EPRI | ELECTRIC POWER RESEARCH INSTITUTE

IntelliGrid Architecture Application Guide: Metering and Consumer Systems

1013610

IntelliGrid Architecture Application Guide: Metering and Consumer Systems

1013610

Technical Update, December 2006

EPRI Project Manager
J. Hughes

ELECTRIC POWER RESEARCH INSTITUTE
3420 Hillview Avenue, Palo Alto, California 94304-1395 • PO Box 10412, Palo Alto, California 94303-0813 • USA
800.313.3774 • 650.855.2121 • askepri@epri.com • www.epri.com

DISCLAIMER OF WARRANTIES AND LIMITATION OF LIABILITIES

ORGANIZATION(S) THAT PREPARED THIS DOCUMENT

EnerNex Corporation

This is an EPRI Technical Update report. A Technical Update report is intended as an informal report of continuing research, a meeting, or a topical study. It is not a final EPRI technical report.

NOTE

For further information about EPRI, call the EPRI Customer Assistance Center at 800.313.3774 or e-mail askepri@epri.com.

CITATIONS

This report was prepared by

EnerNex Corporation
170C Market Place Boulevard
Knoxville, Tennessee 37922-2337

Principal Investigator
Jerry Melcher

With contributions from

Grant Gilchrist

Ben Rankin

Darren Highfill

This report describes research sponsored by EPRI.

The report is a corporate document that should be cited in the literature in the following manner:

IntelliGrid Applications Guide EPRI, Palo Alto, CA: 2006. Product ID 1013610

PRODUCT DESCRIPTION

This report is directed to Utilities and energy service companies applying the results of EPRI's Integrated Energy and Communications Systems Architecture Project, now known as the IntelliGrid Architecture.

The Initial IntelliGrid Architecture documents were prepared by a team of experts from several companies and many stakeholder groups. The initial project was titled "The Integrated Energy and Communications Architecture" (IECSA), sponsored by the Electric Power Research Institute (EPRI) in a public private partnership.

The objective of the guidelines presented in this report is to assist power system personnel in using the ideas and concepts presented in the IntelliGrid Architecture documents.

Results & Findings

These guidelines provide techniques that are being applied to the adoption of the IntelliGrid Architecture by utilities. The methods are the result of refinements and extensions by utilities involved in projects both large and small that will make use of emerging technologies that can be integrated through open standards. These guidelines include the systems engineering methods and notation used in emerging utility projects. The findings also identify where additional work is required to assist in the maturation of the overall process of defining, specifying, procuring, installing and managing intelligent systems across the enterprise.

Challenges & Objectives

These guidelines represent a stepping stone in the development of an industry-level architecture for intelligent equipment and systems. The intelligent systems that are possible with emerging technology represent challenges to utilities on several levels. These challenges include mastering a minimal level of the technology related issues in addition to understanding how the technology may influence business and even organizational modes of operation. It should also be noted that the standards used to integrate these systems are still working toward their own levels of maturity to provide a foundation for vendors to build the necessary equipment. Several challenges are ahead including the resolution of key technical issues and details remaining within the key standards and defining key elements necessary for management and security within the systems.

Applications, Values & Use

These guidelines can be used to better understand how to describe and specify systems that make use of the standards and recommended practices that comprise the IntelliGrid Architecture

documents. Using these guidelines will also provide an understanding of the state of the development of the critical portions of the architecture. Using these guidelines can contribute to a utility's own set of requirements development processes for specifying equipment that makes use of the recommendations within the IntelliGrid Architecture

EPRI Perspective

The IntelliGrid Architecture represents on going work to establish an industry level architecture for intelligent equipment that interoperates. To assist the ultimate deployment and cost effectiveness of intelligent systems it is important to develop and understand the infrastructures that will enable the integration of equipment. The development of open systems will assist in competitive procurement of intelligent equipment and assist in the long term ability to cost-effectively build and maintain systems that meet both today's and tomorrows demands for power system operations. These guidelines can serve to provide a better understanding of the tasks involved in reaching these goals as an industry.

Approach

These guidelines build upon the initial systems engineering approaches taken by the original team in developing the IntelliGrid Architecture documents. The guidelines in this report reflect the experience of team members in working with utilities to apply the concepts put forward in the IntelliGrid Architecture.

Keywords

Consumer Gateway, Consumer Portal, Advanced Metering, Consumer Services, Networking, Communications, Standards, Systems Engineering, Communications Protocols, Distributed Computing, Architecture

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

IntelliGrid is an information system engineering approach consisting of a set of processes and tools applied by an electric utility for increasing operational efficiency, improving capital utilization, and enhancing customer satisfaction. This IntelliGrid Application Guide will provide users:

1. IntelliGrid System Architecture Concepts
2. IntelliGrid Methodology Cookbook: Plan, Operate, and Execute
3. Requirements Capture Processes & Methodologies
4. Conceptual Architecture & Design Process
5. Security Best Practices
6. Technology Assessment and Mapping

Appendices

A. IntelliGrid Case Studies and Examples

B. Products, Tools, & Templates

IntelliGrid is a term that can and should imply many things – important things – to anyone in the electric power industry. IntelliGrid embodies a wide range of methodologies, technologies, standards, business models, and other practical yet forward thinking concepts that are necessary to propel the power industry forward into the 21st century. IntelliGrid focuses the full resources of the Electric Power Research Institute (EPRI) onto those new technology elements that will allow all components and business entities in the electric power infrastructure to share information in a seamless and cost effective manner. The result is a solid foundation – the architecture - of a power system that can meet the needs of a changing, and increasing technology oriented society.

