loaded from c_sentinel_tag to table stations. Distinct (sentinel) stations, intendedlat, intendedlon are loaded from c_sentinel_tag to table station_history ([create_stationsstation_history.sql](#)).

Table c_tag_meta is loaded into obis like tables (obis field assignments are noted in the next section under obis_detailcache, though are implemented during load to) [tagcache and animalcache](#).

VIEWS

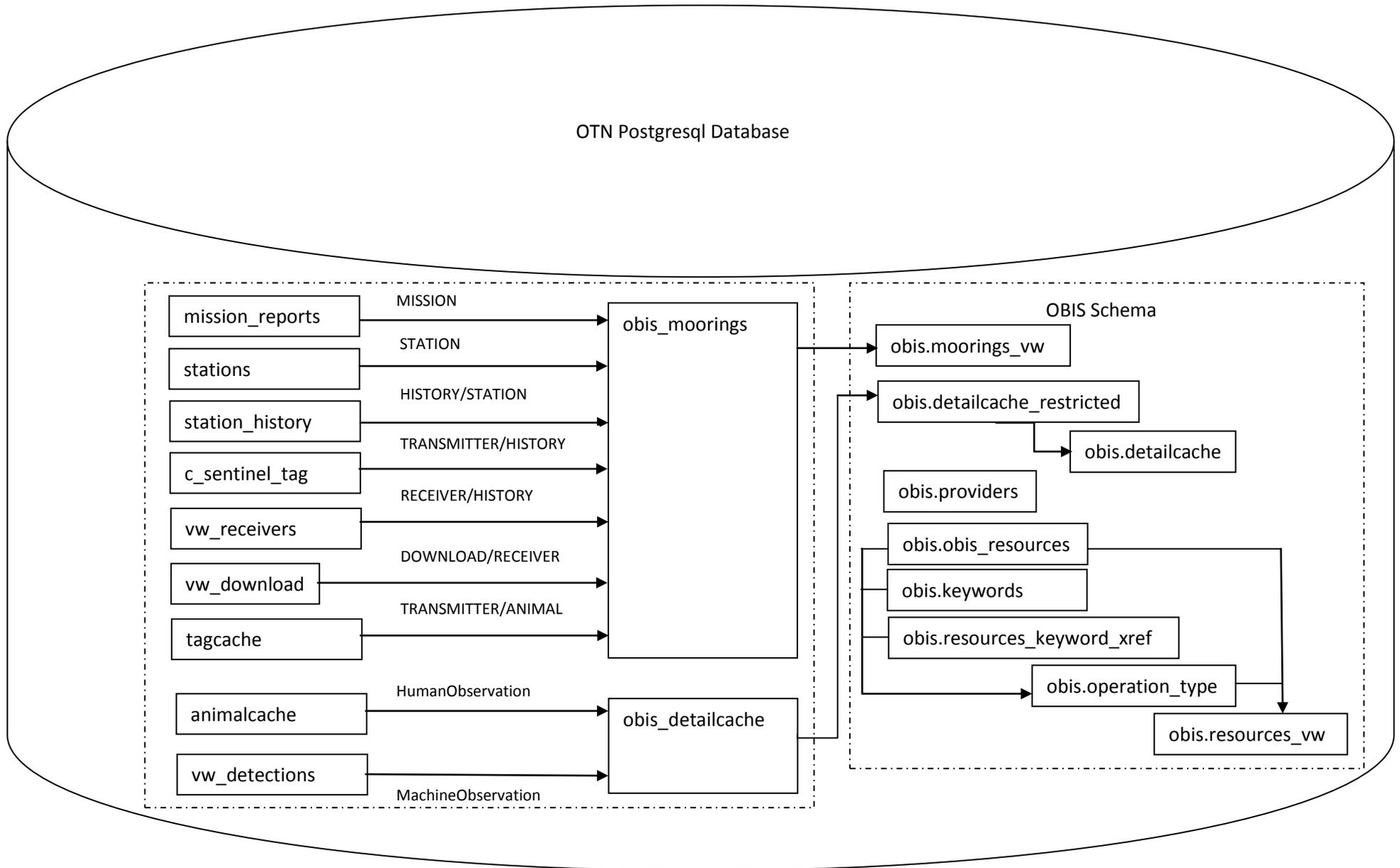
Tables rcvr_meta_deploy and rcvr_meta_recovery are joined where array, station_no, consecutive_deploy are equal and recover_date_time is not NULL and recover_ind=y to create view [vw_receivers](#). And field gis_point is also created for later use in Geoserver.

