The CENSAM Cyberinfrastructure Architecture

Last Edited: 2012-11-07
by Juebo Wu and Chih-Yuan Chen

Byounghyun Yoo
Liang Yu (Alumni, has contributed to the first edition, 2010-05-21)
Singapore-MIT Alliance for Research and Technology / Center for Educational Computing Initiatives, Massachusetts Institute of Technology

Juebo Wu
Chih-Yuan Chen
Department of Geography, National University of Singapore

Judson Harward
Center for Educational Computing Initiatives, Massachusetts Institute of Technology

Chen-Chieh Feng
Department of Geography, National University of Singapore

Philip Bailey
Kimberly DeLong
James Hardison
Center for Educational Computing Initiatives, Massachusetts Institute of Technology
Contents

Abbreviations ................................................................................................................. 1
1. Introduction .................................................................................................................. 2
2. Goals and Objectives ................................................................................................. 2
3. Metadata Standards and Ontologies .......................................................................... 6
   3.1. Basic standards ....................................................................................................... 6
   3.2. Ontology Design .................................................................................................... 8
4. CCI Components ......................................................................................................... 11
5. Data Producers .......................................................................................................... 15
   5.1. Provider Workflow ............................................................................................... 16
   5.2. Data Alignment ..................................................................................................... 18
6. CCI Data Discovery ..................................................................................................... 20
7. The Urban Meteorology CDR ...................................................................................... 21
8. CENSAM Cyber Infrastructure CCI Walk Through ...................................................... 25
   8.1. Interfaces for data providers .................................................................................. 25
   8.2. Interfaces for data users ....................................................................................... 39
References ......................................................................................................................... 44
Figures

Figure 1 General organization of the CENSAM Cyberinfrastructure showing the separation of the CCI and federated data repositories ........................................... 3
Figure 2 Metadata standards and ontologies for CCI .................................................. 8
Figure 3 The Structure of Ontology Modeling in CCI .................................................... 9
Figure 4 A portion of concepts from the CCI ontology, showing the is-a relation between concepts .................................................................................................................. 9
Figure 5 CCI block architecture .................................................................................... 11
Figure 6 The ontology-related components of the CCI. ................................................ 12
Figure 7 Diagram of the CCI three-layer architecture and usage paths for data users and data providers ........................................................................................................... 14
Figure 8 Work flow for a data provider ......................................................................... 17
Figure 9 Data access work flow from a data user’s point of view .................................. 20
Figure 10 Water-vapor transport model image ................................................................ 22
Figure 11 Prototype of the data storage service for meteorology dataset ...................... 23
Figure 12 Interactive visualization interface of sensor data in time series ...................... 24
Figure 13 Overall work flow of the data registration ...................................................... 25
Figure 14 Top menu and sub menu of the CCI data source management ...................... 26
Figure 15 Metadata input form for CCI data source registration .................................... 26
Figure 16 CCI Web user interface for importing the GSDGM metadata file ................. 27
Figure 17 The user interface of MetaVist 2 .................................................................. 28
Figure 18 Registered data sources in CCI .................................................................... 29
Figure 19 The basic metadata information about a data source .................................... 29
Figure 20 Process for adding Keywords to a data source .............................................. 30
Figure 21 Process for adding Entities to a data source .................................................. 31
Figure 22 Create a new connection .............................................................................. 32
Figure 23 Current list of saved connections .................................................................. 32
Figure 24 Process for selecting the connection for a data source ................................. 33
Figure 25 List of loaded tables ..................................................................................... 35
Figure 28 List of fields in a specified table................................. 35
Figure 29 Initial interface for matching the Entity and the Schema........ 36
Figure 30 Properties of Temperature entity.................................. 37
Figure 31 Matching the Entity and the Schema.................................. 38
Figure 32 Entities tab after completion of matching the Entity and the Schema .......... 39
Figure 33 CCI user interface for data search by keyword and results window .......... 40
Figure 34 Entity query process.......................................................... 41
Figure 35 Summary of the entity query and sample of the result query............... 42
Figure 36 The page of “Lookup Keyword”.............................................. 43
Abbreviations

CCI: CENSAM Cyberinfrastructure
CDR: CENSAM Data Repository
CENSAM: Center for Environmental Sensing and Modeling
CSDGM: Content Standard for Digital Geospatial Metadata
CSV: Comma Separated Values
FGDC: Federal Geographic Data Committee
HDB: Housing and Development Board
ODBC: Open Database Connectivity
OWL: Web Ontology Language
OWL-DL: OWL-Description Logics
PHP: Hypertext Preprocessor
SPARQL: SPARQL Protocol and RDF Query Language
SWEET: Semantic Web for Earth and Environmental Terminology
SWRL: Semantic Web Rule Language
URI: Uniform Resource Identifier
XML: Extensible Markup Language
1. Introduction

Scientific data is created, collected, stored and exchanged across many research groups. As the volume of data has increased and the range of applications has widened over the past decade, the challenge of finding and accessing the data we need has increased [1]. Although existing data resources can usually be accessed through the Internet, it is often difficult to utilize them due to a lack of semantic information detailing what the data represents and how it was collected. This has been a persistent issue in the data engineering field.

2. Goals and Objectives

The CENSAM Cyberinfrastructure (CCI) provides the following functionality for researchers:

- to extend the longevity of their data,
- to facilitate data discovery, and
- to provide data access for authorized users.

The CCI is not intended to serve as a universal data repository for CENSAM and associated projects. We therefore utilize a federated data storage model in which individual research projects within CENSAM will provide their own data storage either on standard file servers or in relational database management systems (RDBMS). For prototyping purposes, the Cyber Infrastructure team has implemented a data repository utilizing initial sensor data from the Urban Airshed Project. Over the length of this project, this data repository may expand or grow into a general CENSAM facility. The prototype CENSAM Data Repository (CDR) will remain a separate entity from the CCI, and architecturally it will behave as a peer to other federated project data repositories within CENSAM (Figure 1).

The main goal of the CCI is to enable the usage and sharing of data between research projects. To ensure usability, the CCI has associated datasets with appropriate geospatial, sensor, accuracy, and access control metadata to optimize the data's utility, security, and longevity for present and future researchers. This project addresses the
following research questions:

- What data model(s) and markup languages can best accommodate the multi-domain and multi-scale character of CENSAM datasets?
- How can data discovery and secure data access be supported in a federated data repository environment?
- How can the cyberinfrastructure be structured to optimize queries in support of typical CENSAM modeling and real-time decision support applications?
- How can the CCI be coupled to a wide variety of domain-specific data analysis and visualization tools?

![Diagram](image)

Figure 1 General organization of the CENSAM Cyberinfrastructure showing the separation of the CCI and federated data repositories.

