Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 1277–1284

ISBN 978-83-60810-51-4

# OpenNode: A Smart Secondary Substation Node and its Integration in a Distribution Grid of the Future

Marta Alberto
Atos Worldgrid
Av. Diagonal 200
08018 Barcelona, SPAIN
marta.alberto@atos.net

Raúl Soriano
Energy Technological Institute (ITE)
Juan de la Cierva, 24
46980 Paterna, SPAIN
raul.soriano@ite.es

Jürgen Götz
Siemens AG, Corporate Technology
Otto-Hahn-Ring 6
81739 Muenchen, GERMANY
juergen.goetz@siemens.com

Ralf Mosshammer
Siemens AG Austria
Siemensstraße 90
1210 Vienna, AUSTRIA
ralf.mosshammer@siemens.com

Nicolás Espejo
Núcleo de Comunicaciones y Control, S.L.
Av. de la industria, 24
28760 Tres Cantos, SPAIN
nicolas.espejo@nucleodf.com

Florent Leménager
EDF R&D
1 av. du Général de Gaulle
92141 Clamart, FRANCE
florent.lemenager@edf.fr

Raúl Bachiller
Iberdrola
Av. de Manoteras, 20
28050 Madrid, SPAIN
rbpr@iberdrola.es

***Abstract*—This paper presents the EU project OpenNode from a technological point of view. OpenNode designs a massively distributed system (both in number of nodes and geographically) for managing the operations of a distribution smart grid. It describes the components, innovations, problems faced and what we call distribution of intelligence. The proposed system is prepared to being a base platform where build a smart grid, able to manage AMM, AMI, DSM, DER, and EV integration. Finally the system is proposed as a base platform to build MAS systems for distribution grid automation and management.**

## I. Introduction

ELECTRICAL Infrastructures are facing important challenges on the last years. Market liberalization, new regulatory framework, new generation technologies, customer centric networks, demand rising, distributed generation, environmental issues, security of supply or infrastructure aging are driving factors that require an action.

The new electricity supply network will be based on incremental changes over traditional grid. Electrical system is too complex to face a revolutionary change, so the introduction of new technologies must interface with old ones. Assets management and return of investments stress the problem too.

In addition, the European energy industry is changing due to the liberalization of the EU energy markets and the new model where prices are determined by supply and demand rather than through regulations.

All these targets lead to high risks associated with a complete or partial collapse of an already fragile energy infrastructure. It is obvious that a massive investment in the distribution network infrastructure by the Utilities is required.

OpenNode project [1] focuses on the electrical distribution grid operation and explores answers on the challenges introduced:

1. How we can improve the distribution grid monitoring to cope with unstable states in the grid.
2. How we can integrate the "smart" substation automation devices to increase the efficiency of the distribution grid.
3. How we can inter-operate with the different roles e.g. smart meters maintenance, grid operation, etc.

OpenNode's R&D efforts converge in the creation of an open ***Secondary Substation Node*** (SSN) as an essential control component of the smart distribution grid, a ***Middleware*** (MW) to couple the SSN operation with the Utility systems for grid operation and a modular standard-based ***Communication Architecture*** to grant the flexibility required by the stakeholder diversification and to cope with massively distributed embedded systems in the distribution grid.

Section II introduces OpenNode's overall architecture and Sections III to V describe the main parts of OpenNode's architecture (Secondary Substation Nodes, Middleware and Communications Architecture respectively), challenges and design decisions. Section VI describes the test procedures, and, finally, Section VII deals with conclusions and next steps.

The research leading to these results has received funding from the European Community's Seventh Framework Program (FP7/2007-2013) under Grant Agreement Number 248119

## II. OpenNode architecture overview

OpenNode brings together the two main aspects of future smart distribution grids [2]:

- The communication with customer meters supported by the use of standardized protocols such as DLMS/COSEM.
- The metering infrastructure goes hand in hand with automation of the distribution network, using legacy (IEC 60870-5-104) and future-oriented protocols (IEC 61850).

OpenNode's overall architecture is presented in Fig 1. The main components of the architecture are the network of ***SSN***, the ***MW*** and a modular ***Communication Architecture***.

The ***SSN*** is the essential part of the system. The innovative outcome is the definition of a modular and extensible software architecture inside the SSN. This allows inclusion of third-party applications (e.g advanced grid control algorithms) which can be dynamically installed during live operation. This extension capability enables a gradual, controlled shift of grid intelligence from centralized systems in the Utility control centre towards lower echelons of the grid hierarchy, and provides means to balance and optimize control intelligence across future distributed grid devices [3].