1 IntelliGrid System Architecture Concepts

The IntelliGrid System Architecture is a process that utilities should use when designing communications-based information systems targeted at transmission and distribution systems as well as at the customer end user. Figure 1-1 shows the domain space over which IntelliGrid Architectures can be applied.

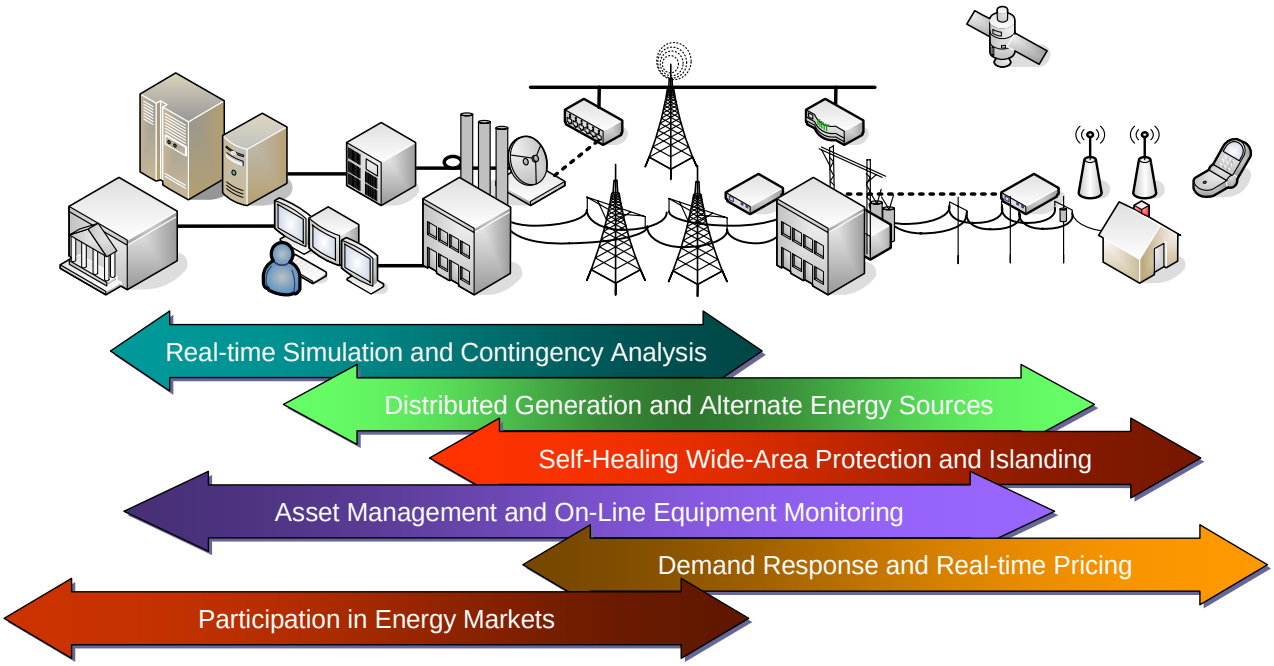

Figure 1-1 IntelliGrid Applications

There are several high-level concepts and principles that are promoted in the IntelliGrid Architecture:

- Integration of systems
- Use of standards-based open systems that interoperate
- User definition of applications and requirements
- Mapping technology solutions to requirements

The IntelliGrid Architecture has developed several tools that can be used when designing communications-related information systems:

- Templates for capturing and defining requirements
- Recommendations of standards and technologies to use
- Strategies for building security into systems
- Strategies for migrating to open, standards-based systems and integrating new open, standards-based systems with existing systems
- Strategies for developing "layered" solutions that minimize the impact of changing technologies in the future

An application of the IntelliGrid Architecture is the use of any or all of the above tools and high-level concepts. Several IntelliGrid Partners initiated projects that implement IntelliGrid Architecture's tools and concepts. These projects include:

- **TXU:** Advanced Metering Infrastructure Development
- **SRP:** Substation LAN Deployment and Equipment Monitoring
- **LIPA:** Utility and Consumer Device SCADA via BPL and Wireless Communications
- **CEC:** Demand Responsive Infrastructure Reference Design
- **SCE:** Advanced Metering Infrastructure System Development and Deployment
- **PPGC-O:** Field Device Communications, Phasor Measurements, and FSM
- **Alliant Energy:** Distribution Monitoring System Replacement

These utility projects were chosen for initial application of IntelliGrid because they share many of the following attributes:

- Clear problem definition
- Implement one or more IntelliGrid principles and/or technologies
- Well defined project timeline
- Buy in from staff across the enterprise
- Plausible value story
- Measurable results during and at end of project

- Ability to clearly transfer the results to other IntelliGrid partners and industry

1.1 IntelliGrid System Architecture Development

IntelliGrid encompasses an inclusive design process approach gathering input from a wide range of stakeholders, both internal and external, to develop effective solutions that meet the defined project scope as shown in Figure 1-2 .

