NRG4CAST
FP7-2012-NMP-ENV-ENERGY-ICT-EeB
Contract no.: 600074
www.nrg4cast.org

NRG4CAST

Deliverable D2.4

Data distribution prototype

Editor:	Ioannis Chamodrakas, SingularLogic
Author(s):	Ioannis Chamodrakas, SingularLogic, Simon Tzanakis, SingularLogic, Klemen Kenda, Jasna Skrbec, Iztok Savnik, JSI
Deliverable Nature:	Prototype (R)
Dissemination Level: (Confidentiality)[1]	Public (PU)
Contractual Delivery Date:	November 2013
Actual Delivery Date:	November 2013
Suggested Readers:	Energy monitoring/forecasting software developers
Version:	1.0
Keywords:	data integration, data distribution,

[1] Please indicate the dissemination level using one of the following codes:
• **PU** = Public • **PP** = Restricted to other programme participants (including the Commission Services) • **RE** = Restricted to a group specified by the consortium (including the Commission Services) • **CO** = Confidential, only for members of the consortium (including the Commission Services) • **Restreint UE** = Classified with the classification level "Restreint UE" according to Commission Decision 2001/844 and amendments • **Confidentiel UE** = Classified with the mention of the classification level "Confidentiel UE" according to Commission Decision 2001/844 and amendments • **Secret UE** = Classified with the mention of the classification level "Secret UE" according to Commission Decision 2001/844 and amendments

Disclaimer

Executive Summary

This deliverable report presents the prototype that has been developed to facilitate data distribution, i.e. to provide a data provision mechanism to other components as well as to external services and systems.

The development of the Data Distribution Prototype has resulted to:

a) The integration of real-time energy consumption sensor data from the MIREN lighting network and the CSI building into the NRG4CAST Data Access & Integration Platform.

b) The design and deployment of an integrated analytical database which is optimized for analytical methods and processes. The integrated analytical database consists of two parts: the dynamic sensor database which stores measurements of sensors and the static sensor database which includes data about structure of the observed environment.

c) The design and deployment of an integrated database for visualization and monitoring purposes which gathers together all necessary metadata within a common data model and enables the easy and efficient execution of queries that join metadata with sensor data. The deployed database has been integrated into the NRG4CAST Data Access & Integration Platform.

d) The specification and implementation of a generic Data Acquisition API through the use of SOAP Web Services.

The sensor data integration approach, the integrated data layer for monitoring and visualization as well as the Data Acquisition API are characterized by flexibility and extensibility so that new data resources may be easily supported by the Data Distribution Prototype.

Table of Contents

List of Figures

List of Tables

Abbreviations

CSV	Comma Separated Values
DAI	Data Access & Integration
DBF	DBASE File
DBMS	Database Management System
EER	Enhanced Entity Relationship Diagram
ETL	Extract-Transform-Load
OGSA-DAI	Open Grid Software Architecture – Data Access and Integration
SOAP	Simple Object Access Protocol
WSDL	Web Services Description Language

1 Introduction

Deliverable D2.4 aims at the development of the Data Distribution prototype that will be providing real data from the first NRG4Cast installed pipeline.

The design and implementation of the Data Distribution prototype dealt with the following challenges:

- The integration of real-time sensor data into the NRG4CAST Data Access and Integration (DAI) Platform. Sensor measurements are provided periodically by the data provider either through FTP or through a Web Service Interface.
- The design and deployment of an integrated database for analytical methods and processes aimed at the facilitation of activities such as statistical analysis, data mining, and forecasting.
- The design and deployment of an integrated database for visualization and monitoring purposes providing unified access to the relevant metadata of the monitored objects (buildings, electrical vehicles, lamps, charging stations, etc). The integrated database may be accessed directly when performance is prioritized or through the DAI platform when there is a need for complex queries joining metadata with sensor data.
- The provision of a simple and common Web Service API to provide access to both metadata and sensor data from the DAI platform to the other NR4CAST components as well as to external services and systems.

Section 2 of this deliverable describes the approach followed in order to meet the challenges described above. Section 3 summarizes the main results of this deliverable.