View vw_receivers and table rcvr_meta_download are joined where array, station_no, model, serial_no, consecutive_deploy are equal creating view [vw_download](#).

Table detections and view vw_receivers are joined where rcvr_serial_no are equal and detection_date between deploy and recovery dates to create view [vw_detections](#). And field the_geom is also created for later use in Geoserver.

Table detections excluding sentinel tags is used to create view [vw_detection_summary](#), providing count by year/tag (code_space, id).

3. Processed tables/views are loaded into OBIS like tables (for specific collaborator schema). Then collaborator schemas are rolled up into OBIS Schema views. Additional OBIS schema tables/views for metadata and internal use.



3. Processed tables are mapped into OBIS like schema using BasisofRecord, RelationshipType and RelatedCatalogItem for distinguishing data types and linking (ex. Receivers to Stations, Animals to Tags/Transmitters).

OBIS_MOORINGS ([create obis moorings.sql](#))

(missions, stations, receivers, downloads, tags/transmitters)

Tables mission_reports and obis.obis_resources are joined (to extract resource's ocean, country, state, county) and loaded into obis_moorings.

BasisofRecord=MISSION,

CatalogNumber= otn_mission_id,

Locality= destination,

FieldNumber= NULL ,

RelationshipType= NULL,

RelatedCatalogItem= NULL

Tables stations and obis.obis_resources are joined (to extract resource's locality, ocean, country, state, county) and loaded into obis_moorings.

BasisofRecord=STATION,

CatalogNumber= station_name,

Locality= from obis.obis_resources,

FieldNumber= NULL ,

RelationshipType=NULL,

RelatedCatalogItem= NULL

Tables station_history and obis.obis_resources are joined (to extract resource's locality, ocean, country, state, county) and loaded into obis_moorings adding field the_geom for later use by Geoserver.

BasisofRecord=HISTORY,

CatalogNumber= station_name || '-' || id,

Locality= from obis.obis_resources,

FieldNumber= NULL ,

RelationshipType=STATION,

RelatedCatalogItem= station_name

Tables c_sentinel_tag and obis.obis_resources are joined (to extract resource's ocean, country, state, county) and loaded into obis_moorings adding field the_geom for later use by Geoserver.

BasisofRecord=TRANSMITTER,

CatalogNumber= model||'-'||serial_number||'-'||VUE_ID||'-'||consecutive_deploy_no,
Locality= 'STATIONS: '||upper(nearby_stations),
FieldNumber= vue_id ,
RelationshipType=HISTORY,
RelatedCatalogItem= station_name

View vw_receivers is loaded into obis_moorings adding field the_geom for later use by Geoserver.

BasisofRecord=RECEIVER,
CatalogNumber= otn_array||'-'|| station_no ||'-'|| rcv_model_no||'-'|| rcv_serial_no||'-'||consecutive_deploy_no,
Locality= 'LINE: {RESOURCE}-L{LINE_NUM}, STATION: '|| otn_array||'-'||station_no,
FieldNumber=rcv_model_no||'-'||rcv_serial_no ,
RelationshipType=HISTORY,
RelatedCatalogItem=otn_array||'-'||station_no

Distinct station_no, otn_mission_id,rcv_serial_no are loaded from view vw_download into obis_moorings.

BasisofRecord = DOWNLOAD,
CatalogNumber = rcv_model_no||'-'||rcv_serial_no||'-'||substring(date, 1,10) ,
Locality= LINE: {RESOURCE}-L1, STATION: '||otn_array||'-'||station_no ,
FieldNumber=filename|NO DETECTIONS,
RelationshipType=RECEIVER,
RelatedCatalogItem= otn_array||'-'||station_no ||'-'||rcv_model_no||'-'||rcv_serial_no ||'-'||consecutive_deploy_no

Tables tagcache and obis.obis_resources are joined (to extract resource's ocean, country, state, county) and loaded into obis_moorings.

BasisofRecord=TRANSMITTER,
CatalogNumber= tag_id_code,
Locality= release_location,
FieldNumber= tag_code_space||'-'||tag_id_code,
*RelationshipType= ANIMAL,
RelatedCatalogItem= resource_acronym||'-'||tag_id_code

*RelationshipType=ANIMAL means the "ANIMAL" record is found int OBIS_DETAILCACHE with BasisofRecord=HumanObservation.