The ***MW*** system is the orchestrator of all OpenNode subsystems: facilitating the integration between the Utility systems and other stakeholders; handling the technical information of all the devices of the system; storing measurements data; managing the software provisioning; supervising the grid status and dispatching SCADA control actions.

The ***Communication Architecture*** is based on open communication standards. The data exchanged between the smart meters (SM) located at client premises and the SSN is based on the outcome of OPENmeter project [4], which, in turn, is based on PRIME PLC (PoweRline Intelligent Metering Evolution).

For the communication between the SSN and the MW it is designed a two-path based communication strategy, each of them with different transmission requirements (see Fig 2):

- the metering data path and
- the grid automation data path.

## III. Secondary Substation Node (SSN)

One of the aims of this system is to serve as aggregation point for status monitoring and metering management. The SSN is also devoted to act as a distributed system for control purposes able to take decisions autonomously as well as remotely controlled. The SSN is integrated in secondary substations (SS), acting as a distributed control system for the medium to low voltage distribution network.

The SSN aggregates information from the connected SMs and the local devices available at the SS, providing metering and automation data. Some examples are: acquisition of customer energy consumption in regular intervals, polling SMs to determine losses of power and real time monitoring of power quality at the SS.

The SSN has interfaces to control the SMs; e.g. remote controlled operation of SMs, remote SM contract management, etc. Moreover, the SSN is able to interact with the local devices, sensors and any electrical equipment available in the SS as well.

The aggregation and control capabilities are based on local application running in the SSN. For instance, based on SM metering measurements and local devices automation data, the SSN can differentiate between the regular operation of the grid and outages which are signaled to the MW. Other verifications that could run locally at the SSN are overloads detection or forecast calculations for maintenance purposes.

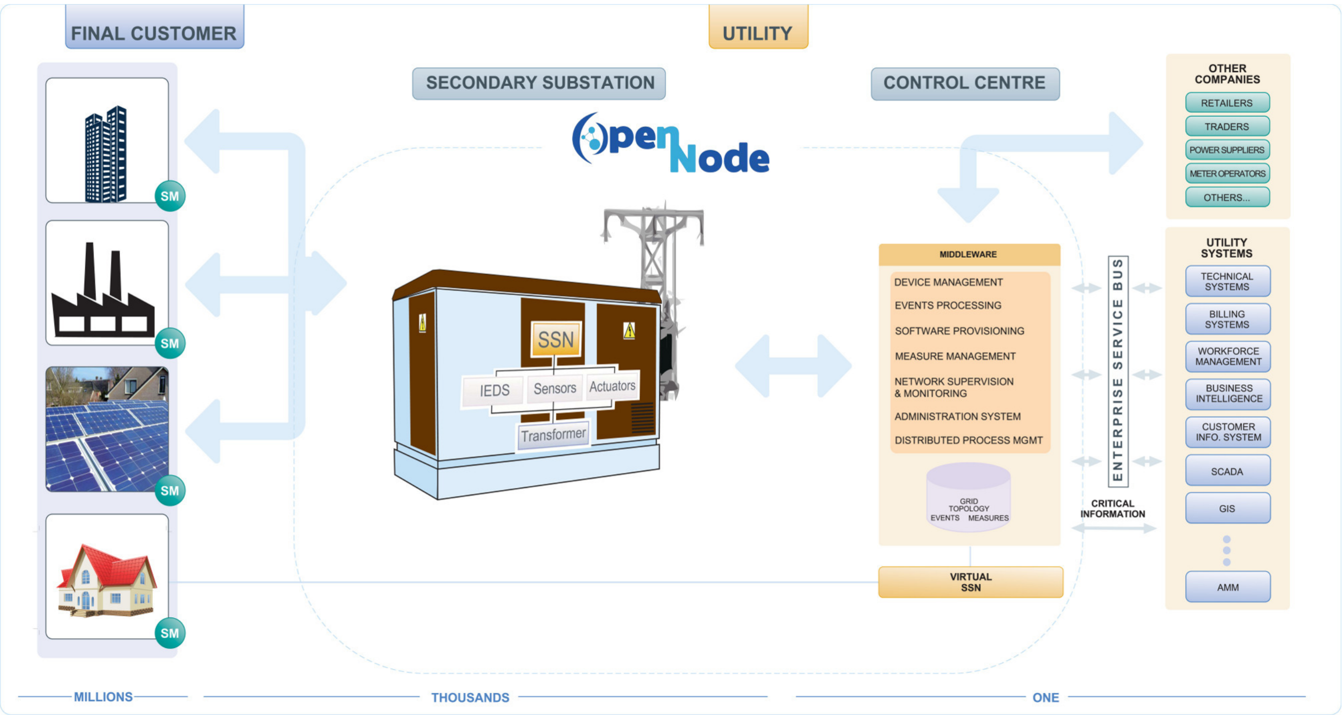


Fig 1: OpenNode's overall framework

Besides, the SSN might receive remote control information from the MW. For instance, to act over local circuit switches based on the overall strategies determined in the Utility control centre.