…

2 Data Distribution Prototype

The design and implementation of the Data Distribution Prototype include the following activities:

1. Integration of real-time sensor data from the different pilot cases into the NRG4CAST DAI Platform.
2. Design of the integrated data layer for monitoring & visualization purposes, deployment of the monitoring database in a MySQL database management system, extraction, transformation and loading (ETL) of the pilot metadata from the provided data resources into the monitoring database and integration into the NRG4CAST DAI platform.
3. Design and implementation of the API for the provision of both metadata and historical sensor data from the DAI platform to the other NRG4CAST components as well as to external services and systems. The API is implemented as a set of SOAP Web Services.

Those activities are reflected in the system's architecture, which is illustrated in Figure 1. Similar picture is used for deliverables NRG4CAST D2.3 and NRG4CAST D6.3 and in each deliverable components relevant to it are coloured with different colours and those non relevant are coloured in grey. Component, relevant for this deliverable are the database and the input coming from Qminer Analytical platform (aggregated measurements, predictions, forecasts, trends), APIs for streaming and static data, and NRG4CAST DAI platform with the incoming raw data.

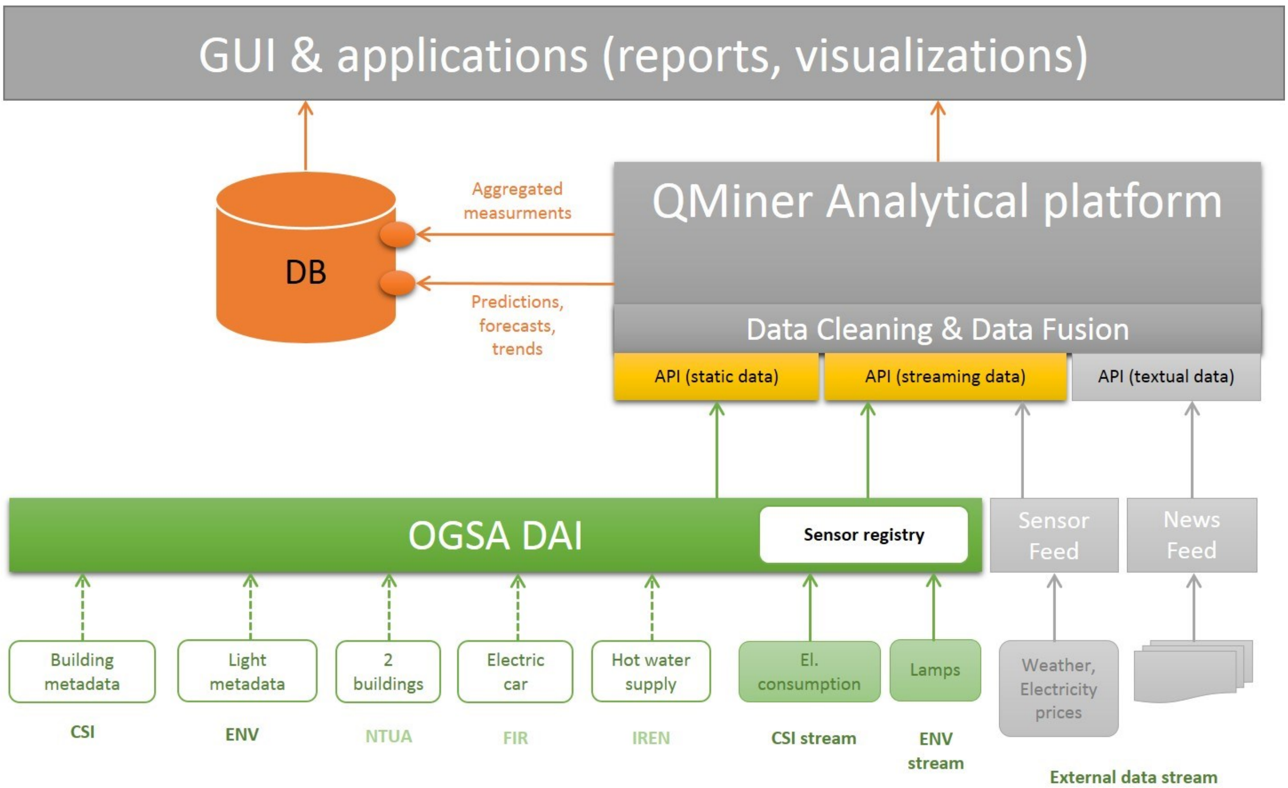

Figure 1. The arcitecture of the NRG4Cast system.

