The SENSEI Real World Internet Architecture

Version 1.0, March 2010

Section 1 - Introduction

The Future Internet is expected to enable a networked digital society by revolutionising key application domains and industry sectors such as energy, transport, mobility and logistics, health care, media content, utilities and environment. The underlying technological transformations in these sectors provide promise to significant productivity gains by increasing the efficiency and effectiveness of delivered services and corresponding service experience.

Much of this transformation will be enabled by making the services and application offered in these domains more aware of the real world information and interaction capabilities. The key to achieving the real world awareness is to make the right information available at the right time to the right decision making processes and take appropriate actions to arising demands within the existing complex real world eco-systems.

Currently, much of the real world information on the Internet is created or entered by humans, and often only available through human readable interfaces. While the available information may certainly serve as a sufficient basis for a variety of different applications, it does not by far capture the real world in its complexity; timeliness of information also remains an issue. True real world awareness can only be achieved if information concerning the physical world is captured in an automatic fashion, ideally in real time with respect to arising demands.

Recent research efforts within the SENSEI project have tackled the challenges of enabling such a Real World (aware) Internet at global scale. SENSEI has developed an architecture and corresponding technological building blocks serving as key enablers for the envisioned future smart environments.

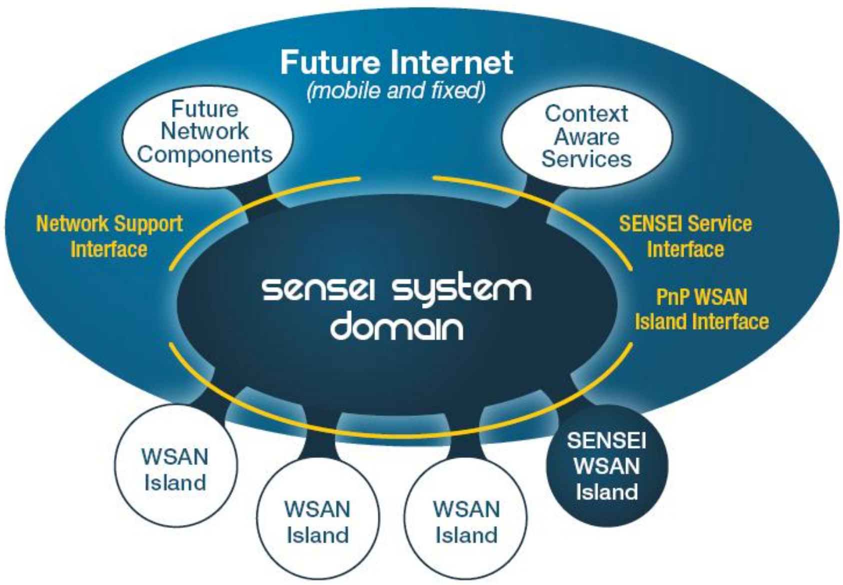

Figure 1-1: SENSEI Vision

As shown in the Figure 1-1, Wireless Sensor and Actuator Networks (WS&AN) provide the essential foundations for such a Real World Internet. Ubiquitously deployed at the edges of the networks, they will provide an infrastructure that enables augmentation of physical world and interaction with it, without the need for direct human intervention. Building on top of the communication service layers of the current or future Internet, our architecture weaves heterogeneous WS&AN and processing resources into a homogeneous real world resource fabric, enabling an open market space for real world-awareness and interactions..

According to SENSEI the world that we live in is divided in a real and a digital world. The real world consists of the physical environment that is instrumented with sensors, actuators and processing elements organized in Wireless Sensor and Actuator Networks (WS&AN) islands in order to monitor and interact with the physical entities that we are interested in: people, places and objects. The digital world consists of a) *Resources* which are representations of the instruments (sensors, actuators, processing elements), b) *Entities of Interest (EoI)* which are representations of people, places and things and c) Resource Users which represent the physical people or application software that intends to interact with Resources and EoI (see Figure 1-2). Bridging the physical and the digital world by allowing users/applications to interact with the Resources and EoI is the main contribution of SENSEI towards a Real World Internet.

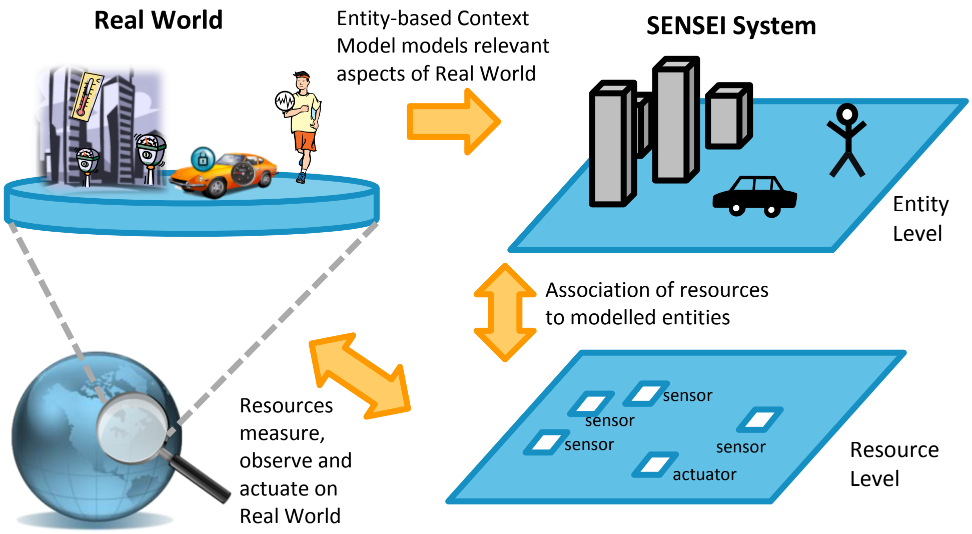

Figure 1-2: SENSEI Abstraction Levels

The SENSEI system provides two abstraction levels for interaction; the *resource level* and the *entity level* (see Figure 1-2). On the *resource level*, resource users directly interact with resources. Sensor-based resources typically provide information in the form of observations or measurements, which include the value, but also some meta-information regarding what type of information is provided, e.g., location or temperature.

As in the early days of the Internet the explosion of Internet sites created the need for rendezvous services (directories, search engines, etc), the expected increase in the number of real world resources and resource users will definitely lead to the need for similar rendezvous functionality. SENSEI offers a rendezvous service to allow a user to locate suitable resources based on desirable properties while allowing resources to advertise these properties.