OBIS_DETAILCACHE ([create obis_detailcache.sql](#))
(tagged animals, detections)

Tables animalcache and obis.obis_resources are joined (to extract resource's ocean, country, state, county) and loaded into obis_detailcache, adding field the_geom for later use by Geoserver.

BasisofRecord= HumanObservation,
CatalogNumber= resource_acronym||'-'||tag_id_code,
Locality=release_location,
FieldNumber= tag_id_code,
RelationshipType=NULL,
RelatedCatalogItem=NULL

View vw_detections and table obis.obis_resources are joined (to extract resource's ocean, country, state, county) and loaded into obis_detailcache, adding field the_geom for later use by Geoserver.

BasisofRecord= MachineObservation,
CatalogNumber= code_space||'-'||transmitter_id||'-'||detection_key,
Locality= station_name,
FieldNumber= transmitter_name,
**RelationshipType= UNQUALIFIED,
RelatedCatalogItem=NULL

**RelationshipType will later be set to one of *ANIMAL|TEST|TRANSMITTER (sentinel tag) and appropriate RelatedCatalogItem will also be set.

OBIS SCHEMA (union from collaborator schemas)

Three views are created in the OBIS schema:

View obis.moorings_vw is a union of the individual collaborator schema's obis_moorings table.

View obis.detailcache_restricted is a union of the individual collaborator schema's obis_detailcache table.

View obis.detailcache is subset of obis.detailcache including only those schema that are public/not restricted. Also includes grant select to logged_in_users.

Additionally, the OBIS Schema contains tables and views for collaborators' metadata as well as lookup tables for OTN internally created/used information.

Table obis.providers (public and logged_in_users) contains OTN list of (data) providers.

Table obis.obis_resources (logged_in_users) contains collaborator/resource metadata (including provider_id).

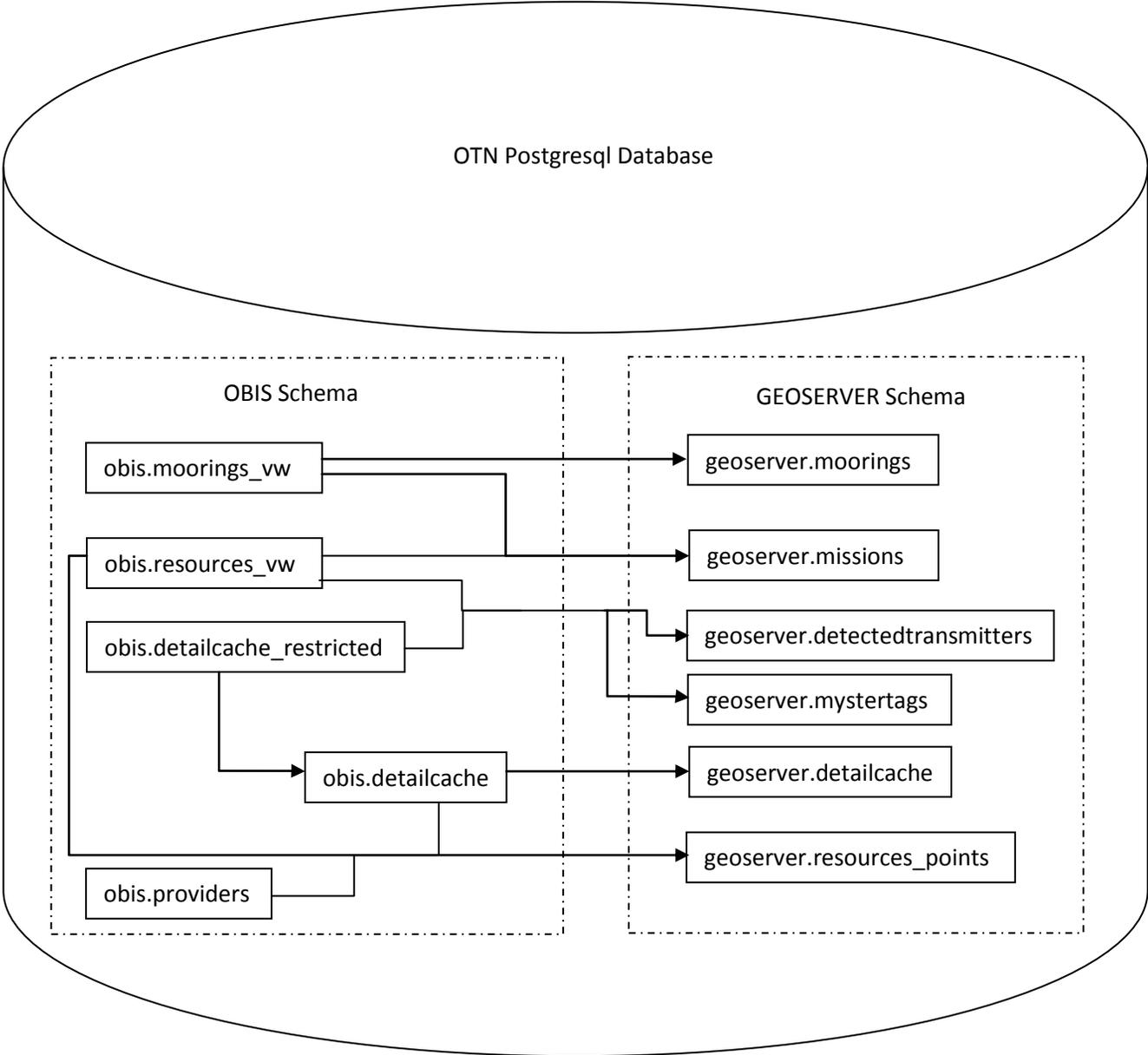
Table obis.keywords (logged_in_users) contains OTN list of keywords (e.g. instrument types).

