Grant Nº 285273

Glocal enterprise network
focusing on customer-centric collaboration

D2.1
Required Information/Knowledge Provision Services Specification

Edited by
UvA – H. Afsarmanesh
Date: 25 May 2012

Project funded by the European Commission under the
ICT – Factories of the Future Programme

Contract Start Date: 1 Sep 2011 **Duration:** 36 months

GloNet WP2

Service Oriented System Architecture for Collaboration

Deliverable data

Deliverable no & name	**D2.1– Required Information/Knowledge Provision Services Specification**
Main Contributors	Afsarmanesh Hamideh, Sijali Petro Korojelo, Ozgul Unal Karakas, Mahdi Sargolzaei – (UvA)
Other Contributors	Martin Kirchner, Holger Bar – (CAS) Luis Camarinha-Matos, Patricia Macedo – (UNINOVA)
Dissemination level (select)	Public
Date	May 2012
Status	FINAL

Deliverable summary

The deliverable D2.1 focuses on identification and analysis of the information/knowledge related to complex products in GloNet, while aiming at the specification of a set of generic services for their provision to the stakeholders. Towards achieving this goal, a ***systematic process*** is defined and followed. In this process, we carefully aim at *generalization,* and not only focusing on a specific product in GloNet. Therefore, the systematic process starts with targeted analysis of the three planned complex products of GloNet, namely: (i) the PV solar plants, (ii) the intelligent buildings, and (iii) the future incubators.

As a ***first step*** of the approach, the main *entities* common among these three products are discovered and extracted. These represent the elements, about which their information and/or knowledge needs to be stored and shared among different users in this environment, constituting the virtual *Bag Of Information/Knowledge Assets* (BOIKA) for each of the complex products. As a ***second step***, the main groups of information users (*including both stakeholders and software systems*) that are common to different GloNet products are addressed. In the ***third step,*** a cross section between entities from the first step and users from the second step is established, and their specific access requirements and/or constraints are identified, aiming to verify their relevance as well as their comprehensiveness. In the ***fourth step***, the results generated by the first step are analyzed and classified by their nature (/form) into the main categories of information/knowledge in the environment. *Our main resulted categories include:* ***atomic data***, ***documents***, *and* ***processes***. The ***fifth step*** primarily analyzes and classifies the results which are generated in the second step, by their nature (/kind), into the main categories of users in the environment. *Our main resulted categories include:* ***Human user*** *and* ***software systems***. In the ***sixth step*** it is discussed that in GloNet, access to information/knowledge from the virtual BOIKA repository, by *software systems* require them to be provided as *web services*. Furthermore it is discussed that access to information/knowledge by *human users* in GloNet requires them to be provided through user-friendly GUIs, where in turn, similar to the above, the software system of the GUI needs to access information/knowledge of the BOIKA, and thus also requires access through *web services*. Therefore, in the ***seventh and final step*** of the systematic approach, **high level generic specification** of the required web services for accessing through the cloud to the - *atomic data, documents, and processes,* related to GloNet products are provided. In the next step of the GloNet project, the specification of services which are provided in D2.1, serves as the base for their development, as will be addressed in D2.2.

TABLE OF CONTENTS

PROJECT-RELATED SUMMARY

GloNet focuses on three main application areas for its demonstration purposes. These areas include: (i) PV solar plants, (ii) Intelligent buildings, and (iii) Future incubators. In relation to every one of the three complex products of GloNet, there is one industry partner representing that product line in the project. Namely, iPLON for the PV solar plant, PROLON for the intelligent buildings, and SKILL for the future incubators. The research performed in WP2 – Service-oriented system architecture for collaboration – considers the heterogeneity among these different kinds of complex service-enhanced products, and aims at generalization of its provided solutions, such that its scientific and technical solutions are reusable and can be applied to other similar cases, i.e. those external to GloNet.

Deliverable D2.1 is the first deliverable of this workpackage and represents the results generated in Task 2.1. This task primarily aims at identification and characterization of the main types of required information/knowledge related to GloNet complex products, which need to be stored and shared among its diverse stakeholders, e.g. information about catalogues of equipment, business services, brochures and company profiles, lessons learned, etc. Furthermore, this task provides high level specification of a set of services (i.e. web services to be accessed through the cloud) that are required to be developed within the GloNet infrastructure, for accessing the needed information/knowledge in this environment.

The findings reported in this deliverable are generated through following a systematic 7-step process. The deliverable describes this applied approach and provides the results generated in Task 2.1, at each step of the process.

Aiming for generalization of the offered solutions, results generated at each step of this process are also verified and validated by the consortium partners representing the three lines of complex products, as mentioned above.

This deliverable has used as a starting point the results provided in D1.1 – Detailed Requirements for GloNet Use Case and Domain Glossary. Although that deliverable is mostly focused on the PV solar plants, it was useful to have the in-depth requirement analysis of one complex product. The identified set of main environment entities, and especially the identified set of main stakeholders for that product, served as the starting point for Task T2.1, to analyze the information/knowledge elements, as well as their access requirements and constraints within the step 1 and 2 of the applied systematic process.

The specification of generic services provided in D2.1 will serve as the base for the development of the interfaces that will be addressed in deliverable D2.2.

1. INTRODUCTION

In information and communication technology areas, information typically means data that has been processed and conveys an intended meaning to its users. Information differ from data, as data are just raw facts that have been collected but unprocessed, and thus without meta-data describing them, data are meaningless to the users [10,15]. Knowledge on the other hand represents some extracted facts from the provided data or information. In some other contexts also related to the GloNet, knowledge also sometimes refers to skills gained by someone from his/her past experience on something, or acquired through education. Scholars argue that both data and information are the means of attaining knowledge [17].

In GloNet project, information and knowledge represent important resources that are managed in a product environment, and are individually owned by the involved stakeholders. These are usually generated during different life cycle stages of complex products, e.g. during the operation, monitoring or control processes at solar PV plants, intelligent buildings, or future incubators.

Each of the above named products is complex, involving distinct stakeholders, including equipment and devices, and requiring business services and software systems that support different stages of its life cycle. The steps and processes involved during the product's design and engineering, construction and commissioning, and operation and maintenance phases are also complex. Furthermore, the above mentioned elements both handle and generate a large amount of information/knowledge that requires to be properly organized, stored and managed, so that it can be accessed and manipulated by different environment stakeholders.

For instance, a part of this information/knowledge needs to be easily shared among stakeholders for the purpose of accomplishing their tasks, where most tasks are complex and provided jointly by several stakeholders. Other parts need to be accessed for internal use of individual stakeholder organizations. The information/knowledge shared by the community of stakeholders around one product needs to be stored in a common virtual repository which we call the GloNet's *Bag of Information/Knowledge Assets (BOIKA)*. The virtual BOIKA repository consists of large amount of info/knowledge related to orders, product, equipment, stakeholders, business services, brochures, and processes descriptions among many others.

Different stakeholders in the GloNet environment, which range from customers, project developers, construction and commissioning firms, equipment suppliers, service provision companies, and many others, are considered to be either a member of a long term alliance of stakeholders involved in the area of the product – i.e. a VBE (Virtual organizations Breeding Environments) member [1,3,8], or a partner in a goal-oriented collaborative network which is established to perform a specific task for the product – i.e. a VO (Virtual Organizations) partner [3,6,9]. For different purposes, these stakeholders need to have access to virtual BOIKA repository, where each one may have specific requirements, preferences and constraints related to its access. As an example, while the customer is just interested to know which equipment, software system and/or business services to buy or acquire, the project developer goes further by being interested in who are the suppliers of these equipment, systems, and services which are needed by the customer, so as to be able to advice the customer on the best fit choices and which orders to make. With all the complex information/knowledge around each product as explained here, and considering that the sources generating them are also distributed, perhaps around the world, the process of providing information access as required by different stakeholders is complicated. That being the case, the information access infrastructure for the complex products needs to be carefully planned, considering the three-fold aspects. This means suitable mechanisms and processes must be defined so that each *stakeholder* is provided with the exact needed *information/knowledge*, from the exact *source* generating it.

This deliverable is centred around first investigating the above complexities for the GloNet complex products, and then proposing generalized reusable solution to the challenge of providing multi-stakeholder information / knowledge access from virtual BOIKA that are physically provided from multi-sources. The proposed approach aims at design and specification of common means to provide the needed information access. We have defined and followed a systematic 7-step process to achieve the aims of this deliverable, which is first defined and then elaborated within the next sections of this deliverable.

2. MOTIVATION AND APPROACH

The life span of service oriented complex products, such as those addressed in GloNet (i.e. PV solar plants, intelligent buildings, and future incubators), is typically more than twenty years. Furthermore, there are usually a large number of independent and autonomous organizations, with heterogeneous sizes, specialties, interests, etc. involved in the environment which supports the *Total Life Cycle* (TLC) of these complex products [2,3]. Therefore, the need for an infrastructure and the tools to support information/knowledge sharing in this environment is inevitable. Additionally, such an infrastructure shall address the physical distribution of the network and its elements, as also addressed below.

The basic need for formation of collaborative networks among the involved stakeholder organizations to support complex products, such as the ones addressed in GloNet is already identified [2,5], and at present some simple forms of such networks already exist. For instance, the InfraNet [10] is established in Germany as a network of enterprises that support PV solar plants' operation and maintenance. In fact, as already proven in both research and practice, within any collaborative networks, the more effective is their information/knowledge exchange, the better is the performance of the network, and thus in this case, the result would be better support for the complex products.

For all products addressed by the GloNet project, there are a number of main common stakeholders who need to collaborate and share their information/knowledge during the product's TLC stages. For example, in D1.1 [3], for supporting the TLC of the PV solar plants, we have identified the large group of involved stakeholders and their roles. Many elements related to the products, e.g. devices, software systems, etc. as well as many stakeholders, e.g. the business service providers, customers, equipment suppliers, etc. generate certain information/knowledge that can be provided as *sharable* to all other authorized stakeholders in the environment.

Information access and sharing plays a major role in collaborative networks, and in gluing the concepts and their definitions among its stakeholders. Some of the main gained advantages through information/knowledge sharing are mentioned below:

i. Information/knowledge sharing enables better understanding of the common project which the members of the network intend to accomplish. Here, the project can be as large as the complex product itself, or any subset of its needed activities and tasks, that can be aimed to be accomplished by a particularly formed goal-oriented VO network of stakeholders.

ii. Information/knowledge sharing brings about better understanding about different stakeholders, equipment, devices, tools, business services, software systems, etc., involved within the project, on which partners collaborate. For instance it provides the opportunity to select the best and most fit companies or organizations to from a goal-oriented VO network to achieve a project.

iii. Information/knowledge sharing can influence decision making within the intended project. For instance, analyzing some accessed information might reveal certain flaws in the performance of the project that requires immediate attention from certain members in the network.

The sharable information/knowledge within the collaborative network, such as those established around supporting the complex products, forms a virtual common pool/repository of information / knowledge resources called BOIKA (Bag of Information/Knowledge Assets) in this deliverable, containing all that can be shared by the stakeholders. Also each stakeholder requires accessing certain information/knowledge, and may have preferences and constraints for each access requirement. However, it should be noted that both the provider points (source) and the consumer points (sink) of the virtual BOIKA repository are typically dispersed geographically, and sometimes even located in different continents. Therefore, suitable interconnection/communication infrastructure and mechanisms shall be used to support this sharing environment among its sources and sinks. With the current state of ICT advances, storage/retrieval through the common virtual BOIKA pool can be supported through the *cloud computing environment*, as it is planned for the GloNet project.

But at the same time, it is too inefficient and cumbersome if the access and sharing of the elements through BOIKA is achieved bilaterally between every two stakeholders involved. Instead, novel *common mechanisms* are required to be introduced, in order to simplify both the provision of information/knowledge (to be shared with others) as well as their retrieval (to be accessed what is shared by others). Such common mechanisms can be provided through applying the *service oriented architecture* approach to accessing information/knowledge from the common pool of stored data. Therefore, the mechanism suggested in this deliverable is to develop *web services* for accessing different kinds of information/knowledge from the common virtual BOIKA repository, which shall be established in the cloud for each complex product.