Direct interaction between a resource user and a resource is suitable for certain types of applications where the provided information or interaction (actuation control) does not need any context (e.g. the name of the person that this information is about). Maintaining the context of real world information or interaction could be left entirely to the applications. However SENSEI offers the option to applications to use the SENSEI context model which is centered on the *entities of interest*. For these entities of interest, relevant aspects like the activity of a person or the current location of a car are modelled as context attributes. Applications can base their requests on entities of interest and context attributes. The underlying requirement is that the resources providing information are associated with the respective entities and attributes, so that the SENSEI system can find the required resources for the entity-level requests. SENSEI offers architectural components that enable contextualized information retrieval and interaction as well as dynamic service composition.

Finally, SENSEI takes into consideration the business and socioeconomic aspects of these interactions by introducing the notion of the SENSEI Community. A SENSEI Community consists of various actors taking up one or more business roles within the context of the Real World Internet. The main roles are the resource providers who are the owners of resources, framework providers who are the owners of SENSEI framework components, service providers who are the owners of services that use the SENSEI resources or support services and Resource Users who are the main user of the SENSEI resources or support services. In order for all these SENSEI community members to interact with each other proper authentication, authorization and access control mechanisms must be in place. SENSEI offers community management services that allow open interactions as well as security enforced interactions, offering the flexibility to realise a variety of business models with the developed architecture.

Section 2 - Architecture

2.1 The SENSEI System model

At its core the SENSEI system model defines a set of entities and their corresponding relationships. These entities form the basic abstractions on which the various SENSEI system functions operate.

The modelling of the system entities and their relationships reflects a clear separation of concerns at the various system levels and resembles their possible real world interrelationships. The system model differentiates between the (physical) instances of system resources (Resource, Figure 2-1) and the software components implementing the interaction endpoints from user perspective (REP, Figure 2-1). Furthermore it differentiates between the devices hosting these resources (Resource host, Figure 2-1) and the network devices hosting the respective interaction end points (REP host, Figure 2-1).

This separation allows the various system functions in the SENSEI system (explained later in the paper) to deal with real world dynamics in an efficient and adequate manner and allows for different deployment models of the system. Figure 2-1 depicts these entities and their inter-relationships.

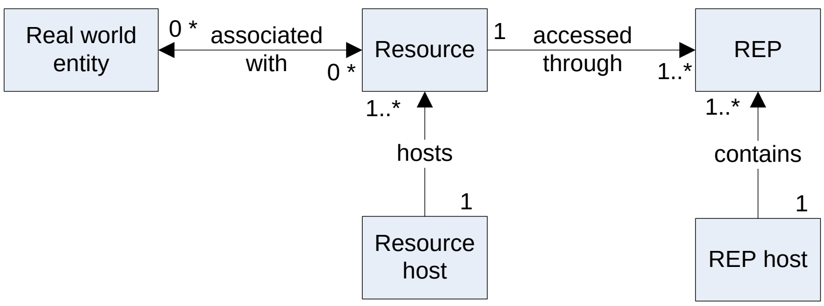

Figure 2-1: Key entities and their relationships in the SENSEI system model

A central entity in the SENSEI system model is the *SENSEI resource*. Conceptually resources in the SENSEI system provide unifying abstractions for real world information and interaction capabilities comparable to the web resources in the current Web architecture: in the same way as a web user interacts with a web resource (e.g. retrieve a web page), she can interact with real world resources (retrieve sensor data from a sensor resource). However while the concept of web resource refers to nearly everything referable via a URI on the Web, a resource in the SENSEI system model is an unifying abstraction for a specific set of physical and virtual resources. SENSEI resources abstract capabilities offered by resources such as sensing, actuation, processing of context and sensor data or actuation loops, and management information concerning sensor/actuator nodes, gateway devices or entire collections of those. Thus a SENSEI resource has an embodiment in the physical world, which could be sensor, actuator, a processing component or a combination of these. In the latter case it is referred to as composite resource. A resource is unique within a SENSEI system (domain) and is described by an associated resource description, whose format is unified for all SENSEI resources.

A Resource End Point (REP) is a software process, which represents an interaction end point for a physical resource. It implements one or more Resource Access Interfaces (RAIs) to the resource. The same resource may be accessible via multiple REPs, either providing the same RAIs or different ones. When realised on top of the current Web architecture, REPs can be considered equivalent to web resources, which are uniquely identified by a URI.

The device hosting a resource is referred to as the *resource host*. Sensor nodes are typical examples for resource hosts, but there can be other devices acting in this role such as mobile phones or access points that embed resources. A *REP host* is a device that executes the software process representing the REP. As mentioned above the resources and REPs are conceptually separated from their hosts to allow different deployment options. In some cases a REP host and a resource host can be co-located on the same physical device (e.g. mobile phone). There may be many cases where the REP is not hosted on the resource host itself; e.g. a computer in the network or an embedded server may act as a REP host for a resource, which is physically hosted on a sensor node connected behind it. This differentiation is important when mobility, disconnections and other system dynamics come into play, in order to effectively keep the system state consistent for correct operation of the overall system. Moreover separation of concerns provides the option of protecting low capability resources (e.g. low power sensor nodes) from external attacks by hosting their REPs on more capable hardware than the resource hosts.

Unlike other systems, SENSEI considers also real world entities in its system model and manages the dynamic associations between real world entities and the sensors/actuators that can provide information about them/act upon them. Examples for real world entities (also known as Entities or Entities of Interest (EoI)) are persons, places, or objects of the real world that are considered relevant to provide a service to users. A SENSEI resource thus provides (context) information or interaction capabilities concerning associated real world entities.

2.2 SENSEI Support Services

As mentioned before, interaction between the real world information and interaction providers (resources) and consumers or service providers (resource users or simply users), can be of different types (direct, via rendezvous and context and EoI oriented). The SENSEI framework [SENSEI-D3.2] provides a set of support services for direct, rendezvous and context/entity based interaction, which are called SENSEI Support Services (Figure 2-2).

The first need that a user of the SENSEI framework may have is to find what information or interaction capabilities the framework can provide. To support this requirement SENSEI includes a basic *rendezvous* function which is split in two components:

- The *Resource Directory (RD)* that stores the descriptions of the resources, and can provide a list of resources that meet the users' criteria.
- The *Entity Directory (ED)* that maintains the associations between the properties of the EoI and the resources supplying information or interaction capabilities related to these EoIs.

The Entity Directory complements the Resource Directory. The focus of the latter is on the Resources while the Entity Directory focuses on the associations between the Entities and the registered Resources in the RD.