Table obis.resource_keyword_xref (logged_in_users) contains OTN crosswalk mapping collaborators/resources with list of associated keywords.

View obis.operation_type (public and logged_in_users) using obis_resources, keywords, and resource_keyword_xref to obtain collaborator/resource operation_type (i.e. Line, Tag or Both)

View obis.resources_vw (public and logged_in_users) joining obis_resources and operation_type.

4. OBIS schema views are then used to create GEOSERVER schema views for use in creating Geoserver Layers.



4. OBIS Schema views are used to create GEOSERVER schema views for use by Geoserver Layers. These were created as Proof of Concept and are under considerable development including cleaning, removing, adding of fields/views/Layers.

All of the GEOSERVER schema views creation SQL may be found here ([link](#)).

View obis.moorings_vw (where basisofrecord=HISTORY and relationshiptype=STATION) is joined with view obis.moorings_vw (where basisofrecord=TRANSMITTER or RECEIVER) to create view geoserver.moorings. Also includes grant select to public.

View obis.moorings_vw (where basisofrecord=MISSION) is joined with table obis.resources_vw (to obtain resource's lat/lon in the case that the mission record is missing lat/lon) to create (XML formatted) view geoserver.missions. Also includes grant select to logged_in_users.

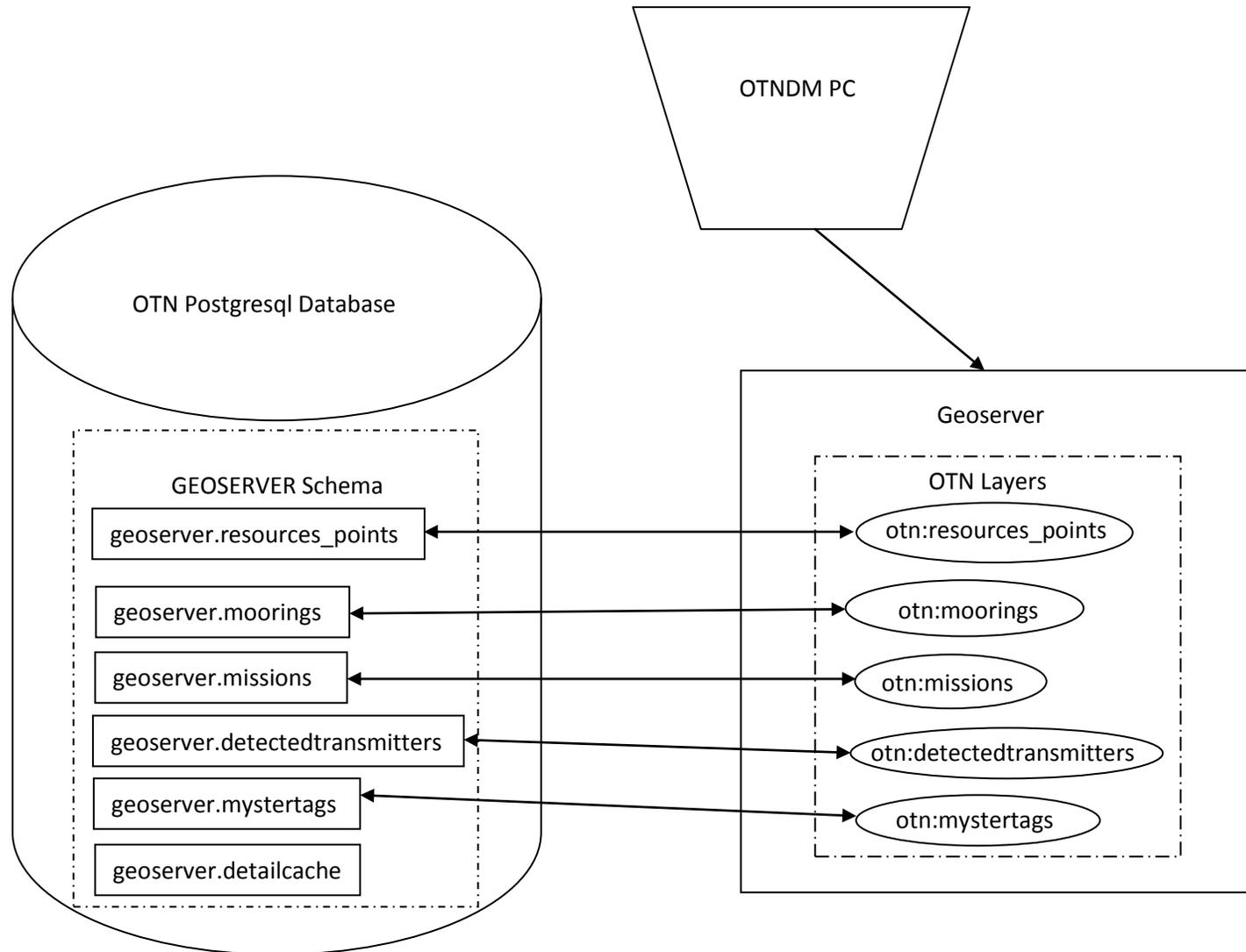
View obis.detailcache_restricted (where basisofrecord=MachineObservation) is joined with table obis.resources_vw (to obtain resource's name and lat/lon) to create summary of detections with "count" (either 1 or >1), by deployment collaborator (for detection receiver), year detected, id creating view geoserver.detectedtransmitters.

View obis.detailcache_restricted (where basisofrecord=MachineObservation and relationshiptype=UNQUALIFIED) is joined with table obis.resources_vw (to obtain resource's name and lat/lon) to create summary of detections with "count" (either 1 or >1), by deployment collaborator (for detection receiver), year detected, id creating view geoserver.mysterytags. Also includes grant select to logged_in_users and public.