The OpenNode project searches for open and testable solutions. So, two levels of integration have been necessary:

- Basic integration: local devices are connected to the SSN. It acquires real time information, might perform some basic actions and informs the MW.
- Full integration: includes connection with standardized Intelligent Electronic Devices (IED) such as the IEC 61850 compliance ones. Thus, modern SS could integrate control and protection within a metering processing unit.

At the highest level, from the software point of view, the SSN can be decomposed in two partitions (Fig 3):

- Base Partition (BP): Contains the software needed to fulfill the SSN basic functionality, as well as the interfaces for communicating with the Extensibility Partition.
- Extensibility Partition (EP): Allocates the software components needed to extend securely the functionality of a SSN, as well as the extension modules themselves.

The name “partition” has been selected on purpose. The intention is to isolate as much as possible the extension modules from the rest of the software, for security reasons. No extension should compromise the basic functionality of the SSN. The concept “partition” is already used in a similar sense in security-based and even safety-critical software.

Each SSN manufacturer must provide a SSN with the hardware, the BP and that part of the EP that enables the deployment of extension modules.

The extension applications are meant to be implemented mainly via extension modules. These extension modules would be deployed inside the EP, and could made use of the functionality served by the BP through well defined interfaces.

The extension modules can be implemented by third parties, as well as by the SSN manufacturers themselves. One goal of this architecture is to facilitate an “application level“ modularity for these modules.

The BP is decomposed in the following layers:

- Operating system.
- Base framework: modularity enabler that eases life cycle management (start, stop, load, unload) of the software modules. The proposed framework has been Qt. Other frameworks could be used seamless as long as they provide the aforementioned capabilities.
- Base foundation: enables the access to SSN resources such as data files, communication services, external sensor states, etc.

The EP is made of the following layers:

- Java Virtual Machine.
- Extension container: modularity enabler that facilitates managing the life cycle of the extended application modules (EAM) which can be started, stopped, loaded and unloaded on top of this container. The framework chosen has been OSGi [5] (Open Services Gateway initiative). However, other container could be used whether it realizes the defined interfaces.

Finally, the main software interfaces defined in this architecture are two:

- Interface between the BP and the EP, named CFI. This interface guarantees that the EAM modules don't compromise the BP functionality.
- Interface between the EAMs and the container, named ECI. Allows that the EAM could access to the SSN functionalities.

## A. EAM/BAM

The extended application modules (EAM) are modules which can be developed by third parties and can be installed started, stopped, and unloaded without rebooting the SSN.

The basic application modules (BAM) implement the SSN core functions.

## B. OSGi for the EAM

The EAM is based on OSGi, which is an industry-strength application framework providing a service oriented architecture (SOA) to applications running on top of it. OSGi manages software packages as “bundles”, where each