The CCI will support two basic operations. It will accept the registration of new datasets with proper metadata, and it will allow users to locate and access data from registered datasets meeting search criteria. There will be many variants of these operations. For example, an authorized user should be permitted to add metadata to a dataset description or to access a set of datasets and then generate a new dataset using a specified computational workflow. The resulting dataset should be automatically registered with the CCI along with the appropriate provenance metadata [2, 3]. Additional requirements may emerge with use, and the functionality of CCI operations will grow incrementally over time.

The fundamental unit recorded in the CCI is the dataset. In this context, a dataset is defined as a body of data with a clear origin and well-defined set of semantics. In one case, a dataset might consist of a set of temperature, pressure, and humidity readings sampled over the course of three months by a weather station installed on the side of a large building. In another case, a dataset might represent the output of an ocean
circulation model whose boundary conditions and forcing functions include previous runs of the same model, other supporting datasets (e.g., bathymetry), the output of other models (e.g., a large scale meteorological model), along with relatively small amounts of sensor-sampled data assimilated into the model run. Thus, datasets can store raw sensor data, modeled data, or a combination thereof. The critical requirements for any body of data to qualify as a dataset are (1) that the process by which the data was acquired is recorded and (2) that the relation between the data and physical phenomena being measured can be described. If the dataset records raw sensor data, then the first requirement will usually be satisfied by describing the sensor platform that generated the data. If it records modeled data, then the computational workflow that generated the dataset must be specified including the original datasets that were processed in the workflow. As an example of the second qualification, let us assume that data is stored as a table in a database where the column names are D, T, P and H and the column types are specified as one DateTime and three Doubles respectively. Without context this table has little meaning to anyone but its creator. This table only becomes a dataset when accompanied by the information that the first column represents the time at which the measurements were taken, the second a temperature measurement in Centigrade, the third a pressure reading in a particular set of units, etc. This context is typically referred to as metadata.

Metadata (conceived as data about data) is “structured information that describes, explains, locates, or otherwise makes it easier to retrieve, use or manage an information resource” [4]. Individual project datasets are typically described in metadata using terminology specific to the project, domain or organization that originally generated the data. Thus, one meteorological dataset may specify its recording position in terms of GPS coordinates and another might use the street address of the building beside which the instrument package was located. Since metadata is typically used in research to help people determine the applicability of a dataset [5], many different metadata standards have been developed to enable the sharing of information within a discipline.

Therefore, a dataset consists of two parts: the bare data usually expressed in structured records and the metadata that records the origins of the data and gives it physical meaning. The CCI will increase the value and longevity of the data by storing
and managing the metadata that describes the data. The CCI will initially record limited
categories of metadata. We have begun with metadata specifying the type, units and
semantics of the fields in the dataset. Looking forward, the CCI will incrementally
expand to include broader categories of metadata including:

1. Data format: How is the data stored? RDBMS, NetCDF file, Excel spreadsheet,
etc.

2. Data access: Who can access the data and how?

3. Data quality: What is known about the quality of the data, its accuracy, the
   characteristics of the sensor platform that recorded the data or the algorithms used
to generate a model?

4. Provenance: What chain of data transmission and calculation produced the data?

A core objective of the CENSAM Cyberinfrastructure (CCI) is to develop an
infrastructure to support semantic data access for pervasive monitoring, modeling and
control within a highly developed and carefully managed environment. To achieve this
objective, the CCI will use ontologies to coordinate the metadata concepts used by
different projects so that related data from separate datasets with different metadata
terminology may be located and compared.

An ontology is an explicit specification that is used to encode inter-connected
concepts in the real world [6]. Often these connections result in an object-oriented
hierarchy of concepts in a project’s metadata schema. Ontologies can be easily combined
to accommodate a set of concepts that cross domain boundaries in contrast to traditional
metadata systems that encode concepts for a single domain. Semantic information from
ontologies can be used to recognize the concepts associated with diverse vocabularies and
data descriptions and build equivalencies between them. The encoding of such
information can be extremely useful for integrating data described by different but related
vocabularies and metadata schemas. Thus, ontologies complement metadata by
establishing equivalencies and other relationships between the concepts in different
domain-specific metadata schemas. This provides an ideal means for representing the
semantic connections between similar elements across distinct datasets. In this way, for
example, temperature measurements across diverse datasets can be identified as referring
to the same type of information.

The ontology’s capability to support reasoning through the relationship between concepts helps to bridge semantic gaps without additional classification or annotation. The existence of such semantic gaps is not an urgent problem in a single domain because researchers in a given domain are expected to share a set of domain knowledge and its associated vocabularies. It is, however, a serious problem for multidisciplinary projects that require the participation of researchers from multiple domains. The problem of the semantic gap prevents these researchers from sharing, understanding, and utilizing each other’s data.

We are using an ontology infrastructure as the core component of the CCI data exchange system in order to bridge the semantic gaps between research specialties and to help users from different professional domains utilize each other’s data efficiently. The CCI will utilize both dataset specific metadata and ontologies to enable dataset discovery, comparison and sharing.

3. Metadata Standards and Ontologies

3.1. Basic standards

The utility of an ontology-driven infrastructure has gained recognition recently [2, 3, 7, 8] especially in geo-spatial sciences [9-11]. Several reusable ontologies have been published [12-14]. We have investigated several different ontologies for the earth and spatial science domain that can facilitate semantic data access in CENSAM since most of the data generated is geo-referenced and describes the physical environment. After careful consideration and investigation, we opted to utilize three standards for the CCI:

1. Content Standard for Digital Geospatial Metadata (CSDGM) from the Federal Geographic Data Committee (FGDC) [15]. CSDGM is designed to promote the coordinated development, use, sharing, and dissemination of geographic data. It defines terminologies for describing the content of a dataset, such as Identification Information, Data Quality Information, Spatial Data Organization Information, Spatial Reference Information, Entity and Attribute Information, Citation
Information, and Time Period Information. In scientific research domains such as meteorology or oceanography, most data have spatial components. Sensor data is often coupled with location information generated by GPS. Thus, we found CSDGM is very suitable to describe these key features of a dataset.

2. The Semantic Web for Earth and Environmental Terminology (SWEET) [16] developed by NASA provides a common semantic framework for Earth Science [17]. It has two versions. SWEET 1.0 consists of several faceted ontologies such as Substance, Space, Time, Unit, Physical Properties, Physical Process and Earth Realm. Based on these ontologies several integrative ontologies were developed (e.g., Natural Phenomena, Human Activity, and Data). SWEET 2.0 inherits most of the basic concepts from SWEET 1.0 but makes clear distinctions between specific domains (e.g., astronomy, biology, and geography). Our ontology design takes the advantage of the simplicity of SWEET 1.0 and the richness of domain concepts in SWEET 2.0. We incorporated the concepts from SWEET 1.0 because they provide the most fundamental framework for modeling the physical environment. We also utilize upper concepts from SWEET 1.0 as CCI root level concepts, such as NumericalEntity, PhysicalProperty, Instrument, HumanActivity, and Unit. In addition, we incorporated some domain concepts applicable to our research domains from SWEET 2.0 such as Temperature, Humidity, and WaterPressure. They help encode the semantics of domain specific data.