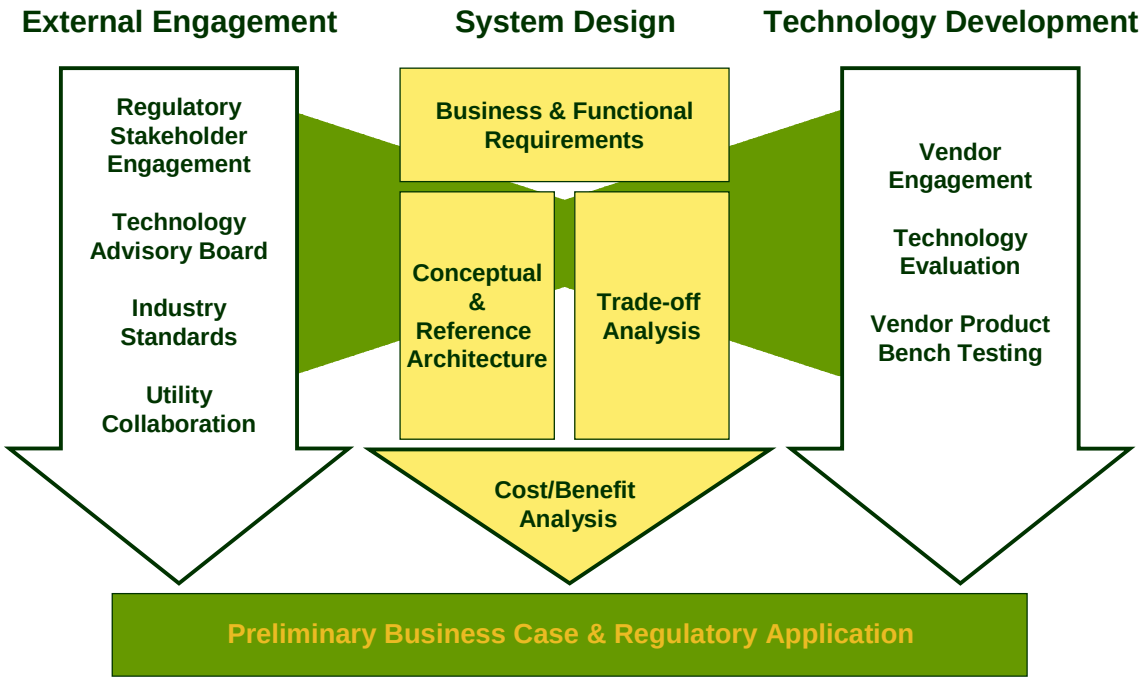

Figure 1-2 Overview of the IntelliGrid Development Process

Broadly IntelliGrid creates a comprehensive framework in which a utility can organize work efforts around in developing new information systems. These are:

System Design

A utility will need to complete a System Design activity in order to effectively communicate "**Why** are we investing in the project?", "**What** functions and capabilities does the system have to perform?", "**Who** is involved in the information system, and **Who** will be the users of the information?", "**Where** are the data sources for the required functions?", "**How** are all the organizational and technology components integrated into a logical architecture?", **Which** systems, subsystems and components do we need in order to meet our utility's

requirements?", and with "**What** standards, technologies, and vendor solutions will our utility use to accomplish the project goals?" With these questions in mind the utility can start the process of capturing the **Business & Functional Requirements**, create a **Conceptual & Reference Architecture**, and perform the **Trade-off Analysis** and **Cost/Benefit Analysis**.

External Engagement

There are many parties who interact with a utility and have an impact on the utility's operations and provide input to its regulation. These stakeholders include **Regulatory personnel, Technology Advisory Board, Industry Standards,** and **Utility Collaboration and user groups.** These external stakeholders are also sources of requirements that need to be taken into account in the development of systems.

Technology Development

The IntelliGrid process requires development of the specifications and technology that enables interoperable equipment from different vendors. Projects deploying systems now will do extensive **Vendor Engagement** in order to assess current market capabilities and ability to meet minimum levels of interoperability. This process also requires in-depth **Technology Evaluation,** along with **Vendor Product Bench Testing** against requirements.

Preliminary Business Case & Regulatory Application

In order for management and regulators to support and approve of any implementation of an information system, a strong business case must be assembled that defines the benefits of putting a new system into place and assessing this against total system costs, both initial and on-going.