With these two components the user is able to find resources that provide some information and interact directly with them. However, these components just support basic search criteria (e.g. one type of criteria is simple keywords), they are not able to reason, combine or compose existing resources to create new ones based on the user needs. For these more advanced interactions SENSEI has designed the *Semantic Query Resolver (SQR)*. The SQR includes an advanced rendezvous function as well as advanced resource tasking and service composition. The SQR is able to receive queries that contain semantic information through a declarative language, analyze the queries and find the appropriate set of resources to satisfy the query using the basic rendezvous components (Resource and Entity Directory). If the query includes any form of aggregation of information or interaction resources, the SQR will construct a plan involving a sequence of resources. If required resources do not exist, a resource provider will attempt to create and deploy them. An example query formulated in English is "Provide the average temperature of Stockholm". The SQR will receive this query in a SENSEI formatted request and deduce that the entity of interest is a place (Stockholm) and the desirable property of the EoI is temperature. The SQR will consult the entity directory to find out the temperature resources associated with the entity Stockholm and according to a plan of action for such types of queries (average temperature), it will compose an aggregation (averaging) tree to produce the result.

The interactions of the users and the resources can be of two types, a one-shot interaction (where the user asks for information and gets it back) and a long term interactions (where the resource can provide additional responses over the time). The long term interactions can be handled by the user itself or can be facilitated by a component called the *Execution Manager (EM)*. The EM can support sessions, events (i.e. notify the user when the temperature reaches 25ºC) and more complicated functionality like the resource adaption (i.e. dynamically change the resource that provides the information back to the user due to a failure without any user interaction).

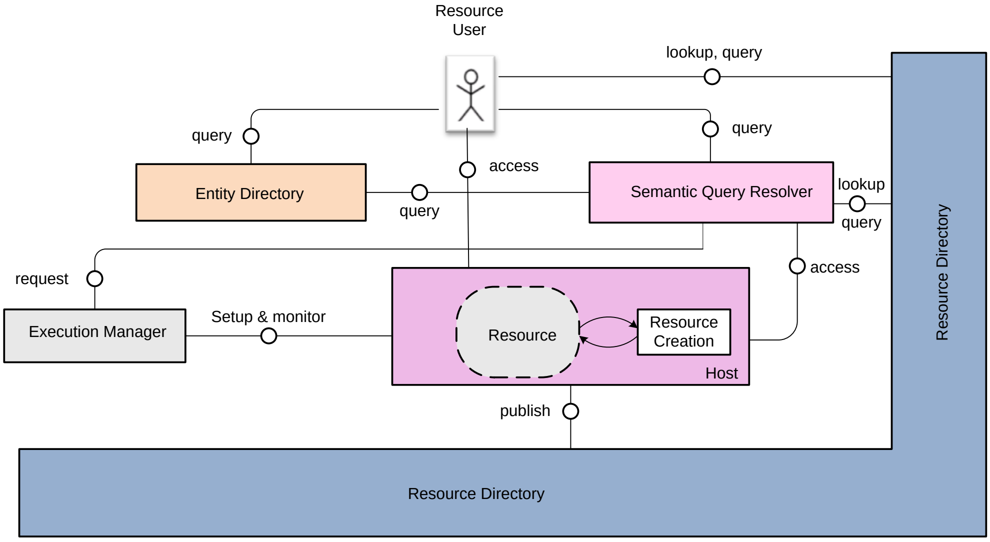

Figure 2-2: SENSEI Support Services

2.3 Operations and Interactions

In this section we present examples of the most common interactions between the resource users and the SENSEI framework. These interactions depend mainly on the following factors:

- The *goal* of *interaction*: The resource user may want to retrieve the resource(s) that provide some information or the direct physical world information. In the latter case, the system retrieves the information from the resource on behalf of the resource user.
- The *level of interaction*: The resource user can ask for a set of resources that meet some criteria or for entities of interest and their context attributes.
- The *type of the interaction*: The information that the resource user asks can be one-shot or long term interactions.

Whether resources are considered for a certain interaction depends on the way in which their capabilities have been published to the Resource Directory (Figure 2-3 a), Step 0): Some interactions require knowledge of the resource semantic description (which is optional information); then just the ones that provide it are considered.

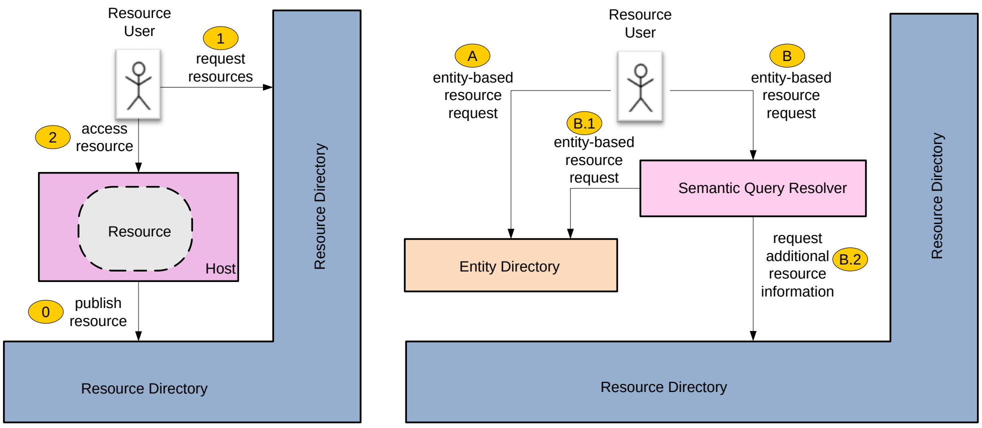

Figure 2-3: a) Simple Resource Request and Access b) Entity-based Resource Request

When the Resource User wants to retrieve the set of resources that met some criteria, which is contained in the tags of the resource descriptions (i.e. the type of information, etc.) to the user requests it directly from the Resource Directory (Figure 2-3 a), Step 1). In this interaction, the resource user just receives the resources (never the information); however, all the resources in the Resource Directory (regardless whether their description contains semantics or not) are considered. The Resource User can then directly access the Resource (Figure 2-3 a), Step 2).

If the Resource User wants to request Resources that provide some information about an Entity of Interest, it has two possibilities: Ask the Entity Directory (Figure 2-3 b), Variant A) or the Semantic Query Resolver (SQR) (Figure 2-3 b), Variant B). The main difference between these interactions is that the Entity Directory is just able to return the information that is stored in the directory (resource IDs of the relevant resources), while the Semantic Query Resolver can perform additional actions described below.