2.1 Sensor Data Integration

Sensor network is a real-time system comprising sensor nodes that are used to collect sensor measurements. Each sensor node is composed of a set of sensors. Each sensor measures the state of some particular property of observed environment e.g. temperature of the observed environment. Collectively, a set of sensor nodes with corresponding sensors are used to measure the state of observed environment.

Our integrated data model is defined in order to complete NRG4Cast platform. Core of integrated data model are sensor nodes and sensor measurements. Core data model represents dynamic data which is gathered from sensor network. All other data is gathered around this core set of sensor nodes for particular application.

Firstly, data gathered around core of sensor database represents the structure and properties of containers of sensor nodes, and their relationships to outside world. By containers we mean rooms, cars, streets, and other entities that are parts of the measured environment and that contain sensor nodes. By the relationships to the outside world we mean contracts with institutions such as electricity providers, cadastre records, etc. This data will be referred to as static data in the sequel.

Secondly, data gathered around the core of sensor database may also represent various kinds of aggregates of sensor data augmented with static data to represent the basis for particular analysis or application on top of sensor network. This data will be referred to as application data.

Data model of sensor database core is common to all data sources, and all locations where sensor data is processed. However, particular locations may have specific view of sensor database depending on the particular aspect managed on location. For instance, data about electricity consumption may be gathered on one location where sensor data about electricity consumption as well as structural data describing the observed environment is stored. To summarize, specific dynamic as well as static data is gathered on particular location for management of specific aspect of observed environment.

Some applications that use sensor database for the performance of activities such as statistical analysis, data mining, reporting or predicting need data from the sensor database to be transformed to the format appropriate for particular application. Examples of application data may be datasets generated for data mining, reporting database intended for the generation of project reports, or data warehouse for analyzing sensor data.

2.1.1 Integrated Analytical Data Model of NRG4Cast

Integrated analytical data model is optimized for analytical methods and processes, while integrated monitoring base is more optimized into reporting activities. Sensor database is composed of two parts: a) dynamic sensor database and b) static sensor database.

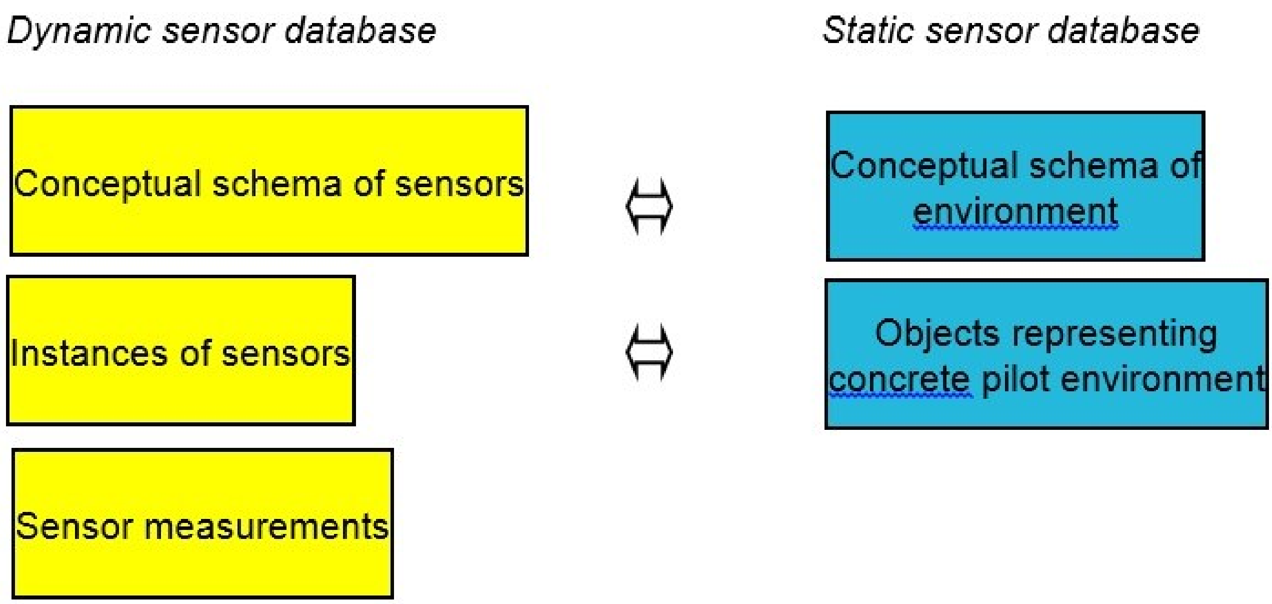

Figure 2. Schematic illustration of sensor database.