1.2 System Design

The IntelliGrid Systems Engineering approach is an iterative process of analyzing a complex problem at successive layers of detail starting with a high-level functional vision, and working down through lower layers of functional decomposition. The systems architects perform this identification of functional requirements in parallel with the identification of technical approaches and mechanisms that support both the functional vision and the utility's enterprise standards.

This process results in a series of engineering decisions which yield a platform-independent, preferred solution. The utility's preferred solution, represented as a Conceptual Architecture and Design, serves as input into vendor evaluations, business case estimates, integration and testing plans, and provides a means to accelerate

platform-specific engineering and deployment tasks once a specific set of technologies has been selected.

The challenge in communicating the conceptual architecture for the information system lies in the complex collection of devices, data, networks, computer systems, protocols, organizational processes and people necessary to provide the operational benefits.

1.2.1 Business and Functional Requirements Gathering

At the start of the IntelliGrid project, cross functional teams need to be assembled to assess what the business and functional requirements are the target of the information system under investigation. Over the last 10 years, a methodology called Use Cases has evolved, which assists these teams. In order to develop requirements based on Use Cases, the utility must organize a process consisting of series of workshops using cross-functional teams. Use Cases place particular emphasis on how the information system will actually be used when deployed rather than being constrained by the design of existing products. The utility's intent is to clearly define the desired requirements, leaving vendors as free as possible to come up with innovative solutions.

1.2.2 Conceptual & Reference Architecture

The Conceptual Architecture identifies what high-level systems, subsystems and components does the utility need in order to meet the functional and non-functional requirements. The systems, subsystems and components are characterized in a platform independent way so that various technologies may be evaluated against the same standard set of requirements for a particular system, subsystem, or component. The answer to the conceptual architecture questions provides the system design team with an understanding of the technical capabilities necessary for various system elements to meet the requirements. The challenge in communicating the Conceptual Architecture lies in the complex collection of devices, data, networks, computer systems, protocols, organizational processes and people necessary to enable the uses of the technology that unlock key operational benefits.

The Reference Architecture determines how the system components are integrated into a logical architecture and what services must be provided to create a viable end-to-end solution to fulfill the utility's goals.

- **Performance, business continuity, high availability, maintainability, analysis and design** – Systems Engineering activities should ensure service levels and performance requirements are well documented and understood. Engineering team members should work to ensure the solution architecture meets the performance requirements

- **Security analysis and design** – Systems Engineering leads to a better understanding of the sensitivity of the data handled by the solution. This understanding will allow the engineering team to work with IT Security to analyze and design an appropriate security solution
- **Operational design and planning** – System Engineering should ensure monitoring, administration tools and guidelines are robust enough to support post implementation operations and solution maintainability. Additionally, the engineers should assist in the development of the implementation plan with a focus on mitigating risk of disruption to either the business process or technical environment.
- **Solution architecture and design** – The System Design Team should work together to develop individual application, information, security, integration, and technical architectures and ensure applicable/available enterprise services are included in the design. These architectures should be packaged into an overall solution architecture document.
- **Application and data integration** – engineers should work with the project team to identify integration requirements and develop solutions that include enterprise standard technologies and services where appropriate. Additionally, engineers should work with other major program Architects to identify opportunities for data integration implementation of formal data management.
- **Enterprise architecture alignment** – The system engineering effort should focus on providing early identification of applicable standards, potential deviations and opportunities to leverage existing products and enterprise services. Additionally, utility engineers should evaluate each enterprise technical service that the utility has already created to determine the appropriate use of that service within the project's solution. Maximizing the utility's return on investment in creating enterprise technical services by reusing and leveraging these services where appropriate and to the fullest extent possible is a primary goal of every SCE System Engineering effort.
- **Schedule, resource planning, Work-Breakdown Structure (WBS) and cost estimation support** –Engineers work with Project Management to determine the appropriate WBS, schedule and cost based on the technical nature of the project and complexity of the conceptual solution architecture.

1.2.3 Trade-off Analysis

With what standards, technologies and vendor solutions will should the utility select to accomplish their goals?" The outcome of this process is the selection of

specific vendor solutions necessary to meet the requirements required to achieve a positive operational benefit. This is the lowest level of design abstraction prior to commencing detailed design and integration activities. Equally important is the ability to balance cost, schedule and technical constraints with a thorough understanding of the maturity level of technologies available in the market place.

1.2.4 Cost/Benefit Analysis

As part of the review of the reference architecture, a refined answer to the question "How much value (costs vs. benefit) does each alternative architecture and component yield and what are the constraints and boundaries?" Utilities must constantly remind themselves that each new component, subsystem, or system brings a long term burden, both initial cost and ongoing O&M cost. For the Reference Architecture, only those items that support the business case in a cost effective manner should be included in the final design.

1.3 External Engagement

In order for a project to be successful project, it must engage in broad consensus building process. This external engagement process reaches out to many parties who are directly involved plus those who have oversight of the utility at management levels as well as those from the regulatory side.

