VO Model Specifications and Example

Version:	2.6.0
Total Number of Pages:	26
File:	VO Model Specifications and Examples

Abstract

This document presents the description regarding the iCore VO Model, including the presentation of VO Model specifications, the detailed description of the VO Model included concepts and properties, as well as various examples on the use of the model by using different description formats, such XML, JSON and RDF.

Keywords List

VO Model, VO Description template, VO Model Ontology, XML, JSON, RDF

Table of Acronyms

Acronym	Meaning
CVO	Composite Virtual Object
ICT	Information and Communication Technologies
RWO	Real World Object
URI	Uniform Resource Identifier
VO	Virtual Object

Table of Contents

1. Introduction

This document is a working document on Virtual Object (VO) Modelling concepts. It aims to help in obtaining a consolidated understanding regarding the VO Model concepts. The document starts from a detailed description of VO Model concepts based on conclusions of the discussions between TNO and UPRC as well as taking into account the conclusions and the results of the collaboration between CREATE-NET, ZIGPOS and UPRC for the Y1 review of iCore Project.

2. VO Model

A Virtual Object (VO) is the virtual abstract representation of a Real World Object (RWO) with Information and Communication Capabilities (ICT), a device such *sensor, actuator, Smartphone, etc*, that in turn is associated to one or more non-ICT Objects, such *buildings, places, persons, etc*.

As already mentioned a **VO** represents an **ICT Object** and is owned by one VO owner who essentially is an **iCore user**. The VO may have one or more VO Parameters that refer on specific information regarding the VO. Each **VO Parameter**, depending on its type, may have specific **Access Rights** as well as specific **Billing Costs**. Furthermore, a VO represents the Functionality that is offered by ICT object. In particular the VO is associated with **VO Functions** that, in turn, have specific Input and Output parameters, whilst are described in terms of **VO Function Features**, such **Utilities** *(add positive meaning on the function)* and **Costs** *(add negative meaning on the function)* as well as it has **Access Rights** and **Billing Costs**. Moreover, the virtual abstract representation includes information for the further description of ICT and **non-ICT Objects**. Such information is classified in **ICT Parameters** and **Geo Location** parameters of objects. An ICT Parameter can include information about the specifications of ICT object and other necessary data regarding the ICT. The parameters that are associated with an ICT object describe essentially the specific features that characterize the ICT Object. For instance, in case where we have as an ICT Object, a sensor, a potential ICT Parameter could be the range or the accuracy of sensor. On the other hand, the objects, which belong in the real world, have a physical location that is described in terms of geographical coordinates through the Geo Location parameters. At this point it should be highlighted that an ICT and non-ICT can have the same or different Geo Location. A typical example for this situation, arises when we have as ICT Object, a *camera* that observes a *building* that is some meter far away of it. In this case the [ICT_Object = Camera] *hasICTLocation* "X" and the [non-ICT_Object = Building] *hasNonICTLocation* "Y".

The information that is described above can be clearly readable and understandable by humans (***human readable and understandable data***) [1] but it is quite difficult to be readable and understandable by machines (***machine readable and understandable data***) [1]. In order to allow the machines to be able to understand the meaning of data and consequently to infer conclusions on them, should be carried out the ***semantic enrichment*** of data. In addition through the semantic enrichment of data, it is allowed the semantic interoperability between heterogeneous entities in the iCore system. A possible and an efficient way to achieve this, is the use of ***ontologies*** that belong to ***semantic web technologies*** [3] [4]. In the following sections there is the detailed description of the iCore ontology, which among others includes meta-data for the annotation of VO Model concepts.

2.1 iCore Ontology and VO Model concepts

The creation of iCore ontology started by the iCore partner VTT (October 2012) where it circulated a light-weight version of ontology. UPRC has used this version in order to add new concepts that are included in the VO Model and they are presented in this document.

The iCore ontology, as each ontology, is comprised of four main components; ***concepts***, ***instances***, ***relations*** and ***axioms,*** while it has been developed using the ***'Web Ontology Language'*** (OWL) [4]. A set of different ***classes (concepts)*** and ***properties*** (***relations*** and ***axioms***) have been defined in the

ontology so as to describe and represent, in terms of meta-data, the related information about VO Model concepts as well as their associations. As is depicted in Figure 1, the taxonomy of VO Model comprises all necessary classes for the description of concepts. The **Thing (owl:Thing)** constitutes the super-class of the taxonomy, whilst there is a number of different classes and subclasses for the definition of components. The Table 1 provides a detailed description of each class that is included in the taxonomy. The description of classes is supplemented, in the next section, through the description and the analysis of meta-data that are expressed as properties of classes.
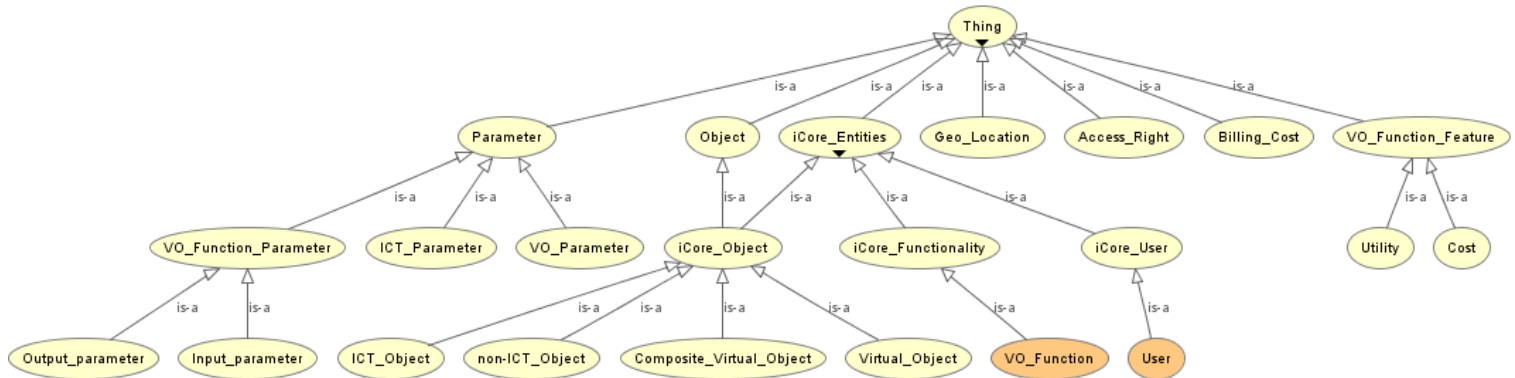

Figure 1: VO Model Taxonomy

VO Model Concept / Class	Description	Subclasses
Thing	In OWL this class is the super-class where all classes fall under it. Consequently, it is the super-class of VO Model taxonomy.	Object, iCore_Entites, Parameter, Geo_Location, Access_Right, Billing_Cost, VO_Function_Feature
Object	This class can be used for the classification of objects in the IoT, such iCore Objects, Social-IoT Entities, etc. For now it includes only iCore Objects.	iCore_Object
iCore_Entities	This class can be used for the classification of different iCore entities, such Objects, Functionalities, etc.	iCore_Object, iCore_Functionality, iCore_User
Parameter	This class can be used for the classification of different types of parameters, such VO Parameter, ICT, Parameter, etc.	VO_Parameter, ICT_Parameter, VO_Function_Parameter
Geo_Location	This class can be used for the description of RWOs physical location in terms of geographical coordinates.	-
Access_Right	This class can be used for the classification and the description of different types of access rights for the VO Functions and the VO Parameters.	-
Billing_Cost	This class can be used for the classification and the description of different types of billing costs for the VO Functions and the VO Parameters.	-

