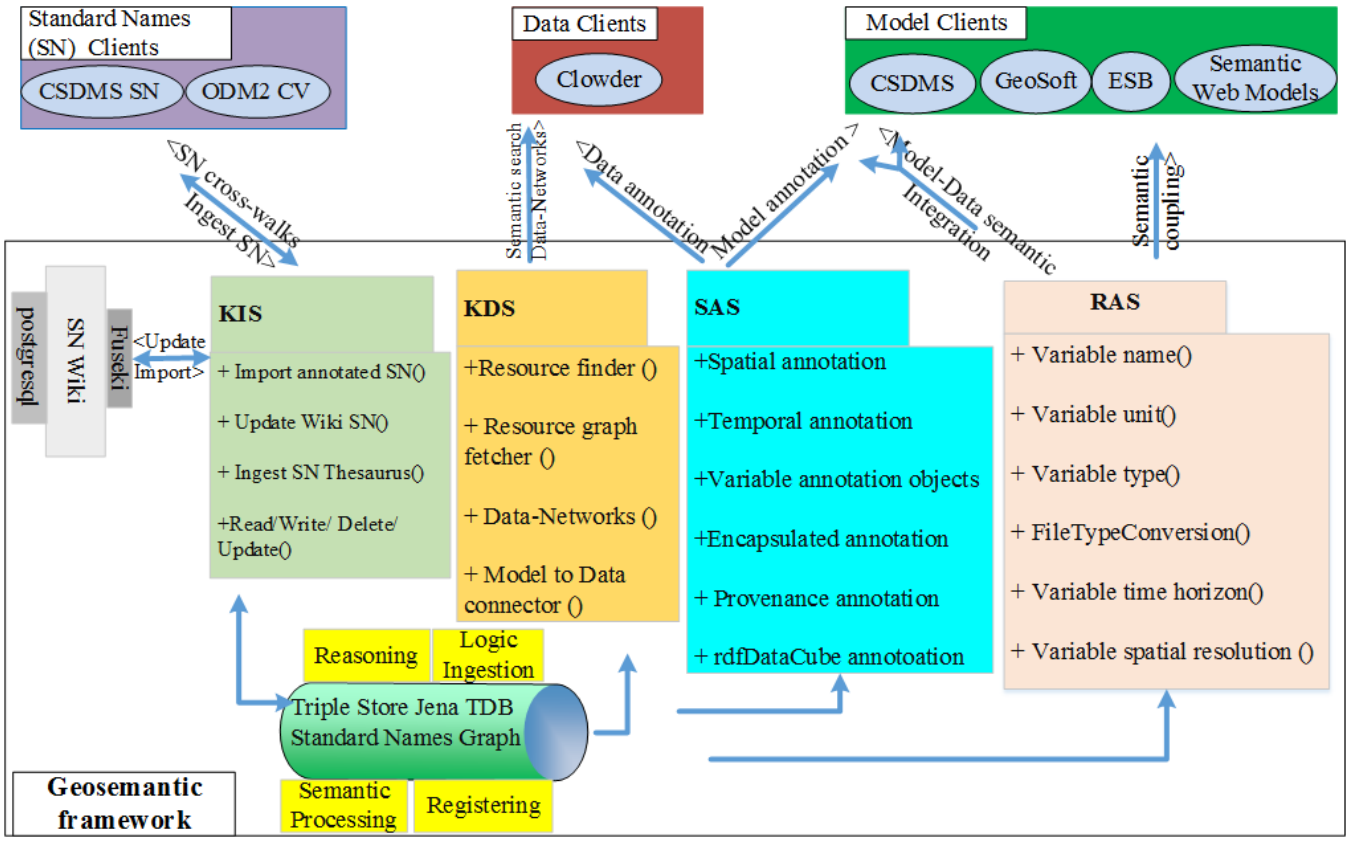
Architecture



The Geosemantics framework skeleton consists of two components:

- Services framework to include the services and their related components. Play framework (https://www.playframework.com/) is used to create the services framework.
- Standard Names wiki (SNW) to annotate SN and allow the crosswalks between them. Semantic Media Wiki (https://semantic-mediawiki.org/) is used to create the wiki system.

A- Services Framework:-

It consists of three layers:

1. **Knowledge-base**:
   a. It uses JenaTDB to develop a graph database to store ontologies and Standard Names.