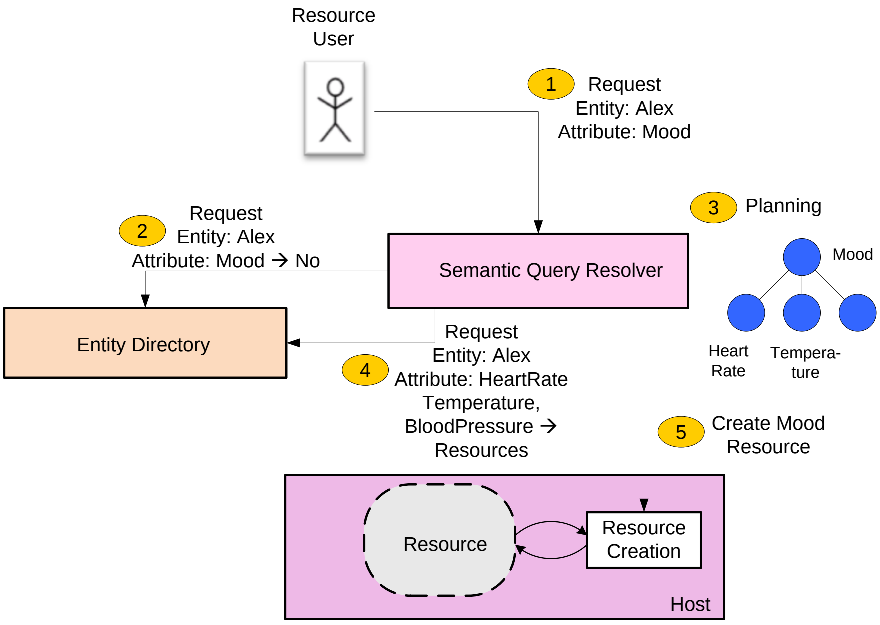

Figure 2-4: Planning and Resource Creation

For illustrating a complex interaction with the SQR, let's assume that the Resource User asks for the mood of Alex (Figure 2-4, Step 1). The SQR, since the query refers to an Entity of Interest, forwards the query to the Entity Directory (Step 2). In this example we assume that there is no resource that provides that information, so the ED does not return a resource. We can also assume that the SQR could create a plan where this information can be obtained by combining three different resources (for example heart rate, temperature and blood pressure, Step 3). If the Resources for providing the required inputs are available (Step 4), the SQR can invoke a resource provider to create, instantiate and deploy the required resources to provide the mood of Alex (Step 5). It is important to note in this example two important details: a) the semantically annotated advanced resource descriptions facilitate the resource composition of the three different resources by the SQR, b) the SQR uses additional information such as abstract task plans to create the required plan (the details are omitted in this paper) c) the real physical sensors are already in place (e.g. heart rate) and instantiation of a resource means instantiation of either a virtual/software resource (e.g processing element) or a REP.

Assuming that a required Resource exists the next important issue to consider is the goal of interaction. In case, the resource user only wants to know the Resource that can provide the interaction, the SQR only needs to return it. In case, the Resource User has requested information, the Execution Manager is provided with the execution plan (Figure 2-5, Step 3) to execute the request on behalf of the Resource User. Here, the type of interaction (one-shot, long term) has to be considered.

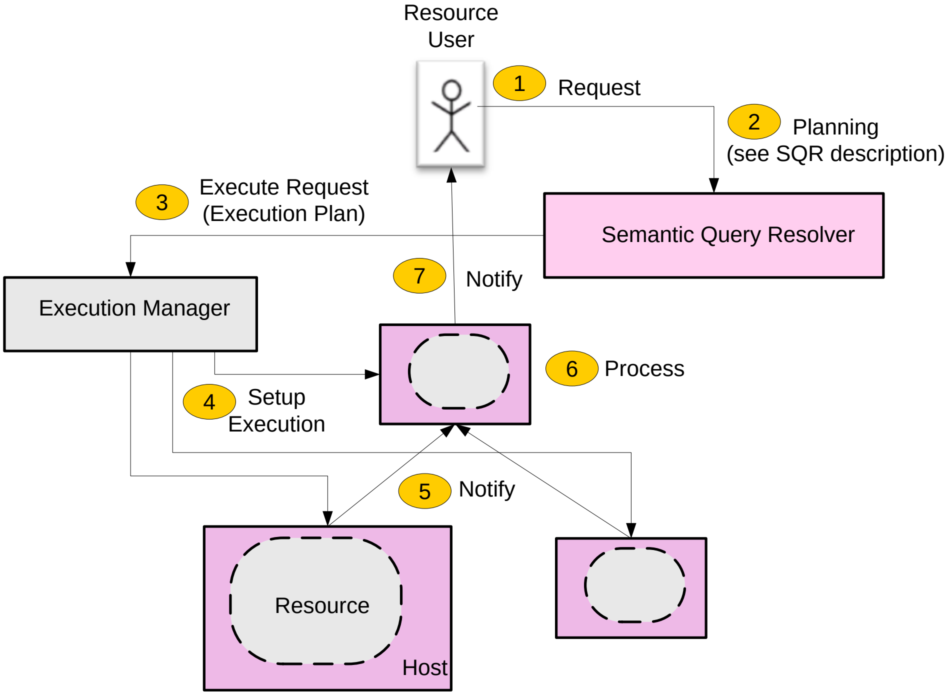

Figure 2-5: Execution of Long-Term Request

For a one-shot interaction, the Execution Manager retrieves the information and returns it to the resource user via the SQR. For long term interactions, the Execution Manager sets up information sessions between the Resources involved (Figure 2-5, Step 4) and the Resource User to allow a direct data flow of notifications (Figure 2-5, Step 5 & 7) , thereby avoiding to have a centralized component that becomes a bottleneck. The Execution Manager can also set up monitoring of resources and relevant context parameters in order to detect changes to availability off initially selected resources and newly appearing resources in the system during request execution time.), Based on this knowledge appropriate actions can be taken by the Execution manager to ensure the continuity and desired quality of the interaction.

The SQR represents a powerful tool for the resource users; however it has to be noted the SQR can only use advanced resources that contain semantically annotated descriptions of their capabilities.

2.4 Community Management Services

Community management services support the relationships between SENSEI community members (i.e. context and actuation service providers and consumers), and also between community members and the *SENSEI Framework Provider*s (the whole framework may not be provided by a single business entity). *Community Management* functions include user account management, identity management, security and privacy functions, etc. This part of the architecture must be very flexible, as the relationships within the SENSEI system will be dependent on the deployment scenario. For example, the "Crisis Management" scenario [SENSEI-D1.1] requires a tightly managed SENSEI system, where the stakeholders are well defined and identities strongly certified. This environment can rely on a centralised trust model (i.e. the framework providers can be a trusted party for all other entities in the system), which shapes many of the security options. Other scenarios, such as the "Multimodal Traveller" [SENSEI-D.1.1], have a much more open community, with a very dynamic and loosely managed *SENSEI Framework*. In this scenario, the trust model cannot rely on a central trusted party, and must adopt different approaches to the security architecture. While all this breadth of scenarios can use the same SENSEI architecture, the *Community Management* functions must be more tailored to take into account specific high level constraints.
The key contributions of SENSEI to the community management services are in identity management and access control approach (Authentication, Authorisation and Accounting, or AAA), privacy and trust management. In this white paper only the AAA and privacy contributions are described while trust management contributions are omitted as the work is still in progress.