3. SensorML[18] from the Open Geospatial Consortium (OGC). Sensor networks are the core infrastructure of CENSAM, around which we expect to construct many sensor-based services. We need a standard to describe the metadata of a sensor, such as its classification, input and output, manufacturer, data precision, and recording frequency. Sensor Model Language Encoding Standard (SensorML) and all its related standards from OGC are designed for this purpose. It provides a framework within which sensor placement as well as the dynamic and observational characteristics of sensors and sensor systems can be defined. SensorML is an encoding standard based on XML that we use to organize database schemas for individual sensor-based datasets. SensorML also provides a hierarchy of object-oriented concepts that describe the current range of sensor technologies and their use. We have taken advantage of this conceptual hierarchy to enrich the CCI’s
ontology.

A general principle when designing an ontology is to distinguish the concepts from the terms. We have adopted a two-tier design, which include a concept tier and a term tier.

1. The concept tier contains concepts in our domains. For example, there are substantial objects such as Sensor, Platform, Ocean, Water, Air, etc, physical properties such as Temperature, Humidity, Velocity, Quality, etc, and human activities such as Experiment, Process, Trial, Research, Publication, etc.

2. The term tier contains terms that are built upon the core concepts by one or more logical expressions. For example, the logical expression “equivalent” can be used to define an alias name for a concept.

Over time utilizing well-accepted ontologies provides for more consistent data
classification and encourages more explicit vocabularies to describe data semantics. The CCI ontology uses concepts based primarily on the SWEET ontology which contains a range of widely used concepts in earth science areas and organizes them into seven major categories – data, property, device, research, space, time, and numeric. These concepts can be used in different scientific domains, such as the land surface and climate domains (Figure 3). Figure 4 shows a portion of the concepts in the land surface domain, which include highway, lane, zebra crossing, HDB, apartment, and the is-a relations between the concepts from the upper ontology and the CCI ontology and between the concepts from the CCI ontology and scientific domains.

Figure 3 The Structure of Ontology Modeling in CCI

Figure 4 A portion of concepts from the CCI ontology, showing the is-a relation between concepts
The main namespaces derived from SWEET include:

1. **Data.** This concept is the root of all elements related to data. For example, data accessing methods (e.g., DataFile and DataService), data format (e.g., Text and Binary), data property (e.g., Size and Format), and spatial data model (e.g., Vector and Raster) are all sub-concepts of Data.

2. **Property.** There are often many properties attached to an object in a scientific database. For instance, physical properties such as temperature, weight, and length, are applicable to most physical objects. Spatial properties such as location, orientation, and elevation, are applicable to objects which are in a space coordinate system. These concepts usually fall under the category of Physical Property in SWEET.

3. **Device.** This concept incorporates all elements related to hardware used in data gathering and research. The most typical ones are Computer, Sensor, Vehicle, and GPS. Most concepts of this category are sub-concepts of sweet:Instrument. Since SWEET’s sciInstrument concept has limited sub-concepts, we have added others from SensorML and from our research domains (e.g., environment and geography).

4. **Research.** This concept incorporates research domain, research action (e.g. Observation and Analysis Fieldwork), and academic activity (e.g. conference and publication). It is mainly related to the HumanActivity concept in SWEET.

5. **Space.** This concept supplies the basic elements describing physical spaces, such as the coordinate system, datum, resolution, and coordinate triple (x, y, z). The basic feature of space is that an object has one or more two- or three-dimension properties that specify its geometric properties such as the location, shape, and orientation. This category defines the basic frames of reference for spatial objects, including topology relations such as containment and located-in. Space is the basic category of SWEET, and it has sub-categories such as spaceCoordinates, spaceDirection, spaceDistribution, and spaceObject.

6. **Time.** Temporal concepts are common in almost all environmental data. Similar to spatial reference systems, the value of time is always associated with a reference system, such as Before Christ and After Christ. Some computer systems may use other reference systems or their own customized systems. This is related to the time
7. **Numeric.** Numeric concepts are used to represent the quantity associated with a property. They are always associated with values, units and reference systems, which are defined here and reused for specific subclasses. This is mainly derived from the NumericEntity and Unit concepts in SWEET.

### 4. CCI Components

The CCI is designed to support two core functions: (1) enabling data providers to publish and specify the semantic meaning for their data and (2) helping data users to search and access data with a set of formal concepts. The architecture is comprised of four main modules (Figure 5): the Web Portal for user access and security, the Ontology for data discovery and reasoning, the Data Broker for data registration and retrieval and the federated project-specific data repositories.

![CCI block architecture](image)

**Figure 5 CCI block architecture.**

The CCI is a server-based application. Client access is currently provided through a web portal, but we envision additional client access methods, e.g., web services, grid, etc. Java is used as the major implementation language because it is widely supported by the open source community. We use a version of OWL (Web Ontology Language) for expressing the ontology. OWL, endorsed by WWW Consortium and widely accepted by the ontology community, is an XML-based language which makes it easy to read and
exchange ontologies. For considerations of efficiency and reasoning support, we chose the OWL-DL dialect of OWL for implementation.

We are using Protégé, an open source ontology editing tool from Stanford University, to create and edit the ontologies since it supports the export of the ontology as OWL files. We selected two ontology engines, Jena [19] and Pellet [20], to process the ontology files. We have adopted Jena, an open source RDF/OWL API, as the major development toolkit. The OWL files are usually generated by Protégé, and then stored and maintained via the Jena APIs. Pellet is a reasoning engine that provides support for SWRL [21] based rules (see below). The major components of the CCI Ontology system are depicted in Figure 6 and Figure 7. In addition, we are using ArcGIS from ESRI, and a Python Spatial Analysis Library (PySAL) from GeoDa center to develop spatial-related functions and have chosen PostgreSQL as the back-end database for persistent OWL storage. Jena provides a built-in API for reading persistent OWL files through PostgreSQL.

![Figure 6 The ontology-related components of the CCI.](image)

Two further technologies are worthy of mention here: SWRL and SPARQL [22]. SWRL is used to build complicated rule expressions, whose interpretation improves the reasoning ability of OWL. SPARQL is a structured query language for OWL files similar to SQL for relational databases. The execution of SPARQL queries depends on the implementing API and its reasoning engine, which means that a single SPARQL clause will return different results from different reasoning engines. We chose Pellet as the CCI reasoning engine since Pellet is a widely accepted reasoning engine, and it works as a plug-in for Jena.