We have developed a systematic process for performing our aimed task related to this deliverable. Below every step of the process is defined and explained:

Step-1. At this step we extensively investigate, discover, and extract the main *entities common* among the three complex products considered in GloNet. These represent the elements about which their information and/or knowledge needs to be stored and shared among different stakeholders. Therefore, these entities constitute the virtual *Bag Of Information/Knowledge Assets* (referred to as BOIKA in this deliverable) for each complex products. **(Section 3)**

Step-2. Similarly, at this step we extensively investigate and identify the main group of users (including both *stakeholders and software systems*) that are *common* to different complex products of GloNet. **(Section 4)**

Step-3. At this step, a cross section is established between the entities identified in Step-1 and the stakeholders identified in Step-2, together with identification of their specific access requirements and/or constraints. The main purpose of this step is to on one hand verify the relevance of both of these identified elements, and on the other hand to insure their comprehensiveness. **(Section 5)**

Step-4. For generalization purposes, at this step, the results generated in Step-1 are further analyzed and classified by their nature (or form) into the main categories of information/knowledge in the environment. **(Section 6.1)**

Step-5. Similarly, for generalization purposes, at this step we primarily analyze and classify the results which are generated in Step-2 by their nature (or kind), into the main categories of stakeholders in the environment. **(Section 6.2)**

Step-6. At this step, we cross check between the categories of information/knowledge generated in Step-4 versus the categories of stakeholders garneted in Step-5, to identify the specific needed forms of access by stakeholders to the information/knowledge stored in the virtual BOIKA repository. After our analyses, we propose the approach to provide the needed access. **(Section 6.3)**

Step-7. Finally at this step, we take the results generated in Step-6, and provide high level generic specifications for the services which enable access to the information/knowledge related to complex GloNet products, as it was aimed. As such the produced specifications are reusable for different complex products. **(Section 7)**

3. GloNet BAG OF INFO/KNOWLEDGE ASSETS (BOIKA)

In business and accounting, an asset is used to refer to anything which can be owned, that can produce economic benefits to its owning entity. It can for instance be cash, buildings, machinery, computers, automobiles, etc.

In collaborative networks literature, an asset is used to refer to for instance something owned by an individual stakeholder, and which is important for task execution by that particular stakeholder. Thus, here the notion of asset also includes the information / knowledge that exist in the environment. However, in collaborative networks, some of the assets owned by different stakeholders are not only useful to their owners, but also for other members collaborating with them, and therefore need to be shared. For the purpose of this deliverable, this is the case for the enterprises which are the members of a VBE alliance [3,8], organized around a complex product.

Assets shared in collaborative networks, are normally assumed to be kept in a common virtual asset repository, known as Bag of Assets (BOA), representing the fact that they can be shared with various authorized VBE members, and may include documents, machineries, software tools, etc. [1,8]. In this deliverable, only a part of the BOA which deals specifically with shared information and knowledge is addressed, and is referred to as the GloNet Bag of Information/Knowledge Asset (BOIKA).

BOIKA contains all valuable information/knowledge elements at the VBE, such as orders, profiles, lessons learned, best practices, etc. that different enterprises may share for their collaborative working. Some of these assets are the property of the VBE administrator, added to BOIKA for access/use by VBE Members, while others are owned by different VBE members and shared with some other authorized users, or with the entire community. The main purpose of sharing these VBE related information/knowledge assets is to speed up and improve co-working and co-innovation among VBE member.

As the ***first step*** of the systematic process defined in Section 2, in this section we concentrate mainly on identification of the main common set of entities related to all GloNet complex products, which their sharing is necessary for execution of various tasks during the TLC of the complex products. Therefore, we identify which information/knowledge is needed to be stored in BOIKA, to potentially be shared by others. In GloNet, normally this information/knowledge is accessed by the VBE members who possess the needed access rights from the owners of each piece of information/knowledge.

From our initial analysis of the three complex products of GloNet, which is performed together with the expert in GloNet consortium representing that complex product, we have identified ten main entities related to these complex products about which their information/knowledge need to be stored in the virtual BOIKA repository of GloNet, these entities include: equipment, support services, stakeholders, orders history, brochures, products, historic sensored data, generated knowledge (represented as processes), product-based VBE network, and the VO networks.
Figure 1 indicates these entities which are the main source of information/knowledge included in BOIKA.

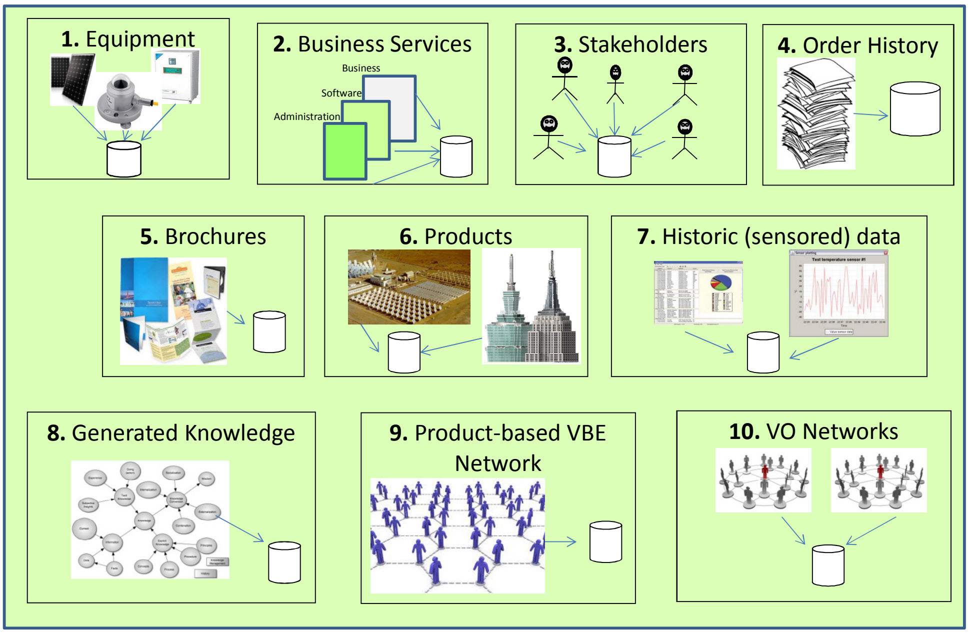

Figure 1: Bag of Info/Knowledge Assets related to Complex Products

The Following sub sections present short descriptions of each of these ten entities, and provide some examples for each within the three complex products of GloNet. It shall be noted that the detailed information/ knowledge related to each of these entities is our main concern, and is considered to be stored in the virtual BOIKA repository.

3.1 Equipment

Equipment here refers to any electrical and non-electrical item used within the intelligent buildings, future incubators, or PV solar plants constituting our complex products. Electrical components, may consist of devices to measure and control various environmental or electrical parameters in a complex product, such parameters may include: temperature or humidity for instance. Taking the example of a PV field here, electrical components may include devices, such as inverters, solar panels, grid connection equipment, etc. For intelligent buildings, they may include devices for Heating, Ventilation, and Air Conditioning (HVAC), lighting equipment, sunblind control devices, metering devices, etc. For future incubators, the equipment includes the computer systems or other devices and machinery needed in the experimental laboratories in the future incubators.
The non-electrical equipment includes all other components present on the three products. For instance, they include mounting structures and tables in both cases of PV plants and intelligent buildings, or office furniture and lab furniture for the case of future incubators.

3.2 Business Services

There are typically a large number of business services supporting different aspects of the TLC of complex products, e.g. supporting their operation and maintenance. These services may be either provided fully manually, or partially augmented with using some software systems.
As examples of manual services the preventive maintenance services, e.g. cleaning the solar panels in PV solar plant product or checking the performance of blinds in intelligent buildings

can be mentioned. As examples of business services that are performed by human, but augmented with the use of certain software system, the calibration of sensors, or the device configuration in an intelligent building, PV solar plant, or future incubator can be mentioned, which are always performed by an engineer that applies some software system and devices.

3.3 Stakeholder

The term stakeholder refers to a person, group of people, or an organization/enterprise that has certain direct or indirect association to another organization, and either has an effect on the organization or is affected by the organization's actions, decisions, and policies.

In GloNet, stakeholders refer to all organizations involved in different stages of the life cycle of complex products. The roles of these stakeholders differ widely, and some are very specific to the complex products. Nevertheless, we have identified a number of main stakeholder organizations that are common to all three complex products, as addressed in details in Section 4. The BOIKA repository needs to store the profile/competency information of these stakeholder organizations, so that they can be shared by all VBE and VO members in the complex product environments. For instance, before suitable stakeholders are selected as partners for a VO, their competencies are fully analysed, to determine if they can fulfil the emerged business objectives.

3.4 Order History

The history of orders for complex products is usually preserved at the VBEs for many years. These may consist of the product requirements, as specified by the customer, as well as a detailed list of all the ordered elements for the product. These include all devices and equipment, business services, software system, etc. which are ordered during the TLC of the complex product. Each order details differ from others, due to specificity of the ordered item, but every order is then detailed in a contract which will be stored in the BOIKA for future references. Preserving the order contract history can play an important role for protecting the rights and enforcing the legal aspects of the product, as well as for improving certain future decision making processes at the VBE level, through learning from the past experiences. For instance, by applying machine learning approaches to learn from the past orders of equipment, business services, and/or software systems when they are compared against the requirements specified by the customer, the system may be able to process a new customer request and make suggestions as to what can be ordered for it.

3.5 Brochures

Brochures are typically advertising tools for the functions of businesses and organizations. It gives business owners and organization managers a chance to sum up everything that customers need to know in an easy, simplified, and understandable way.

Storing brochures in GloNet's virtual BOIKA repositroy can play the following roles:

- Informative role: used to inform VBE stakeholders about other companies (i.e equipment or services providing companies). Also all updates to companies information are reflected in changes to their brochures.

- Advertising role: act as an advertising tool, which attractively allows a company to promote one or more of its offerings.

3.6 Products

When we speak of products in GloNet we refer to the three main kinds of complex products: PV solar plants, intelligent buildings, and future incubator facilities. The information/knowledge common to all 3 complex products is in fact gradually collected through the entire TLC, which is longer than 20 years. In other words, at every step of the way, information is generated and needs to stored, among which we can mention the physical design of the complex product, and all the planned devices and equipment, as well as all the needed procurements and hiring of business services and software systems, to all the operational and maintenance details performed at the site of the product. These and many other pieces of information/knowledge must be stored in the virtual BOIKA, so that when and if needed they can be retrieved by the users (addressed in Section 4) in the environment. We do not repeat these common aspects of the information/knowledge for the different 3 complex products below, rather just give a simple overview of the Intelligent buildings and the Future incubators complex products, while a long detailed definition of the PV solar plants is already provided in D1.1.

Intelligent Buildings

An intelligent building, also referred to as a smart home, is the building that is equipped with building automation features. Building automation represents advanced intelligent functionality provided and controlled/activated by an installed control system within the building [19]. The Building Automation System (BAS) is an example of a distributed control system. The control system is computerized, and is equipped with an intelligent network of electronic devices specially designed for monitoring and controlling the mechanical electronics in the building, e.g. the lighting systems and the sunblinds within the building [11,12].

The core functionalities typically provided in intelligent buildings include: i) keeping the building climate (e.g. temperature) within a specified range, ii) providing lighting based on an occupancy schedule, and iii) monitoring the system performance and device failures, while issuing warnings, e.g. email and/or text message notifications to the building engineering/maintenance office and staff once failure happens. The BAS which is established in intelligent buildings generates the advantage of reducing building's energy and maintenance costs compared to a non-controlled building. The complexity of information/knowledge related to the TLC of intelligent buildings is similar to that of the PV solar plants, which in turn requires effective storage and sharing of the information about this complex product among all its users.

Future Incubators

A future incubator, also referred to as a business incubator, is a physical environment (a building located usually within an industry/science park) and a program, which exists with the main purpose of supporting the establishment and growth of future innovative but small businesses, through offering different kinds of supporting services to their entrepreneurs [16]. Business supporting services offered usually include: equipped working space (e.g. use for design, prototype, demonstrate, and test new products), technical assistance, advice, coaching, business management skills, and mentorship among many others. Most future incubators offer such support during the early stages of entrepreneurship, where the involved

individuals wish to start and run start-up companies, while they do not have adequate finances to acquire these needed base services. The aim in most cases is to support the start and growth of Small and Medium Enterprises (SMEs) [13].
The so called Technology incubators support businesses in the area of Information and Communication Technologies (ICT), by offering to them services, such as internet connectivity, technical hardware needed for innovation purposes, as well as software solutions for these businesses [7].
One advantage of joining future incubators is the ability to meet experts who are able to help entrepreneurs in finding solutions to the specific challenges that a business faces. Due to the connections that future incubators have with financial organizations, a joining organization will also receive help and guidelines for acquiring funds needed for the business. Most future incubators also organize pitch sessions, to incubate entrepreneurs' pitches before a panel of financiers for demonstrations of their business ideas to them. Another advantage is capacity building and networking. Future incubators normally organize training sessions touching on areas like business plan writing, management, marketing and research, which go hand in hand with the running of any business. Some of them even provide special software solutions for dealing with such business issues.

3.7 Historic (Sensored) data

A PV solar plant, an Intelligent building, as well as a Business incubator facility all have control systems, typically SCADA systems that analyse large amount of captured data, collected from various sensors placed at the sites of the complex products and monitor their operation. These collected data (being through sensors, counting, or otherwise) are usually analysed to determine the performance of different aspects of the complex product and when and if necessary, to take appropriate measures to rectify the factors affecting their performance. There are various data collected from sensors, such as data about the temperature, humidity, irradiation, energy yield, etc. in the case of PV solar plant fields, about light, temperature, ventilation, etc. in the case of intelligent buildings, and about facility usage, light, ventilation, etc. in the case of future incubators.
The collected and analysed data at these products are usually not deleted; instead, they are backed up and saved in the backup storage devices for potential future analysis and learning purposes, thus constituting historic data for complex products. For instance, when a new project for a complex product is about to start, the previously saved data from other similar products can serve as the base for learning how to set up, configure, and/or tune and parameterize the devices and software systems for the new product. Another example is at the daily operation stage of each product, when the current collected data are normally compared to the saved data of the past in order to identify if there is any drop in the performance. Also the past data acts as the base for various maintenance activities carried out at the product site. For instance, when data indicates there is a low yield of electricity in a PV solar field, through analysis of the historic data, the plant administrator can determine that panels need to be cleaned, if they have covered by dust or snow.