VO_Function_Feature	This class can be used for the classification of different types of function features that add positive and/or negative meaning on the function.	Utility, Cost
Utility	The class of VO Function Feature 'Utility' that describes the function in terms of utility values, such Quality, Security, etc.	-
Cost	The class of VO Function Feature 'Cost' that describes the function in terms of cost values, such Energy, Performance, etc.	-
iCore_Object	This class can be used for the classification of objects, in context of IoT, as iCore Objects.	Composite_Virtual_Object, Virtual_Object, ICT_Object, non-ICT_Object
Composite_Virtual_Object	The class of Composite Virtual Object that is classified as iCore Object. It can be used for the description of CVOs.	-
Virtual_Object	The class of Virtual Object that is classified as iCore Object. It can be used for the description of VOs.	-
ICT_Object	The class of ICT object that is classified as iCore Object. It can be used for the description of ICTs.	-
non-ICT_Object	The class of non-ICT object that is classified as iCore Object. It can be used for the description of non-ICTs.	-
iCore_Functionality	This class can be used for the classification of potential functionality types, such VO/CVO Functionality, etc.	VO_Function
VO_Function	This class can be used for the classification and the description of different types of functionality that is offered by available VOs.	-
iCore_User	This class can be used for the classification of different types of users in the iCore system.	User
User	This class has been defined so as to define a User who is either *Human* or *Software Agent*, under iCore user specifications.	
VO_Parameter	This class can be used for the classification and the description of different parameters that refer to specific VO characteristics.	-
ICT_Parameter	This class can be used for the classification and the description of different parameters that refer to specific ICT characteristics.	-
VO_Function_Parameter	This class can be used for the classification of function parameters, such Input and/or	Input_Parameter, Output_Parameter

	Output parameters.	
Input_Parameter	The class of Input parameters that describes parameters those constitute function inputs.	-
Output_Parameter	The class of Output parameters that describes parameters those constitute function outputs.	-

Table 1: Description of VO Model taxonomy

Bearing in mind the description about high-level concepts that are included in the VO Model and are defined into iCore ontology, in the next section will describe a number of different meta-data that are included in the iCore ontology so as to describe the concepts and their relationships.

2.2 VO Model meta-data

Different types of meta-data have been defined for the description of VO Model concepts and their relations. Figure 2 presents a ***Graph Data Model*** of the VO Model that comprises all concepts and their properties, which in turn can be used for the semantic enrichment of available VOs and their components. Additionally, Figure 3 depicts the visualization of VO Model concepts and their associations.
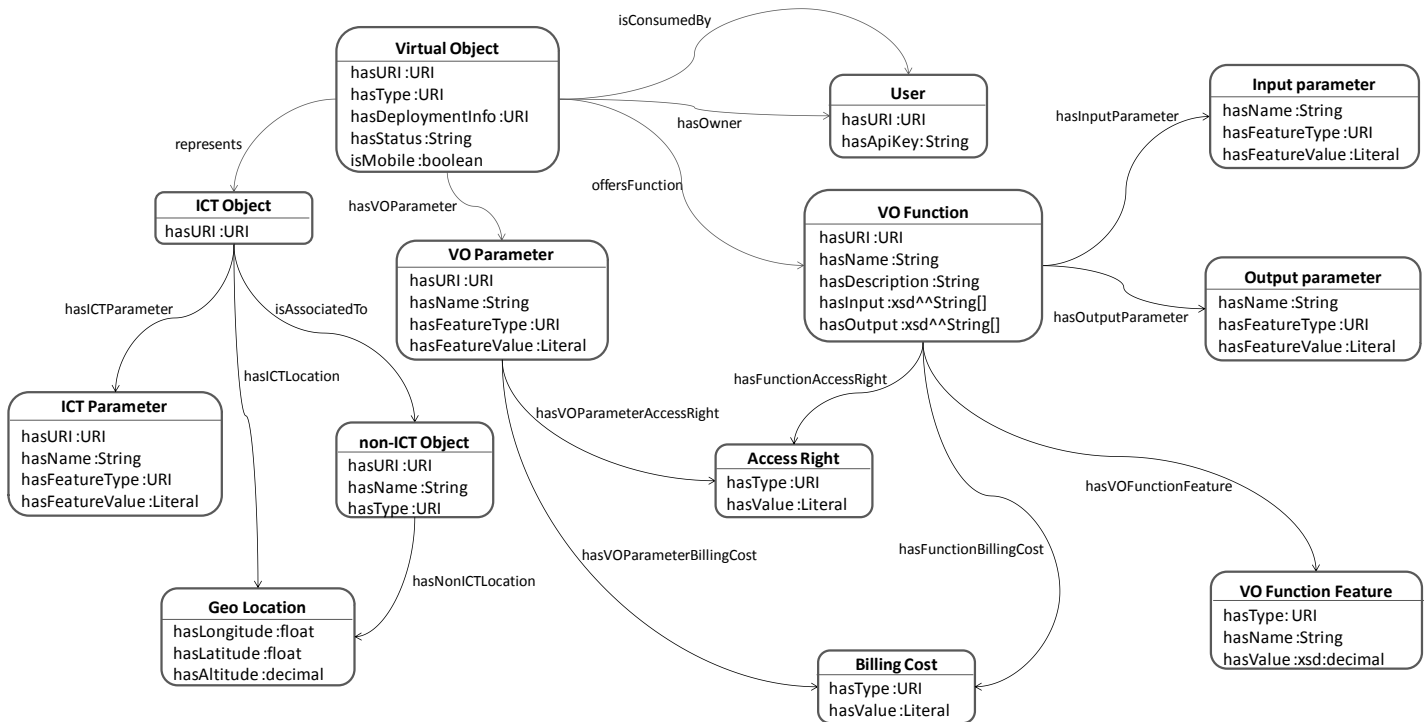

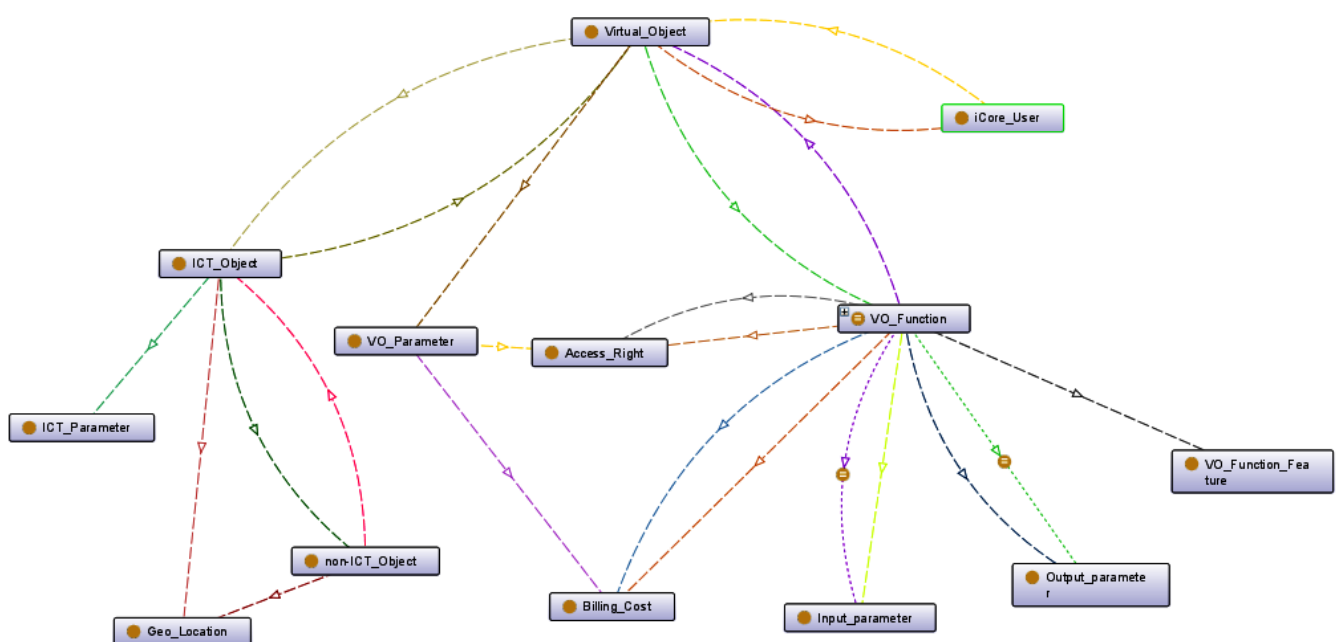**Figure 2:** VO Model - The Graph Data Model

Figure 3: VO Model - Visualization of main concepts

For the better explanation and understanding of different concepts and their meta-data, we have divided the Graph Data Model (Figure 2) in three main parts; (a) *Virtual Object*, (b) *ICT Object* and *non-ICT Object* and (c) *VO Function*. In the following subsections there is the detailed description of each part.