1.3.1 Stakeholder Engagement Process

A list of stakeholders needs to be assembled early in the project. There will be multiple stakeholders related to all of the system, subsystems, and components. Each level will have stakeholders with varying interest. Some of these stakeholders will have interests that are highly technical, some financial, others focused on business management practices.

1.3.2 Technology Advisory Board

Due to the complex nature of advanced information systems, utilities should form an advisory board to keep track of industry developments and how newer technologies can impact utility operations.

1.3.3 Industry Standards

Many standards have been developed to support the IntelliGrid concept of open architecture. Utilities are encouraged to develop an understanding of the key standards and their function in developing a reference architecture.

1.3.4 Utility Collaboration

Many utilities share ideas at open industry forums hosted by industry organizations such IEEE, EEI, APPA, NRECA, UTC, and other more specific conferences focusing on transmission, distribution, and customer automation. In addition, key user groups such as the Association of Edison Illuminating Companies (AEIC) metering services group, and recently formed Open AMI and Utility AMI under UCA International Users Group, provide open forums for resolution of technical issues and reaching agreement of applications of key standards. These meetings are useful for the introduction to common issues, exchange of lessons learned.

1.4 Technology Development

The Reference Architecture will identify with what standards, technologies, and vendor solutions will the utility select to accomplish their system goals.

1.4.1 Vendor Engagement

The utility should engage the vendor community early in the project. Surveys of technology offerings should be compiled from Request for Information bids. Vendor presentations and factory visits are encouraged.

1.4.2 Technology Evaluation

Once the Conceptual Architecture is complete, the utility should perform technology evaluation as an integral part of developing the Reference Architecture. The utility system engineering team can identity capabilities that meet functional and non-functional requirements and areas where the technology is currently not available or needs further development.

1.4.3 Vendor Product Bench Testing

Utility should consider developing or working with an organization that does system integration testing to evaluate vendor products in a lab setting before committing to large scale deployments. Components can be interchanged and tested against performance requirements.

1.5 Business Case Development and Regulatory Application

Any new system requires justification using business case development. Each subsystem and component should support the overall benefits for the total system in the business case. Most systems will have multiple applications that utilize the same

platform. Any one application may not be justified, but the sum of the applications in total can result in benefits that exceed the base system cost. Since most utilities operate in a regulated environment, business cases must be scrutinized by appropriate regulatory agencies for validation.

2 IntelliGrid Methodology Cookbook

This is the process recommended by the IntelliGrid Consortium for performing project planning, requirements definition, architecture development, technology selection and deployment. The steps described in this "cookbook" are illustrated in Figure 2-1 and Figure 2-2. Figure 2-1 illustrates the initial requirements definition and systems architecture development processes, while Figure 2-2 illustrates the later steps of business case analysis, technology selection, and deployment. This document is intended to be an overview, for the purposes of understanding the general concepts of the process, and a checklist for monitoring the progress of a project.