3.8 Lessons Learned (Generated Knowledge)

The term knowledge is defined in Section 1. During the long TLC of the complex product, some pieces of knowledge are generated gradually that needs to be stored and shared through the virtual BOIKA at the product's VBE. The main kinds of gathered knowledge will include: i) lessons learned during different stages of the product life cycle, ii) results learned from the previous history of orders, iii) best achieved practices within the project, etc. Different formalisms may be used to specify each piece of knowledge, for instance representing them with a process description, as workflows, or with textual description (in documents), and even through audio/visual means. Knowledge can play an important role in a product VBE, if properly stored and shared by all VBE members. For instance, through sharing the lessons learned and best practices within the VBE, the provision of business services by service providers for the product can be very much improved.

3.9 Product-based VBE Network

As mentioned earlier, and in details in D1.1, the Virtual Breeding Environments (VBEs) [1,8] represent the long-term strategic alliances of organizations that offer necessary conditions to support the configuration of Virtual Organizations (VOs) [6,9]. For supporting the complex products, the established VBEs are product-based, meaning that a single VBE is formed around one product, and typically supports the total long life cycle of the product.
VBEs are centred around establishing an environment for the formation of: i) cooperation agreements, ii) common operation principles, iii) common interoperable infrastructures, iv) common ontologies, and v) trust among their members, among others. The main objective of VBEs is to prepare member VOs to collaborate in case any business opportunities arises [8]. As such large amount of information is typically gathered inside the VBEs, in relation to their establishment, regulation, and past organized VOs (VO inheritance data), which are valuable to be shared by the user community around the product.

3.10 Task-Based VO Networks

A VO is a temporary alliance of independent organizations that share common resources and skills to achieve certain common goal or mission, and whose cooperation is supported by computer networks [6,9].
The VO networks are goal-oriented, meaning that the network partners are driven by continuous production and/or service provision activities until the goal of the VO is achieved, at which time the VO dissolves.
For supporting the complex products, and within the VBE alliance which is formed with all the organizations and enterprises related to one product, different goal oriented VOs are formed, towards achieving specific goals, such as performing different specific business services for the product. For example, a VO will be formed to perform the commissioning of an intelligent building, another VO can be formed to deliver the required monitoring and control system of a PV solar plant, or a VO can be formed to install certain laboratory at a Future incubator, etc. Clearly large amount of information/knowledge is incrementally generated inside each VO, which needs to be stored and shared by all its partners. After completing its tasks and reaching its goal, the VO will dissolve and all its relevant data will be transferred to the VBE, which will be accessible only for authorized organizations either within the respective VO or within the VBE. Clearly some data in the VO over which there may be non-disclosure agreement (NDA), will be handled accordingly.

4. MAIN USERS OF INFORMATION/KNOWLEDGE

The ***second step*** of the process towards the main goal of this deliverable which is providing service specification for BOIKA access is to identify the *common set of users of information and knowledge among the three kinds of complex products of GloNet*, as identified in section 3.

For this purpose we investigated each of these complex products, namely the PV solar plants, intelligent buildings and future incubators, and identified which kinds of users there are in their VBE environment that need to access the information through BOIKA. We also identified the user requirements, when accessing the information/knowledge associated to the product. Furthermore, we narrowed down our study by only concentrating on the common set of usages/users as they can be identified in all three products. Several of these usages/users come from the stakeholder organizations in the environment, e.g. from the equipment suppliers, or consultants, etc., while some other usages/users are initiated within some ICT-based systems that require to access one or more piece of information / knowledge associated with a product.

Ten sets of specific usages/users were identified to be common among the 3 kinds of products, as represented in Figure 2, where seven types of them represent different stakeholder organizations, and the last three constitute the main ICT-based systems.

As further explained in this section, the seven common stakeholder organizations who need to access information/knowledge include: i) Project developing firms and Engineering, Procurement and Construction (EPC) companies, ii) Consultants, iii) Equipment suppliers, iv) Service provision companies, v) Software system provision companies, vi) VBE administrator, and vii) VO coordinator, The three ICT-based systems include: i) Monitoring systems, ii) Controlling systems, and iii) Operation management systems.

MAIN INFO/KNOWLEDGE USERS

1. EPC / Project Developing Firms
2. Consultants
3. Equipment Suppliers
4. Business Services Companies
5. Software Systems Companies
6. VBE Administrator
7. VO Coordinators

---

8. Monitoring Systems
9. Controlling Systems
10. Operation Management Systems

Figure 2: Main Information/Knowledge Users

This section provides a brief overview of each of these kinds of users. But prior to that, as described in D1.1 [3] and also mentioned earlier in this deliverable, all stakeholders in the environment of a complex product are a member of that product-based VBE network, and also some of these VBE members will be involved in different task-based VO networks. However, there are two specific stakeholders in each of these two networks, e.g. the administrator of the VBE and the coordinator of the VO, who may need to access certain proprietary information/knowledge in the environment, which requires special authorization that other members of these two networks cannot acquire.
In Section 5, we will further discuss the individual requirements, needs, and constraints for each of these users when accessing information/knowledge in BOIKA.

4.1 Project Developing Firms / Engineering, Procurement & Construction (EPC) Companies

For establishing all three identified complex products in GloNet, the project developing firm and the EPC perform the same role.
The project developing firm is typically the company that is responsible for coordination and management of the whole project for a complex product, from its initial set up phase to its operation and maintenance phase. The project developer has the role of finding all other players to be involved in the intended project, and provides several other needed tasks within the project [3]. As an example, if the project deals with establishing a new intelligent building, the project developer is responsible for finding, contracting, as well as monitoring and coordinating all activities performed by other companies, e.g. in dealing with equipment suppliers, service providers, construction and commissioning companies, etc. For this matter, the project developing firm is considered as the central body "the heart" of the whole project development and operation. Nowadays, usually instead of hiring the project developing firms, customers contact the EPC firms, which also perform all of the above, plus if needed also themselves get involved in the technical work needed for the project, such as the design and manufacturing of the needed control system, or even get involved in performing some needed business services for the product.

4.2 Consultants

Due to the complexity of the products, consultancy companies are typically hired for providing different kinds of consultancy services on different aspects of the on-going project related to establishing and assessment of readiness and/or performance of a complex product. The consultancy services can be provided to an investing firm (project owner company) directly, or to the project developing firm/EPC, which are responsible for establishing the technical details of the entire project.

4.3 Equipment Suppliers

The equipment related to each complex product are manufactured and provided by different manufacturing companies.
When there is a new project to be started, the project developer and/or the EPC, with the agreement of the investing company (the customer), contact the equipment suppliers, so as to purchase and deliver the intended equipment for the project.

Typically three distinct categories of equipment supplier companies exist in the VBE established around the product. These include: companies supplying electrical equipment and devices (e.g. computers, AC/DC inverters, etc.), companies supplying the non electrical equipment (e.g. tables, blinds, lab equipments, cables, etc.), and companies supplying the monitoring and control equipment and tools (e.g. temperture sensors, cameras, etc.). Details about the three categories of equipment are addressed in section 3.1.

4.4 Business Service Companies

A number of service provision companies provide different business services to the established projects.

The business services are the services that require human involvement in carrying them out, and sometimes additional special tools may be required for providing these service. To give a few examples of business services, consider: i) maintenance and repair services, ii) training services, iii) consultancy services, iv) sensor calibration services, v) product construction services, vi) site monitoring of the product, etc. All these types of business services are provided by specialized companies. While all business services include human involvement, some of these services in fact need to use some software systems to perform their tasks (e.g. calibration of sensors), but the others may be fully manual (e.g. product construction).

4.5 Software system Companies

A number of software system provision companies are also members of the product-based VBEs. These companies provide different software systems that are needed to support different functionality in the product environment.

Three main kinds of software systems can be identified for supporting the complex products, which include: monitoring systems, controlling systems, and operation management systems. These software systems provide fundamental support to the proper operation of the products, and are procured or hired by the product owner. Usually these systems run continuously at the site of the product, measuring different aspects of the product and taking actions when the gathered data falls outside acceptable boundaries. The software system providers for complex products work closely with the customers, and frequently need to innovate, considering that the complex products considered in GloNet are still not fully mature products. Many new requirements and needs pop up every day in their environment, and customers frequently ask for new or better assistance or automated support through the software systems.

4.6 VBE Administrator

Members of product-based VBEs shall constitute all stakeholders involved with the product. Their involvement might be for instance due to the provision of certain business services, dealing with product development, supplying some equipment to the product, or even coordinating and managing the operation of product, etc. As such, normally the VBE members support all different aspects and requirements of the product during its TLC.

Within such VBE network, the VBE administrator is the organization that is responsible for its effective operation [1,3,8], as mentioned below. Besides managing its members and providing their structured profiles, the VBE administrator identifies relevant market opportunities for the VBE members to get involved. It also provides a common infrastructure as well as working and sharing policies among autonomous organizations, in order to increase their chances of successfully co-working in potential future VOs. Furthermore, the VBE administrator provides ICT-based support tools which assist the brokers of VOs with finding most-fit partners for the VOs that can together address the emerged market opportunity.

4.7 VO Coordinators

Organizations that form a VO are known as VO partners. Each partner in the VO has specific roles and responsibilities towards achieving its common set of goals. VOs are goal-oriented, meaning that they are targeted at achieving certain specific goal, and once the goal is achieved the VO will be dissolved. A goal can be for instance, construction of a new power plant, and so once the plant is built, the established VO has fulfilled its task and hence it is dissolved.
However one important aspect of the VOs is for them to operate successfully and to cope with all unexpected events during its life cycle. While some of these events are caused by the market dynamism, many others are in fact due to the everyday exceptions raised within the organizations that are involved in the VO. The organization in charge of handling all the raised exceptions and coping with the required dynamism, as well as monitoring and controlling the planned activities of the VO during its life cycle, is called the VO coordinator.

4.8 Monitoring systems

These are the software systems, which are developed and/or provided by some enterprises that are members of the product-based VBE. These systems are specifically designed to deploy the devices and equipment included in the product, and sometimes they need to also exchange data with other software systems related to the product. Primarily, the important task of observing and gathering important information about the operation and performance of the product is performed by the monitoring systems. A monitoring system itself constitutes one of the different kinds of information/knowledge users centred around the product, and need to receive certain data (from the BOIKA repository) from the product site as direct input, through its installed devices and equipment, e.g. through its sensors, counters, etc.
As such monitoring systems need to access a large amount of data from the product environment in near real-time. The monitored data then gets analysed by these systems, while the generated analysis results themselves constitute input for other users involved with the product.

4.9 Controlling systems

Control software systems receive their input from the devices, and equipment, as well as other software systems related to the product. These systems are also developed and/or provided by some of the enterprises that are the members of the product-based VBE. Control systems are specifically designed to work with the devices and equipment used in the product, as well as with the other software systems running at the product. Primarily, the important task of controlling the monitored information about the operation and performance of the product is performed by these systems. As such control systems also constitute one of the different kinds of information/knowledge users centred around the product, and need to receive data from the product site as direct input, either through the devices and equipment installed at the site, or through other software systems. Similar to monitoring system, the controlling systems also need to access and analyse a large amount of data from the product environment in near real-time. The results generated by this software system aim at assisting the control process of certain functionality at the product. For instance, during summer, when the sun shines directly into the offices of an intelligent building (through the windows), it causes the rise in temperature in those offices, and therefore in such case, the input from the monitoring sensor data causes the control system to shut the blinds at the windows, in order to avoid the sun from shining in the rooms.

4.10 Operation management systems

These software systems are also developed and/or provided by some enterprises that are members of the product-based VBE. These systems are also specifically designed to work with the devices and equipment set in the product, as well as with other software systems in the product environment. The operation management systems perform certain tasks such as the generation of: product performance reports, billings, planning of next periodic maintenance, etc., based on their received information from the product environment. As such, these systems also constitute one of the different kinds of information/knowledge users centred around the product, and need to receive data from the product site, as direct input, through the installed devices and equipment, e.g. the sensors, the counters, as well as information about the stakeholders, business services, and orders related to the product, etc. Some of this information need to be accessed from the product environment in near real-time, i.e. the sensored and other gathered data. The gathered data are then analysed by the operation management systems, and the generated results, either printed forms or on-line documents, constitute important elements for some stakeholders in the product environment.