2.4.1 AAA Functional view

The AAA functionality is organised into the architecture depicted in Figure 2-6. Solid rectangles represent functional blocks, circled lines between them are interfaces and grey pipes are security layers on existing interfaces. A security layer on an existing interface provides standard security-related support to the interface e.g. authentication of endpoints, confidentiality, integrity, etc.

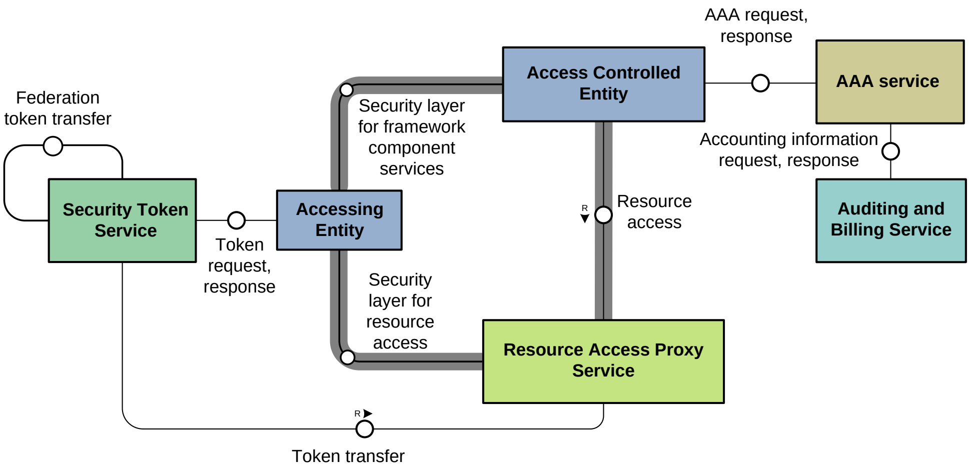

Figure 2-6: AAA functional architecture

The two parties being supported by the AAA services are the Accessing Entities (AE, e.g. resource users or resources) and the Access Controlled Entities (ACE, e.g. framework components or REPs). The architecture uses the same interfaces for all access controlled entities, although different implementations may be used for different resources and framework components.

The first AAA support service is the Security Token Service (STS), which provides the AE with the security assertions they will need to access the ACEs. The nature of the tokens being provided is dependent on the type of STS. The STS provides interfaces for token transfer within one administrative domain or between STS across administrative domains (federation).

The AAA service is the second major component of the architecture. Tokens have been acquired by the AE and provided with a service request to the ACE, which must establish whether or not the request should be granted. This decision is performed by the AAA service, and the answer returned to the ACE (resource end point or framework component).

The Auditing and Billing Service is a central point for collecting accounting information and performing analysis, including auditing, on it. It allows the framework provider to obtain an overall view of the actions being performed by its registered users and resource providers. Additionally this service provides a billing capability, access control to be based purely on payment and ensuring that users can perform the required transactions with maximum privacy.

When AEs (e.g. resource users) wish to access ACEs (e.g. resources) in another domain, they have several steps to perform. Firstly, they need to obtain translated tokens suitable for use in that domain. Translation may involve a change in format (for example from Kerberos to SAML) and mapping of attributes (for example the “manager” and “engineer” roles in one domain may map to a “general staff” role in another). Then, they need to go to each of the resources they wish to access, provide the tokens (to allow an access control decision to be made), negotiate a security session and agree keys, set up the security session and, finally, obtain the sensor data or perform the actuation. The Resource Access Proxy Service is provided to handle this efficiently on behalf of the user.

2.4.2 Privacy support

As the *SENSEI Framework* is mainly about sensing and communicating information, it is natural that privacy concerns arise. Since privacy protection (or data protection), in addition to being desired by users, is also required by European Union directives and national legislation, adequate protection has to be provided by SENSEI. There are two categories of privacy issues in the SENSEI system:

- **Real world privacy issues:** Real-world privacy issues include the privacy of personal information collected by sensors. Access to this information via the SENSEI framework is controlled by use of the AAA architecture to ensure that only those who are authorised to see and/or use the data can do so.
- **Electronic privacy issues:** Electronic privacy issues include people leaving digital traces of their movement and actions in various places. Privacy protecting systems aim at concealing the connection between the traces and the people who caused them. While traceability is necessary under certain considerations (legal proceedings, business process accountability, etc), this capability must in general be restricted to specific entities at specific times. The management of this capability is central to the flexibility of the overall privacy framework within the system. The AAA architecture provides a range of features to allow users to control how difficult it is to link their traces to them, such as use of pseudonyms or attributed instead of recognisable identities, single use tokens, etc.

2.5 Resource Model

Resources in SENSEI represent the Real World Information providers, which offer information through a set of operations (or services). Taking into account the heterogeneity of existing protocols, technologies and information providers (resources), it cannot be assumed that a unique interface will be able to meet all the requirements. For this reason, and considering that no uniformity can be expected at the interface level (as happens usually in the systems), the main goal of the Resource Model is to provide homogeneity at the description level.

An important design decision has been to keep the balance between an easy integration of existing resources, while providing powerful information about their capabilities.

The resource model [SENSEI-D2.3] is divided in two parts:

- The *basic* resource description: This part contains information such as an identifier (ID), a name for the resource, a set of tags where the basic capabilities can be described, and also the syntactic description of the interfaces to access the resources (usually the WSDL[WSDL], WADL[WADL] or WIDL[WIDL] description of the resource).
- The *advanced* resource description: This part is considered optional and includes semantic descriptions for the operations and information provided by the resource. Utilising the advanced description, some powerful functions such as dynamic resource composition offered by SENSEI components can be provided to the final users.

2.6 Information Model

The SENSEI Information Model [SENSEI-D2.3] is shown in Figure 2-7. It consists of three layers: the *raw data* provided by sensors, the *observation and measurement* (O&M) provided by sensor resources, and the *context information* provided by advanced system components or context-level resources.