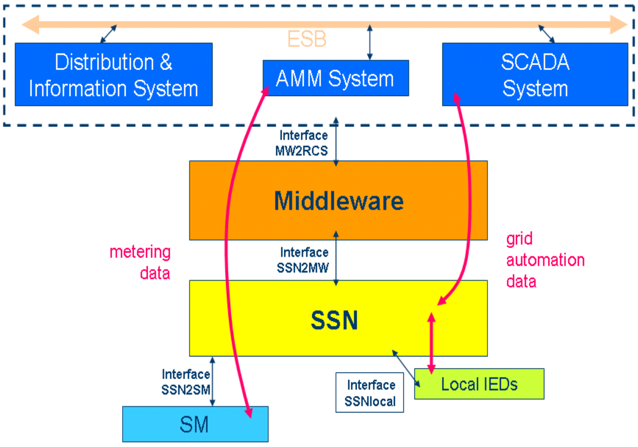


Fig 2: Data paths in the OpenNode project

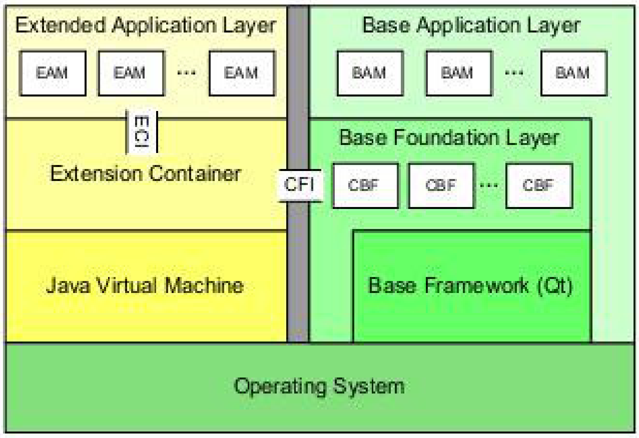


Fig 3: SSN Software decomposition inside partitions

bundle represents a stand-alone process or function which exposes (or hides) services to other bundles, in accordance with the SOA principle. All software bundles run in parallel on the framework and can individually be installed, uninstalled, started and stopped. This facility makes OSGi extremely attractive for the SSN. To move intelligence away from central systems, such as the SCADA or MW, algorithms and logic are downloaded on-the-fly, with no need to reboot or flash the system. Once the algorithm is distributed to all SSNs, it can be launched in parallel. This is made possible via a special configuration Web Service.

An example of trivial usage of these facilities would be critical value monitoring, e.g. on a busbar voltage. If the busbar voltage exceeds a critical threshold or experiences a fast up- or downward slope, there should be an appropriate reaction, e.g. activating circuit breakers to split currents across multiple transformers. In a centralized set-up, the MW or the SCADA would have to constantly monitor (poll) the busbar voltage value. To distribute this intelligence, an algorithm can be downloaded to the SSN to autonomously monitor the busbar voltage and take an appropriate action. The higher-tier systems would only be notified with an alarm flag, rather than having the actual values transmitted constantly across a potentially bandwidth-limited channel.

### *C. Data oriented Java API*

Programming of algorithms for the SSN is done in the industry-standard Java language. Via the OSGi interface, the SSN exposes its internal data model to algorithm bundles. The data exposition process is event-based where changes in the data model trigger events; e.g. due to a switch changing position or a voltage changing value. The user algorithm subscribes to a specific set of events related to data points; it is immediately notified of the value change and can react accordingly in a callback function.

Internally, the data model is based on the industry standard IEC 61850. All data points are represented in the canonical tree structure of IEC 61850. Data points are addressed with a simplified path syntax, e.g. the following expression could be used for a phase-to-phase measurement value between phases A and B:

```
IED002MEAS/MMXU1.PPV.phsAB.cVal.mag.f
```

Aside from reacting to value changes, the algorithm can also write back into the data model to effect changes in underlying systems or notify the MW. Thus, all operations of the algorithm happen solely on the data model, completely abstracting underlying automation hardware and higher-tier system architectures.

In conformance with the BAM/EAM security concept, the user algorithm operates solely on the data model, with the underlying hardware being completely abstracted. Together with the service-oriented mechanism of interface exposition OSGi provides, this not only allows vendor-specific hardware on the automation side to be swapped without any changes to the algorithm; it also precludes users from inadvertently or maliciously manipulating core automation functionality.

## IV. Middleware (MW)

The Middleware (MW) is the OpenNode system responsible of interacting with the SSNs through the communications network. It's located at the Utility control centre facilities.

The goals accomplished by the MW are:

1. Acting as a gate to the SSNs, delivering orders to the SSNs that come from the stakeholders and storing the information that comes from the SSNs.
2. Facilitate the integration to couple the power grid operation with the stakeholders.
3. Provide a suitable storage repository that can cope with the foreseen big data amount of measurements: 50 TB a day.
4. Introduce intelligent modules that allows monitoring the whole system in near real time.

To achieve these goals, the following approaches have been adopted:

1. Employment of standard communication and data model protocols, including future-oriented data model protocols such as IEC 61850. Also a special COSEM to Web Service mapping has been designed to overcome the gap of remote data exchange of COSEM data with the SSNs.
2. Mapping data models (COSEM to CIM) to facilitate the integration with the stakeholders.
3. Evaluation and implementation of a NoSQL [6] repository competent enough to deal with the big data and availability requirements.
4. Usage of an event correlator that allows including state aggregation and monitoring functions in a dynamic manner.

### *A. Integration strategy - CIM*

#### *1) Challenges*

The MW must provide means to integrate any Utility system with the SSN network in both directions.

Utilities' systems may have different and proprietary information models that have no relationship between them, but also they have their own set of specialized services which will need to interact with the OpenNode system .

In addition, some Utilities' systems may need to be aware of the current grid status, so they will require receive OpenNode information in order to be up-to-date.