5. ACCESS REQUIREMENTS BY MAIN USERS OF INFO/KNOWLEDGE

In the first two steps of the systematic process, we identified the main entities for which we store information/knowledge in the virtual storage repository (BOIKA) in Section 3, and the main kinds of users needing to access the information/knowledge around a complex product in Section 4. In this section, we provide the results of step-3 of our process, and we identify the requirements and constraints for each of the users in relation to accessing their needed information/knowledge. This step also verifies the relevance of our findings in steps 1 and 2, through the mappings that are defined between the users and their needed information/knowledge as shown in Figure 3. Further to their cross-reference identification, also identify the varying requirements and preferences of different users when it comes to their needed access to information/knowledge. Taking the example of stakeholders and their needs and requirements for accessing information/knowledge about a complex product, normally the need and requirements are linked to the task in which the stakeholder is involved within the on-going project. For instance, for the purpose of coordinating the project, the project developing firm or EPC is interested in the information about all other various stakeholders who are involved in the current on-going project, however, the project consultant is not interested in such information, rather it is interested in the information about the complete product itself, as well as what is so far being ordered, so that it can propose what other equipment, business service, software system etc. are suitable to be ordered next for this product.

This section is centered on addressing the mappings or direct inter-relationships between different users in the complex product environment and the BOIKA repository elements. It also provides some description of the specific requirements and constraints identified for different stakeholders.

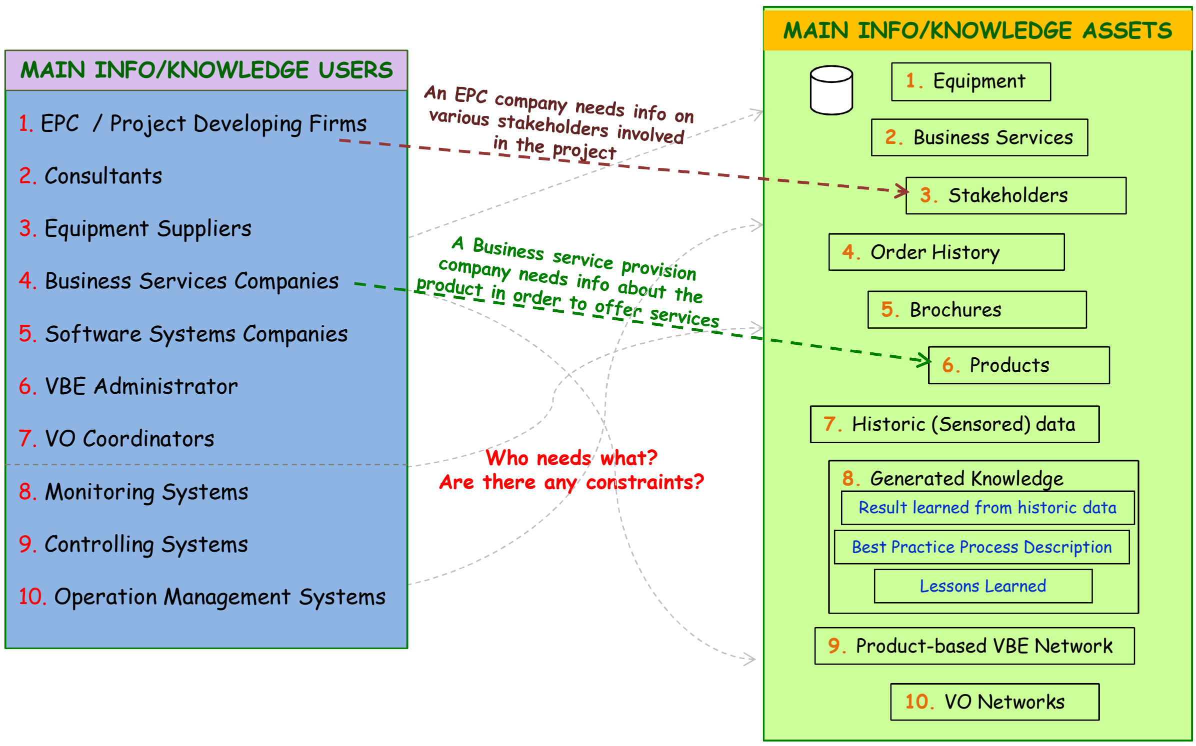

Figure 3: Identification of Main Stakeholders Requirements for BOIKA elements

In our analysis, we have come to the following findings:

5.1 EPC / Project Developing Firm

Both the project developing firm and EPC require accessing large amounts of information/knowledge to perform their tasks. For instance, the following types of information are needed to be made available through the BOIKA repository: equipment, business services, stakeholders, brochures, historic sensored data, product-based VBE network, and VO networks. Access to this information enables this user to perform its task that is to order the most suitable equipment, business service, software service, etc. for the product, as well as selecting the most suitable stakeholders among the VBE members to form the new VO for performing specific task in the product.

5.2 Consultant

A consultant usually deals with consultancy services at different phases of the product's TLC, once it is hired by an owning firm or the project developer/EPC. Normally most of today's complex products have more than one consultant, each providing consultancy services on specialized areas. The role of the consultant is usually either on what is the most suitable solution to order for the product, or on how to excel the performance of its current status [18]. Considering the nature of their tasks, it is then clear that consultants might need to access all the information about the product from the virtual BOIKA repository, to facilitate the provision of their consultancy services.

5.3 Equipment Supplier

Considering the complexity of the products, usually when the equipment supplier is contacted for supplying certain equipment, it has to understand the project needs so as to be able to supply needed equipment efficiently. Understanding the project is only possible through access to certain information/knowledge available related to the project. As such the need to access the following information/knowledge about the product is raised: product information, stakeholders, and brochures.

5.4 Business Service Provision Company

Business service provision companies require accessing the following types of information/knowledge for their effective involvement in providing the needed business services to the project: product information, equipment information, and order history.

5.5 Software system provision company

Software system provision companies require accessing the following types of information/knowledge for their effective involvement in providing the needed systems to the project: product information, equipment information, and order history.

5.6 VBE Members

Product-based VBEs aim to support the entire life cycle of the complex product. Within such a VBE, if authorized, its members require to access all information/knowledge on anything related to the product, to the extent that they help with accomplishing their intended tasks within the project. For this reason, the infrastructure for establishing and using the virtual BOIKA repository is in fact often managed by the VBE administrators, who also provide the common BOIKA access mechanisms and tools to their members.

5.7 VO Partners

Similar to VBE members, in the course of the life cycle of VOs, the VO partners require access to the information/knowledge stored in BOIKA repository, in order to carry out their VO tasks in a more effective way. Therefore, if authorized, the VO members might require accessing all the stored information/knowledge in BOIKA repository.

5.8 Monitoring systems

As an information/knowledge user in the product environment, the monitoring systems require access to the equipment information, as well as the historic sensored data, related to the product.

5.9 Controlling systems

Similar to the monitoring systems, as an information/knowledge user in the product environment, the control systems also require access to the equipment information, as well as the historic sensored data, related to the product.

5.10 Operation management systems

As another information/knowledge user in the product environment, the operation management systems require access to a number of different information/knowledge at this environment, including information about: equipment, stakeholders, business services, order history, as well as the historic sensored data, related to the product.

6. GENERALIZATION OF BOTH INFO/KNOWLEDGE ENTITIES AND USERS AND THEIR CROSS SECTION

This section addresses the results of Step-4, Step-5, and Step-6 of the systematic process, as introduced in Section 2, respectively within subsections 6.1, 6.2 and 6.3. As addressed in Section 5, different kinds of users need to access different kinds of information/knowledge related to complex products. This makes the task of defining access mechanisms (or access services) by each of the identified stakeholders to each of their needed data a binary effort, and if you add the autonomy and heterogeneity of each data providing node as well as the geographic distribution of these nodes to this picture, the problem of data access provision for complex products becomes challenging, requiring a large number of bilateral solutions between every pair of providing (source) and requesting (sink) nodes, and these solutions will not be reusable and generalizable to other cases.

Instead of the above solution, the steps 4 and 5 of the systematic process respectively focus on generalization of the types of entities and the types of users, in order to on one hand simplify the proposed solution and on the other hand make the solution reusable and applicable to other cases.

6.1 Nature-based categorization of information/knowledge entities

In Step-4, for generalization purposes, the results generated in Step-1, i.e. the ten kinds of information/knowledge to be stored in BOIKA, are further analyzed, and classified by their ***nature*** (or forms) into the main categories of existing information/knowledge related to the complex product environments. This in turn simplifies the tasks of defining access mechanism (access services) to various categories of information/knowledge.

The purpose as well as the gained advantage of this categorization by nature includes:

i. Simplifying the process of defining access mechanisms (access services) to different information/knowledge pieces by different users.
ii. Providing reusable access mechanisms.
iii. Enabling generalization of common control features for information protection.

As a result of our analysis, the information/knowledge about the entities identified in Step-1 is classified into 3 main categories, which represent the nature of their content. These categories include:

- **Atomic data** (representing data elements in BOIKA, typically retrieved from a database)
- **Documents** (representing files stored in BOIKA, can be textual, or including design diagram, etc.)
- **Processes** (representing the knowledge stored in BOIKA, in form of process descriptions)

Figure 4 shows the nature-based categories of information/knowledge related to complex products of GloNet, and Figure 5 represents the nature of the information/knowledge related to each of the 10 entities identified in Step-1.

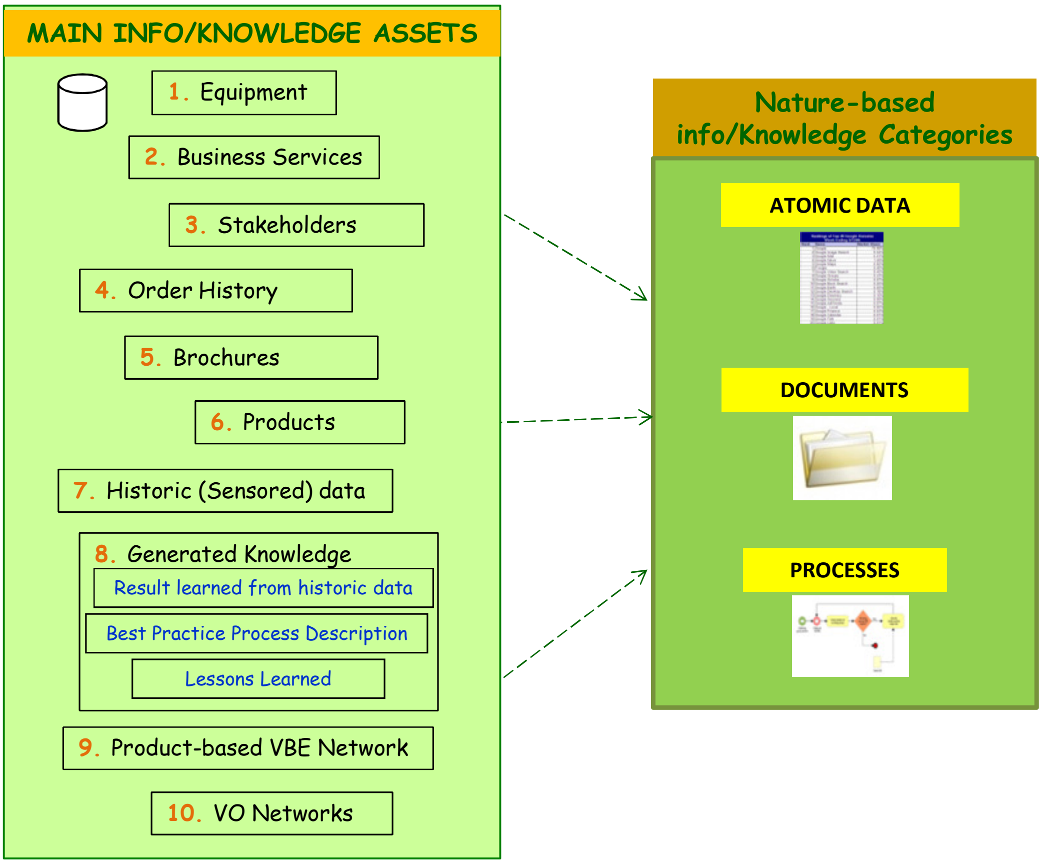

Figure 4: Three nature-based categories of the Information/Knowledge in BOIKA

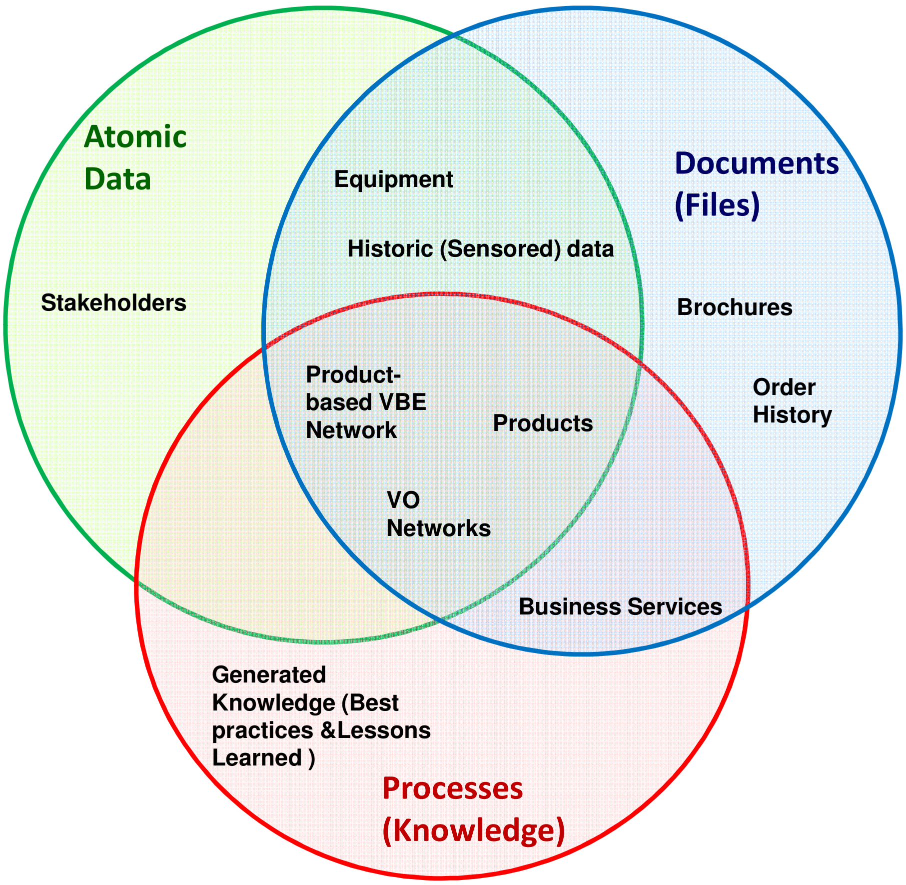

Figure 5: The nature of the Information/Knowledge related to the ten entities of Step-1

Based on the three needed categories of information/knowledge named above, in the next subsections we address which of the main entities introduced in Step-1 fall in each of these categories.