2.2.1 Virtual Object meta-data analysis

The first part of information that is included in the VO Model refers to the VO and its properties and its direct associations with other entities. The *"Virtual Object"* concept, that constitutes the root element in the VO Model, is directly associated with the following concepts/entities; (a) *"ICT Object"*, (b) *"VO Parameter"*, (c) *"VO Function"* and (d) *"User"*. The tables below represent the *relationships/associations* (Table 2) and the *properties* (Table 3), that are included in the first part of the VO Model (Figure 4) and constitute the VO Model meta-data for the VO.
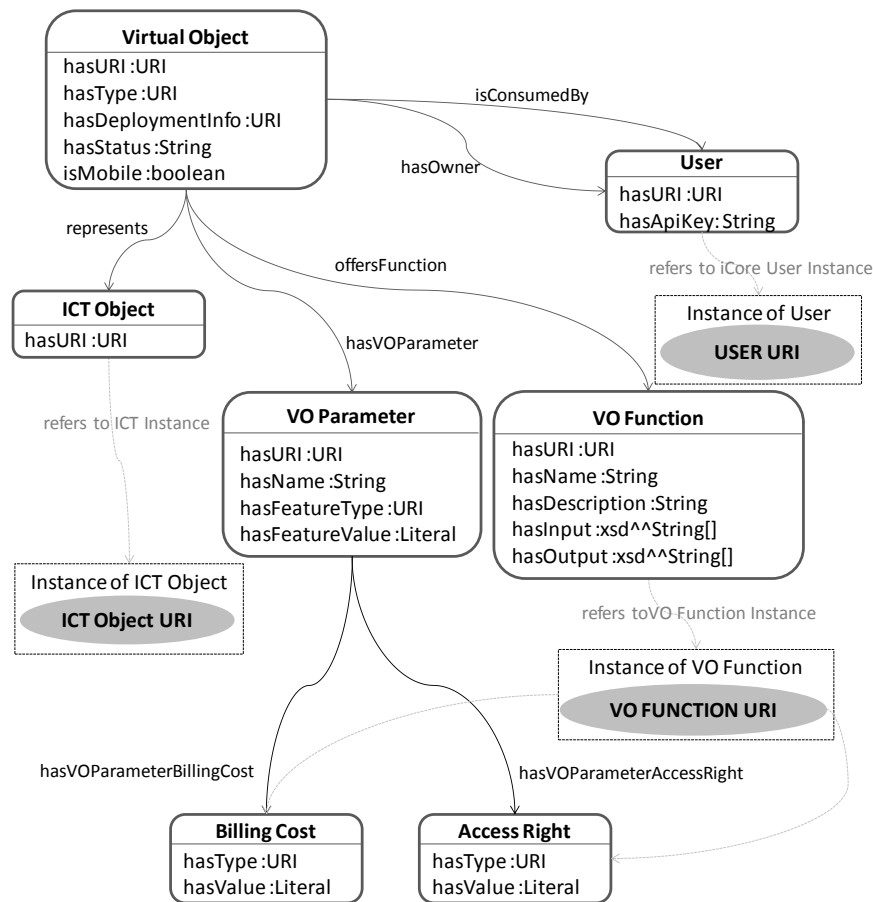
Figure 4: Virtual Object Graph Data Model part

Association	Concept / Entity	Description of association
represents	ICT Object	Each VO is the virtual representation of an ICT object that in turn is associated with a non-ICT object
hasVOParameter	VO Parameter	Each VO may have 1 or more specific characteristics that can be presented by VO Parameter entity.
offersFunction	VO Function	Each VO is associated with a set of VO Functions that are offered by itself in the virtual world, supporting the control

		and management of ICT functions.
hasOwner	User	Each VO has an owner who installs it in the iCore system and has administrative rights on it. Each user is unequally represented in the iCore system by a unique URI and a unique API-KEY.
isConsumedBy	User	A VO may be consumed by one or more end-users that may be either human users or software agents. Each user is unequally represented in the iCore system by a unique URI and a unique API-KEY

**Table 2:** Virtual Object direct relationships/associations

Concept	Description / Properties
Virtual Object	**hasURI:** works as; (a) a unique identifier for the VO and (b) refers on the description of VO as Web Resource (*dereferenceable URI*). **hasType:** defines the type of VO that essentially is the type of represented ICT Object,(e.g.: *sensor, actuator, tag, etc*). **hasDeploymentInfo:** defines a URI that refers on a file that constitutes the description of VO deployment, access and use as SW Agent, *(e.g.: OWL-S, WSDL 2.0, etc).* The deployment info is part of an instantiated VO Template and it is out of the VO Registry so as to reduce the total number of RDF Triples / VO data that is stored in it. **hasStatus:** refers to the current status of the VO that can take different values, (currently either "*AVAILABLE*" or "*UNAVAILABLE*"). **isMobile:** a Boolean parameter that defines whether the represented entity is a mobile object, (e.g.: *smart phone, car, etc*).
ICT Object	<u>**hasURI:**</u> identifier of ICT object
VO Parameter	*It is a meta-data container that can include various and diverse parameters that describe the VO. Each parameter can be described, in details, by specific features.* **hasURI:** the identifier of parameter. **hasName:** the name of the parameter. **hasFeatureType:** refers to specific features that can be used so as to describe a parameter, such as *Communication Protocol, Deployment descriptor, etc*. **hasFeatureValue:** defines the value of the feature that is referenced by Feature Type.
VO Function	**hasURI:** identifies the resource and points to the description of resource. **hasName:** an indicative name for the function. **hasDescription:** a textual description of what the function does. **hasInput:**. textual description of function inputs **hasOutput:** textual description of function outputs.
Access Rights	*Description of access rights for the VO Function and/or VO Parameter (e.g.: a VO Parameter that describes data that could be accessible only by VO Owner, such*

	as API Keys for the VO Usage, etc). It should be highlighted that the access rights are distributed dynamically by the iCore system based on a specific Role-based security scheme that enable the RBAC security mechanism of the VO Registry.
Billing Costs	*Description of billing costs for the VO Function and/or VO Parameter , (e.g.: 2 VO Parameters that refer to different Communication Protocol (REST and SOAP). The billing cost can be different for the use of each protocol).*

Table 3: Virtual Object properties

2.2.2 ICT Object and non-ICT Object meta-data analysis

The second part of information that is included in the VO Model refers to the ICT that is represented by a VO, its properties and its direct associations with other entities. The *"ICT Object"* concept is directly associated with the following concepts/entities; (a) *"ICT Parameters"*, (b) *"Geo Location"* and (c) *"non-ICT Object"*. In addition the "*non-ICT Object*" concept in turn is associated with a specific physical location, which in some cases can be different by the *"ICT Object"* physical location. The tables below represent the *relationships/*associations (Table 4) and the properties (Table 5), that are included in the second part of the VO Model (Figure 5) and constitute the VO Model meta-data for the ICT and non-ICT object.
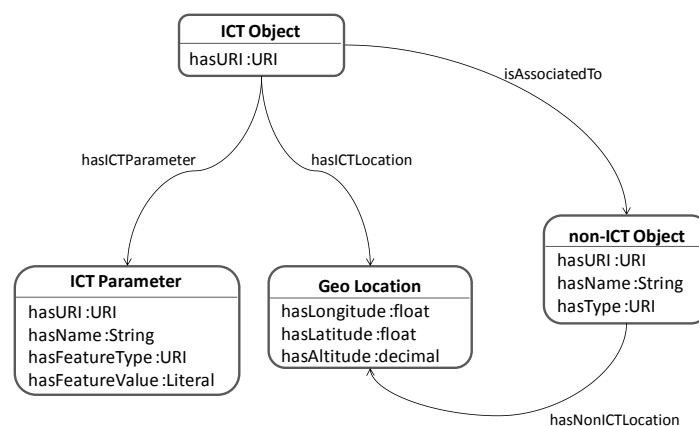
Figure 5: ICT Object & non-ICT Object Graph Data Model part

Association	Concept / Entity	Description of association
hasICTParameter	ICT Parameter	Each ICT Object is associated with a set of specific parameters that represent its specific features as a RWO.
hasICTLocation	Geo Location	The ICT Object as a RWO has a specific location in the real-world that is described by Geo Location container in terms of geographical coordinates.
isAssociatedTo	non-ICT Object	Each ICT Object is associated with one or more non-ICT Objects that are available in the real-world and its/their features can be represented in the virtual world, indirectly though the ICT Object, *(e.g.: monitoring of temperature in a room by a temperature sensor).*

hasNonICTLocation	Geo Location	The non-ICT Object, similarly with ICT Object has a specific physical location in the real-world that is described by Geo Location container.