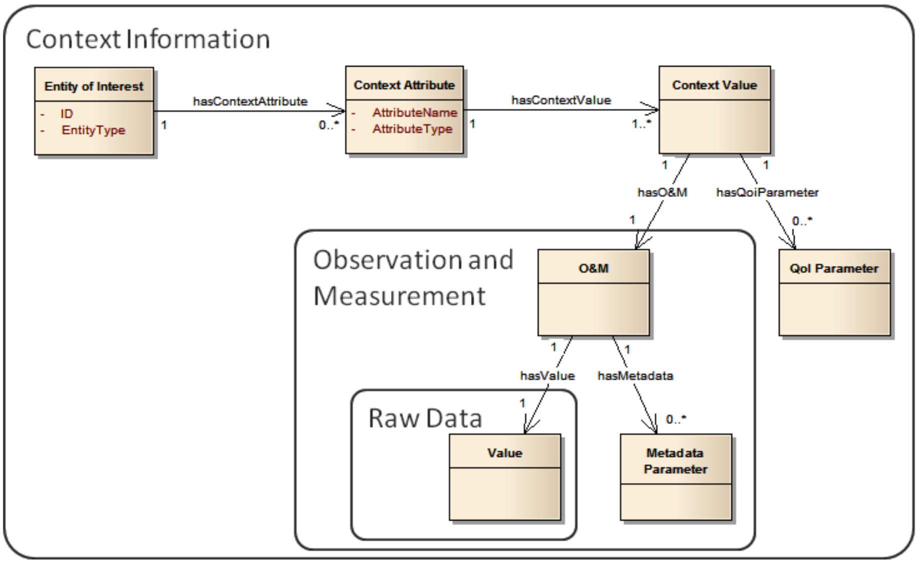

Figure 2-7: SENSEI Information Model

The raw data consists of the value that the sensor provides, e.g., the numerical value 25. A resource may augment this information with meta-information, i.e., that the measured value is a temperature, that it is in degrees Celsius, that it was measured by sensor X at a certain point in time etc. We call the resulting information observation and measurement (O&M).
This information is not yet contextualized, i.e., we may know that it is a temperature, but not what this temperature describes, e.g., is this the indoor temperature of a room or the temperature within my fridge? This information is modelled by the context information. The real world is modelled as Entities of Interest, which are further described by context attributes. The Entities of Interest have an identifier and an entity type. The entity type, e.g., person, place, or car, defines the context attributes of an entity of interest. For example, a person can have a blood pressure; whereas a car can have a certain fuel level. The context attributes have an attribute name, e.g., "hasIndoorTemperature", an attribute type, e.g., "temperature" and a context value. The context value consists of an O&M plus some quality of information parameters. This quality of information may be different from the quality information that was provided by the observation and measurement, e.g., the accuracy for a room temperature may be calculated as a function of the accuracy of the O&M and the reliability that the temperature sensor provides the temperature of the room.
In the SENSEI project, we have decided to represent information using an RDF [RDF] encoding, taking into consideration the advantages of a semantic representation that allows automated reasoning.

2.7 Actuation

Besides gathering information about the real world from sensors, SENSEI enables its users to control and interact with the real world via an actuator infrastructure (e.g. thermostats, heating/cooling elements, servo-motors, vibrating elements, speakers, electronic displays)

Requirements: Actuation has quite similar requirements to sensing. Actuators need to be abstracted as resources so that they are discoverable and usable by the SENSEI framework components (SQR, AAA). Scalability to large numbers of independent actuators and resource users is also a common requirement to sensing.
In addition to sensing, *conflict resolution* needs to be considered for actuation (i.e. in case of conflicting requests to the same actuator, one can for example cancel lower priority requests).

Approach: similar to sensor and context retrieval, the SENSEI approach is to provide both direct access to actuator REPs (direct actuation), and a high level approach, where users formulate their requests to the SENSEI system via the SQR.

Direct actuation
An example of a direct access to an actuator is when the user issues a command set the power level of the actuator *speaker5* to 10, or the comment to turn on *lightX.* In this approach, the user sets the property **p** of *actuatorA* to value **v,** by directly accessing the respective actuator REP. The user contacts the REP of the actuator, and invokes the corresponding operation (e.g. by doing a POST on http://sensei.org/room1/lightSwitch1/status/ with "status=on"). Assuming the user has the required credentials (AAA framework) and there is no conflict with another request, the actuator performs the required task.

High-level actuation
High-level actuation can be of three types: a) entity/attribute-based (e.g. "*Turn on light in roomX"),* b) control loop (e.g. "m*aintain temp of room X to 20 degrees Celsius until told*

otherwise/until time x/for a duration of time d", or c) conditional (e.g. "*If Bob is in room X then turn on light in roomX")*

The user can formulate such actuation requests using the SENSEI Actuation Task Model which allows declarative specification of a desired outcome, without necessarily specifying how this outcome is to be achieved. Similar to the high-level context requests, these actuation requests are submitted to the SQR. The SQR then analyzes the request, searches for the relevant actuator(s) or tries to construct a complex orchestration of actuators in case no suitable one already exists, and finally delegates the execution and the monitoring to the Execution Manager.

Actuator Discovery

Relevant actuator resources are discovered similarly to sensing and processing resources, via searches in the Entity/Resource Directories. Nevertheless, for actuators it is relevant to express and query for their capabilities to actuate the real world, which is described via "pre-conditions" and "post-conditions". For example, a PUMP1 actuator may have an operation fillTank() with (precondition="tank is empty"), and (postcondition="tank is full"). Assuming the user requests the tank to be filled, it is easy to find how to do it automatically, by invoking the operation fillTank() on PUMP1.

Conflict resolution

For actuators special Resource End-Points are designed which include arbitration capabilities and prioritisation of actuation requests. In the simplest form, the arbitration can be performed using a FIFO mechanism, that simply delays all tasks until the currently running one is finished, then proceeds with the next, etc. Higher priority tasks can pre-empt lower priority ones.

Enhanced actuation resources

An advanced type of resources has been proposed in the SENSEI framework to provide enhanced actuation services. These resources are called *Actuation Space Resources (ASR).* An ASR is responsible for providing a set of actuation services, for example for a specified administrative domain (a company), a physical location (a building), or a type of actuation (heaters). ASRs provide insert/update/delete operations, and they accept as input actuation tasks, as specified by the Actuation Task Model. As all resources, they register with the RD, and with the ED the entities (and possibly actuation tasks) they are responsible for.

2.8 Management

The scope of management in SENSEI covers the management of WS&AN islands and the management of framework components. Management functionality is useful for lifecycle operations on the WS&AN islands or framework components (e.g. software updates), deployment planning and monitoring (performance monitoring, fault detection) and deployment provisioning (e.g. configuration). In SENSEI we assume that these operations can be abstracted as special types of resources (called management resources) which can be accessed by specific resource users (e.g. operator or administrator of an WS&AN island) as opposed to any resource user. Therefore management in SENSEI is an intrinsic part of the architecture and usage of management resources is not different than normal interaction with a resource.