6.1.1 Atomic data category - information stored in a database

Among the identified ten main entities about which their information/knowledge is stored in the virtual BOIKA repository, several of them have certain data/information which needs to be stored within a database, these include the following:

i. *Equipment information* – A part of the information related to all equipment used in the product must be stored in a database. These specify for instance the manufacturer of the equipment and its other characteristics. These data are typically provided by equipment supplier companies.
ii. *Stakeholders information* – This includes storing in a database the stakeholder information e.g. related to profiles, roles and responsibilities of the stakeholders with respect to the product.
iii. *Product information* – A part of the information about the product needs to be stored in databases. As an example, this will include the information about the site of the product, its owner, and the devices and equipment which are used for developing it, among many others.
iv. *Historic (sensored) data* – A part of the information related to the generated and/or gathered data from equipment at the product needs to be stored in a database, for example, the information related to the name/type of the sensor from which certain data is gathered.
v. *Product-based VBE network information* – Large part of information about the VBEs are typically stored in a database. This will include for example the name of its members, the coordinator, as well as the names of the task-based VO networks which are established within the VBE.
vi. *VO Networks information* – A part of the information about each VO-network is stored in a database. This includes among others the name of its partners, the goal of the VO, etc./

6.1.2 Document (file) category - information stored in a content management system

Several of the ten entities identified in Step-1 have information in form of documents and/or files, which need to be stored in a document management system. By document we refer to a piece of written, printed, or electronic matter that provides information or evidence, and for instance can serve as an official record. The following information about entities in a product is document related:

i. *Brochures Information* – This includes all the brochures for all the stakeholder organizations involved in the VBE of the product.
ii. *Order History Information* - All the information related to the history of orders for complex products is represented as documents.
iii. *Equipment information* – A part of the information related to equipment used in the product constitutes documents. These include for instance the document representing the technical design of the equipment.

iv. *Product information* – A part of the information about the product needs to be stored as documents. As an example, this will include the information about the land ownership documents, signed legal permissions, etc.

v. *Historic (sensored) data* – The main part of all gathered data from the product are in files, which need to be stored in a document management system.

vi. *Product-based VBE network information* – A part of the information about the VBEs may include documents which need to be stored in a document management system. These include for instance the generated reports at the VBE and all the signed agreements within the VBE.

vii. *VO Networks information* – A part of the information about each VO-network constitutes document, e.g. the design diagrams for an innovative new solution prepared within a VO, etc.

viii. *Business Services* – A part of the information about the business services related to product, which is usually provided through a VO, is typically described in a document, which needs to be stored in a document management system.

6.1.3 Process (Knowledge) category - information stored in form of a process

A process can be defined as a collection of activities designed to produce a specific output, and implies a strong emphasis on how certain activity is performed. Process definitions focus mainly on formalizing the work activities, and indicate the time inter-relationships among the steps of the work. Each process definition indicates all the steps from the beginning to the end, and clearly defines the inputs and outputs, as well as a structure for actions to be taken [4]. Around the product, there are clearly several processes that need to be stored and shared by the users in the community.

Further to the processes, during the life cycle of a product there is a large amount of knowledge which is generated, mostly in forms of processes, e.g. representing best practices, or lessons learned on how to perform certain physical process or how to make certain decision in the environment. In all above cases, the generated knowledge can be formalized as a process. These processes can be effectively stored in form of either a workflow diagram, or a business process, etc.

The following information about entities in a complex product needs to be stored as a process:

i. *Business Services* – The main part of the information about the business services related to product, typically constitutes a process and can be represented in terms of a sequence of steps to accomplish a specified task.

ii. *Generated Knowledge (lessons learned, best practices, etc.)* – The generated knowledge or lessons learned related to product addresses how to best perform certain physical process or how to make certain decision in the product environment. Thus, they can be stored as a sequence of steps constituting a process.

iii. *Product information* – A part of the information about the product may consist of a set of processes, e.g. for ordering, purchasing, maintenance etc. that need to be stored as processes.

iv. *Product-based VBE network information* – A part of the information related to the VBEs may include certain processes, such as those for registering members, configuring a new VO, etc.

v. *VO Networks information* – A part of the information generated at each VO-network constitutes certain processes, for example in a VO focused on providing certain business service, the process for performing this activity needs to be stored as a business process, e.g. the business process for cleaning the solar panels at a PV solar plant product.

6.2 Nature-based categorization of users

Similar to the above, in Step-5, for generalization purposes, the results generated in Step-2, i.e. the ten kinds of users that need to access the information/knowledge stored in the BOIKA, are further analyzed, and classified by their ***nature*** (or kind) into the two known main categories of existing users related to the complex product environments. These include:

- **Human user** (representing the set of all common stakeholders)
- **Software system** (representing the 3 kinds of identified systems)

This in turn similarly simplifies the tasks of defining reusable access mechanism (access services) to various categories of information/knowledge.

Human users (stakeholders) typically access BOIKA to retrieve both information and knowledge that they need for making decision. On the other hand, a number of software systems running at the product site need to access BOIKA and retrieve only information as input for running their programs. A good of example of a software system is the Supervisory Control and Data Acquisition (SCADA) system for monitoring the performance of different complex products in GloNet. This system needs to be able to fetch the stored data from databases for monitoring the product's operation, and analyzing its related data so as to determine the performance of the product, and or controlling its operation. SCADA systems are auto-systems and directly interact with databases without being controlled by the human user. Figure 6 shows the classification of the main users of information/knowledge in GloNet into the two mentioned categories.

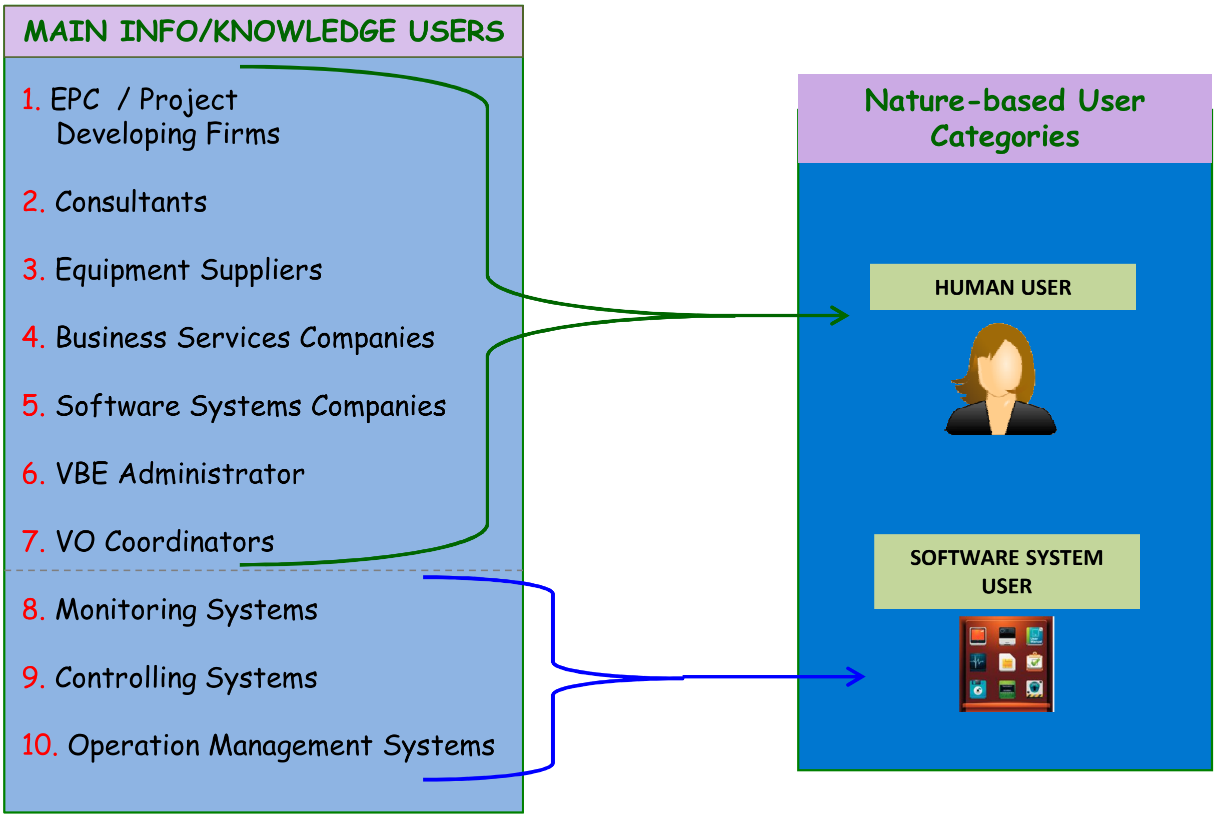

Figure 6: Categorization of the Main Users of Information/Knowledge in BOIKA

6.3 Cross section and analysis of step-4 and step-5 results

In Step-6, we cross check between the categories of information/knowledge generated in Step-4 versus the categories of users generated in Step-5, to identify the specific needed forms of access by stakeholders to the information/knowledge stored in the virtual BOIKA repository. After our analyses, at this stage we end up with the finalized proposal for a set of generalized/reusable access mechanism (access services) which need to be specified, but they will reusable and applicable to different cases.

First we give a bit of background. The complexity involved in specification of access mechanisms for different kinds of user to the variety of data at the complex products necessitates the development of generalized access mechanisms. During this process we managed to end up with the two main categories of users composed of the human users and the software systems. But from the practical cross section of the 2 kinds of users (from Step 5) and when considering their preferences, requirements, and constraints, we realized that all stakeholders' access to information/knowledge in BOIKA must be provided through a user-friendly GUI, which is itself a software system, and therefore, we concluded that even the human-user access to BOIKA would be indirectly performed through the software systems.

At this point, it should also be remembered that different stakeholder organizations, both the providers (source) and the requesters (sink) of information/knowledge in GloNet, are geographically distributed worldwide, e.g. currently some of them are in India while others are in Europe. Therefore, defining access to the shared information/knowledge at BOIKA requires provision of the cloud based infrastructures to store the virtual BOIKA repository,

and then applying service-oriented architecture approaches, web services can be developed as the means of access to BOIKA in the cloud.
As such, access to the information/knowledge stored in the virtual BOIKA repository is not a bilateral *physical access* from a source node to a sink node, rather it will be provided through web service calls that fetch the required information or knowledge through certain cloud storage, and return the results to the user. In other words, with this access mechanism, a user in any geographic location can send a request to the virtual repository for accessing certain information or knowledge, this request will be channelled through a web service which fetches the required information/knowledge and returns the answer to the user.

Figure 7 shows the described access mechanism for the two categories of users, to the three categories of information/knowledge. As presented, the access mechanisms are not completely the same for the two categories of users.
A human user uses a GUI through which to specify the needed access to certain information/knowledge category from BOIKA; this request goes from the GUI through a web service to BOIKA that retrieves the intended information/knowledge and returns the answer to the user. A software system user on the other hand, sends a direct request to BOIKA on the needed information/knowledge, and this request goes directly through a web service that fetches the required information/knowledge from the virtual storage and returns the answer to the software system.