Table 4: ICT Object & non-ICT Object direct relationships/associations

Concept	Description / Properties
ICT Object	*The RWO (with Information and Communication Capabilities) that is represented as a full abstraction in virtual world by VO.* • **hasURI:** works both as an identifier as well as the locator of the ICT Object description that can be represented as an instance of ICT Object *owl:Class* or by an ontology that is defined by ICT owner/manufacturer, (e.g.: WASPMOTE may have its own ontologies for the description of sensor/actuator devices).
ICT Parameter	*It is a meta-data container that can include data regarding the available ICT Object, (e.g.: Specifications of a Sensor, etc.). Essentially this container in combination with ICT Object container constitutes the description of specific features of the ICT Object.* • **hasURI:** identifies each parameter through a URI • **hasFeatureType:** refers to specific features that can be used so as to describe a parameter, such as *Sensor Accuracy, Sensor Range (OGC ontologies)*. • **hasFeatureValue:** defines the value of the feature that is referenced by Feature Type.
GeoLocation	*Includes data regarding the Physical Location of RWOs. An ICT and a non-CIT can have the same or different physical location. For this reason there are two different labelled associations that can link RWOs with different and diverse Geo Location instances.* • **hasLongitude:** defines the longitude • **hasLatitude:** defines the latitude • **hasAltitude:** defines the altitude
Non-ICT Object	*It describes the RWO without ICT capabilities that is associated with an ICT.* • **hasURI:** identifier of non-ICT Object. • **hasName:** the name of non-ICT Object. • **hasType:** the type of non-ICT Object, such as Person, Place, an Indoor location, etc.

Table 5: ICT Object & non-ICT Object properties

2.2.3 VO Function meta-data analysis

The third part of information that is included in the VO Model refers to the VO Function(s) that is/are offered by a VO, its properties and its direct associations with other entities.
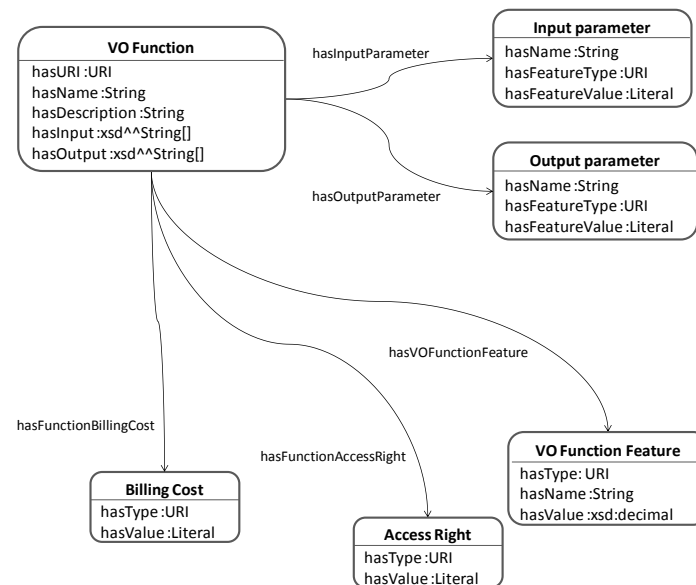
Figure 6: VO Function Graph Data Model part

The *"VO Function"* concept is directly associated with the following concepts/entities; (a) *"Input Parameter"*, (b) *"Output Parameter"*, (c) *"VO Function Feature"*, (d) *"Access Rights"* and (e) *"Billing Costs"*. The tables below represent the relationships/associations (Table 6) and the properties (Table 7), that are included in the third part of the VO Model (Figure 6) and constitute the VO Model meta-data for the ICT and non-ICT object.

Association	Concept / Entity	Description of association
hasInputParameter	Input Parameter	The VO Function has a set of Inputs that are described in terms of specific Input parameters.
hasOutputParameter	Output Parameter	The VO Function has a set of Outputs that are described in terms of specific Output parameters.
hasVOFunctionFeature	VO Function Feature	Each VO Function is associated with a set of special features that may have positive or negative meaning for the function.
hasFunctionAccessRight	Access Right	Each function may have specific access rights.
hasFunctionBillingCost	Billing Cost	Each function may have specific billing costs.

Table 6: VO Function direct relationships/associations

| Concept	Description / Properties
VO Function	*It describes the VO Function* • **hasURI:** identifies the resource and points to the description of resource. • **hasName:** an indicative name for the function. • **hasDescription:** a textual description of what the function does. • **hasInput:**. textual description of function inputs, *(e.g.: keywords)*

|	
	• **hasOutput:** textual description of function outputs, *(e.g.: keywords)*
Input Parameter	*The description of function Input(s), in terms of input parameters.* • **hasName:** textual description of parameter name. • **hasFeatureType:** refers to specific features that can be used so as to describe a parameter, such as *Measured Range, Datatypes (ontology: uo.owl, dtype.owl).* • **hasFeatureValue:** defines the value of feature that is referenced by type.
Output Parameter	*The description of function output(s), in terms of output parameters.* • **hasName:** textual description of parameter name. • **hasFeatureType:** refers to specific features that can be used so as to describe a parameter, such as *Units of Measurements, etc.* • **hasFeatureValue:** defines the value of feature that is referenced by type.
VO Function Feature	*The description of VO Function Features that can have either positive meaning or negative meaning. The meaning of the feature is identified by a specific Feature Type that can be either 'Utility' or 'Cost' respectively.*
Access Right	*The description of access rights for the function, (e.g.: specific user roles can have specific access rights on offered functions). It should be highlighted that the access rights are distributed dynamically by the iCore system based on a specific Role-based security scheme that enable the RBAC security mechanism of the VO Registry.*
Billing Cost	*The description of billing costs for the function, (e.g.: Function billing with price per function call, or per minute of function usage, etc).*