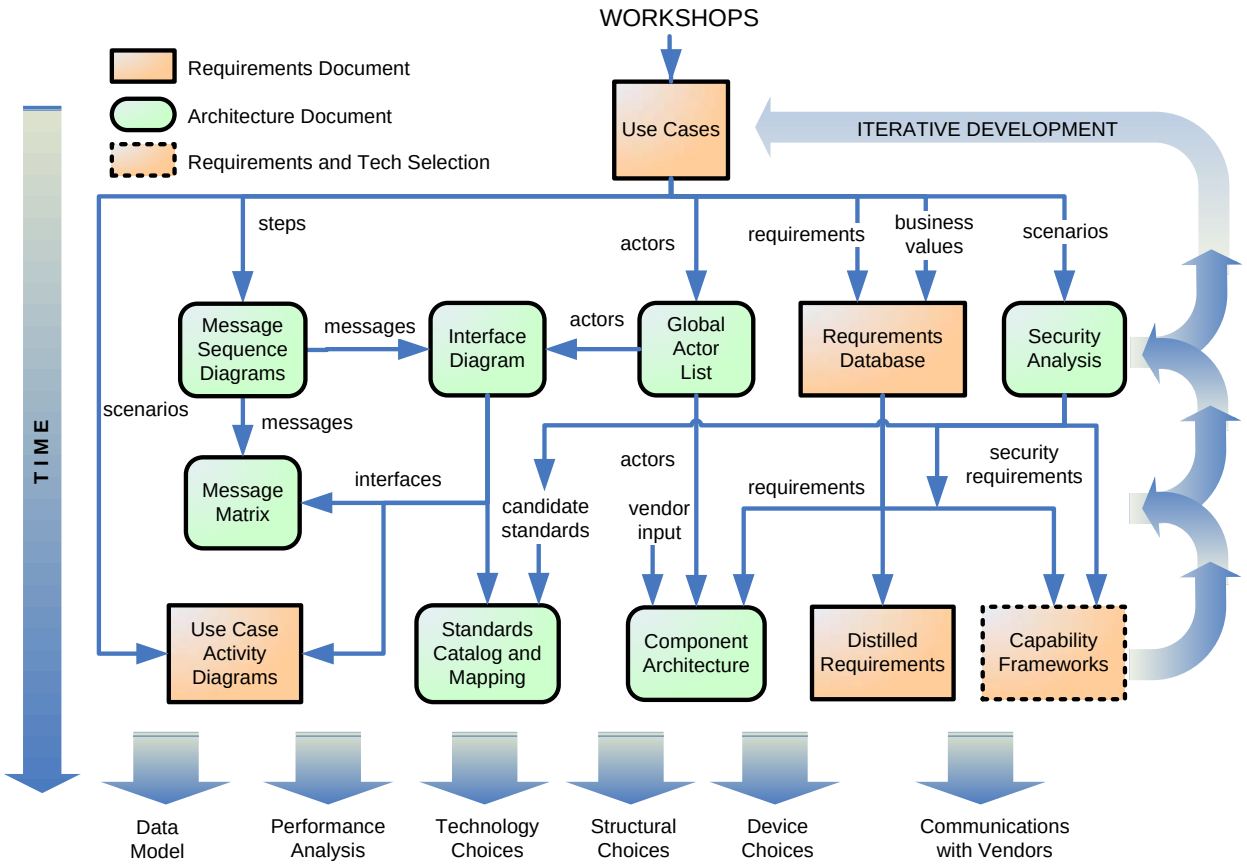

Figure 2-1 Requirements and Systems Architecture Process

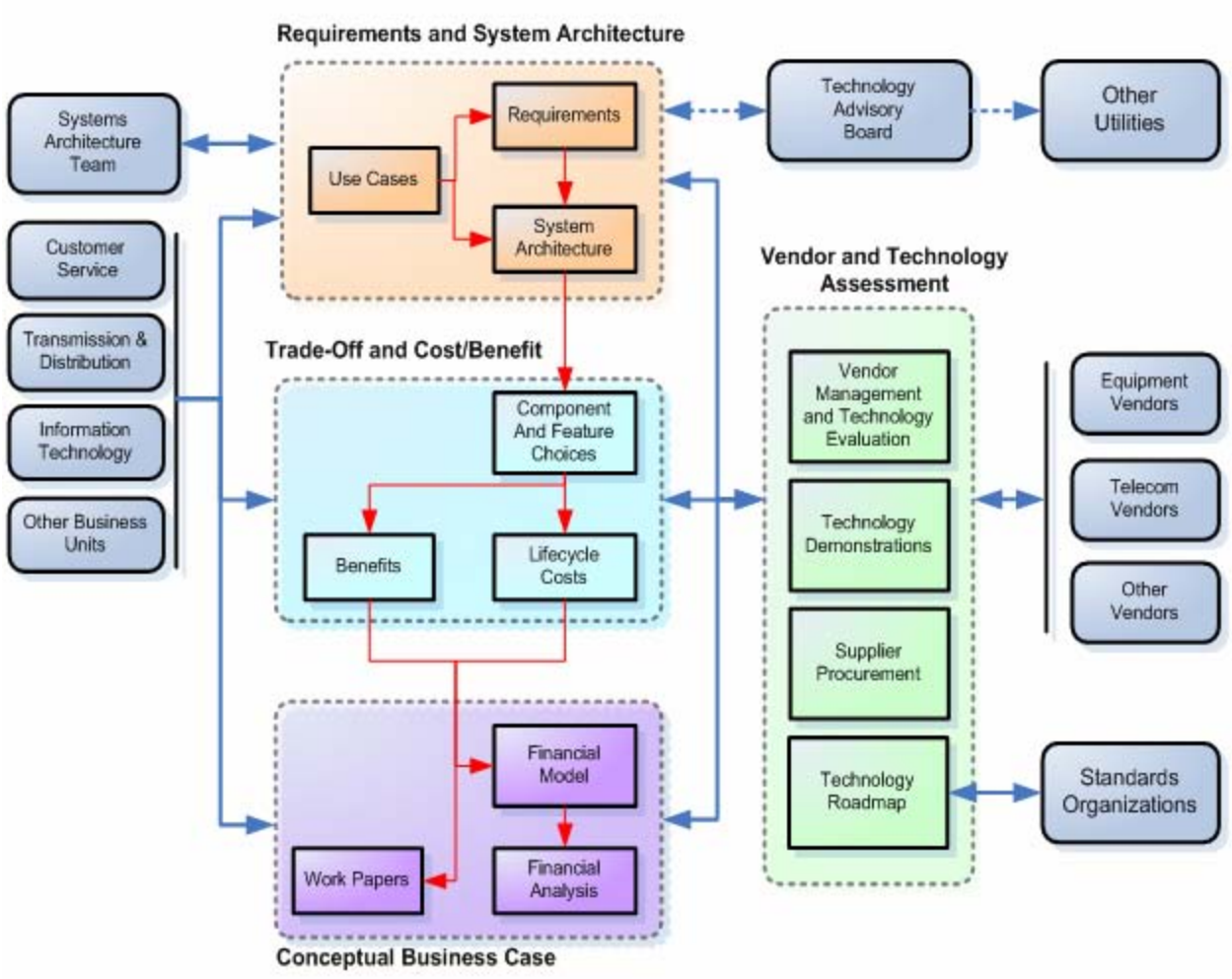

Figure 2-2 Technology Selection, Business Case, and Deployment Process

2.1 Plan Projects

This section describes the initial planning phases of the project, to be performed by upper management.