a) Dynamic sensor database stores sensor measurements grouped into nodes. Dynamic database can be further seen as composed of: i) data presenting the structure of sensors and nodes, including the definition of sensors and their types, and ii) the definitions of nodes comprising sets of sensors.

b) Static sensor database stores data about static structure of measured environment i.e. infrastructure where sensor nodes are installed. Measured environment can for instance represent a university campus with different types of buildings that include rooms where sensors are installed. Static sensor database can be also called metadata about the observed environment since it includes static structural information about sensors and nodes.

2.1.1.1 Dynamic sensor database

Dynamic sensor database stores measurements of sensors. Sensors are described by specifying sensor ID, sensor name and sensor sampling period. Nodes are employed to group sensors. They are defined by node ID, node name, node type, node location and node sensors.

Sensor measurements are collected in sensor nodes that have enough computation power to compute selected aggregates of sensor measurements on site. Sensor nodes collect data in flat tables for a set of sensors. Flat table contains aggregates of sensors measurements together with timestamps stored for a selected time periods. Table is sent to the central site repetitively in selected time intervals.

There are more ways to store sensor measurements. Column tables including timestamp, identification of sensor and sensor value is the most flexible way of storing measurements for a set of different sensors. The structure of column tables is uniform and simple; all tables including sensor measurements have the same structure. In the case a table stores measurements of a given set of sensors it can remain the same in the case the set of sensors changes. The most important negative aspect of column tables is the complexity of gathering groups of sensor measurements for presentation or for further analysis.

Another way to store sensor measurements is to group selected sets of sensors into flat tables where sensor measurements are stored in columns together with column for timestamp. The main advantage of flat tables is efficiency: since logically connected sensors are already stored together in one table and there is no need for join processing that can take considerable time for very large column tables. The most important negative aspect of flat tables is in more complex management from application programs.

The design of dynamic sensor database of NRG4Cast is given in Figure 2. The first solution is used for storing dynamic sensor data. The most important reason for the selection is simplicity of structure, however, larger amounts of sensor data would require to consider using flat tables for frequently queried sets of sensors.

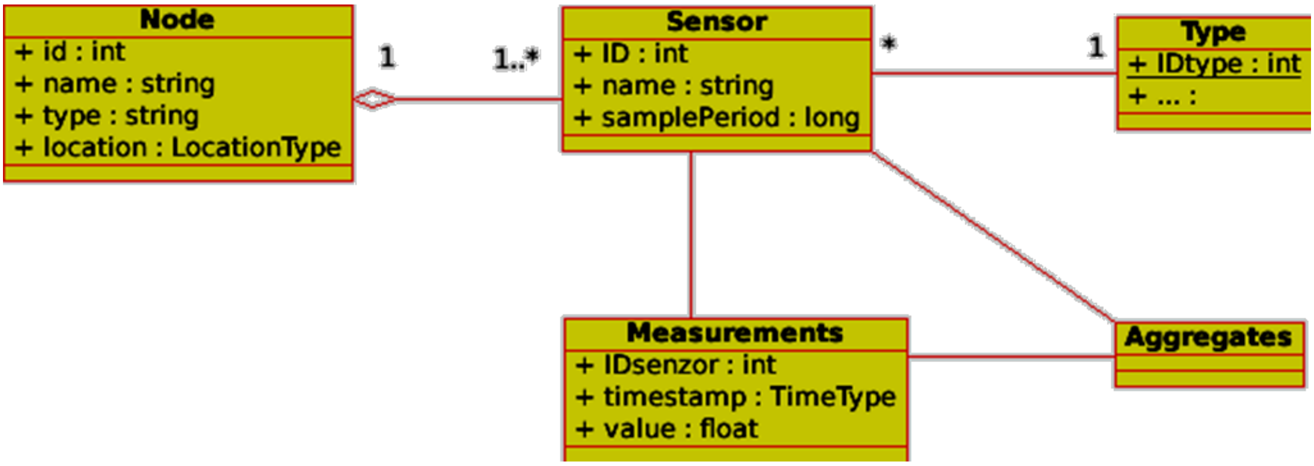

Figure 3. Dynamic sensor database

Sensor measurements are stored on one central location that can be queried via Internet to retrieve selected dynamic data. Integrated dynamic sensor database uses schema presented in Figure 3.