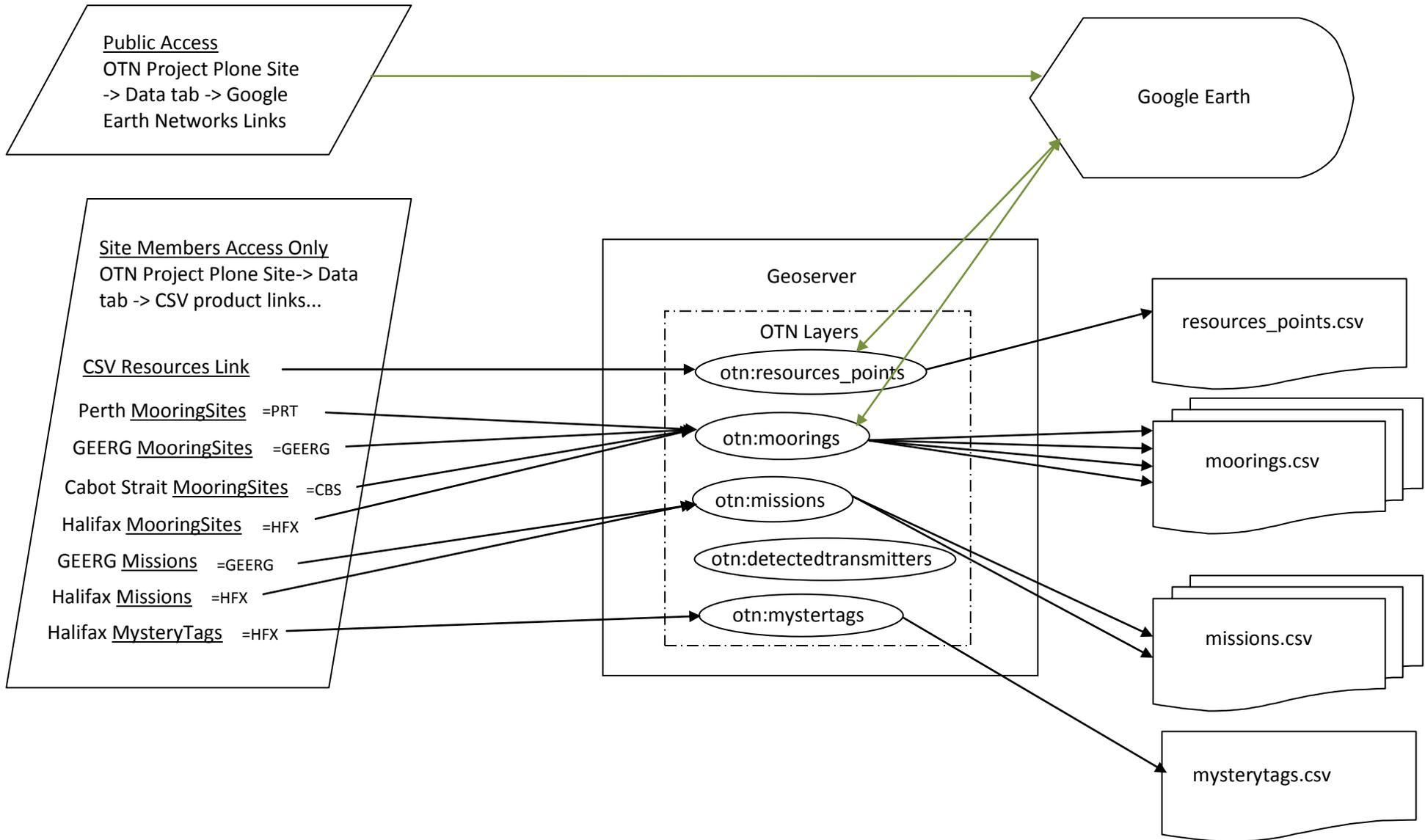
View obis.detailcache is used (basically copied) to create view geoserver.detailcache . Also includes grant select to logged_in_users.

View obis.resources_vw (obtaining resources' full name and operation type) is joined with view obis.detailcache (obtaining resources' record count and count of distinct scientificname) and table obis.providers (obtain resources' provider name) creating view geoserver.resources_points. Also includes grant select to logged_in_users and public.

5. The Geoserver Layers (Store otn) are then created using the OTN Postgresql database GEOSERVER Schema views. Only two of the Geoserver Layers (resources_points, moorings) may be accessed without a valid Geoserver username/password.



6. OTN Project Plone Data folder includes publically accessible specially constructed KMZ and OTN Site Members only access to URLs under folder CSV product links for some of the OTN Geoserver Layers (only otn:resource_points and otn:moorings are accessible without a Geoserver username/password).



6. Though the Geoserver Layers were created as Proof of Concept and are under considerable development there are URLs on the OTN Project Plone site which are using these layers.

There is a publically available link on the OTN Project Plone Data folder to a KML/KMZ file which is constructed from the two unrestricted OTN Geoserver Layers (otn:resources_points and otn:moorings), with addition of a Legend and many folders (using filters) for selecting Resources with Data/by Status/by Region/by Provider and/or selecting Moorings Operational (with further breakdown by Deployed(VR2/VR3/VR4)/Sentinel/ADCP/CPOD) and/or Proposed (with further breakdown by plan, i.e. 1st plan, 2nd plan ...).

Google Earth Network Links ...