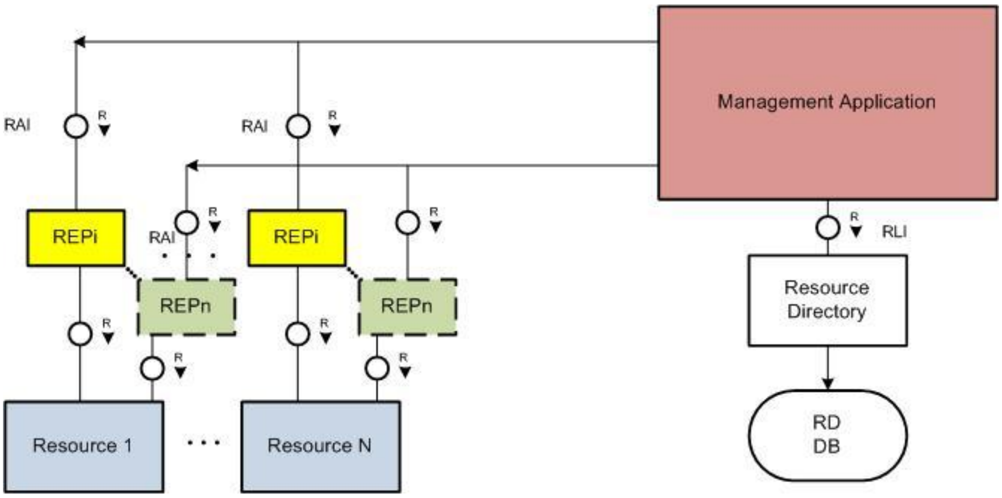

Figure 2-8: Management mechanism overview.

As Figure 2-8 illustrates, management interaction with a resource is performed using the standard resource endpoints (REP) and resource access interfaces (RAI) that are part of the SENSEI architecture defining a resource. A management application is just a resource user (RU) as seen from the SENSEI architectural point of view. As other resource users, a management application will do resource discovery of the resources it intends to manage. Once the needed resource descriptions have been retrieved, the management application can use management associated resource access interfaces hosted by the resource endpoint to perform management. For example, if the management application provides a Graphical User Interface (GUI) that displays the values of configurable parameters (e.g. sample rate, power consumption settings) then the management application will use the management associated interfaces to retrieve this information from the endpoint in order to acquire the values before they are displayed. In the same manner, if the management application allows the end user to perform code updates to a selected set of resources (and the resource provides such management functionality) then the management applications may allow the end user to select the wanted resources, select the file where the new resource code is located and allow the user to activate the code update. The management application will then use the associated code update interfaces of all the selected resources to transfer the new code and to activate the code update to the selected resources.

As an example, consider a temperature sensor resource. To read the temperature value from the sensor a resource user needs to access (read from) the interface associated with the temperature value. In a similar fashion, some interfaces will be associated with management functionality. If for example the temperature sensor resource supports configurations of its sampling rate (how often the underlying temperature sensor actually samples the temperature), then the temperature sensor endpoint will have to support interfaces that allows a resource user to read and set the sample rate. Further, if a resource supports code update as a management option, the resource endpoint will have to support a code update interface. These interfaces will be regarded as (part of) the endpoints' management interface. In order to make these management interfaces discoverable for management resource users, the resource descriptions need to be specifically tagged with information that identifies them as management resources.

In SENSEI, the unique resource identifier is formed as a URN where the domain is the first part after the "urn:sensei" namespace identifier. Following this, a valid resource identifier for a temperature sensor resource allowing users to set the temperature sampling rate could have the following name:

urn:sensei:**telenor.com**:NanoSensorN740:Temperature:SampleRate:4e3a-41aa-0123-e412

The naming defines that the administrative domain of this resource is *telenor.com.* All resources in SENSEI (including management resources) belong to a domain and the unique resource identifier thus holds meta-information that pinpoints a (management) resource to a domain. In addition to this, the resource description contains other types of management tags.

Section 3 - Key Features and Benefits

SENSEI offers several technological benefits which are summarized in the following.

Flexible access to real word information and automated interaction: SENSEI offers real world resources that expose raw data as well as processed information from sensors. Interaction with the real world through simple actuators as well as control loops is also possible with SENSEI. Each resource provider has the option to expose simple or complex interfaces for information retrieval and interaction. This makes integration between SENSEI and other systems easier and smoother.

Decoupled Resource Endpoints: SENSEI offers information and interaction capabilities through resource endpoints that are not necessarily collocated with the actual real world resources. This provides the resource providers the option to deploy the resource endpoint on a more capable hardware than the one that typically hosts a simple sensor and actuator. In turn this enables protection of the simple resources from attacks such as denial of service, energy draining or protection of human users from harmful actuation: the underlying assumption is that simple sensors or actuators have enough processing power or the situation/context awareness to detect these attacks.

Real world resource discovery: SENSEI offers multiple directories for finding the appropriate real world resources. These directories maintain increasingly complex information that can be used by different other services and external-to-SENSEI systems: They offer simple resource descriptions containing resource endpoint URLs or tags summarizing the functionality of resources as well as complex meta-data to facilitate resource composition.

Sophisticated information processing and actuation control capabilities: SENSEI enables the creation of complex information processing and actuation control. Raw data from simple sensor resources can be processed and aggregated by different processing resources. SENSEI offers functional components to dynamically instantiate such processing plans. Likewise complex actuation control loops can be constructed dynamically using SENSEI support services.

Resource and Service composition: SENSEI allows the integration of sensor information and actuator control resources and services into more complex composed services.

Plug-and-play: SENSEI offers functional support for implementation of automatic configuration of a resource in order to join the SENSEI support services if desirable.

Entity context model: SENSEI offers the flexibility to associate real world entities with resources and enable the access and discovery of information and interaction services by using the SENSEI context model.

Flexible business model support: SENSEI offers the option for resource providers to offer the resource information or interaction services with the desired access control and compensation (free vs for a fee). The SENSEI community management support services enable authentication, authorization, accounting and access control functions that are essential in an information marketplace. The same functions can also be used for offering free services. The difference in the free access case is that there are no access control rules and the accounting data are not used for billing.

Section 4 - An example use case from the automotive industry

This scenario is entirely fed by interviews conducted by SENSEI partners towards industrial stakeholders about their perception of the SENSEI framework.

Main actors and roles. The scenario describes the benefits of the SENSEI system in automotive industry, in particular for car manufacturers which play the role of the application service provider. A current trend in the automotive industry is to connect vehicles with other entities in the physical environment in order to provide drivers context information and actuation services. Other actors in this domain are mainly entities that provide context information and actuation services: road infrastructure owners, buildings owners, meteorological institutes (content provider), telecom operator (connectivity provider)or electricity network owner/manager (third party service provider). The users' preferences and behavior could be factored in their profiles in order to align services to the drivers' liking.

Before SENSEI – Before SENSEI, simultaneous access in real time to WSAN services was impossible. It was unfeasible to access the needed information separately beforehand. Vehicles already embed local AmI systems fed by integrated WSANs. Nowadays, the trend is to mesh services together and to connect them to the physical environment.

With SENSEI - The future vehicle is considered as WSAN-enhanced entity that is able to exploit information coming from other entities and to feed other entities with its own data. The intention of the car manufacturers is to develop this vision of the car-provider/consumer for context information/actuation. The future vehicle needs a SENSEI framework to become a terminal connected to the Internet.