2.1.1.2 Static sensor database

Static sensor database includes data about structure of observed environment. Data in the static sensor database can be divided into parts representing the structure of environment for specific measured aspects.

For example, one measured aspect can be electricity consumption in city areas, buildings, apartments and rooms. In this case the finest level of measurement is room. Rooms are grouped into departments, departments are grouped into buildings and buildings into areas. Each measured aspect is therefore related to containers of nodes where measurements are obtained. In the case of electricity consumption the containers are rooms, apartments, buildings, etc.

Schema of Figure 4 can be seen as a star with class Node in the center. The points defining the star are classes Building, Carbox, WeatherStation and Lamp. In general, star points represent the containers of sensor nodes. Each container is a physical entity that usually has location and can be many times ordered in composition (aggregation) hierarchy. For instance, buildings are parts of campuses, campuses are parts of cities, etc. Similarly, lamps are located on streets, streets are parts of the cities, etc.

Measurements of sensors are stored in class Measurements and are linked to sensors in class Sensors that are linked to nodes in class Nodes. Furthermore nodes are related to containers of nodes i.e. buildings, weather stations, carboxes, lamps, etc. Let us now present classes from the point of view of data warehouses.

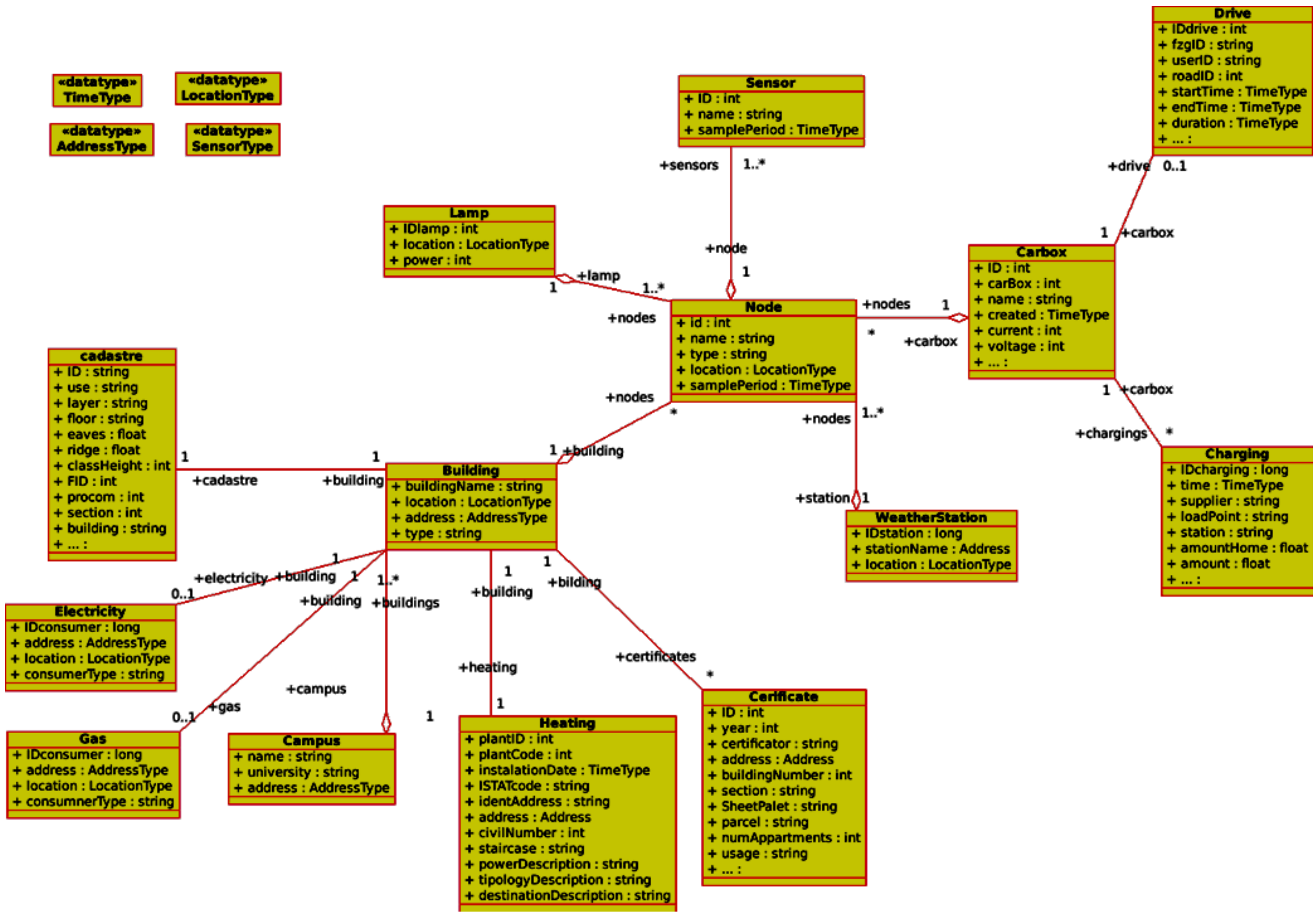

Figure 4. Static sensor database

Sensor measurements represent the source for data facts of data warehouse fact table. Each sensor can be the topic of interest represented in fact table as data fact column. Dimensions of data warehouse star schema can be designed from the properties of containers of nodes that are included in fact table.

For instance, hierarchical structure of containers can form a dimension. An example of such dimension can be the hierarchy: room, house, area and city. Other possible sources of data warehouse dimensions may be the properties of containers such as for example elements of cadastre records of buildings.

It seems beneficial to see the design of sensor database through the eyes of data warehouse designer. All possible data facts of sensor database can be selected from the set of sensor values. For example, data facts may represent electricity consumption, air temperature, weather, gas consumption, etc. The properties that are related to measured parameters are containers of sensor nodes, their properties and their relations to institutions and individuals. This properties are the candidates for the dimensions of data cube.

Efficient generation of data warehouse for a selected set of data facts and dimensions is important for on-line analysis of sensor database. To speedup the generation of fact tables aggregates of critical views can be generated incrementally when new data is obtained from nodes. Incremental computation of selected aggregates can be implemented by means of stored procedures in SQL. Another possibility to implement partial aggregates is to automatically select and construct levels of aggregation based on user demands i.e. queries posed to sensor database.