Reasoning is a powerful function which enables inferences based on relations between one or more basic concepts [21, 23]. A concept is represented by a class in OWL. For example, the concept “parent”, “person”, and “child” can be represented as OWL
classes parent, person, and child. Assume that in OWL the parent is defined as a subclass of person who has at least one child, an individual of person who has children can be readily inferred as an individual of parent without explicit declaration. In this way, reasoning allows a user to customize concepts without changing existing ontology data. This is very effective in an open system designed for people with different backgrounds.

The CCI ontology-based data registration system targets two user groups: data providers and data users. The CCI adopts a three-layer architecture – the UI and Application Layer, the Data Registration Layer, and the Resource Layer as shown in Figure 7. The Data Registration layer uses the ontologies and other APIs to process project data via two paths, one for the data providers and the other for the data users:

- A data provider must provide sufficient metadata based on uniform standards, such as an XML schema. Such a schema can be used to standardize the format of metadata, and it can also support standard-dependent programs. The metadata can be directly based on ontology concepts, or on some other standard vocabulary like GCMD [24], which would enable them to be recognized and aligned automatically. Alignment is the process of establishing a correspondence between the concepts in an ontology and the way the data is actually stored in a dataset. When the dataset is a relational database, this usually involves mapping ontology concepts to database tables and the properties of those concepts to table columns. The data provider can also add more alignments or alter existing ones manually. Both the original metadata and generated alignments are stored in the database for future applications.

- For a data user, it is usually easier and more straightforward to search by concepts rather than by the project-specific details of data. The user, therefore, often uses the ontology to develop a request, which the reasoning engine in the CCI will processes to achieve richer semantics (e.g. more concepts and more specific queries). The ontology alignment service original then translates the original query to different forms suitable for querying each different data source using the ontology alignment service.
Figure 7 Diagram of the CCI three-layer architecture and usage paths for data users and data providers.

Data providers simply want to publish their data and make them searchable by authorized users. To achieve this, they are required to provide metadata to describe the content and essence of their data, as specified in the standards and base ontologies we mentioned above. The collection of metadata for registration is often contained in an XML file, so that the registration of a dataset can be carried out by just uploading a XML file. However, data providers might not know all the technical details of how metadata are encoded in XML or how concepts in an ontology are linked to the specific types in their metadata. In response to this, a user interface based on web forms has been created to provide producers a simple way to register their data.

There are two sections of metadata that are worthy of mention: keyword and entity/attribute. The keywords indicate the most important features of the data, such as theme and place. They are useful for free-text searching and can be improved by ontology reasoning if the keyword refers to an ontology concept. A dataset may possess as many keywords as necessary. Entity/attribute metadata indicate the structure of the content in the dataset, e.g. Road/(num, length, start_point, end_point). The storage structures in heterogeneous databases can be totally different, but we can make data access transparent by building up the alignment between ontology concepts and database schemas. Figure 7 depicts the work flow of registering a data source. This includes the submission of
metadata, the specification of keywords, entities and a database connection, loading schemas from the database, and aligning schemas to ontology-based classes.

Once the data source is registered and the appropriate metadata supplied, users may search over the information the data source contains. There are two ways for a data user to query the data: a keyword based query or a structured query. A keyword search simply identifies datasets that are tagged with the specified keywords. Ontology reasoning can broaden the query to locate datasets tagged with equivalent or more specific keywords. A structured query is based on formal concepts and is used to query any registered databases that hold pertinent data using the appropriate mappings between the query terms and the database fields.

5. Data Producers

Many interdisciplinary research groups generate scientific data in the CENSAM project. Researchers conduct observation through various scientific activities and generate large amounts of data. Several research groups perform measurement with physical sensors and generate data with physical values. Other research groups assimilate the data into large computational models and generate model-derived datasets that may refer to the same physical properties as the original sensor data. For example, temperature and humidity data can be measured by physical sensors in a weather station, or they can be generated from an urban model. Thus, the CCI architecture must recognize and be able to handle different types of data being produced including raw data from physical sensing and processed data from simulation, assimilation, and other data processing.

1. Examples of CENSAM raw data from physical sensing and measurement:
   a) Kayaks and AUVs logging time-stamped and geolocated sensor data. Temperature and velocity of seawater are examples of typical sensor measurements. The AUV trials will also include the measurement of chemical properties and generate range scanning data using lasers and sonar.

   b) The Urban Meteorology group is deploying weather stations into Singapore Housing Development Board (HDB) blocks in the Punggol residential area located in the northeastern part of Singapore. These stations generate
temperature and humidity data in time series. These weather station sensors employ a specialized binary file format (.HoBo).

c) The Urban Hydrology monitoring group is using a wireless sensor network to monitor the status of a water distribution system. Typical measurements include pressure, water quality, and acoustic waveforms. Algorithms are being designed to predict water flow in the system and to detect leaks.

d) The Urban Meteorology group records sky luminosity by capturing a fisheye image of the entire sky every 2 minutes and storing raw JPEG images with 9 different levels of exposure that are processed into a high dynamic range (HDR) image.

2. Scientific data from simulation, assimilation, and data processing include:
   a) Groups using standard text (e.g., CSV) or commercial (e.g., Excel) intermediate data formats.
   b) NetCDF, and MATLAB file formats widely used by researchers in oceanography and climate change fields using models like FVCOM.

In order to provide mechanisms for data discovery by researchers as well as data access for authorized users, the CCI will maintain an ontology-based registration system for the data stored in the federated repositories. Individual research projects within CENSAM will provide their own data storage on standard file servers or in RDBMSs. The CCI is using RDBMSs for prototyping a data publication service and an integrated search capability through the ontology-based data registration system.

5.1. Provider Workflow

From a data provider’s point of view, there are four main tasks required to add data to the CCI.

Step 1: Data production. This is the process used to generate the raw data. Usually the data provider deploys a set of sensors or schedules the deployment of a mobile sensor. The raw data is collected from the sensor’s built-in storage using its manufacturer specific tools.
**Step 2: Data Process/Management.** The process usually involves two steps before the raw data are registered with the CCI. First, the data are converted from the sensor manufacturer’s specific format to an open format. Second, the data are organized into different datasets according to spatial and temporal range, theme, or sensor parameters.

**Step 3: Data Registration.** The metadata of these data are generated using standards and ontologies. Either by uploading an XML file or manually through an editing interface, the researcher registers the dataset and its metadata. The user can also specify the access interface, such as a web site URI or ODBC connection. Concepts from ontology can be added to the dataset to enable semantic searching and reasoning.

**Step 4: Data Alignment.** Data alignment is used to build up connections between the well-developed ontology classes of the CCI and the various data structures of a particular dataset. This process lets the registration program record the details of the dataset and makes the real data, which can be distributed across different databases and other data repositories, searchable by users from different domains.