#### *2) Current trends*

Nowadays many Utilities are facing the same aforementioned challenges. Therefore, a standard has emerged to solve them: the Common Information Model, CIM [7].

CIM is a standard for power system data representation and exchange in an Utility control centre. Nowadays, some Utilities are working on gradually adopting CIM[8], manufacturers are developing CIM compliant applications and the standardization bodies are working on the CIM evolution.

#### *3) Approach*

CIM provides an abstract unified power system data model that allows the representation of all the major objects that any Utility system information model could need. Therefore CIM enables integration in any domain by means of a common information model that facilitates interoperability and exchange of information between systems.

Moreover, CIM defines a group of standard interface specifications for "plug-in" applications, therefore any application that is compliant with these interfaces will be able to communicate with any other CIM compatible application. With this approach it is possible to integrate, homogeneously, any disperse Utilities' system.

The main motivations of using CIM are :

- It is gaining acceptance on the Utilities [8].
- It is not necessary to implement the whole standard, only the required sections.

#### 4) CIM reference implementation

A COSEM measurement was selected as a representative information to be exchanged with the MW throught CIM to assess the feasibility of the CIM integration approach.

The class hierarchy for achieving the association of the metering measurements with the rest of CIM classes model has been specified. From this definition, the base information was used for exchange COSEM measurements.

### B. NoSQL storage engine

The big data issue is a challenge addressed currently in many business fields, but also in OpenNode, where a huge amount of information related to measurements and events was foreseen and must be stored. It was estimated that dozens of millions of measurements will be gathered daily (50 TB approx) and the format of this data is flexible (object-oriented following COSEM definition). Consequently, a classical approach of a relational database has been discarded in favour of a more revolutionary approach: NoSQL storage engines.

NoSQL is a new database generation that is facing the same challenges addressed at OpenNode: big data storage, flexible content and scalability in a cost effective fashion.

Table I Shows the five NoSQL solutions analysed. The main requisites that have been stressed at the evaluation are:

1. Store a huge amount of data in a competent way.
2. Flexible content type. Different types of data should be stored: events, and meter and sensors measurements.
3. Open solution and technologically promising.
4. Demonstrated effectiveness in real environments: scalability and high-availability is a must.
5. Product support and simple architecture.

The chosen candidate used at OpenNode is Apache Cassandra. Cassandra is a distributed storage system for managing structured data that is designed to scale to a very large size across many commodity servers, with no single point of failure. The main reasons for this selection are:

- Realizes the above requirements.
- Impressive real references: Facebook (until Nov 2010), Twitter, Cisco WebEx, Rackspace.

### C. Complex Event Processing

One of the main goals of the MW is to be able to monitor the whole power grid status.

The MW receives many information from the SSNs (measurements, status changes, configuration information, events, etc) and stores the asset information managed by the OpenNode system (discoveries, technical status, configurations, software versions, etc). Therefore it has a privileged position for providing an up-to-date picture of the power grid status by analysing the real time data that arrives to the MW jointly with the local asset information.

Moreover, the monitoring capability can be easily configurable, allowing non technical staff add or change the monitoring algorithms to be verified by the MW.

#### 1) Approach

The solution designed relies on a complex event processing engine that provides the base platform to allow monitoring in near real time all the information available (the data just arrived and those already stored locally).

The engine allows to analyse data by means of Event Processing Language (EPL) [9] rules. EPL is a declarative language, with a syntax like SQL, that allows grouping, aggregation, sorting or filtering event streams, include mathematical functions, access to relational databases and introduce temporal conditions. For instance, this EPL sentence generates an alarm if a certain value is exceed during 5 seconds:

```
Select ssnId from Event.win:time(5 sec)
where 104data.var='I' and 104data.value>
9f;
```

## V. Communications Architecture

The Communications Architecture from and to the SSN is crucial. OpenNode developed concepts and real implementations for the upper and lower layers of the communication links, taking into account legacy and future oriented technical possibilities and cost aspects. The rationale for these decisions are described below.

### A. Lower Layer Protocols

In OpenNode, we have provided a detailed description of the required lower layer protocols (layer 1-4) and network technologies in Deliverable 4.1 [10]. Table II summarizes the results.

TCP/IP is not mentioned in the table because the OpenNode solutions are planned to build on the Internet communication protocols.

Powerline fits well to the communication aspects for the interface between SM and the SSN. Also it would be expensive to build up a new communication infrastructure for the huge amount of SMs connected to an SSN in parallel to the already existing electrical grid infrastructure. The available data rate of a few 100 kbit/s is sufficient.