2.1.1 Determine Business and Regulatory Drivers

It is important that all subsequent steps be driven by the business needs of the organization. Before beginning a utility automation project, it is important to be clear what financial problems or regulatory compliance issues are being addressed. Some examples include:

- Requirements for information sharing with other organizations
- Requirements for improving energy efficiency or reliability
- Potential new service offerings or revenue streams

- Requirements for improving energy efficiency or reliability
- Potential new service offerings or revenue streams
- Reduction of costs by automating processes that were previously manual

2.1.2 Choose Focus Areas

IntelliGrid projects generally focus on one of the following areas of utility automation:

- Consumer Participation
- Network Optimization
- Wide-Area Reliability
- Real-Time Simulation
- Energy Markets

Choose one or more of these focus areas to be the subject of your automation project.

2.1.3 Choose Projects

Within the chosen focus areas, select particular projects to be implemented; for instance, implementing a demand response program or advanced metering system would fall within the category of Consumer Participation.

2.1.4 Identify Candidate Technologies

Determine which international and national standards, industry agreements, best practices, and de facto standards may apply in this environment. The list of applicable technologies on the IntelliGrid Architecture web site may serve as a starting point.

2.1.5 Define a High-Level Business Case

Based on your business and regulatory drivers, determine the high-level benefits and costs you expect to find as a result of implementing the project. This is the first decision point where your organization must determine whether it is feasible to proceed with the project.

2.1.6 Refine Process for Your Organization

Examine the remainder of this process and determine which steps are applicable to your organization and project within your organization. As illustrated in Figure 2-3

the project should follow three concurrent streams of effort: external engagement, system design, and technology development.

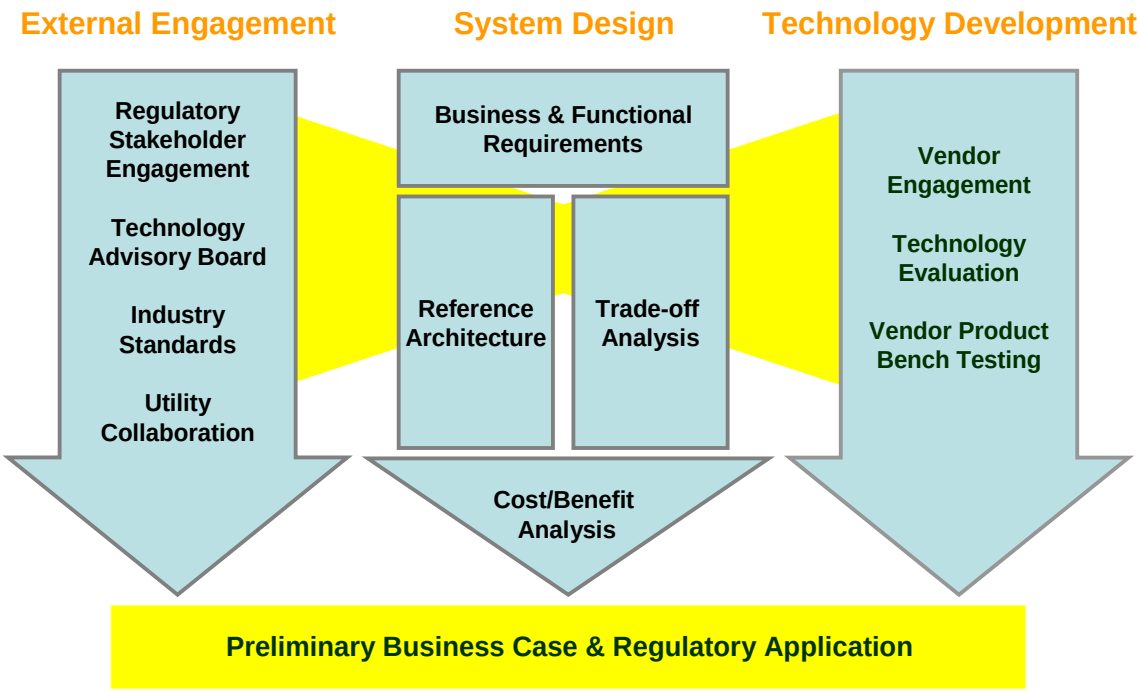

Figure 2-3 Project Streams

2.2 Define Requirements

This section describes the steps to ensure that the remainder of the process is driven by a complete and comprehensive set of requirements, established by those within the organization having the most knowledge on the subject. This part of the process is illustrated in the "Requirements Documents" portions of Figure 2-1 and the "Use Cases" and "Requirements" rectangles in Figure 2-2.