![Figure 8 Work flow for a data provider.](image-url)
5.2. Data Alignment

Both users and data providers need to deal with real data. Users want to retrieve the real data in a desired spatiotemporal scope and in a data format suitable for analysis. The data providers want to publish their data independently of the requirements of different users. Alignment is the necessary technology for converting between specific concepts and arbitrary data structures or between different metadata standards. Ontology alignment is the process of establishing a correspondence between two similar concepts, including any subordinate and related concepts. The process of identifying possible alignment is a technology known as ontology matching. We designed a mechanism to extract information from original data sources, to perform reasoning, and to translate ontology query requests to query requests against specific data sources.

In our work, we focus on the alignment between ontology and some general data models including the particular relational data model of the data publisher’s database, a semi-structured data model like XML, and an object-oriented model of a programming language like Java. Three types of alignment are incorporated in our system according to the level of information where the alignment occurs:

1. Concept alignment is data alignment between similar concepts in an ontology or different sets of ontology. It identifies the corresponding concepts by the comparison of text such as keywords. For example, the keyword Depth can be aligned with WaterDepth, and Temperature with WaterTemperature. The identification of the concept correspondence is usually facilitated by referring to the context, i.e., the standard vocabulary or the ontology used by the users. In OWL the alignment between ontologies can be expressed by subClassOf, equivalentClass, equivalentProperty, and SameAs [25]. We can also use SWRL rules to add restrictions on an alignment.

2. Relation alignment is alignment between the model of the concepts of an ontology and a general data model such as the data schema of a database. It identifies the correspondence between the explicit semantic relation in our ontology and the implicit binary relation in a general data model. A data model usually does not specify the semantics of relation between different components. For example, a
temperature data table of a database might have a sensor_id, but it has no implication that the temperature record is generated by which sensor. The relation is specified as the generatedBy relation in a concept of ObservedData and Device in our ontology. This alignment actually clarifies the relation between each object and its properties, and makes the data model more understandable to those who are not familiar with the backgrounds of the data model design. Here is another example, the Temperature concept in our ontology can be matched with the relational data model of a database. But a relational database usually does not have the semantics of concept such as hasValue (e.g. Temperature concept has hasValue relation with NumericalEntity property). Consequently, we need the alignment between the properties of ontology concept and the properties of specific data model of the database as the following alignment.

3. Property alignment identifies the correspondence between the properties of a concept and the data attributes of a relational data model. The NumericEntity property of Temperature concept can be matched with the specific column temp in the sensor_data table of a database. A dataset is usually generated by a sensor or a process using other data sources. The data producer using the CCI user interface for entity matching provides the information for property alignment when the dataset is registered. Property alignment is more important when the corresponding data attributes are distributed in multiple tables of a database related by a foreign key. For example, the property NumericEntity of the Temperature concept can be aligned to the field temp in the sensor_data table of a database, but the Location property of the Temperature concept should be aligned with the field location in the sensor table. The alignment between the two properties is distributed in separate tables and related by a foreign key. By using the alignment information, the query criteria can be translated to a joint query on multiple tables in the database. Sometimes not all the properties are available or explicit from the data source. For example, we need to supply the unit of a physical property, e.g., Centigrade for Temperature, to make sure the data will be properly used when the unit information for a temperature value is not explicitly specified in the data source.

Alignment can help us integrate data from different sources without first loading their data into a central data store [26]. In our project, since the data sources are
distributed over the network and possess heterogeneous data models, we can use ontology to align and convert data from diverse datasets to a uniform data model.

6. CCI Data Discovery

The CCI system is distributed and the data in the system is organized in separate databases, each of which stores data developed and processed by researchers of individual research groups. The reasoning engine and validation component serve as the middleware. Each group should use an appropriate and specific set of concepts and constraints to describe the meaning of its datasets. Similarly, when a user issues a query to the system, the concepts used are domain specific.

![Diagram of Data Access Work Flow]

The basic steps for data access are as following:

**Step 1: Generate ideas and requests.** This is a process in which a user specifies the data they are searching for. For example, the user may be looking for data that is connected to concepts such as Pollution, Rainfall, and Temperature, and other data constraints such as the year, spatial domain, the value range. The user does not know which datasets are related to these concepts and how they are stored in the data
The system then performs the following two steps to support intelligent query:

**Step 2: Parse and analyze requests.** This is a process in which the reasoning engine translates the user’s ideas to an ontology query using the semantic rules defined in the ontology. Below is an example of a semantic rule about pollution data:

污染数据 = ?x is a Data ?x has keyword Heavy Metal
  
  Cu is Heavy Metal;
  Fe is Heavy Metal;
  Pb is Heavy Metal;

A heavy metal pollution data request would be translated into the following:

?x is a Data and ?x has_keyword “Cu” or “Fe” or “Pb”

The reasoning engine will modify the query so that it includes keywords from a list of specific concepts that are child concepts to “heavy metal” and thus can be regarded as Pollution Data.

**Step 3: Retrieve Data.** This is the process by which the computer system locates and queries heterogeneous data sources and then integrates these data into a unified query response. In order to query multiple data sources, the CCI first uses the reasoning engine to translate the query terms into parameters suitable for every data access interface, e.g., into a SQL clause for an ODBC database or a SOAP query for a web service. This process uses the alignment between an ontology and a data source’s data schema. The result can be merged to a uniform data model according to the inverse structure of the ontology concepts that distributed the original query. Note that the process includes semantic conversion (e.g., units for scalars).

---

7. The Urban Meteorology CDR

The Urban Airshed project involves the measurement and modeling of airflows in the urban canopy layer. For the first task, data measurement, MEMS-based airflow sensors will be developed and used in conjunction with conventional sensors to monitor outdoor
airflows in selected areas in urban canyons and housing developments to obtain fine-grained data. For the second task, modeling will be carried out with a selected set of computational fluid dynamics (CFD) programs and simpler parameterized methods. Models will include heat and water-vapor transport. The goal of the work is to inform the design and operation of buildings in an urban context (Figure 10).

Data collection involves time series measurements of temperature and humidity that are stored in the memory of the weather stations and are periodically downloaded via a USB cable. The data is downloaded from the weather station in proprietary HoBo file format and converted into a CSV format in the laboratory. The data is used to generate a CFD simulation of the urban micro climate. A total of 41 weather stations are planned to be deployed in Punggol Singapore area. Currently, 12 weather stations have been deployed for a week long test measurement.

The cyberinfrastructure team has implemented a data repository utilizing initial sensor data from the test deployment in the Punggol HDB area. This CDR is being used as an example of the federated repositories for prototyping the CCI architecture. The CDR uses an open-source database, PostgreSQL, for data storage and PHP for the design and implementation of the interface between the CCI and the dataset. Working with the Urban Airshed team, we have collected the semantics of and the metadata about the datasets collected from sensors. This information was used to design the project specific database.