Table 7: VO Function properties

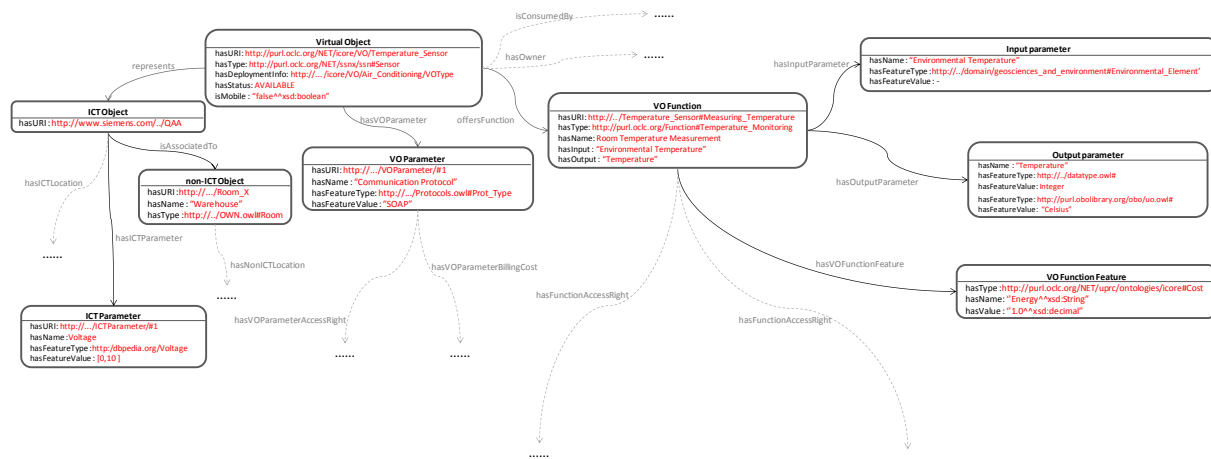# 3. Examples

3.1 Graph Representation Examples

3.1.1 VO: SIEMENS HVAC System
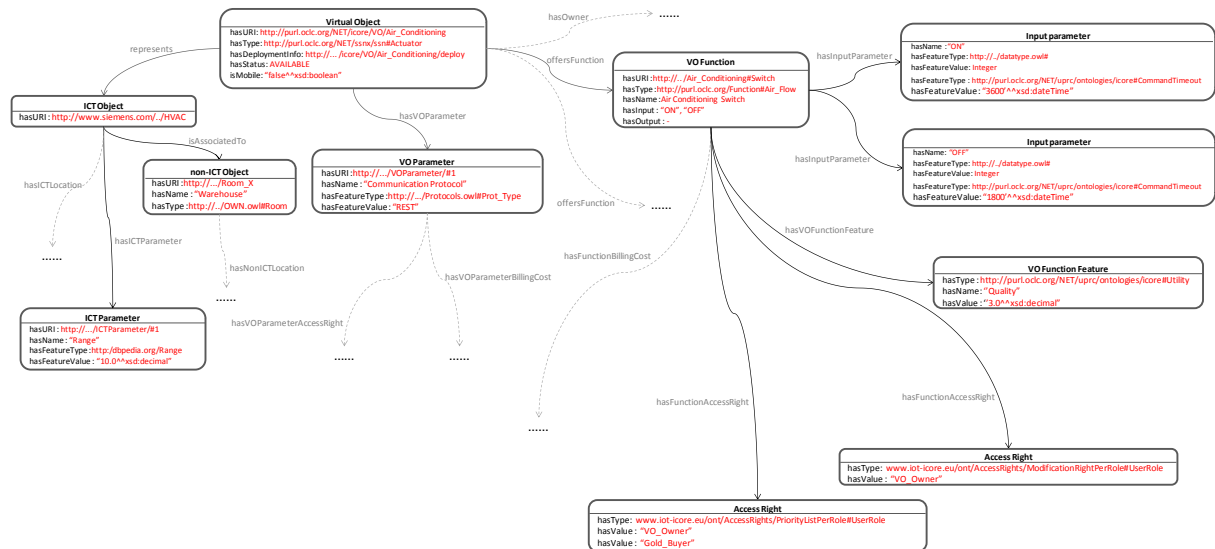
3.1.2 VO: Smart Camera
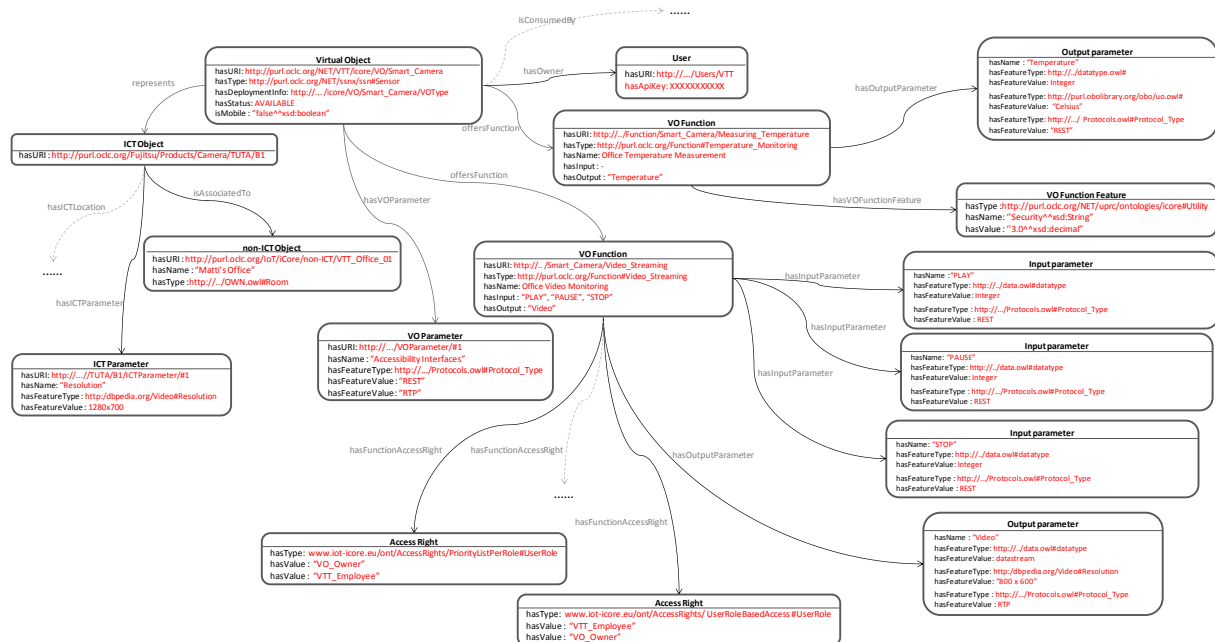
3.1.3 VO: SIEMENS Temperature Sensors