Extraction of selected views from sensor database is an important part of information system of sensor network. Extracted views are used as input of analysis and reporting tools. For instance, datasets have to be constructed for data mining tools that can determine complex events and rules from data. Similarly, tools for analysis of statistical characteristics of selected sensor values require generation of large amounts of different views on sensor database.

2.1.1.3 External Data Sources

The following external data sources have been integrated into the QMiner Analyitical Platform for planned analytics purposes in the later phases of NRG4CAST project: NewsFeed, SensorFeed and data about energy consumption and prices. All the sources are further explained in Deliverable D2.3.

NewsFeed brings textual data into the picture. NewsFeed is a service that continuously polls tens of thousands of news sites, blogs and other news generating sites in the World Wide Web. The text is then cleaned, saved into a DB, enriched, stored in XML format and published to the single feed, that a service can subscribe to. Newsfeed will be used to approximate different social and economic indexes that will be later used in different prediction scenarios, mainly with the energy price prediction.

SensorFeed is a piece of software that is designed to continuously poll open sensor data sources like weather data sources, astronomy data sources and others. SensorFeed stores this data into a local SQL database and detects new measurments on the sources which are then pushed to any subscribed services. In the NRG4CAST this data is pushed to the QMAP and is available for any models that might be derived for solving the prediction problems in the project. Currently OpenWeatherMap and WorldWeatherOnline parsers are implemented and the system is continuously polling them for weather contidions in the places of interest for NRG4CAST use cases.

Data on energy prices and consumption has been imported into QMAP. It consists of spot market prices for electrical energy and adjoint data about energy consumption. Data is from German market.

2.1.1.4 Integration of data from case studies

In this phase of the project, MIREN real-time lamp power consumption sensor data and CSI real-time building energy consumption data have been integrated into the DAI platform. However, the followed approach may be used for any kind of real-time data related to the pilot cases.

A Relational Resource is deployed within the NRG4CAST DAI platform according to the process described in Deliverable D1.5. An open source JDBC driver for CSV and DBF files is utilized for this purpose.

Then, in the case of:

MIREN lighting network sensor data

1. Sensor measurements are uploaded every 15 mins to the FTP server of the NRG4CAST toolkit.
2. A set of cron jobs run every 15 mins in order to synchronize the Relational Resource with the last measurements.