The database schema for the dataset generated by the weather stations includes:
- Station information including geographic location.
- Sensor information including sensor identifier, product model, product name, manufacturer, version, device memory, and sensor serial number.
- Deployment information including station identifier, start time, end time, number of samples, interval, and object identifier for raw data such as Hobo, CSV, TXT files.
- Sensor data including time, station identifier, and battery level.
- Sensed data including temperature, humidity, dew point.
- Resource data including station identifier, resource type, and object identifier for resource data.

The CENSAM Data Repository has two major components. For the data provider, there is a web service interface for data submission and publication, and for data discovery and sharing through the CCI, there is a separate web service interface that communicates with the CCI Distributed Data Broker.

![Weather Station Overview](image)

The data provider uses a server-side application that provides a web-based user interface to enable uploading, parsing and storing of data in the CDR. Figure 11 shows
the user interface for the overview of location information about the deployed weather stations. The interface accepts raw data in HoBo file format, sensor data in CSV format, and metadata from the sensors in TXT format. The metadata in the TXT file is generated by specialized software called HoBoWare used by the data provider and provided with the commercial sensor products.

A web-based query interface supports data exports in a variety of common file formats such as Excel, Word, XML, CSV, and PDF. Other available tools enable interactive web-based visualizations and geospatial representations of datasets. This allows query output to be immediately displayed as a 2D time series plot with the location of the sensors mapped on top of a web mapping service (Figure 11 and Figure 12).

![Interactive visualization interface of sensor data in time series.](image-url)
8. CENSAM Cyber Infrastructure CCI Walk Through

There are two groups of CENSAM users: data providers and data users. Data providers are responsible for collecting, storing, and registering their data. The registered data is then annotated with semantic information and made accessible to the data users. Once the data is registered, the data users can search, query, and retrieve the registered data through CCI interfaces.

8.1. Interfaces for Data Providers

Once the data has been collected and stored, the data provider needs to register their data and metadata with the CCI. While the CCI will eventually be able to handle datasets contained in a wide variety of formats including relational databases and various file formats including NetCDF, the current prototype implementation only supports database datasets. Hence, the remainder of this section will assume that the dataset exists in a database. The overall work flow is depicted in Figure 13. This is a multi-step process: (1) Register the data source, (2) Add context and metadata through Keywords and Entities, (3) Load the data repository schema and (4) Match entities to data location for data retrieval. This typically involves matching entities with database fields. At the end of the data registration process, the data is ready to be searched.

![Diagram](image)

Figure 13 Overall work flow of the data registration.
8.1.1. Register the data source.

The first step is to register a data source by providing the metadata, which conforms to the CSDGM standard. As it is shown on the “Data Source” menu of the CCI system (Figure 14), we provide two ways to generate the metadata: manual input or import from a GSDGM metadata file. To manually enter the data, select the “Data Source” and then “New DataSource” link to access the metadata input form. In the Data Source form, (as Figure 15 shows), enter the information which describes the data source and data owner.

![Figure 14 Top menu and sub menu of the CCI data source management](image1)

![Figure 15 Metadata input form for CCI data source registration](image2)
Alternatively, the data provider might decide to use a third-party application to generate an XML metadata file that can then be uploaded. In this case, the user selects the “Upload Metadata” option from the CCI main menu, clicking the “Add” button (figure 16) to select the appropriate XML metadata file for uploading(Figure 17).

Figure 16 CCI Web user interface for importing the GSDGM metadata file

MetaVist [27] is a possible solution for generating an XML metadata file with the GSDGM standard that is well-accepted in the geospatial metadata community. Figure 18 shows basic user interface of MetaVist 2. There is a correlation between items in the data source registration form shown previously in Figure 15 and MetaVist 2 in Figure 18 since both user interfaces follow the same metadata standard.