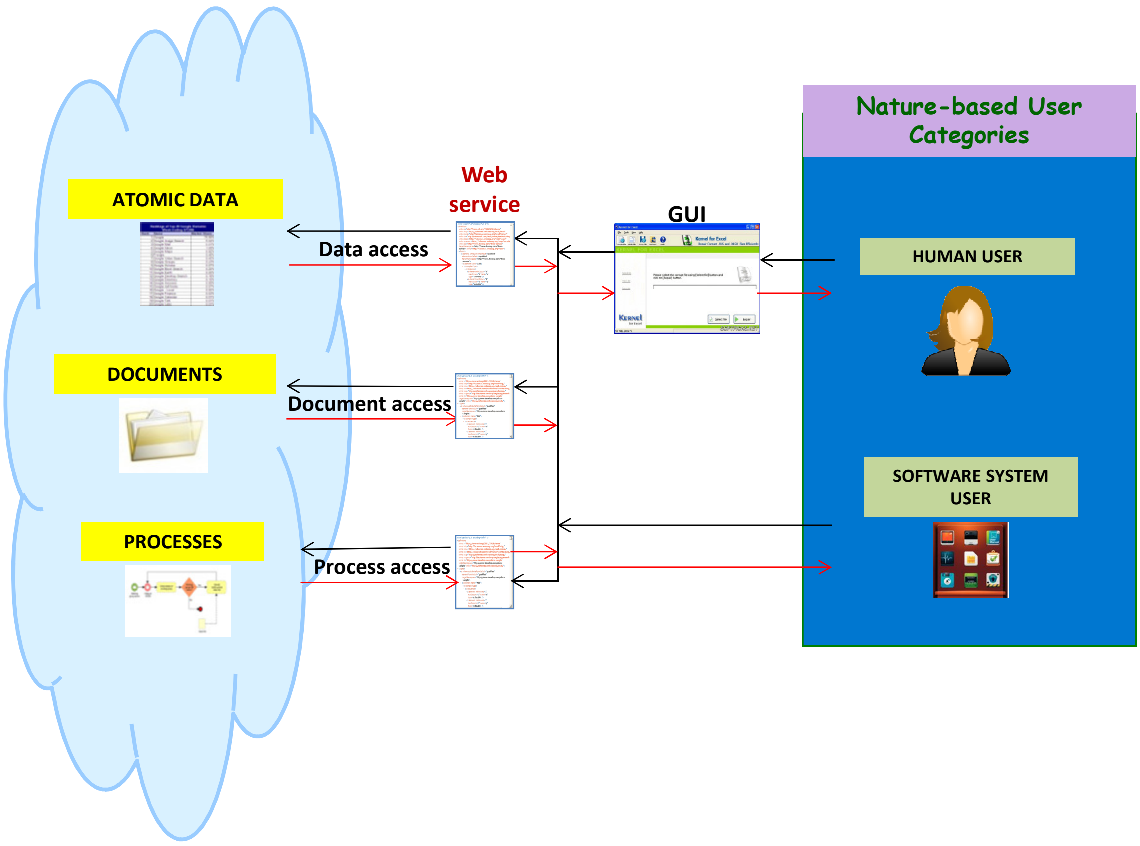

Figure 7: Access Mechanisms to BOIKA from two categories of Users

7. GENERIC SERVICE SPECIFICATION FOR BOIKA ACCESS

This section is centered on providing specifications for the three software services that we concluded at the end of Section 6. These three service specifications provide the generic access/retrieve, as well as the manipulation (creation, update, and delete) of the required information/knowledge in virtual BOIKA repository.
The challenging task in Step-7 of our systematic process is to provide the specification of a set of web services (marked as web services in Figure 7) to enable access to each of the intended information/knowledge category (i.e. atomic data, document, or process) in BOIKA, and that are generic and can be reused.

In our approach to specifying the needed software services when appropriate the following means are applied:

- Data model specifications for the needed services (UML based)
- Tabular description addressing specific features of the needed services (IOPE based)
- A set of example CRUD (Create, Retrieve, Update, Delete) operations (Soap based)

It should be noted that the specifications intended in this document are generic, meaning that they are neither centered on a single complex product (rather applies to all products), nor centered on any specific real piece of information/knowledge. They rather support all possible categories of information/knowledge. Therefore, they are reusable for different complex products and can support all possible information/knowledge related to them.

Generic tabular frame for service description

The detailed description we have designed for specification of software services addresses a set of their features, as introduced and described in Table 1. Generic software specification template shown in this table is rooted in IOPE (Input, Output, Preconditions, Effects) properties, which are used for functional description of services in Web Ontology Language for Services (OWL-S) [14].

Service Name	Human-readable label assigned to the service
Service Description	A brief textual description of the service
Data/Information Type	Type of data that the service operates on
Operation	Set of operations defined in a service
Input	Inputs provided as parameters to the service
Output	Expected outcome of the service
Pre-conditions	Conditions that need to be met for the service to execute
Effects	Effects of the service after its execution
Exceptions	Abnormal execution paths during the execution of the service
Timeliness Quality of Service	Required timing (i.e. responsiveness) QoS of the service
Synchronous/asynchronous access	Whether the service could be invoked in a synchronous fashion, or whether the execution is of 'fire & forget' nature, which could be as such that the invoker is synchronously provided a 'ticket', which he uses asynchronously to check the outcome

Table 1: Generic Software Service Specification Template

Generic data models for services manipulating information/knowledge in BOIKA

For BOIKA we specify a generic data representation to allow us handling all various types of entities (e.g. equipment, historic (sensored) data, products, stakeholders, VO networks, product based VBE networks, etc.) in a common generic way, and as a generic entity.
Such a generic entity, from the perspective of the web service, is called a *DataObject* which basically consists of an *objectType*, a set of fieldname-value-pairs and a set of access permissions (see Figure 8). To enable type-safety a fieldname-value-pair is encapsulated in a *Field* with a sub class for every supported primitive type (e.g. String, Integer, Long etc.).

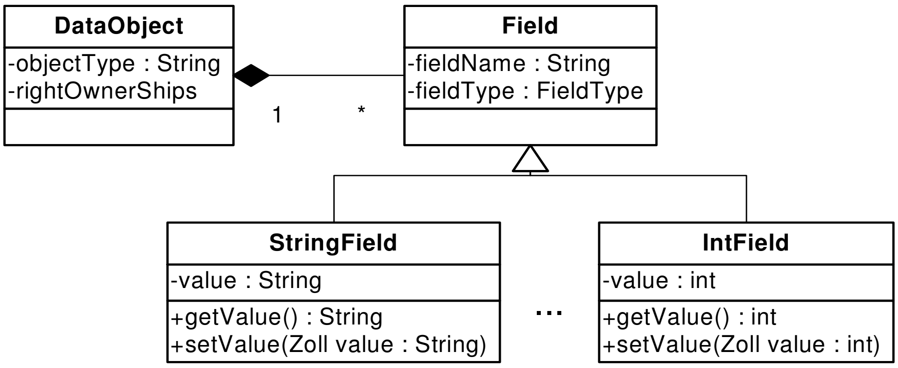

Figure 8: Generic data model for DataObject

A *DataObject* is identified by a globally unique identifier - a GGUID.

From the perspective of a web service consumer a generic *DataObject* has the disadvantage that the available fields of an entity is not described in the web service specification, and therefore in any generated software stub. We address this issue by providing a generic *DataObjectDescription*. In the same way that a *DataObject* provides type safe *Fields,* the *DataObjectDescription (see* Figure 9*)* provides type specific *FieldDescription*s of the entity's fields along with additional meta information such as field lengths, localized display names and constraints.

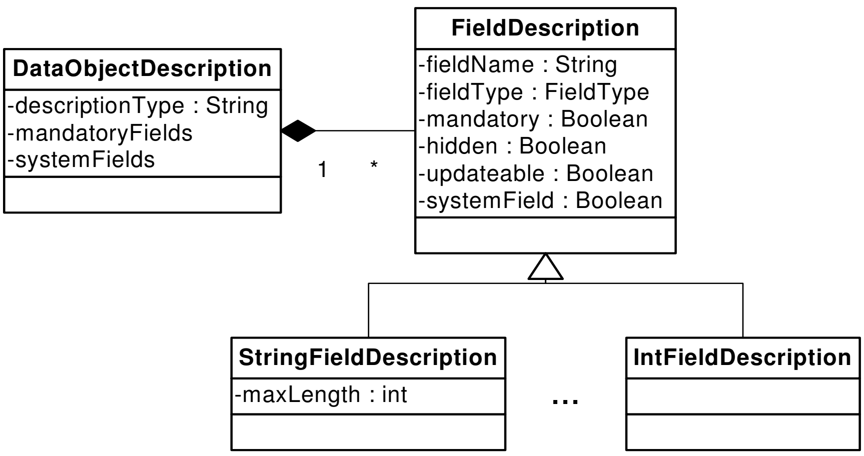

Figure 9: Generic data model for DataObjectDescription

7.1 Service Specification for Atomic Data Access and Manipulation

Two forms of access and manipulation are considered here:

(i) direct access to atomic data through its GGUID
(ii) access to atomic data through a database query

(i) Direct access to atomic data through its GGUID

Using the generic data model specified in Figure 8 andFigure 9, we can limit the required operations for atomic data manipulation to a minimal set of operations. Following the DAO-pattern (*DataAccessObject*), the so called CRUD-operations suffice. For this purpose, four operations are specifically addressed in the service specification template of Table 2, including: 1) CreateObject, 2) GetObject, 3) SaveAndRunObject, and 4) DeleteObject. provides the detailed service specification for atomic data access and manipulation.

Service Name	Access & Manipulation of Atomic Data			
Service Description	Enables read and write access to data stored in the database			
Data/Information Type	Generic *DataObject*s (e.g. Equipment, historic sensored data, stakeholders, products, product-based VBE network, VO networks)			
Operation	CreateObject	GetObject	SaveAndReturnObject	DeleteObject
Input	objecttype of generic object (e.g. Equipment)	objecttype and primary key (GGUID)	*DataObject*	objecttype and primary key (GGUID)
Output	A new unsaved, but initialized *DataObject*	An existing *DataObject*	The saved *DataObject*	
Pre-conditions	objecttype must be valid	objecttype and GGUID must be valid	The *DataObject* must be initialized according to *DataObjectDescription*	objecttype and GGUID must be valid
Exceptions		- Invalid GGUID	- *DataObject* is not initialized correctly - Update conflict (optimistic locking): a record has been updated by another process	- Invalid GGUID
Timeliness Quality of Service	Result should be returned in 10 seconds	Result should be returned in 10 seconds	Result should be returned in 10 seconds	Result should be returned in 10 seconds
Synchronous/ asynchronous access	Synchronous	Synchronous	Synchronous	Synchronous

Table 2: Service Specification for Atomic Data Manipulation through GGUID

Besides requiring only a small web service interface, the biggest advantage of a generic data model and therefore also of the generic web service specification, is the fact that these specifications become independent of any future changes to the any specific data model, such as its modification with additional fields.

Some Example CRUD-operations

Example 1 below outlines the creation of a new *DataObject.* In this example, in (1) the client asks for creation of a *specific order* DataObject of type *MAINTENANCE APPOINTMENT.* In (2) the server responds by sending the *DataObject,* which has a start_date, an end_date (and it cannot be wholeday), and a keyword. In (3) the client submits values for this *DataObject,* including the keyword *Lighting Sensor*, as well as the start and end date for the maintenance appointment. Finally in (4), the server updates the *DataObject* in the database and returns its new GGUID in BOIKA (i.e. the primary key in the virtual BOIKA database).

Example 1 – Creating a data object

This is the example of steps involved in creating a new order through a *maintenance appointment* data object.