Table I:
NoSQL considered solutions for OpenNode

| Name | Developer | Type |
|---|---|---|
| Cassandra | Apache Foundation | BigTable |
| HBASE | Apache Foundation | BigTable |
| HyperTable | HyperTable.com | BigTable |
| MongoDb | 10gen | Document |
| CouchDb | Apache Foundation | Document |

Only WiFi or low-bandwidth cellular communications may be an alternative for those SM that will have problems to be connected by PLC technology.

Cellular networks like GPRS / UMTS are available all over Europe and offer a relatively high data rate from several 100 kbit/s up to about 7 Mbit/s. In future, UMTS LTE will even offer several tens of Mbit/s. The advantage of cellular networks is that no new IT network infrastructure must be built up by Utilities. However, the economic aspect here is crucial. In many countries, UMTS is still rather expensive.

## B. Data Models and Upper Layer Protocols

Before describing the preferred solutions, we have to mention the following restrictions that are important to keep in mind:

- It was necessary to provide a first prototype to be able to start the integration and test phase in an early stage of the project. It is important to keep the time constraints for OpenNode. Furthermore, it is helpful for the development of a good OpenNode prototype to get practical experiences in field and/or labo tests very soon in order to improve the prototype as much as possible. With this procedure we could initiate an iterative process of implementation, integration and evaluation between manufacturers and Utilities for a high-quality SSN prototype.
- Further reasons for a first prototype are the technological constraints which are found today in a real SS: there are no IEC 61850 devices, there are no good communications, etc.; so the only way to test a prototype in a real SS is to have legacy protocols and basic functions.
- This first prototype is focused on the provisioning and testing of the new OpenNode functions and use cases. The focus of the first prototype was not on evaluating and collecting experience with new protocols, since referring to newer standards will be a challenge.
- Furthermore, the integration conditions to Utilities' systems for realizing the field and lab tests are often based on "legacy" control centre systems. For instance, grid automation communication has to use IEC 60870-5-104 protocol, just because the SCADA systems of today don't provide other protocols. To integrate the OpenNode solution to the current control centres, we need to provide the connection to today's control centre data models, messages and protocols.
- The expectations of Utilities and project partners in the project are different, and therefore it was not sufficient for OpenNode only provide the first prototype. In addition, we needed a second, "enhanced", prototype as well to show the features of new protocols (like IEC 61850) and new communication possibilities (like Web Services).

Based on these aspects we follow a "migration path" towards a final vision for the OpenNode SSN with all the new OpenNode functionality and the introduction of new standards at all levels.

TABLE II:
LOWER LAYER PROTOCOLS IN OPENNODE

| Interface | Distance | Technology |
|---|---|---|
| SSN2SM | < 500 m | Narrowband Powerline (PRIME or G3-PLC)<br>GSM or WLAN (IEEE 802.11) |
| SSN2MW | 10's to 100's of km | Cellular wireless network (GPRS / UMTS) |
| MW2RCS | Local or 10's to 100's of km | Industrial Ethernet or optical fiber / WiMAX / UMTS |
| SSNlocal | Local (100's of m) | Industrial Ethernet<br>Sometimes also Zigbee / WLAN |
| SSNmain | Local | Standard Ethernet (IEEE 802.3) |
| Inter-SSN | 10's of km | GPRS / UMTS<br>WiMAX<br>Broadband PLC over MV lines |

## C. Metering Data Transmission

The metering data must be transferred from the SMs up to the MW system and the Utility systems via the SSN.

The fundaments for the solution to transmit these metering data is OPENmeter project. Consequently, a WebService based on DLMS/COSEM is the selected data model and upper layer protocol solution for this type of data. Furthermore, DLMS/COSEM is made especially for SMs data transferring, so this decision is also future oriented.

DLMS/COSEM is well suited for the communication to and from the meters and contains a lot of important data models that are adapted to the metering environment.

However, we face the problem that none of the consortium Utilities are using DLMS/COSEM to connect directly to their AMM system to transport the data. This integration issue has been solved defining a Web Service based solution which consists of a Web Services Description Language in combination with an XML Schema.

Below, we include the main reasons behind the choice of using Web Services to exchange the DLMS/COSEM data:

- All the layers of the Web Services protocol stack are standards.
- No other short-medium term standard alternative is foreseen to address this capability. For the long term, full CIM integration is envisioned.

## D. Grid Automation Data Transmission

As already discussed, we need a migration strategy for the grid automation data transport.

To ensure availability of a fist prototype as soon as possible, we concentrate on the functionalities and use cases related to the use of "legacy" standard IEC 60870-5-104 for industrial automation as upper layer communication protocol. This standard doesn't need a predefined data model, but uses ASDU (Application Service Data Units) messages for transporting the data.