3.2 VO Description Templates Examples

3.2.1 VO Description XML Templates – Sensor

```xml
<?xml version='1.0' encoding="UTF-8"?>
<icore.voregistry.api.VirtualObject>
  <uri>http://127.0.0.1:8040/accelerometer</uri>
  <type>http://purl.oclc.org/NET/ssnx/ssn#Sensor</type>
  <deploymentInfo>http://127.0.0.1:8040/accelerometer/deploy.wsdl</deploymentInfo>
  <status>AVAILABLE</status>
  <mobility>false</mobility>
  <ict__object>
    <uri>http://127.0.0.1:8040/accelerometer/ict</uri>
    <non__ict__objects>
      <icore.voregistry.api.NonICTObject>
        <uri>http://127.0.0.1:8040/accelerometer/ict/non_icts/Smart_Home</uri>
        <type>http://www.loa.istc.cnr.it/ontologies/OWN/OWN.owl#Room</type>
        <name>Smart Home</name>
        <geoLocation>
          <uri>http://127.0.0.1:8040/accelerometer/ict/non_icts/Smart_Home/geolocation</uri>
          <longitude>300.0</longitude>
          <latitude>300.0</latitude>
          <altitude>0.0</altitude>
        </geoLocation>
      </icore.voregistry.api.NonICTObject>
    </non__ict__objects>
    <geoLocation>
      <uri>http://127.0.0.1:8040/accelerometer/ict/geolocation</uri>
      <longitude>260.2</longitude>
      <latitude>1160.0</latitude>
      <altitude>0.0</altitude>
    </geoLocation>
    <ict__parameters/>
  </ict__object>
  <vo__parameters>
    <icore.voregistry.api.VOParameter>
      <uri>http://127.0.0.1:8040/accelerometer/vo_parameters/Primary_Communication_Protocol</uri>
      <name>Primary Communication Protocol</name>
      <meta__features>
        <icore.voregistry.api.MetaFeatureSet>

<uri>http://127.0.0.1:8040/accelerometer/vo_parameters/Primary_Communication_Protocol/meta_features_set
s/Protocols.owl#Protocol_Type</uri>
          <type>http://www.iot-icore.eu/ontologies/Protocols.owl#Protocol_Type</type>
          <value class="string">HTTP-REST</value>
        </icore.voregistry.api.MetaFeatureSet>
      </meta__features>
      <access__rights/>
      <billing__costs/>
```

```
        </icore.voregistry.api.VOParameter>
      </vo__parameters>
      <vo__functions>
        <icore.voregistry.api.VOFunction>
          <uri>http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring</uri>
          <name>Accelerometer Monitoring</name>
          <description>This function is used for the monitoring of the space coordinates.</description>
          <input>
            <string>3D Point</string>
          </input>
          <output>
            <string>Accelerometer</string>
          </output>
          <access__rights>
            <icore.voregistry.api.AccessRight>

<uri>http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/access_rights/DISCOVERY<
/uri>
              <type>http://www.iot-icore.eu/ont/AccessRights/DISCOVERY</type>
              <value    class="string">[PREMIUM_USER,    ADMINISTRATOR,    THIRD_PARTY_USER,    BASIC_USER,
VO_OWNER]</value>
            </icore.voregistry.api.AccessRight>
            <icore.voregistry.api.AccessRight>

<uri>http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/access_rights/DELETE</ur
i>
              <type>http://www.iot-icore.eu/ont/AccessRights/DELETE</type>
              <value class="string">[ADMINISTRATOR, VO_OWNER]</value>
            </icore.voregistry.api.AccessRight>
            <icore.voregistry.api.AccessRight>

<uri>http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/access_rights/USE</uri>
              <type>http://www.iot-icore.eu/ont/AccessRights/USE</type>
              <value class="string">[PREMIUM_USER, ADMINISTRATOR, BASIC_USER, VO_OWNER]</value>
            </icore.voregistry.api.AccessRight>
            <icore.voregistry.api.AccessRight>

<uri>http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/access_rights/UPDATE</ur
i>
              <type>http://www.iot-icore.eu/ont/AccessRights/UPDATE</type>
              <value class="string">[PREMIUM_USER, ADMINISTRATOR, VO_OWNER]</value>
            </icore.voregistry.api.AccessRight>
          </access__rights>
          <billing__costs/>
          <input__parameters>
            <icore.voregistry.api.InputParameter>

<uri>http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/input_parameters/3D_Poin
t</uri>
              <name>3D Point</name>
              <meta__features>
                <icore.voregistry.api.MetaFeatureSet>

<uri>http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/input_parameters/3D_Poin
t/meta_features_sets/planning#pre</uri>
                  <type>http://www.uprc.gr/ontologies/planning#pre</type>
                  <value class="string">sensorsOK(SH)</value>
                </icore.voregistry.api.MetaFeatureSet>
              </meta__features>
            </icore.voregistry.api.InputParameter>
          </input__parameters>
          <output__parameters>
            <icore.voregistry.api.OutputParameter>

<uri>http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/output_parameters/Accele
rometer</uri>
              <name>Accelerometer</name>
              <meta__features>
                <icore.voregistry.api.MetaFeatureSet>

<uri>http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/output_parameters/Accele
rometer/meta_features_sets/planning#post</uri>
                  <type>http://www.uprc.gr/ontologies/planning#post</type>
                  <value class="string">acceleration(SH)</value>
```

```
            </icore.voregistry.api.MetaFeatureSet>
          </meta__features>
        </icore.voregistry.api.OutputParameter>
      </output__parameters>
      <function__features>
        <icore.voregistry.api.VOFunctionFeature>

<uri>http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/function_features/Energy
</uri>
          <type>http://www.iot-icore.eu/ontologies/features.owl#Cost</type>
          <name>Energy</name>
          <value>1.0</value>
        </icore.voregistry.api.VOFunctionFeature>
        <icore.voregistry.api.VOFunctionFeature>

<uri>http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/function_features/Networ
k</uri>
          <type>http://www.iot-icore.eu/ontologies/features.owl#Cost</type>
          <name>Network</name>
          <value>1.0</value>
        </icore.voregistry.api.VOFunctionFeature>
        <icore.voregistry.api.VOFunctionFeature>

<uri>http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/function_features/Perfor
mance</uri>
          <type>http://www.iot-icore.eu/ontologies/features.owl#Utility</type>
          <name>Performance</name>
          <value>3.0</value>
        </icore.voregistry.api.VOFunctionFeature>
        <icore.voregistry.api.VOFunctionFeature>

<uri>http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/function_features/Expend
iture</uri>
          <type>http://www.iot-icore.eu/ontologies/features.owl#Cost</type>
          <name>Expenditure</name>
          <value>1.0</value>
        </icore.voregistry.api.VOFunctionFeature>
        <icore.voregistry.api.VOFunctionFeature>

<uri>http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/function_features/Qualit
y</uri>
          <type>http://www.iot-icore.eu/ontologies/features.owl#Utility</type>
          <name>Quality</name>
          <value>3.0</value>
        </icore.voregistry.api.VOFunctionFeature>
        <icore.voregistry.api.VOFunctionFeature>

<uri>http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/function_features/Securi
ty</uri>
          <type>http://www.iot-icore.eu/ontologies/features.owl#Utility</type>
          <name>Security</name>
          <value>3.0</value>
        </icore.voregistry.api.VOFunctionFeature>
      </function__features>
    </icore.voregistry.api.VOFunction>
  </vo__functions>
  <vo__owner>
    <uri>http://www.iot-icore.eu/Users#DimitrisKelaidonis</uri>
    <api__key>                          api__key>
  </vo__owner>
  <vo__consumers/>
</icore.voregistry.api.VirtualObject>
```

Table 8: XML VO Description Example - Accelerometer Sensor

3.2.2 VO Description XML Template – Actuator

```
<?xml version='1.0' encoding="UTF-8"?>
<icore.voregistry.api.VirtualObject>
  <uri>http://127.0.0.1:8040/lamp_1</uri>
  <type>http://purl.oclc.org/NET/ssnx/ssn#Actuator</type>
  <deploymentInfo>http://127.0.0.1:8040/lamp_1/deploy.wsdl</deploymentInfo>
  <status>AVAILABLE</status>
  <mobility>false</mobility>
```

```
<ict__object>
  <uri>http://127.0.0.1:8040/lamp_1/ict</uri>
  <non__ict__objects>
    <icore.voregistry.api.NonICTObject>
      <uri>http://127.0.0.1:8040/lamp_1/ict/non_icts/Smart_Home</uri>
      <type>http://www.loa.istc.cnr.it/ontologies/OWN/OWN.owl#Room</type>
      <name>Smart Home</name>
      <geoLocation>
        <uri>http://127.0.0.1:8040/lamp_1/ict/non_icts/Smart_Home/geolocation</uri>
        <longitude>300.0</longitude>
        <latitude>300.0</latitude>
        <altitude>0.0</altitude>
      </geoLocation>
    </icore.voregistry.api.NonICTObject>
  </non__ict__objects>
  <geoLocation>
    <uri>http://127.0.0.1:8040/lamp_1/ict/geolocation</uri>
    <longitude>302.0</longitude>
    <latitude>536.9</latitude>
    <altitude>0.0</altitude>
  </geoLocation>
  <ict__parameters/>
</ict__object>
<vo__parameters>
  <icore.voregistry.api.VOParameter>
    <uri>http://127.0.0.1:8040/lamp_1/vo_parameters/Primary_Communication_Protocol</uri>
    <name>Primary Communication Protocol</name>
    <meta__features>
      <icore.voregistry.api.MetaFeatureSet>

<uri>http://127.0.0.1:8040/lamp_1/vo_parameters/Primary_Communication_Protocol/meta_features_sets/Proto
cols.owl#Protocol_Type</uri>
        <type>http://www.iot-icore.eu/ontologies/Protocols.owl#Protocol_Type</type>
        <value class="string">HTTP-REST</value>
      </icore.voregistry.api.MetaFeatureSet>
    </meta__features>
    <access__rights/>
    <billing__costs/>
  </icore.voregistry.api.VOParameter>
</vo__parameters>
<vo__functions>
  <icore.voregistry.api.VOFunction>
    <uri>http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting</uri>
    <name>Indoor Lighting</name>
    <description>This function is used for the indoor lighting.</description>
    <input>
      <string>TURN ON</string>
      <string>TURN OFF</string>
    </input>
    <output>
      <string>Light</string>
    </output>
    </access__rights/>
    <billing__costs/>
    <input__parameters>
      <icore.voregistry.api.InputParameter>
        <uri>http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/input_parameters/TURN_ON</uri>
        <name>TURN ON</name>
        <meta__features>
          <icore.voregistry.api.MetaFeatureSet>

<uri>http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/input_parameters/TURN_ON/meta_features_s
ets/www.datatype.org#</uri>
            <type>http://www.datatype.org#</type>
            <value class="string">Integer</value>
          </icore.voregistry.api.MetaFeatureSet>
          <icore.voregistry.api.MetaFeatureSet>

<uri>http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/input_parameters/TURN_ON/meta_features_s
ets/planning#pre</uri>
            <type>http://www.uprc.gr/ontologies/planning#pre</type>
            <value class="string">ambient_luminosity(SH)</value>
          </icore.voregistry.api.MetaFeatureSet>
        </meta__features>
```

```
          </icore.voregistry.api.InputParameter>
          <icore.voregistry.api.InputParameter>

<uri>http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/input_parameters/TURN_OFF</uri>
            <name>TURN OFF</name>
            <meta__features>
              <icore.voregistry.api.MetaFeatureSet>

<uri>http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/input_parameters/TURN_OFF/meta_features_
sets/www.datatype.org#</uri>
                <type>http://www.datatype.org#</type>
                <value class="string">Integer</value>
              </icore.voregistry.api.MetaFeatureSet>
              <icore.voregistry.api.MetaFeatureSet>

<uri>http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/input_parameters/TURN_OFF/meta_features_
sets/planning#pre</uri>
                <type>http://www.uprc.gr/ontologies/planning#pre</type>
                <value class="string">ambient_luminosity(SH)</value>
              </icore.voregistry.api.MetaFeatureSet>
            </meta__features>
          </icore.voregistry.api.InputParameter>
        </input__parameters>
        <output__parameters>
          <icore.voregistry.api.OutputParameter>
            <uri>http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/output_parameters/Light</uri>
            <name>Light</name>
            <meta__features>
              <icore.voregistry.api.MetaFeatureSet>

<uri>http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/output_parameters/Light/meta_features_se
ts/planning#post</uri>
                <type>http://www.uprc.gr/ontologies/planning#post</type>
                <value class="string">light(SH)</value>
              </icore.voregistry.api.MetaFeatureSet>
            </meta__features>
          </icore.voregistry.api.OutputParameter>
        </output__parameters>
        <function__features>
          <icore.voregistry.api.VOFunctionFeature>
            <uri>http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/function_features/Energy</uri>
            <type>http://www.iot-icore.eu/ontologies/features.owl#Cost</type>
            <name>Energy</name>
            <value>1.0</value>
          </icore.voregistry.api.VOFunctionFeature>
          <icore.voregistry.api.VOFunctionFeature>

<uri>http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/function_features/Quality</uri>
            <type>http://www.iot-icore.eu/ontologies/features.owl#Utility</type>
            <name>Quality</name>
            <value>3.0</value>
          </icore.voregistry.api.VOFunctionFeature>
          <icore.voregistry.api.VOFunctionFeature>

<uri>http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/function_features/Network</uri>
            <type>http://www.iot-icore.eu/ontologies/features.owl#Cost</type>
            <name>Network</name>
            <value>1.0</value>
          </icore.voregistry.api.VOFunctionFeature>
          <icore.voregistry.api.VOFunctionFeature>

<uri>http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/function_features/Security</uri>
            <type>http://www.iot-icore.eu/ontologies/features.owl#Utility</type>
            <name>Security</name>
            <value>3.0</value>
          </icore.voregistry.api.VOFunctionFeature>
          <icore.voregistry.api.VOFunctionFeature>

<uri>http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/function_features/Expenditure</uri>
            <type>http://www.iot-icore.eu/ontologies/features.owl#Cost</type>
            <name>Expenditure</name>
            <value>1.0</value>
          </icore.voregistry.api.VOFunctionFeature>
          <icore.voregistry.api.VOFunctionFeature>
```

```
<uri>http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/function_features/Performance</uri>
        <type>http://www.iot-icore.eu/ontologies/features.owl#Utility</type>
        <name>Performance</name>
        <value>3.0</value>
      </icore.voregistry.api.VOFunctionFeature>
    </function__features>
  </icore.voregistry.api.VOFunction>
</vo__functions>
<vo__owner>
  <uri>http://www.iot-icore.eu/Users#DimitrisKelaidonis</uri>
  <api__key>███████████</api__key>
</vo__owner>
<vo__consumers/>
</icore.voregistry.api.VirtualObject>
```