2. **Knowledge management:**
   a. **Reasoner:** ECGS-19 - Getting issue details... STATUS
      i. It validates the categories and datatypes of the SN stored in the Knowledge-base.
      ii. Pellet reasoner will be used but, we can add more reasoners such as KAON2
   b. **Semantic processor:** ECGS-21 - Getting issue details... STATUS
      i. It identifies semantic similarity and matching between resources (semantic search for SN, matching between models and data, matching between models)
      ii. A resources matchmaker script is required
      iii. It provides semantic mediation between SN based on SKOS standards
      iv. SPARQL query
   c. **Logic ingestion:** ECGS-22 - Getting issue details... STATUS
      i. Bulk upload of SN to the SNW
      ii. Grouping of SN and identification of the crosswalks to create the Linked Vocabularies network
      iii. Annotation of a SN in its original schema.
   d. **Registrar:** ECGS-20 - Getting issue details... STATUS
      i. It is a catalog for storing information about external resources including services, data, models, and SN (e.g. the database of Geosoft)
      ii. It provides a form to register a new resource
3. **Web services:**
   a. **Semantic Annotation Service (SAS):**
      i. It contains five functions, each function needs to be linked with specific standard
      ii. Spatial Annotation: It connects to the GML standards to bring predicates and objects associated with a specific geometric shape (http://www.opengeospatial.org/standards/gml) ECGS-36 - Getting issue details... STATUS
      iii. Temporal Annotation: Same as above but using predicates and objects from the time ontology(http://www.w3.org/TR/owl-time/) ECGS-25 - Getting issue details... STATUS
      iv. Standard Names Annotation: It brings an object from the related stand name schema (e.g. CSDMS standard names ontology) ECGS-26 - Getting issue details... STATUS
      v. Provenance Annotation: It connects with provenance ontology to annotate simulation results (http://www.w3.org/TR/prov-o/) ECGS-28 - Getting issue details... STATUS
      vi. Statistical data Annotation: it uses predicates and objects from RDF Data Cube Vocabulary to annotate statistical data (http://www.w3.org/TR/vocab-data-cube/) ECGS-29 - Getting issue details... STATUS
      vii. Encapsulated annotation: Similar to the SNA but this function needs to be connected with our triple store to bring the attributes of a SN, such as units. ECGS-27 - Getting issue details... STATUS
   b. **Resources Alignment Service (RAS):**
      i. It checks the consistency of the attributes of quantities exchanged between two resources (model and/or data)
      ii. Variable names: It finds synonymous of a SN using SKOS or NGram ECGS-31 - Getting issue details... STATUS
      iii. Unit conversion: converts units of exchanged items based on SWEET unit ontology and NetCDF unit schema. SWEET is used to address URL-based units and NetCDF is used to address string-based ECGS-32 - Getting issue details... STATUS
      iv. Temporal Alignment: This function can do the temporal alignment between two variables using CSDMS and OpenMI external packages ECGS-35 - Getting issue details... STATUS
      v. Spatial Alignment: same as above but for spatial attributes of two variables ECGS-36 - Getting issue details... STATUS
      vi. Variable Type: This function checks the type of variables and can do conversion (e.g. string to float) ECGS-33 - Getting issue details... STATUS
      vii. File Type: Same as above but for the text-based files (RAS/FT). It may call the Brown Dog service to convert file types ECGS-34 - Getting issue details... STATUS
   c. **Knowledge Discovery Service (KDS):**
      i. Searches resources: it searches Clowder for resources. This function needs to add more search functions in the Cloweder API such as search by metadata and spatial predicates ECGS-42 - Getting issue details... STATUS

ii. Finds a resource graph: It searches in the registered database for data and models that are related to a specific resource

ECGS-43 - Getting issue details... STATUS

iii. Data Networks: It pragmatically aggregates data around environmental events

ECGS-44 - Getting issue details... STATUS

iv. Model to Model annotation: It annotates models that can be coupled together based on their attributes

ECGS-45 - Getting issue details... STATUS

v. Model to Data: It identifies serviced models that can run on a specific data and annotate and assemble the related data that are required to complete a simulation ECGS-46 - Getting issue details... STATUS

d. **Knowledge Integration Service (KIS):**

i. It is internal service to collect vocabularies from the SNW and recommend relationships between controlled vocabularies

ii. Imports annotated SN from the Wiki: This function imports the attributes that are associated with a SN in the wiki database (Fusaki) and store them in the JenaTDB ECGS-38 - Getting issue details... STATUS

iii. Updates the wiki with relationships between SN: This function return grouped and linked Standard Names based on their attributes and update pages that are associated with the related SNs. ECGS-39 - Getting issue details... STATUS

iv. Ingests SN Thesaurus: Convert SN in an ontology to a wiki page ECGS-40 - Getting issue details... STATUS

B- Standard Names Wiki (SNW):-

1. http://ecgs-dev.ncsa.illinois.edu/wiki
2. It has three levels of annotation: Primary, Secondary, and Inferred annotations as shown in the below figure
3. The annotation is implemented in a SN annotation wiki template