The solution of transmitting data with this protocol is widely used in many areas of industry but not specialized for exchanging information for electrical device monitoring and control, and can be used for this purpose. Since IEC 60870-5-104 is established in many devices and there is a lot of experience with this standard, it seems reasonable to use it for the first prototype. An additional advantage of IEC 60870-5-104 standard is its low bandwidth consumption.

IEC 60870-5-104 is broadly disseminated in all parts of the industry, therefore seems realistic to build a first prototype based on this protocol solution. In addition, the protocol is used in today's SCADA systems and no conversion of data models and messages is required to develop a first prototype. Furthermore IEC 60870-5-104 supports real-time and synchronous data transfer.

In addition to the first prototype, it is necessary to develop an "enhanced" prototype as well. Its main objective is to provide enhancements concerning data models and upper layer protocols.

The "enhanced" prototype will be established mainly in lab tests because it provides a test offer for collecting experiences with the relatively new IEC 61850 environment in future SS to the consortium Utilities.

The "enhanced" SSN is based on the IEC 61850. In this context, a data oriented Java API was specified based on an IEC 61850-7-3/-7-4 data model [11], [12], as explained in Section III.

The IEC 61850 data model is transported to the MW system using the flexible, future-oriented Web Service as defined in the standard IEC 61400-25-4 Annex A. This standard is not specifically related to wind turbines only and can be used seamless for all other smart grid devices.

The solution presented in Fig 4 is a long-term vision in the sense that some components have to be developed and not within the scope of the OpenNode project, mainly an IEC 61850 based SCADA system.

For the "enhanced" prototype, the conversion from IEC 61850 to IEC 60870-5-104 data messages for the SCADA is done in the MW.

## VI. Testing

At the beginning of the project, functions have been defined for the system according to operational needs, sorted in Use Cases to regroup scenarios carrying out similar functionalities and detailed by means of class and sequence diagrams. The diagrams describe the information exchanges between the different actors of the system. Once finished the development phase, started the testing process to assess the system's compliance of these Use Cases.

### A. Test process

Assessment criterias have been defined to evaluate both functional validation and performance. The functional criteria, of ranked relevance, identify the important steps in a test for the system to accomplish the desired actions. Time constraints on scenarios make sure that the system answers to the operating needs of a Utility (Fig 5).

Detailed test reports are put together as the tests are performed to be able to retrace precisely the test steps in order to have a clear view of the test process, identify potential malfunction, and provide valuable feedback.

### B. Laboratory tests

The equipment first undergoes laboratory tests before being experimented on the field. The lab tests ensure a safe behavior of the system, as no risk can be taken on the field in regard to the clients. They also allow tests on more advanced functions and the use of future-oriented standards.

These lab tests are to start first with functional tests, during which each actor of a scenario is tested independently by being stimulated thanks to protocol emulators. After this first step validating the individual behavior of each piece of equipment the complete scenarios are to be run, to test the coordination of the different elements and assess the system's operational capabilities.

The lab tests are almost finished now (July 2012). They have been carried out in an EDF laboratory.

### C. Field tests

Once they have been validated through lab testing, the scenarios that are testable on the field are then to be performed in real-life conditions in a SS in Madrid, hence testing the scalability, efficiency and reliability of the system.

The value added of performing field tests can be summarized as follow:

- Real deployment of SMs at client premises.
- The OpenNode system is subjected to real network behavior: noise, couplings, poor signal quality, etc.
- Real location and distances between equipments.
- Two power transformers with two SSNs from different manufactures that work at the same time.
- The number of signals managed is similar to those used in any SS.
- The SS provides a real environment for the tests: underground room, heat, humidity, electrical disturbances, wiring distances, real incidents, etc.

Currently, July 2012, the field test is being done in Madrid at an Iberdrola's operational substation and will be finished by September 2012.