Table 9: XML VO Description Example - Lamp Actuator

3.2.3 VO Description JSON Template – Sensor

```
{
    "type":"http://purl.oclc.org/NET/ssnx/ssn#Sensor",
    "deploymentInfo":"http://127.0.0.1:8040/accelerometer/deploy.wsdl",
    "status":"AVAILABLE",
    "mobility":false,
    "ict_object":{
        "non_ict_objects":[
            {
                "type":"http://www.loa.istc.cnr.it/ontologies/OWN/OWN.owl#Room",
                "name":"Smart Home",
                "geoLocation":{
                    "longitude":300.0,
                    "latitude":300.0,
                    "altitude":0.0,
                    "uri":"http://127.0.0.1:8040/accelerometer/ict/non_icts/Smart_Home/geolocation"
                },
                "uri":"http://127.0.0.1:8040/accelerometer/ict/non_icts/Smart_Home"
            }
        ],
        "geoLocation":{
            "longitude":260.2,
            "latitude":1160.0,
            "altitude":0.0,
            "uri":"http://127.0.0.1:8040/accelerometer/ict/geolocation"
        },
        "ict_parameters":[

        ],
        "uri":"http://127.0.0.1:8040/accelerometer/ict"
    },
    "vo_parameters":[
        {
            "name":"Primary Communication Protocol",
            "meta_features":[
                {
                    "type":"http://www.iot-icore.eu/ontologies/Protocols.owl#Protocol_Type",
                    "value":"HTTP-REST",

"uri":"http://127.0.0.1:8040/accelerometer/vo_parameters/Primary_Communication_Protocol/meta_features_s
ets/Protocols.owl#Protocol_Type"
                }
            ],
            "access_rights":[],
            "billing_costs":[],
            "uri":"http://127.0.0.1:8040/accelerometer/vo_parameters/Primary_Communication_Protocol"
        }
    ],
    "vo_functions":[
        {
            "name":"Accelerometer Monitoring",
            "description":"This function is used for the monitoring of the space coordinates.",
            "input":[
                "3D Point"
            ],
```

```
        "output":[
            "Accelerometer"
        ],
        "access_rights":[],
        "billing_costs":[],
        "input_parameters":[
            {
                "name":"3D Point",
                "meta_features":[
                    {
                        "type":"http://www.uprc.gr/ontologies/planning#pre",
                        "value":"sensorsOK(SH)",

"uri":"http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/input_parameters/3D_Po
int/meta_features_sets/planning#pre"
                    }
                ],

"uri":"http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/input_parameters/3D_Po
int"
            }
        ],
        "output_parameters":[
            {
                "name":"Accelerometer",
                "meta_features":[
                    {
                        "type":"http://www.uprc.gr/ontologies/planning#post",
                        "value":"acceleration(SH)",

"uri":"http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/output_parameters/Acce
lerometer/meta_features_sets/planning#post"
                    }
                ],

"uri":"http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/output_parameters/Acce
lerometer"
            }
        ],
        "function_features":[
            {
                "type":"http://www.iot-icore.eu/ontologies/features.owl#Cost",
                "name":"Energy",
                "value":1.0,

"uri":"http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/function_features/Ener
gy"
            },
            {
                "type":"http://www.iot-icore.eu/ontologies/features.owl#Cost",
                "name":"Network",
                "value":1.0,

"uri":"http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/function_features/Netw
ork"
            },
            {
                "type":"http://www.iot-icore.eu/ontologies/features.owl#Utility",
                "name":"Performance",
                "value":3.0,

"uri":"http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/function_features/Perf
ormance"
            },
            {
                "type":"http://www.iot-icore.eu/ontologies/features.owl#Cost",
                "name":"Expenditure",
                "value":1.0,

"uri":"http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/function_features/Expe
nditure"
            },
            {
                "type":"http://www.iot-icore.eu/ontologies/features.owl#Utility",
```

```
                    "name":"Quality",
                    "value":3.0,

"uri":"http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/function_features/Qual
ity"
                },
                {
                    "type":"http://www.iot-icore.eu/ontologies/features.owl#Utility",
                    "name":"Security",
                    "value":3.0,

"uri":"http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring/function_features/Secu
rity"
                }
            ],
            "uri":"http://127.0.0.1:8040/accelerometer/vo_functions/Accelerometer_Monitoring"
        }
    ],
    "vo_owner":{
        "api_key":"███████████████████████",
        "uri":"http://www.iot-icore.eu/Users#DimitrisKelaidonis"
    },
    "vo_consumers":[

    ],
    "uri":"http://127.0.0.1:8040/accelerometer"
}
```

<div align="center">Table 10: JSON VO Description Example - Accelerometer Sensor</div>