(1) Client calls *createObject (of type MAINTENANCE APPOINTMENT)*

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header/>
  <S:Body>
    <ns23:createObject xmlns="...">
      <ns23:typeName>MAINTAPPOINTMENT</ns23:typeName>
    </ns23:createObject>
  </S:Body>
</S:Envelope>
```

(2) Server sends initialized *DataObject*

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns29:createObjectResponse xmlns:ns="...">
      <ns29:DataObject>
        <ns5:fields  xsi:type="ns5:BooleanField">
          <ns5:name>WHOLEDAY</ns5:name>
          <ns5:fieldType>BOOLEAN</ns5:fieldType>
          <ns5:value>false</ns5:value>
        </ns5:fields>
        <ns5:nullFields>KEYWORD</ns5:nullFields>
        <ns5:nullFields>END_DT</ns5:nullFields>
        <ns5:nullFields>START_DT</ns5:nullFields>
        <ns5:objectType>MAINTAPPOINTMENT</ns5:objectType>
      </ns29:DataObject>
    </ns29:createObjectResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

(3) Client calls *saveAndReturnObject* and sends changed *DataObject (to be saved)*

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header/>
  <S:Body>
    <ns23:saveAndReturnObject xmlns="...">
      <ns23:dataObject>
        <ns2:fields  xsi:type="ns2:StringField">
          <ns2:name>KEYWORD</ns2:name>
          <ns2:fieldType>STRING</ns2:fieldType>
```

```
          <ns2:value>Lighting Sensor</ns2:value>
        </ns2:fields>
        <ns2:fields  xsi:type="ns2:BooleanField">
          <ns2:name>WHOLEDAY</ns2:name>
          <ns2:fieldType>BOOLEAN</ns2:fieldType>
          <ns2:value>false</ns2:value>
        </ns2:fields>
        <ns2:fields  xsi:type="ns2:DateField">
          <ns2:name>END_DT</ns2:name>
          <ns2:fieldType>DATETIME</ns2:fieldType>
          <ns2:value>2012-05-18T14:46:59.098+02:00</ns2:value>
        </ns2:fields>
        <ns2:fields  xsi:type="ns2:DateField">
          <ns2:name>START_DT</ns2:name>
          <ns2:fieldType>DATETIME</ns2:fieldType>
          <ns2:value>2012-05-18T13:46:59.098+02:00</ns2:value>
        </ns2:fields>
        <ns2:objectType>MAINTAPPOINTMENT</ns2:objectType>
        <ns2:client>TEAMCRM1</ns2:client>
      </ns23:dataObject>
    </ns23:saveAndReturnObject>
  </S:Body>
</S:Envelope>
```


(4) Server returns saved *DataObject* with new primary key (GGUID)


```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns29:saveAndReturnObjectResponse xmlns="...">
      <ns29:return>
        <ns5:fields  xsi:type="ns5:StringField">
          <ns5:name>KEYWORD</ns5:name>
          <ns5:fieldType>STRING</ns5:fieldType>
          <ns5:value>Lighting Sensor</ns5:value>
        </ns5:fields>
        <ns5:fields  xsi:type="ns5:BooleanField">
          <ns5:name>WHOLEDAY</ns5:name>
          <ns5:fieldType>BOOLEAN</ns5:fieldType>
          <ns5:value>false</ns5:value>
        </ns5:fields>
        <ns5:fields  xsi:type="ns5:DateField">
          <ns5:name>END_DT</ns5:name>
          <ns5:fieldType>DATETIME</ns5:fieldType>
          <ns5:value>2012-05-18T12:46:59.000Z</ns5:value>
        </ns5:fields>
        <ns5:fields  xsi:type="ns5:DateField">
          <ns5:name>START_DT</ns5:name>
          <ns5:fieldType>DATETIME</ns5:fieldType>
          <ns5:value>2012-05-18T11:46:59.000Z</ns5:value>
        </ns5:fields>
        <ns5:fields  xsi:type="ns5:GGUIDField">
          <ns5:name>GGUID</ns5:name>
          <ns5:fieldType>GGUID</ns5:fieldType>
          <ns5:value>17FD3C73B19732439DEA3FF2341C58D2</ns5:value>
        </ns5:fields>
        <ns5:objectType>MAINTAPPOINTMENT</ns5:objectType>
      </ns29:return>
    </ns29:saveAndReturnObjectResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

(ii) Access to atomic data through a database query

When accessing/retrieving data stored in a database through a database query, the result is always returned in a relation. To support this, we first define in Figure 10 a generic data model for database relations (as they are defined in a relational database), for the purpose of returning the result of a database query, and second we define a web service (called MassQueryResult) to retrieve query results.

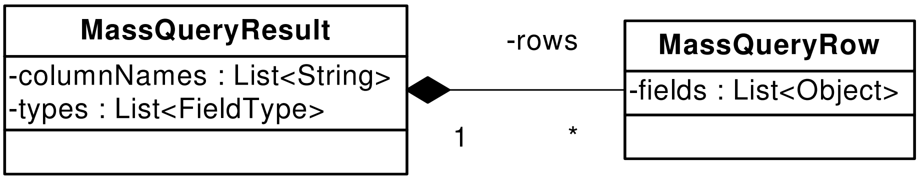

Figure 10: Generic data model for database result-relation

Therefore, the *MassQueryResult* works as a container for such a relation. It also transports some metainformation with it, such as the names and types of the returned columns, as well as the actual rows consisting the query results, which are represented as *MassQueryRows*.

Table 3 provides the service specification for access/retrieve of data through a database query.

Service Name	Query access to atomic data stored in relations
Service Description	Enables read access to data stored in the database, through a query
Data/Information Type	Generic *MassQueryResult*
Operation	Query
Input	SQL query
Output	*MassQueryResult* (i.e. the result of the given query)
Pre-conditions	SQL query must be valid
Exceptions	- Invalid columns / tables
Timeliness Quality of Service	Depending on the complexity of the query
Synchronous/ asynchronous access	Synchronous

Table 3: Service Specification for Atomic Data Access/retrieve through queries

7.2 Service Specification for Document(File) Access and Manipulations

Unlike accessing atomic data, directly or through the database query, documents (files) are typically handled through uploads and downloads. In BOIKA a document is represented by a specific kind of *DataObject called DOCUMENT, while* providing some of its related information (e.g. a title, permissions, and tags), as well as the actual document content itself, i.e. a PDF file. But with the web service operations defined so far in this Section, it is not yet feasible to implement the use cases necessary for up- and downloading documents and linking information throughout the BOIKA platform.

Therefore, following the concept of a *generic web service*, we specify an additional web service operation called ***execute*** that allows us to invoke the so called ***business operations*** *(see*
Table 4*)*. With this concept we follow the design pattern *Command* where the input data (which is a given sub class of *RequestObject)* specifies which operation shall be executed, and transports the required parameters. In turn, the server returns an appropriate sub class of *ResponseObject*.

Service Name	Execute
Service Description	Invoke a business operation
Data/Information Type	Generic *RequestObject* wrapping the processed parameters
Operation	Execute
Input	A sub class of *RequestObject*
Output	A sub class of *ResponseObject*
Pre-conditions	
Exceptions	- Invalid parameters - Operation specific exception
Timeliness Quality of Service	Depending on the complexity of the operation
Synchronous/ asynchronous access	Synchronous

Table 4: An Auxiliary Service Specification (Execute) for Document handling

Document (file) handling - using the generic Execute web service

For handling objects of type DOCUMENT, create, retrieve, update, delete operations are considered to be necessary. In order to support the two operations of: (i) *retrieve document*, and (ii) *create and/or update document*, we have defined two business operations called *CheckInFile and CheckOutFile,* with the corresponding *RequestObject and ResponseObject, as applied in Example 2 below*. For the Operation (iii) *delete document,* this is accomplished by deleting the corresponding *DataObject.*

Example 2 outlines the creation of a new *product-specification* document *DataObject.* In (1) the client asks for creation of a DataObject of type *DOCUMENT.* In (2) the server responds by sending the *DataObject,* with all its fields, e.g. the keyword, date, size, etc. In (3) the client executes the *CheckInFile* to upload this document *DataObject,* by submitting its keyword *product specification*, as well as where this document should be uploaded from. Finally in (4), the server uploads this document *DataObject* in the database and returns values for all of its fields, including its new GGUID in BOIKA (key for accessing the content of this document). Complementary to this, the Example 3 (*appearing in this section following the Example 2*) outlines the access/download of the *product specification* document, which is created/uploaded in Example 2.

Example 2 – Creating a document and uploading the file's content

This is the example of steps involved in creating and uploading a *product specification* document.

(1) Client calls *createObject (of type DOCUMENT)*


```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
    <S:Header/>
    <S:Body>
      <ns23:createObject xmlns="...">
        <ns23:typeName>DOCUMENT</ns23:typeName>
      </ns23:createObject>
    </S:Body>
</S:Envelope>
```


(2) Server returns initialized *DataObject*


```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns29:createObjectResponse xmlns:ns="...">
      <ns29:DataObject>
        <ns5:nullFields>KEYWORD</ns5:nullFields>
        <ns5:nullFields>ARCHIVEFILE</ns5:nullFields>
        <ns5:nullFields>FILETYPE</ns5:nullFields>
        <ns5:nullFields>GGUID</ns5:nullFields>
        <ns5:nullFields>DOCDATE</ns5:nullFields>
        <ns5:nullFields>DOCUMENTSIZE</ns5:nullFields>
        <ns5:nullFields>DOCVERSION</ns5:nullFields>
        <ns5:objectType>DOCUMENT</ns5:objectType>
      </ns29:DataObject>
    </ns29:createObjectResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```


(3) Client executes business operation *CheckInFile* to upload the file and save the document


```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
      <S:Header/>
      <S:Body>
        <ns23:execute xmlns="...">
          <ns23:requestObject  xsi:type="ns10:CheckInFileRequest">
            <ns10:dataObjectToSave>
              <ns2:fields xsi:type="ns2:StringField">
                <ns2:name>KEYWORD</ns2:name>
                <ns2:fieldType>STRING</ns2:fieldType>
```

```
          <ns2:value>Product specification</ns2:value>
        </ns2:fields>
        <ns2:nullFields>ARCHIVEFILE</ns2:nullFields>
        <ns2:nullFields>FILETYPE</ns2:nullFields>
        <ns2:nullFields>GGUID</ns2:nullFields>
        <ns2:nullFields>DOCDATE</ns2:nullFields>
        <ns2:nullFields>UPDATETIMESTAMP</ns2:nullFields>
        <ns2:nullFields>DOCUMENTSIZE</ns2:nullFields>
        <ns2:nullFields>DOCVERSION</ns2:nullFields>
        <ns2:objectType>DOCUMENT</ns2:objectType>
      </ns10:dataObjectToSave>
      <ns10:documentGGUID>1</ns10:documentGGUID>
      <ns10:fileType>txt</ns10:fileType>
      <ns10:keepCheckedOut>false</ns10:keepCheckedOut>
      <ns10:documentContent>6162636465666...</ns10:documentContent>
    </ns23:requestObject>
  </ns23:execute>
 </S:Body>
</S:Envelope>
```


(4) Server returns *CheckInFileResponse* that wraps the saved meta record as a document *DataObject and provides its new GGUID*


```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
 <SOAP-ENV:Header/>
 <SOAP-ENV:Body>
  <ns29:executeResponse xmlns:ns="...">
   <ns29:responseObject  xsi:type="ns20:CheckInFileResponse">
    <ns20:savedDataObject>
     <ns5:fields xsi:type="ns5:StringField">
      <ns5:name>FILETYPE</ns5:name>
      <ns5:fieldType>STRING</ns5:fieldType>
      <ns5:value>txt</ns5:value>
     </ns5:fields>
     <ns5:fields xsi:type="ns5:StringField">
      <ns5:name>KEYWORD</ns5:name>
      <ns5:fieldType>STRING</ns5:fieldType>
      <ns5:value>Product specification</ns5:value>
     </ns5:fields>
     <ns5:fields xsi:type="ns5:StringField">
      <ns5:name>ARCHIVEFILE</ns5:name>
      <ns5:fieldType>STRING</ns5:fieldType>
      <ns5:value>/2012/5/18/6B484986252F376E858BC3ED08838627_0.txt</ns5:value>
     </ns5:fields>
     <ns5:fields xsi:type="ns5:GGUIDField">
      <ns5:name>GGUID</ns5:name>
      <ns5:fieldType>GGUID</ns5:fieldType>
      <ns5:value>6B484986252F376E858BC3ED08838627</ns5:value>
     </ns5:fields>
     <ns5:fields xsi:type="ns5:IntField">
      <ns5:name>DOCUMENTSIZE</ns5:name>
      <ns5:fieldType>INT</ns5:fieldType>
      <ns5:value>59</ns5:value>
     </ns5:fields>
     <ns5:fields xsi:type="ns5:IntField">
      <ns5:name>DOCVERSION</ns5:name>
      <ns5:fieldType>INT</ns5:fieldType>
      <ns5:value>0</ns5:value>
     </ns5:fields>
     <ns5:nullFields>DOCDATE</ns5:nullFields>
     <ns5:objectType>DOCUMENT</ns5:objectType>
```

```
        </ns20:savedDataObject>
      </ns29:responseObject>
    </ns29:executeResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Example 3 – Retrieving a document through downloading the file content

This is the example of steps involved in downloading a *product specification* document, by giving its GGUID. In (1) the client executes the CheckOutFile to download the document object that is identified by its GGUID. In (2) the server wraps and returns the downloaded document.

(1) Client executes business operation *CheckOutFile to download the file*

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
        <S:Header/>
        <S:Body>
          <ns23:execute xmlns="...">
            <ns23:requestObject  xsi:type="ns10:CheckOutFileRequest">
              <ns10:GGUID>6B484986252F376E858BC3ED08838627</ns10:GGUID>
              <ns10:readOnly>true</ns10:readOnly>
            </ns23:requestObject>
          </ns23:execute>
        </S:Body>
</S:Envelope>
```

(2) Server returns *CheckOutFileResponse* that wraps the meta record and the file content