### D. Feedback loop

A feedback loop has been set up for the manufacturers to be able to update their prototypes in regard to the malfunctions revealed by the tests. At the end of the project, accep-

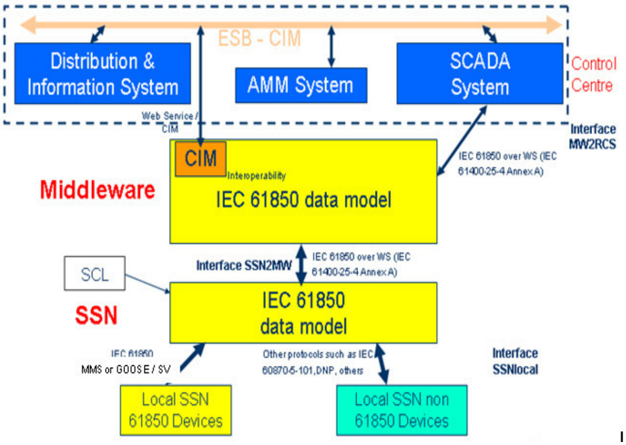

ESB - CIM
Distribution & Information System
AMM System
SCADA System
Control Centre
Web Service / CIM
Interface MW2RCS
CIM
Interoperability
IEC 61850 data model
Middleware
IEC 61850 over WS (IEC 61400-25-4 Annex A)
SCL
Interface SSN2MW
IEC 61850 over WS (IEC 61400-25-4 Annex A)
SSN
IEC 61850 data model
IEC 61850 MMS or GOOSE / SV
Other protocols such as IEC 60870-5-101, DNP, others
Interface SSNlocal
Local SSN 61850 Devices
Local SSN non 61850 Devices

Fig 4: Grid Automation Data Transmission – Long-term Vision

tance tests will be run with the last versions of the prototypes, both in the lab and on the field, providing a final view of the functions that the system can accomplish.

### *E. Conformance tests*

In parallel, conformance tests are run at the moment at KEMA premises, to certify the compliance of the data models and protocols used by the prototypes. The observance of standards ensures the interoperability of the various agents of the system, one of the goals of the OpenNode project.

## VII. Conclusions and next steps

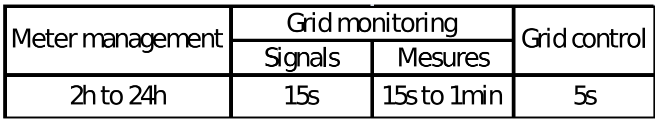
In this paper the OpenNode project has been introduced: its main components, their interfaces, its close-to-market view, the open and distributed architecture foundation and how legacy, current and future requirements can be deployed in the system.

The SSN modularity and extensibility (based on OSGi) concepts have been introduced and how they promote the distribution of intelligence. Consequently, Utilities can take advantage of them to accommodate and increase the level of grid automation introducing new operational services and decentralizing the intelligence according to efficiency criterion.

Moreover, the MW has been explained as an integration enabler, tackling the heterogenous Utililies' system integration issue, that hosts a NoSQL storage for the meter measurements and supports dynamic monitoring algorithms.

The Communications Architecture has been detailed and how it comprises solutions for current (PRIME, DLMS/COSEM, IEC 61870-5-104) and future-oriented protocols (CIM, IEC 61850).

Finally, the OpenNode testing process has been explained as the key task that allows the assessment of the OpenNode system with future-oriented protocols (at a laboratory) and real life operation conditions (at a real Utility network).



| Meter management | Grid monitoring | | Grid control |
|---|---|---|---|
| | Signals | Mesures | |
| 2h to 24h | 15s | 15s to 1min | 5s |

Fig 5: Main operating performance requirements

The SSN has been designed to be an open platform where it's possible to deploy more advanced distributed concepts like intelligent agent systems, advanced grid control algorithms or monitoring functions in a secure way and independently of the SSN manufacturer.

## References

[1] OpenNode, http://www.opennode.eu/

[2] J. Aguiaret al., J. Götz et al., “DER and OpenNode: integration of dg in an open architecture for secondary nodes in the smart grid”, CIRED Workshop proceedings, Lisbon 29-30 May 2012, to be published

[3] R. Soriano, M. Alberto et al., “OpenNode. Open Architecture for Secondary Nodes of the Electricity Smartgrid”, Proceedings CIRED 2011 21st International Conference on Electricity Distribution, CD1.

[4] OpenMeter, http://www.openmeter.com/

[5] OSGi, http://www.osgi.org

[6] NOSQL Databases, http://nosql-database.org/

[7] IEC, 2009. “61970-301 Energy management system application program interface (EMS-API) – Part 301: Common information model (CIM) base”.

[8] OpenNode WP3, D3.1, “Middleware Architecture Requirements and Strategy, September 2010

[9] Event Processing Language - EPL, http://esper.codehaus.org/

[10] OpenNode WP4, D4.1, “Communication Architecture”, September 2010

[11] IEC 61850-7-3 ed2.0, Communication networks and systems for power utility automation - Part 7-3: Basic communication structure - Common data classes, 2010

[12] IEC 61850-7-4 ed2.0, Communication networks and systems for power utility automation - Part 7-4: Basic communication structure - Compatible logical node classes and data object classes, 2010