3.2.4 VO Description JSON Templates - Actuator

```
{
    "type":"http://purl.oclc.org/NET/ssnx/ssn#Actuator",
    "deploymentInfo":"http://127.0.0.1:8040/lamp_1/deploy.wsdl",
    "status":"AVAILABLE",
    "mobility":false,
    "ict_object":{
        "non_ict_objects":[
            {
                "type":"http://www.loa.istc.cnr.it/ontologies/OWN/OWN.owl#Room",
                "name":"Smart Home",
                "geoLocation":{
                    "longitude":300.0,
                    "latitude":300.0,
                    "altitude":0.0,
                    "uri":"http://127.0.0.1:8040/lamp_1/ict/non_icts/Smart_Home/geolocation"
                },
                "uri":"http://127.0.0.1:8040/lamp_1/ict/non_icts/Smart_Home"
            }
        ],
        "geoLocation":{
            "longitude":302.0,
            "latitude":536.9,
            "altitude":0.0,
            "uri":"http://127.0.0.1:8040/lamp_1/ict/geolocation"
        },
        "ict_parameters":[

        ],
        "uri":"http://127.0.0.1:8040/lamp_1/ict"
    },
    "vo_parameters":[
        {
            "name":"Primary Communication Protocol",
            "meta_features":[
                {
                    "type":"http://www.iot-icore.eu/ontologies/Protocols.owl#Protocol_Type",
                    "value":"HTTP-REST",
                    "uri":"http://127.0.0.1:8040/lamp_1/vo_parameters/Primary_Communication_Protocol/meta_fe
atures_sets/Protocols.owl#Protocol_Type"
                }
            ],
            "access_rights":[],
```

```
                "billing_costs":[],

                "uri":"http://127.0.0.1:8040/lamp_1/vo_parameters/Primary_Communication_Protocol"
        }
    ],
    "vo_functions":[
        {
            "name":"Indoor Lighting",
            "description":"This function is used for the indoor lighting.",
            "input":[
                "TURN ON",
                "TURN OFF"
            ],
            "output":[
                "Light"
            ],
            "access_rights":[],
            "billing_costs":[],

            "input_parameters":[
                {
                    "name":"TURN ON",
                    "meta_features":[
                        {
                            "type":"http://www.datatype.org#",
                            "value":"Integer",
                            "uri":"http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/input_parameters/
TURN_ON/meta_features_sets/www.datatype.org#"
                        },
                        {
                            "type":"http://www.uprc.gr/ontologies/planning#pre",
                            "value":"ambient_luminosity(SH)",
                            "uri":"http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/input_parameters/
TURN_ON/meta_features_sets/planning#pre"
                        }
                    ],
                    "uri":"http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/input_parameters/TURN_O
N"
                },
                {
                    "name":"TURN OFF",
                    "meta_features":[
                        {
                            "type":"http://www.datatype.org#",
                            "value":"Integer",
                            "uri":"http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/input_parameters/
TURN_OFF/meta_features_sets/www.datatype.org#"
                        },
                        {
                            "type":"http://www.uprc.gr/ontologies/planning#pre",
                            "value":"ambient_luminosity(SH)",
                            "uri":"http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/input_parameters/
TURN_OFF/meta_features_sets/planning#pre"
                        }
                    ],
                    "uri":"http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/input_parameters/TURN_O
FF"
                }
            ],
            "output_parameters":[
                {
                    "name":"Light",
                    "meta_features":[
                        {
                            "type":"http://www.uprc.gr/ontologies/planning#post",
                            "value":"light(SH)",
                            "uri":"http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/output_parameters
/Light/meta_features_sets/planning#post"
                        }
                    ],
                    "uri":"http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/output_parameters/Light
"
                }
            ],
```

```
        "function_features":[
            {
                "type":"http://www.iot-icore.eu/ontologies/features.owl#Cost",
                "name":"Energy",
                "value":1.0,
                "uri":"http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/function_features/Energ
y"
            },
            {
                "type":"http://www.iot-icore.eu/ontologies/features.owl#Utility",
                "name":"Quality",
                "value":3.0,
                "uri":"http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/function_features/Quali
ty"
            },
            {
                "type":"http://www.iot-icore.eu/ontologies/features.owl#Cost",
                "name":"Network",
                "value":1.0,
                "uri":"http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/function_features/Netwo
rk"
            },
            {
                "type":"http://www.iot-icore.eu/ontologies/features.owl#Utility",
                "name":"Security",
                "value":3.0,
                "uri":"http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/function_features/Secur
ity"
            },
            {
                "type":"http://www.iot-icore.eu/ontologies/features.owl#Cost",
                "name":"Expenditure",
                "value":1.0,
                "uri":"http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/function_features/Expen
diture"
            },
            {
                "type":"http://www.iot-icore.eu/ontologies/features.owl#Utility",
                "name":"Performance",
                "value":3.0,
                "uri":"http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting/function_features/Perfo
rmance"
            }
        ],
        "uri":"http://127.0.0.1:8040/lamp_1/vo_functions/Indoor_Lighting"
    }
    ],
    "vo_owner":{
        "api_key":"                                    ",
        "uri":"http://www.iot-icore.eu/Users#DimitrisKelaidonis"
    },
    "vo_consumers":[

    ],
    "uri":"http://127.0.0.1:8040/lamp_1"
}
```

Table 11: JSON VO Description Example - Lamp Actuator

3.3 VO Description RDF Example

```
3   <rdf:RDF
4       xmlns:icore="http://purl.oclc.org/NET/uprc/ontologies/icore#"
5       xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
6
7       xml:base="http://purl.oclc.org/NET/uprc/icore/VO/VO001">
8
9
10      <!-- http://purl.oclc.org/NET/uprc/icore/VO/VO001 -->
11      <rdf:Description rdf:about="http://purl.oclc.org/NET/uprc/icore/VO/VO001">
12          <rdf:type rdf:resource="http://purl.oclc.org/NET/uprc/ontologies/icore#Virtual_Object"/>
13          <icore:hasURI rdf:resource="http://purl.oclc.org/NET/uprc/icore/VO/VO001" />
14          <icore:hasType rdf:resource="http://purl.oclc.org/NET/ssnx/ssn#Sensor" />
15          <icore:hasDeploymentInfo rdf:resource="http://purl.oclc.org/NET/uprc/icore/VO/VO001/Deploy"/>
16          <icore:hasStatus>AVAILABLE</icore:hasStatus>
17          <icore:isMobile>false</icore:isMobile>
18          <icore:offersFunction rdf:resource="#Temperature_Monitoring"/>
19          <icore:represents rdf:resource="#Temperature_Sensor"/>
20      </rdf:Description>
21
22      <!-- http://purl.oclc.org/NET/uprc/icore/VO/VO001#Temperature_Sensor -->
23      <rdf:Description rdf:ID="Temperature_Sensor">
24          <rdf:type rdf:resource="http://purl.oclc.org/NET/uprc/ontologies/icore#ICT_Object"/>
25          <icore:hasURI rdf:resource="#Temperature_Sensor" />
26          <icore:hasICTLocation rdf:resource="#Physical_Location"/>
27          <icore:isAssociatedTo rdf:resource="#UPRC_Lab_Room"/>
28      </rdf:Description>
29
30      <!-- http://purl.oclc.org/NET/uprc/icore/VO/VO001#Temperature_Monitoring -->
31      <rdf:Description rdf:ID="Temperature_Monitoring">
32          <rdf:type rdf:resource="http://purl.oclc.org/NET/uprc/ontologies/icore#VO_Function"/>
33          <icore:hasInput>Air Temperature</icore:hasInput>
34          <icore:hasInput>Ambient Temperature</icore:hasInput>
35          <icore:hasInput>Environmental Temperature</icore:hasInput>
36          <icore:hasName>Monitoring Ambient Temperature</icore:hasName>
37          <icore:hasOutput>Temperature</icore:hasOutput>
```

> Information about the VO, that refers on VO URI, VO Type, the functions that it offers, etc

> Information about the ICT object that is represented by VO, e.g.: Location, association with non-ICT Object, etc.

> Description of the function, which is offered by VO, in terms of its inputs, outputs, etc.

Figure 7: RDF Description of Temperature sensor

4. References

[1] Taye, M. (2010). Understanding Semantic Web and Ontologies: Theory and Applications. Journal of Computing 2(6). ISSN 2151-9617. HTTPS://SITES.GOOGLE.COM/SITE/JOURNALOFCOMPUTING/. WWW.JOURNALOFCOMPUTING.ORG.

[2] Matthews, B. (2005). Semantic Web Technologies. JISC Technology and Standard Watch.

[3] W3C Standards – Semantic Web. http://www.w3.org/standards/semanticweb/. [Accessed: 12/2012].

[4] OWL Web Ontology Language Overview. http://www.w3.org/TR/owl-features/. [Accessed: 12/2012].