<http://www.marinebiodiversity.ca/OTN/data/google-earth-and-maps-products>

URLs for some of the OTN Geoserver Layers have been posted on the OTN Project Plone Data folder under CSV product links (OTN Site Members only accessible). All but resources_points include filter for collaborator/resource (e.g. Halifax Line filter where collectioncode='HFX'). Only resources_points and moorings are available without Geoserver username/password. There is also another link to the above KML/KMZ.

CSV product links...

Resourcepoints

CSV Resources Link: [http://kil-otn-](http://kil-otn-1.ucis.dal.ca/geoserver/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=otn:resources_points&outputFormat=csv)

[1.ucis.dal.ca/geoserver/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=otn:resources_points&outputFormat=csv](http://kil-otn-1.ucis.dal.ca/geoserver/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=otn:resources_points&outputFormat=csv)

Moorings

Perth: [http://kil-otn-](http://kil-otn-1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:moorings&outputFormat=CSV&propertyName=collectioncode,latitude,longitude,depth&cql_filter=collectioncode+EQ+%27PRT%27)

[1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:moorings&outputFormat=CSV&propertyName=collectioncode,latitude,longitude,depth&cql_filter=collectioncode+EQ+%27PRT%27](http://kil-otn-1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:moorings&outputFormat=CSV&propertyName=collectioncode,latitude,longitude,depth&cql_filter=collectioncode+EQ+%27PRT%27)

St Lawrence River Estuary Greenland Sharks: [http://kil-otn-](http://kil-otn-1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:moorings&outputFormat=CSV&propertyName=collectioncode,latitude,longitude,depth&cql_filter=collectioncode+EQ+%27GEERG%27)

[1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:moorings&outputFormat=CSV&propertyName=collectioncode,latitude,longitude,depth&cql_filter=collectioncode+EQ+%27GEERG%27](http://kil-otn-1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:moorings&outputFormat=CSV&propertyName=collectioncode,latitude,longitude,depth&cql_filter=collectioncode+EQ+%27GEERG%27)

Cabot Strait: [http://kil-otn-](http://kil-otn-1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:moorings&outputFormat=CSV&propertyName=collectioncode,latitude,longitude,depth&cql_filter=collectioncode+EQ+%27CBS%27)

[1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:moorings&outputFormat=CSV&propertyName=collectioncode,latitude,longitude,depth&cql_filter=collectioncode+EQ+%27CBS%27](http://kil-otn-1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:moorings&outputFormat=CSV&propertyName=collectioncode,latitude,longitude,depth&cql_filter=collectioncode+EQ+%27CBS%27)

Halifax: [http://kil-otn-](http://kil-otn-1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:moorings&outputFormat=CSV&propertyName=collectioncode,latitude,longitude,depth&cql_filter=collectioncode+EQ+%27HFX%27)

[1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:moorings&outputFormat=CSV&propertyName=collectioncode,latitude,longitude,depth&cql_filter=collectioncode+EQ+%27HFX%27](http://kil-otn-1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:moorings&outputFormat=CSV&propertyName=collectioncode,latitude,longitude,depth&cql_filter=collectioncode+EQ+%27HFX%27)

Missions

St Lawrence River Estuary Greenland Sharks: [http://kil-otn-](http://kil-otn-1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:missions&outputFormat=CSV&cql_filter=resource_acronym+EQ+%27GEERG%27)

[1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:missions&outputFormat=CSV&cql_filter=resource_acronym+EQ+%27GEERG%27](http://kil-otn-1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:missions&outputFormat=CSV&cql_filter=resource_acronym+EQ+%27GEERG%27)

Halifax: [http://kil-otn-](http://kil-otn-1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:missions&outputFormat=CSV&cql_filter=resource_acronym+EQ+%27HFX%27)

[1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:missions&outputFormat=CSV&cql_filter=resource_acronym+EQ+%27HFX%27](http://kil-otn-1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:missions&outputFormat=CSV&cql_filter=resource_acronym+EQ+%27HFX%27)

Mysterytags

Halifax: [http://kil-otn-](http://kil-otn-1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:mysterytags&outputFormat=CSV&cql_filter=collectioncode+EQ+%27HFX%27)

[1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:mysterytags&outputFormat=CSV&cql_filter=collectioncode+EQ+%27HFX%27](http://kil-otn-1.ucis.dal.ca/geoserver/wfs?request=getfeature&service=wfs&version=1.1.0&typename=otn:mysterytags&outputFormat=CSV&cql_filter=collectioncode+EQ+%27HFX%27)