Figure 17 An example of the XML-based metadata file

```xml
<xml-stylesheet href="NML_classic.xsl" type="text/xsl?">
<metadata>
<info>
<station>
<clientinfo>
<origin>Bongyun Yoo, Bernard Belz</origin>
<pubdate>20091224</pubdate>
<pubtime>21/12/2009</pubtime>
<title>Pungol Weather Data</title>
<edition>
</edition>
</goform>
<database>
<info>
<theName>Pungol Weather Data</theName>
</info>
</database>
</info>
</metadata>
</xml>
28

After a data source is registered, the data provider needs to add more details to it by using the different interfaces provided. This involves adding keywords, entities, and specifying database connection information for the data source. From the CCI main menu, shown in Figure 14, they can select the “Data Source” and then “Manage DataSource” link to retrieve a list of the registered data sources. Using the CCI Manage DataSource interface, (Figure 19), the data provider can add semantic context to the data source.
By clicking on the data source’s associated “Metadata” button from Figure 19, we can view the basic metadata entered during the registration process (Figure 20). This basic information is available to all users. Using the Metadata Management interface, data providers can click one of the other four tabs (Keywords, Entities, Connection and Schemas) and complete the metadata information for this data source by adding features such as keywords, entities, and database connections.

![Figure 19 Registered data sources in CCI](image)

![Figure 20 The basic metadata information about a data source](image)
Adding keywords to a data set is a two-step process. In the Metadata Management interface, select the “Keywords” tab. All keywords currently associated with the data source will be displayed. Figure 21 shows the process of adding keyword to a data source. Beginning with a new data source where no keywords are currently associated, the user clicks the “Add keywords” button to retrieve a list of all keywords available in the CCI ontology. The virtue of using ontology information is that it, along with a reasoning engine, can be leveraged to perform semantic-aware searches across registered datasets. The user selects the appropriate keywords from the list by clicking on the “add” buttons. Once the keywords have been selected, the user clicks the “save” button to update the list of current keywords associated with the data source and displays them in the Keywords tab.

Figure 21 Process for adding Keywords to a data source.

After the data provider has entered the associated keywords, the next step is to enter the Entities. The difference between an “Entity” and a “Keyword” is that an “Entity” describes a portion of the data model in which the data source’s essential attributes and
relations are specified, while a “Keyword” is a term drawn from a concept in our ontology that describes the data set. Thus, in our system, “Entity” specifies the meaning of the different elements of a structured data record in a database or other data repository while “Keyword” acts as a semantic annotation for data sources. Adding entities is similar to adding keywords. Figure 22 shows the process of adding entities to a data source. Beginning with a new data source with no currently associated Entities, the user clicks on the “Add entities” button to retrieve the list of available entities from the CCI ontology so that new entities can be associated with the data source. Candidates entities are selected clicking the “add” button and they are associated with the data source by clicking the “save” button. The updated list of added Entities for the Data Source is displayed in the Entities tab.

![Figure 22 Process for adding Entities to a data source.](image)

### 8.1.3. Load the data repository schema

Data providers typically have their own designed database. Thus, the data providers must specify the information needed to access their datasets if they want to share their data. Users can select one of the existing database connections from the “Connection” tab of the CCI Metadata Management interface or create a new database connection by clicking “New Connection” from the CCI main menu (Figure 14) if the connection is not
defined in advance. The “New Connection” link from the CCI main menu provides an input form for specifying connection information for new database connections (Figure 23). Once the information is entered and saved, then the current list of saved connection is shown as Figure 24.

![Data Connection Form](image)

Figure 23 Create a new connection.

![Connection List](image)

Figure 24 Current list of saved connections.

As an alternative, the users can select one of the existing database connections from the “Connection” tab of the CCI Metadata Management interface. Clicking the “Select” button, shown in Figure 25, retrieves a list of available connections. The data provider selects the specific connection for the data source they are working on by clicking the “Select” button in the list of available connections. The current connection information of the data source is then available in the “Connection” tab.
Once the data provider specifies the database connection for the data source, the CCI can use the connection information to load the available database schemas from the data source. This step is triggered by clicking the “Import” button on the “Schemas” tab as shown in Figure 26. It also shows the results of a list of available schemas. The data provider should select the database schema or schemas that match the registered data source from this list and then click the “Load tables” button.
The CCI will then query the database for the available tables from the selected schemas and display them. The provider then specifies which tables contain the data that they want to publish to the CCI and make available for searching.

After the schemas are selected, the CCI will then query the selected table and retrieve the table field information. The selected tables (and field information) become part of the metadata and can be viewed from the “Schemas” tab as Figure 27 shows. By clicking on the “view” button, the user can list the data fields of a specified table (Figure 28).
The last step in this process is to match the entities previously added to the data source with the schemas loaded from the database. The matching process links the database and the data model, which will enable users to query datasets without having to know the details of the individual databases.

The matching process begins by clicking the “match” (or “rematch”) button associated with an entity on the “Entities” tab of the Metadata Management interface (Figure 22). This brings up the matching interface for the associated entity (Figure 29). For this example we will work with Temperature. Our task is to match the various
properties of the Temperature entity that have been defined within the ontology (Figure 30) to the database fields of the current data source. The properties of an entity can match fields from multiple schemas and tables. These properties generally fall into two categories. Primitive properties are those that map to a standard data type like integer or timestamp (often known as a datetime). These properties are always contained in a single database column and are known to ontology users as has-a relationships.

The other category of entity properties are object properties. An entity could be a component of properties of another entity. Object properties are entity properties that can be whole entities by themselves. These are compound properties that map to multiple fields.

![Humidity Mapping Table](image)

Figure 29 Initial interface for matching the Entity and the Schema

These fields may be contained in the same table as the containing entity, but more often they correspond to an object/entity specified in a different table. If we are mapping the entity Temperature as in Figure 29, a good example of an object property is “location”. The Temperature entity needs to record where the temperature was measured, and locations are often specified as a latitude/longitude pair where each part is specified by a real number. The location in this context really belongs more directly to the sensor that measures the temperature rather than to the temperature itself. Many temperature readings can come from the same location if the sensor is stationary. And location is defined as available entity in CCI as shown in Figure 22. So in this case, location is
recorded as an object property that leads to the location entity that forms part of the
sensor entity. An object-oriented programmer would describe this by saying that the
temperature entity contains a reference to the sensor that specifies the location where the
reading was taken. A database designer would describe the situation in parallel terms by
saying that the temperature entity contains a foreign key into the sensor data table, which
contains another foreign key into the sensor location table. Ontologists simply call them
uses-a relationships. Because of these dependencies, we recommend matching entities to
database schemas “from the bottom up”. That is, the temperature entity might be matched
after the sensor entity has been matched, and the sensor location entity should be matched
at first.

![Diagram of Temperature entity properties]

Figure 30 Properties of Temperature entity

An example should make this clearer. Figure 30 shows the structure of the
Temperature entity, and Figure 31 shows the process of matching the temperature entity