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns29:executeResponse xmlns:ns="...">
      <ns29:responseObject  xsi:type="ns20:CheckOutFileResponse">
        <ns20:GGUID>6B484986252F376E858BC3ED08838627</ns20:GGUID>
        <ns20:fileType>txt</ns20:fileType>
        <ns20:documentContent>6162636465666...</ns20:documentContent>
        <ns20:savedDataObject>
          <ns5:fields xsi:type="ns5:StringField">
            <ns5:name>FILETYPE</ns5:name>
            <ns5:fieldType>STRING</ns5:fieldType>
            <ns5:value>txt</ns5:value>
          </ns5:fields>
          <ns5:fields xsi:type="ns5:StringField">
            <ns5:name>KEYWORD</ns5:name>
            <ns5:fieldType>STRING</ns5:fieldType>
            <ns5:value>Product specification</ns5:value>
          </ns5:fields>
          <ns5:fields xsi:type="ns5:StringField">
            <ns5:name>ARCHIVEFILE</ns5:name>
            <ns5:fieldType>STRING</ns5:fieldType>
            <ns5:value>/2012/5/18/6B484986252F376E858BC3ED08838627_0.txt</ns5:value>
          </ns5:fields>
          <ns5:fields xsi:type="ns5:GGUIDField">
            <ns5:name>GGUID</ns5:name>
            <ns5:fieldType>GGUID</ns5:fieldType>
            <ns5:value>6B484986252F376E858BC3ED08838627</ns5:value>
          </ns5:fields>
          <ns5:fields xsi:type="ns5:IntField">
```

```
        <ns5:name>DOCUMENTSIZE</ns5:name>
        <ns5:fieldType>INT</ns5:fieldType>
        <ns5:value>59</ns5:value>
      </ns5:fields>
      <ns5:fields xsi:type="ns5:IntField">
        <ns5:name>DOCVERSION</ns5:name>
        <ns5:fieldType>INT</ns5:fieldType>
        <ns5:value>0</ns5:value>
      <ns5:nullFields>DOCDATE</ns5:nullFields>
      <ns5:objectType>DOCUMENT</ns5:objectType>
    </ns20:savedDataObject>
  </ns29:responseObject>
 </ns29:executeResponse>
</SOAP-ENV:Body></SOAP-ENV:Envelope>
```


7.3 Service Specification for Process (knowledge) Access and Manipulation

Service specification for process access and manipulation is required to be supported in relation to several common entities in a complex product, such as the business services, as well as for representing the knowledge generated around the complex product in the VO or VBE, e.g. best practices or lessons learned, which can be represented in form of processes.
Different stakeholders may define/model their generated processes differently. For example some stakeholders may represent processes as diagrams (e.g. BPMN diagrams), while others may for instance formalize them as a specific XML document that contains a BPMN-definition of the process in it. These presentations can all be stored in BOIKA as document files, for which their needed access/manipulation services were addressed in Section 7.2. But in addition to the document file, a process also contains other descriptive elements, e.g. name, short-description, provider, constraints, etc. which are best stored in a database. These elements can be stored in BOIKA as atomic pieces of data, for which their needed access/manipulation services were addressed in Section 7.1. As such what remains as needed for proper representation of processes is a service that links the *document* part of a process to its *atomic data* part.

Linking different information throughout the platform

As a generic approach for this purpose, BOIKA provides a way to link any entity with another. Using this feature, a complete process e.g. a best practice definition can be presented in BOIKA through linking its *document part* with its *atomic data part.* Furthermore, with this 2-part definition, the processes can be fully accessed/manipulated by the generic services defined earlier in this Section.

To link information in the context of BOIKA, with the following generic service, we can establish relations between any two kinds of *DataObjects*, through the so called *LinkObjects* or simply *links,* as they are called in the following example. This is outlined in Example 4 below.

Example 4 - Creating a link between two different data objects

The following example outlines the creation of a link between two *DataObject*s of types *BEST PRACTICE* and *DOCUMENT.* In (1) the client asks for creation of a *link* between a data object and a document object. In (2) the server responds by creating the *link* and returning its new GGUID.

(1) Client sends SaveLinkRequest


```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header/>
  <S:Body>
    <ns23:execute xmlns="...">
      <ns23:requestObject  xsi:type="ns3:SaveLinkRequest">
        <ns3:links>
           <ns2:objectType1>BESTPRACTICE</ns2:objectType1>
           <ns2:objectType2>DOCUMENT</ns2:objectType2>
          <ns2:GGUID1>331993431F9C3D4CB2ED0C692F60F0F1</ns2:GGUID1>
          <ns2:GGUID2>A46A33F150A230C0AFDB86B60DD0D0AC</ns2:GGUID2>
        </ns3:links>
      </ns23:requestObject>
    </ns23:execute>
  </S:Body>
</S:Envelope>
```


(2) Server returns *SaveLinkResponse* that contains the saved link and provides its new GGUID


```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ns29:executeResponse xmlns:ns="…">
      <ns29:responseObject xsi:type="ns6:SaveLinkResponse">
        <ns6:savedAndFailedLinks>
          <ns6:savedLinks>
            <ns5:GGUID>BD60675B9EF33C03AC49C2401C96D662</ns5:GGUID>
            <ns5:objectType1>BESTPRACTICE</ns5:objectType1>
            <ns5:objectType2>DOCUMENT</ns5:objectType2>
            <ns5:GGUID1>331993431F9C3D4CB2ED0C692F60F0F1</ns5:GGUID1>
            <ns5:GGUID2>A46A33F150A230C0AFDB86B60DD0D0AC</ns5:GGUID2>
            <ns5:linkDirection>BIDIRECTIONAL</ns5:linkDirection>
            <ns5:isHierarchy>false</ns5:isHierarchy>
          </ns6:savedLinks>
        </ns6:savedAndFailedLinks>
      </ns29:responseObject>
    </ns29:executeResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

8. CONCLUSION

This deliverable is one of the initial outcomes of WP2, and it is centred around providing details on generic specification of services, to access BOIKA in GloNet.

To approach this task, we aim at generalization and not focusing on any specific complex product. As a result we have focused on identification of commonalities, and classification of elements into their nature-based categories. In other to identify the needed access mechanisms for complex products in GloNet, we have developed a systematic 7-step process. This process is described in details in Section 2 of this document and followed step by step, as reported in different sections of this document.

The systematic process starts with targeted analysis of the three planned complex products of GloNet, namely: (i) the PV solar plants, (ii) the intelligent buildings, and (iii) the future incubators. In Section 3, as a first step, the main entities common among these three products are discovered and extracted. These represent the elements about which their information and/or knowledge needs to be stored and shared among different users. In Section 4, as a second step, the main common group of users (including both stakeholders and software systems) are addressed.

In Section 5, in the third step a cross section between entities from the first step and users from the second step is established, aiming to verify their relevance as well as comprehensiveness. In Section 6.1, in the fourth step, the results generated by the first step are analyzed and classified by their nature (or form) into the main categories of information/knowledge in the environment. Our main resulted categories include: atomic data, documents, and processes. In Section 6.2, the fifth step primarily analyzes and classifies the results, which are generated in the second step, by their nature (or kind), into the main categories of users in the environment. Our main resulted categories include: Human user and software systems. In Section 6.3, in the sixth step it is discussed that in GloNet, access to information/knowledge from the BOIKA by software systems require them to be provided as web services. Furthermore, it is discussed that access to information / knowledge by human users in the GloNet requires the provision of user-friendly GUIs, where in turn similar to above, the software system of the GUI needs to access to information / knowledge of the BOIKA, and thus also requires access through web services.

Therefore, in Section 7, in the seventh and final step of the procedure, high level generic specifications of the 3 required web services for accessing: atomic data, documents, and processes related to GloNet products through the cloud are provided. For Specifying the needed software services, the following techniques are applied: Data model specifications (UML based), Tabular description addressing their specific features (IOPE based), and A set of example CRUD (Create, Retrieve, Update, Delete) operations (Soap based). The specified web-services are generic and reusable, as such they are applicable to accessing the information/knowledge about different entities in any complex product environment, where the information/knowledge can have different nature and be located in different types of sources.

The proposed information/knowledge access/manipulation approach and the provided service specification are important for service implementation in the project. In the course of this deliverable preparation, the D1.1 deliverable is used as the base input for identification of the main information/knowledge entities in complex products' environment, to be stored in virtual BOIKA repository, for the purpose of being accessed and shared by different users (stakeholders and software systems). The results from this deliverable will act as the base input, for producing the next three deliverables of WP2 in the GloNet project.

9. REFERENCES

1. Afsarmanesh, H., Camarinha-Matos L. and Msanjila, S. (2009). "On management of 2nd generation Virtual Organizations Breeding Environments." Annual Reviews in Control 33(2): 209-219.

2. Afsarmanesh, H. and Thamburaj, V. (2012). “ICT requirements for enterprise networks supporting solar power plants”. 13th IFIP WG 5.5 Working Conference on Virtual Enterprises, PRO-VE 2012, BourneMouth, UK, October 01-03, 2012.

3. Afsarmanesh, H. and Thamburaj, V. (2012). “Detailed Requirements for GloNet use case and Domain Glossary”. EU Project (285273) - Deliverable 1.1, Global enterprise network focusing on customer-centric collaboration.

4. Aguilar-Savén, R. S. (2004). "Business process modelling: Review and framework." International Journal of Production Economics 90 (2): 129-149.

5. Camarinha-Matos, L., Afsarmanesh, H. and Bernhard, K. (2011). Collaborative Networks in Support of Service-Enhanced Products. Adaptation and Value Creating Collaborative Networks. L. Camarinha-Matos, A. Pereira-Klen and H. Afsarmanesh, Springer Berlin Heidelberg. 362: 95-104.

6. Camarinha-Matos, L. M. and H. Afsarmanesh (2007). "A framework for virtual organization creation in a breeding environment." Annual Reviews in Control 31(1): 119-135.

7. D'Cruz, C. and T. O'Neal (2003). Integration of technology incubator programs with academic entrepreneurship curriculum. Management of Engineering and Technology, 2003. PICMET '03. Portland International Conference on Technology Management for Reshaping the World.

8. Ermilova, E. and Afsarmanesh, H. (2007). "Modeling and management of profiles and competencies in VBEs." Journal of Intelligent Manufacturing 18(5): 561-586.

9. Ermilova, E. and Afsarmanesh, H. (2010). "Competency modelling targeted on boosting configuration of virtual organisations." Production Planning & Control 21(2): 103-118.

10. Feamster, N. and Balazinska, M. (2002). Infranet: Circumventing Web Censorship and Surveillance. In Proceedings of USENIX Security Symposium. 2002, 247-262.

11. Gomes, L., Costa, A., Barros, J., Pais, R., Rodrigues, T. and Ferreira, R. (2007). Petri Net based Building Automation and Monitoring System. 2007 5th IEEE International Conference on Industrial Informatics.

12. Granzer, W., Kastner, W., Neugschwandtner, G. and Praus, F. (2006). A modular architecture for building automation systems. IEEE International Workshop on Factory

Communication Systems, 2006.

13. Hongwei, W., Dechang, L., Hong, Y., Qiang, L. and Haiqing, C. (2008). Linking incubator services to the performance of incubated firms: A review. 4th IEEE International Conference on Management of Innovation and Technology, 2008. ICMIT 2008.

14. Martin, D., Burstein, M., Hobbs, J., Lassila, O., McDermott, D., McIllraith, S., Narayanan, S., Paolucci, M., Parsia, B., Payne, T., Sirin, E., Srinivasan, N. and Sycara, K. (2004). OWL-S: semantic markup for web services. W3C Member Submission 22 November 2004. Available at: http://www.w3.org/Submission/OWL-S/

15. Min, C., Ebert, D., Ebert, D., Hagen, H., Laramee, R., van Liere, R., Ma, K., Ribarsky, W., Scheuermann, G. and Silver, D. (2009). "Data, Information, and Knowledge in Visualization." Computer Graphics and Applications, IEEE 29(1): 12-19.

16. Mitchell, R., Keyes, B., Ullal, H., von Roedern, B., VanSant, K., Symko-Davies, M. and Stephens, S. (2009). The PV Technology Incubator project. Photovoltaic Specialists Conference (PVSC), 2009 34th IEEE.

17. Stenmark, D. (2002). Information vs. knowledge: the role of intranets in knowledge management. Proceedings of the 35th Annual Hawaii International Conference on System Sciences, 2002. HICSS.

18. Sturdy, A. (1997). "The Consultancy Process — An Insecure Business?" Journal of Management Studies 34(3): 389-413.

19. Ziyan, J. (2005). An information platform for building automation system. IEEE International Conference on Industrial Technology, 2005. ICIT 2005.

CONSORTIUM

CAS Software AG, Germany
Project coordinator: Dr. Bernhard Koelmel

UNINOVA – Instituto de Desenvolvimento de Novas Tecnologias, Portugal
Technical coordinator: Prof. Luis M. Camarinha-Matos

Universiteit van Amsterdam, Netherlands

iPLON GmbH The Infranet Company, Germany

Steinbeis GmbH & Co., Germany

SKILL Estrategia S.L., Spain

Komix s.r.o., Czech Republic

Prolon Control Systems, Denmark

Member of the:

FInES Future Internet Enterprise Systems

www.glonet-fines.eu