Horizontalisation – The vehicle manufacturers foresee that the SENSEI framework could enable innovative cars to establish a multiplicity of RWI interactions with sensing and actuation entities. For instance:

- *Weather information (content provider)* - The obvious next step will consist in interfacing the weather stations to the car navigation systems so that they will exchange information about condition.
- *Road infrastructure (WSAN service providers)* - It is also expected that an interaction between vehicles and the road infrastructure information will become reality (about traffic, state of roads etc.).
- *Other vehicles (WSAN service providers)* – At mid-term each car will start to interact with other vehicles and finally at long terms will support more ambitious traffic schemes were vehicles are chained.
- *Built environment (WSAN service providers)* - Reusing context information and actuation deployed in the urban and built environment is a next step of deployment of RWI applications in vehicles.
- *Electricity network (3rd party service provider)* - Interfacing cars with the environment infrastructure could enable huge energy savings. For instance, industrials are considering future generation of vehicles able to communicate with the lighting so that lights are adjusted to the traffic conditions and presence of vehicles.

- *User dynamic profile (Application service consumer)* - Considered as any Internet terminal, the future vehicles will be interfaced with personalised context aware applications adapted to the user profile, behaviour or preferences. For instance, individual or shared vehicles could be adjusted to the drivers needs, connected to their professional or personal social networks (e.g. staff/friends/family members' vehicle), to home/company's automated applications or to other internet devices.

In this horizontalised framework, interactions created between WSANs entities enable each actor to become a potential data consumer and provider for each other party.

Benefits - Along with comfort and security gains for the drivers and car passengers, further rationales are:

- *Sustainable business models*: enabling more intelligent vehicles will permit producing safer vehicles and to shift from passive security to active security. Active security means intelligent embedded systems preventing from crashes. Moreover, saving raw materials (initially meant to protect passengers) in vehicles means reducing the weight of cars to make more efficient and appealing electric cars for customers in terms of performances.
- *Smart grid*: making intelligent and lighter vehicles will promote electric vehicles that will require energy schemes to provide, exchange and harvest power. Connected to intelligent networks, vehicles will be able to share and trade the consumed and produced power together or with other entities.
- *Carbon footprint*: by encouraging the development of greener vehicles and new transport schemes, the SENSEI system will contribute to the reduction of the environmental impact of the transport activity. Networked and context aware vehicles will require fewer extracted natural resources, will reduce the raw material supply chain and will decrease the power consumption of vehicles.

Deployment - Industrials account that the innovation trend is launched in their companies, supported by political voluntarism and heavy societal trends. The deployment of horizontalized information and actuation schemes will be incremental, each step supporting the feasibility of the next one. Interaction between vehicles with weather or road infrastructure is an ongoing trend. More ambitious schemes such as integrating vehicles in the smart grid framework is long-term perspective.

Section 5 - Conclusion

The next generation of smart applications and business processes will increasingly rely on the availability of real world information and interaction capabilities on the Internet. The SENSEI architecture provides the necessary foundations for enabling such a real world Internet, allowing heterogeneous sensor and actuator networks (SAN) deployment of different ownerships to be integrated into a global market place for real world information and interaction. SENSEI breaks up the current vertically integrated domain specific silos and allows SAN infrastructure to be re-used across a variety of different application domains.

The SENSEI architecture adds to the current and future Internet various essential functions that are useful for services and applications, as well as operators of sensor or actuator network infrastructure.

Applications and services can discover suitable sensor and actuator resources by simple lookup and declarative queries or can directly query for context information and inject actuation tasks concerning entities of the physical world. SENSEI smoothly handles the continuity of longer lasting queries and system interactions, despite complex physical world dynamics while maintaining wherever possible a request quality of information and actuation level. SENSEI provides these functional features via expressive interfaces, greatly simplifying the development of services and applications.

The management support of the SENSEI architecture lowers the barriers of deployment for operators of wireless sensor and actuator networks. The flexible community management functions and corresponding AAA framework ensures the realisation of a variety of different business models and provides necessary economic incentives to further fuel the deployment of WSAN infrastructure. SENSEI is thus suitable for closely secured business and enterprise systems or for open user communities.

A PAN European testbed is currently under construction across various partners of the SENSEI consortium. Many of the aforementioned functionalities and features have been implemented on top of emerging web technologies and are currently being tested for a variety of different field trials.

This white paper provided only a high level view of the SENSEI architecture and its offered service capabilities. Further information can be obtained on the SENSEI website (www.sensei-project.eu). The website provides access to a variety of different reports and publications, ranging from more detailed technical specifications to analyses of market opportunities and corresponding business models.

Section 6 - References

[SENSEI-D1.1] D. Wilson (Editor), " D.1.1 - SENSEI Scenario Portfolio, User and Context Requirements", SENSEI, Public Deliverable D.1.1, 2009, http://www.sensei-project.eu/

[SENSEI-D.2.3] M. Rossi (Editor), "D.2.3 - Preliminary Context Model, Interfaces and Processing Mechanisms for Sensor Information Services", SENSEI Public Deliverable D.2.3, 2009, http://www.sensei-project.eu/.

[SENSEI-D.3.1] R. Gold (Editor), D.3.1 "State of the art – Sensor frameworks and future Internet", SENSEI Public Deliverable D.3.1, 2008, http://www.sensei-project.eu/

[SENSEI-D.3.2] F. Carrez (Editor), "D.3.2 –Reference Architecture". SENSEI, Public Deliverable D.3.2, 2009, http://www.sensei-project.eu/

[SENSEI-D.3.3] T. Bauge (Editor), "D.3.3 - Components for End-to-End Networking, Management and Security and Accounting", SENSEI, Public Deliverable D.3.3, 2009, http://www.sensei-project.eu/.

[WADL] Web Application Description Language: https://wadl.dev.java.net/

[WIDL] Web Interface Definition Language: http://www.w3.org/TR/NOTE-widl-970922

[WSDL] Web Services Description Language: http://www.w3.org/TR/wsdl

[RDF] Resource Description Format: http://www.w3.org/TR/REC-rdf-syntax/

Section 7 - Table of Acronyms

AAA	Authentication, Authorization, Accounting
AE	Accessing Entity
ACE	Access Controlled Entity
ASR	Actuation Space Resource
ED	Entity Directory
EM	Execution Manager
EoI	Entities of Interest
GUI	Graphical User Interface
O&M	Observations and Measurements
PET	Pan-European Testbed
RAI	Resource Access Interface
RD	Resource Directory
RDF	Resource Description Format
REP	Resource End Point
RWI	Real World Internet
SQR	Semantic Query Resolver
STS	Security Token Service
URI	Universal Resource Identifier
URN	Universal Resource Name
WADL	Web Application Description Language
WIDL	Web Interface Definition Language
WSDL	Web Services Description Language
WS&AN	Wireless Sensor and Actuator Network