just before the matching is saved. The drop-down menu at the top has been used to select
one of the table schemas which were loaded in the previous step (see Figure 27). The
“Primitive Properties” form is used to match the fields in the selected data schema to the
temperature entity’s primitive properties, i.e., literal or numerical properties attached to
classes in our ontology. For example, the temperature column in the public.sensor_data
The table schema is to be mapped to temperatureValue, a numeric property of the “Temperature” concept in our ontology associated with the concept of temperature values. The “Unit Category” and “Unit” drop-down menu are used to specify a unit definition for a numeric property. For example, the unit of “temperatureValue” can be specified as Centigrade or Fahrenheit. We call these “has-a” relationships.

The “Object Properties” form is used in a similar way to link a loaded table schema to other classes in an ontology. Figure 31 shows an example of matching the source class Temperature (in the upper left corner of the figure) and two associated classes, location and sensor (under Ref Entity field in the lower center of the figure).

![Temperature Mapping Table](image)

Figure 31 Matching the Entity and the Schema

The processes for specifying the mappings for the two relations are similar. Once the source class is selected (Temperature in this case), the system brings the object properties or uses-a relations associated with the source class (i.e., location and bySensor under Name column) to the user interface under the Name columns. For each relation a user can then select from the Ref Entity column the entity that corresponds to the relation’s target class and the columns used for matching individual records, which in database terms are known as the foreign keys. Figure 31 shows that the sensor_id is chosen as the column for matching the entity location to the relation property location as well as for matching the entity sensor to the relation property bySensor. Figure 32 shows the “Entities” tab after completion of the matching process.
The user interface for data searching and retrieval allows the data users of the CCI to search data either by using keyword search or entity query. Keyword search is a simple query that searches for data sources by the text terms specified by the user. Note that this type of search does not return any real data but only links to datasets registered with the CCI that match the text keywords specified by the user.

Figure 33 shows the user interface for text search that is called up by clicking “Keyword Search” which can be found from the CCI main menu, and this figure also shows the search results returned by the CCI.
An entity query offers a more intelligent way than the text query, as it is capable of returning semantic-related items. In our system each entity represents a data model of a particular type of data stored in the CCI. Clicking the “Entity Search” menu from the CCI main menu calls up the list of available entities.

Figure 34 begins with four examples of such entities, including Sensor, Location, Humidity, Temperature, etc. To search for temperature data, a data user can click on Temperature (in the Name column) and then specify the properties associated with Temperature in the following page. For example, Figure 34 shows an example where a user is interested in temperature values (id and temperatureValue) and the locations where these temperature values were measured (x and y). The user can also specify filters that restrict the range of the data values. For example, we can restrict the temperature between 31 and 33 Centigrade, where Fahrenheit is the default unit for the “temperatureValue” property.
Query processing proceeds in a number of stages. First, the query engine determines which datasets possess pertinent data. In the example of the temperature search above, this step would identify datasets that possess temperature data associated with locations. Then in a second stage, the query engine translates the original query string into separate queries. For each of the potential datasets the query engine will search for results according to the mapping information that has been registered by the data provider. All the data records that match the query are then selected and shown on a summary window, which lists the identified data sources and the total number of matching records for each data source.

Figure 35 shows the summary of such an entity query. This general entity query searches all the data sources available in the CCI, but a user can restrict a query to a subset of the available data sources (datasets). The lower right corner of the Figure shows part of the result of the second data source returned by the system. The “Result Count” column shows a hyperlink that indicates 340188 records of data were found. When the
link is clicked, both the primitive property `temperatureValue` and object property `location` are returned. The `temperatureValue` is converted from Centigrade to Fahrenheit according to the predefined unit information.

<table>
<thead>
<tr>
<th>Num</th>
<th>Entity Name</th>
<th>Data Source</th>
<th>Result Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>temperature</td>
<td>Punggol Weather Data</td>
<td>97103</td>
</tr>
<tr>
<td>2</td>
<td>temperature</td>
<td>Sandbox 2 - CENSAM Data Repository</td>
<td>240100</td>
</tr>
<tr>
<td>3</td>
<td>temperature</td>
<td>Sandbox - CENSAM Data Repository</td>
<td>6481</td>
</tr>
<tr>
<td>4</td>
<td>temperature</td>
<td>RSNC Best Track Data</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Num</th>
<th>id</th>
<th><code>temperatureValue</code></th>
<th>location.x</th>
<th>location.y</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>297</td>
<td>31.009999</td>
<td>103.9099872</td>
<td>1.400044444</td>
</tr>
<tr>
<td>2</td>
<td>299</td>
<td>31.052</td>
<td>103.9099872</td>
<td>1.400044444</td>
</tr>
<tr>
<td>3</td>
<td>301</td>
<td>31.103001</td>
<td>103.9099872</td>
<td>1.400044444</td>
</tr>
<tr>
<td>4</td>
<td>302</td>
<td>31.254999</td>
<td>103.9099872</td>
<td>1.400044444</td>
</tr>
<tr>
<td>5</td>
<td>303</td>
<td>31.561001</td>
<td>103.9099872</td>
<td>1.400044444</td>
</tr>
<tr>
<td>6</td>
<td>304</td>
<td>31.636999</td>
<td>103.9099872</td>
<td>1.400044444</td>
</tr>
<tr>
<td>7</td>
<td>305</td>
<td>31.791</td>
<td>103.9099872</td>
<td>1.400044444</td>
</tr>
<tr>
<td>8</td>
<td>306</td>
<td>31.996</td>
<td>103.9099872</td>
<td>1.400044444</td>
</tr>
<tr>
<td>9</td>
<td>307</td>
<td>32.124001</td>
<td>103.9099872</td>
<td>1.400044444</td>
</tr>
<tr>
<td>10</td>
<td>308</td>
<td>32.201</td>
<td>103.9099872</td>
<td>1.400044444</td>
</tr>
<tr>
<td>11</td>
<td>309</td>
<td>32.227001</td>
<td>103.9099872</td>
<td>1.400044444</td>
</tr>
<tr>
<td>12</td>
<td>310</td>
<td>32.201</td>
<td>103.9099872</td>
<td>1.400044444</td>
</tr>
<tr>
<td>13</td>
<td>311</td>
<td>32.201</td>
<td>103.9099872</td>
<td>1.400044444</td>
</tr>
<tr>
<td>14</td>
<td>312</td>
<td>32.484001</td>
<td>103.9099872</td>
<td>1.400044444</td>
</tr>
<tr>
<td>15</td>
<td>313</td>
<td>32.570002</td>
<td>103.9099872</td>
<td>1.400044444</td>
</tr>
<tr>
<td>16</td>
<td>314</td>
<td>32.655001</td>
<td>103.9099872</td>
<td>1.400044444</td>
</tr>
<tr>
<td>17</td>
<td>315</td>
<td>32.535999</td>
<td>103.9099872</td>
<td>1.400044444</td>
</tr>
<tr>
<td>18</td>
<td>316</td>
<td>32.355</td>
<td>103.9099872</td>
<td>1.400044444</td>
</tr>
</tbody>
</table>

Figure 35 Summary of the entity query and sample of the result query

Besides the entity search, we have also developed the interface for the keywords search. To help users better perform the keyword search, “Lookup Keyword” is provided on the top-level menu of the CCI system. A list of the candidate keywords is attached to help users understand what kinds of keywords are using in CCI, as shown in Figure 36.
<table>
<thead>
<tr>
<th>Num</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>millampere</td>
<td><a href="http://censam.mit.edu/numeric.ow#millampere">http://censam.mit.edu/numeric.ow#millampere</a></td>
</tr>
<tr>
<td>2</td>
<td>Senkang</td>
<td><a href="http://censam.mit.edu/space.ow#Senkang">http://censam.mit.edu/space.ow#Senkang</a></td>
</tr>
<tr>
<td>3</td>
<td>Redhill</td>
<td><a href="http://censam.mit.edu/space.ow#Redhill">http://censam.mit.edu/space.ow#Redhill</a></td>
</tr>
<tr>
<td>4</td>
<td>Europe</td>
<td><a href="http://censam.mit.edu/space.ow#Europe">http://censam.mit.edu/space.ow#Europe</a></td>
</tr>
<tr>
<td>5</td>
<td>Sensor</td>
<td><a href="http://censam.mit.edu/device.ow#Sensor">http://censam.mit.edu/device.ow#Sensor</a></td>
</tr>
<tr>
<td>6</td>
<td>minute</td>
<td><a href="http://censam.mit.edu/numeric.ow#minute">http://censam.mit.edu/numeric.ow#minute</a></td>
</tr>
<tr>
<td>7</td>
<td>South_Korea</td>
<td><a href="http://censam.mit.edu/space.ow#South_Korea">http://censam.mit.edu/space.ow#South_Korea</a></td>
</tr>
<tr>
<td>8</td>
<td>pair</td>
<td><a href="http://censam.mit.edu/numeric.ow#pair">http://censam.mit.edu/numeric.ow#pair</a></td>
</tr>
<tr>
<td>9</td>
<td>Eastern_Singapore</td>
<td><a href="http://censam.mit.edu/space.ow#Eastern_Singapore">http://censam.mit.edu/space.ow#Eastern_Singapore</a></td>
</tr>
<tr>
<td>10</td>
<td>French</td>
<td><a href="http://censam.mit.edu/space.ow#French">http://censam.mit.edu/space.ow#French</a></td>
</tr>
</tbody>
</table>

Figure 36 The page of “Lookup Keyword”
References