In the case of:

CSI building energy consumption data

1. Energy consumption measurements are provided through a Web Service which has been specially implemented by CSI for this purpose. The data are provided every hour.
2. A Web Service java client has been implemented in order to retrieve the last measurements which runs periodically every hour.
3. A set of cron jobs run every hour in order to synchronize the Relational Resource with the last measurements.

After the integration of both MIREN and CSI real-time data within the NRG4CAST DAI platform, any kind of data-centric workflow may be submitted to the OGSA-DAI engine which is the core of the DAI platform in order to transform, perform data pre-processing and join operations on them and, finally, retrieve the data with the appropriate form and content (see Deliverable D1.5). Such workflows are used both by the streaming mechanism described in Deliverable D6.3 and the Data Acquisition API presented in Section 2.3.

2.2 Design & Deployment of the Integrated Monitoring Database

In order to gather together all the necessary metadata of the pilot cases (e.g. energy certificate information, building information, energy provider metadata, geographical location information, etc.) as well as to join the metadata of the objects (electrical vehicle, building, charging station, lamp, etc) of each pilot case related with energy consumption measurements with the sensor data, an integrated database has been designed and deployed within the NRG4CAST MySQL DBMS.

This integrated database serves an easier and more efficient functioning of the Visualization component since it constitutes a single point for metadata acquisition. Apart from direct access to the MySQL DBMS, the integrated monitoring database has been deployed as a Relational Resource within the DAI platform in order to permit the execution of data-centric workflows on both the metadata and the sensor data.

Figure 5 depicts the Enhanced Entity Relationship Diagram (EER) of the integrated monitoring database.

The database has been populated with relevant metadata through an ETL process which has been executed on the available data resources.

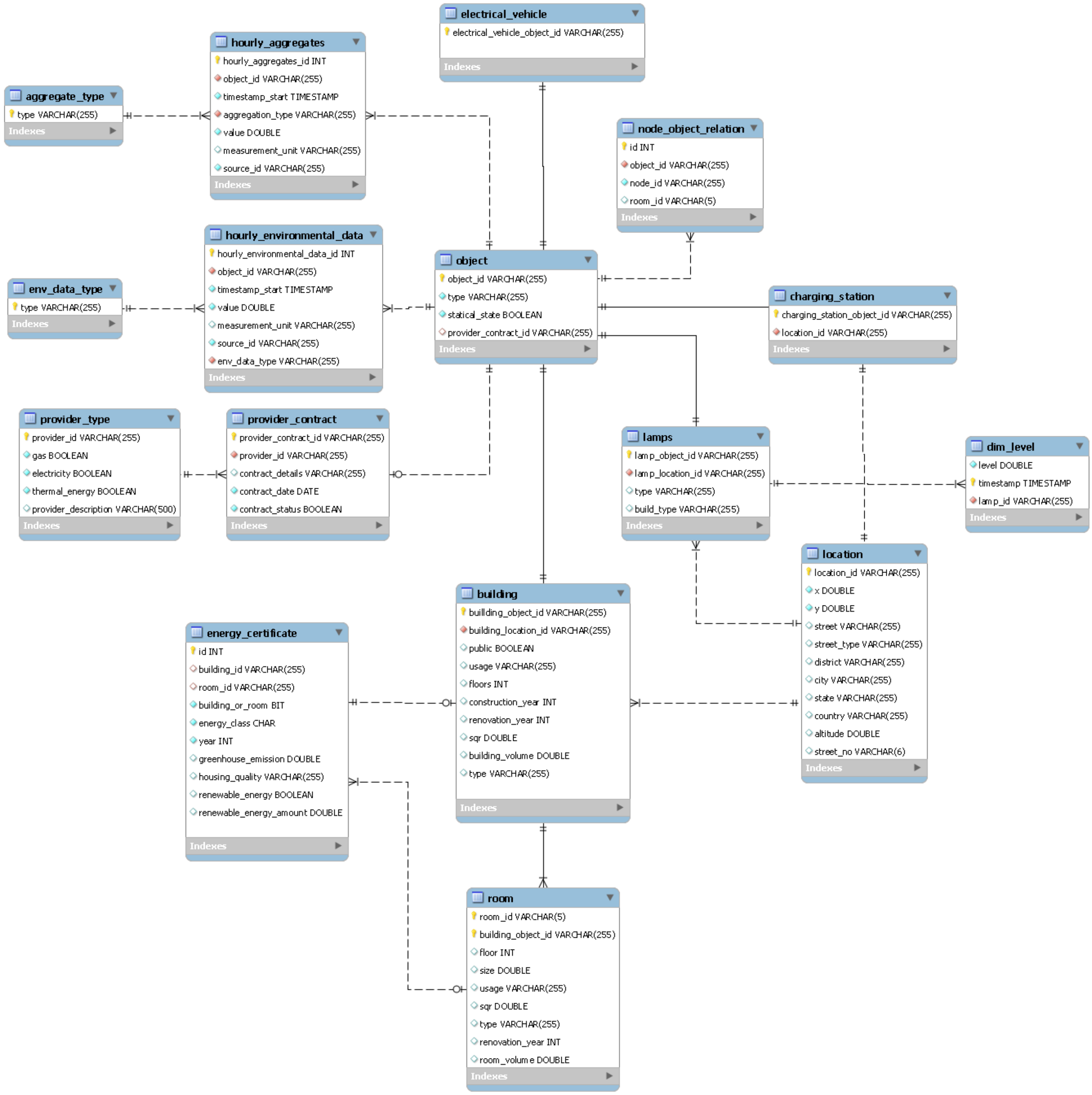

Figure 5. Integrated Monitoring Database EER Diagram

2.3 Data Acquisition API

The specification of the API is provided by Tables 1, 2, 3, 4, 5, 6. Operations are divided into two categories: **historical sensor data / metadata**. The operations are exposed through a Web Service interface which is described in the WSDL file provided online at:

http://83.212.123.209:8080/NRG4CASTServices/services/Nrg4CastServicesPort?wsdl

2.3.1 Historical Sensor Data

Table 1. listSensors Operation

Function:	listNodes
Description:	*Lists all the nodes with sensors that provide historical data, registered in the Sensor Registry database (see Deliverable D6.3).*
Input:	N/A
Output:	List of Node Objects (Node: {nodeId, nodeName, latitude, longitude})

Table 2. listSensorsOfNode Operation

Function:	listSensorsOfNodes
Description:	*Lists all the sensors of a node that provide historical data, registered in the Sensor Registry database (see Deliverable D6.3).*
Input:	nodeId
Output:	List of Sensor Objects (Sensor: {sensorId, sensorType})

Table 3. getMeasurementsOfSensor Operation

Function:	getMeasurementsOfNode
Description:	Returns sensor measurements of a particular node.
Input:	• nodeId – id of the sensor • start_timestamp • end_timestamp
Output:	List of Measurement Objects (Measurement: {sensorId, value})

2.3.2 Metadata

Table 4. listNRG4CASTObjects Operation

Function:	listNRG4CASTObjects
Description:	*Lists all the objects, registered in the DAI.*
Input:	N/A
Output:	List of NRG4CASTObjects (NRG4CASTObject: {objectId, type, providerId})

Table 5. listPropertiesOfNRG4CASTObject Operation

Function:	listPropertiesOfNRG4CASTObject
Description:	*Lists all the properties of a particular object.*
Input:	objectId
Output:	List of columnName Strings

Table 6. getPropertyOfNRG4CASTObject Operation

Function:	getPropertyOfNRG4CASTObject
Description:	*Get a particular property of a particular Object.*
Input:	propertyId, objectId
Output:	Value of a property as a String (numeric values and timestamps are transformed into a String representation)

Table 7. getNodesOfNRG4CASTObject Operation

Function:	GetNodesOfNRG4CASTObject
Description:	*Get nodes of a NRG4CAST Object.*
Input:	objectId
Output:	List of nodeId Strings

3 Conclusions

In the previous sections of this deliverable, work performed with regard to the design and development of the data distribution prototype has been reported.

The main results of this deliverable are the following:

1. A generic method for sensor data integration into the NRG4CAST DAI platform which has been applied to MIREN lamp and CSI building sensor data.
2. An integrated analytical database which is optimized for analytical methods and processes and which is composed by the dynamic sensor database and by the static sensor database.
3. An integrated monitoring database which is optimized for monitoring and visualization which has been designed in an open and extensible way in order to model all possible metadata and has been deployed to the NRG4CAST MySQL DBMS.
4. A generic Data Acquisition API which has been implemented as a SOAP Web Service.