2.2.1 Identify Stakeholders

Determine which parts of your organization, and which other organizations, are affected by the proposed project. Pay particular attention to traditional "silos" within your organization and ensure that even groups that would not traditionally be involved in an information technology project are considered. Figure 2-4 illustrates some groups that should be considered, using advanced metering as an example.

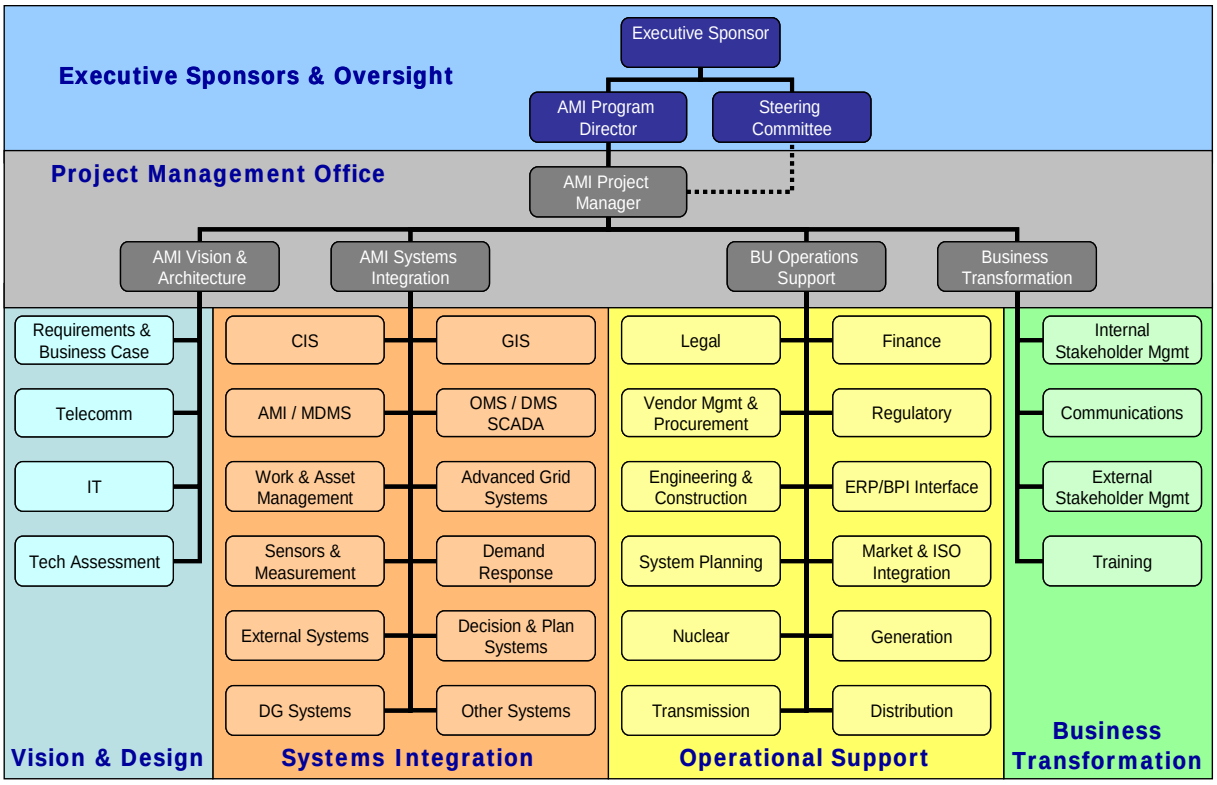

Figure 2-4 Potential Stakeholders and Requirements Team Structure

2.2.2 *Select Teams*

As suggested by Figure 2-4, develop cross-functional teams for performing the requirements definition process. These should include both the "go-to" people with technical knowledge, and their managers who can commit to whether a particular idea is a true requirement for the organization, or just a "would be nice" feature.

Key to the team selection is the choice of a "systems architecture" or "vision and design" team, whose responsibilities will be to lead the process, provide guidance, develop and document the business cases and the architecture from the information that the requirements teams will produce.

The roles of the systems architecture team and the cross-functional requirements teams are illustrated on the left side of Figure 2-4.

2.2.3 *Choose Top-Level Use Cases*

Based on the high-level business case, the architecture team chooses the organizational goals and therefore the top-level "use cases" of the requirements process.

A ***use case*** is simply a "story" that includes various "actors", and the "path" they take to achieve a particular functional goal. By considering the actions of the actors working to achieve this functional goal, a completed use case results in the documentation of several scenarios, each containing a sequence of steps that trace an end-to-end path. These sequential steps describe the functions that the proposed systems and processes must provide, directly leading to the requirements for the given use case.

Each use case has a descriptive name indicating the individual or group that has the primary stake in completing the use case, e.g. "Operator locates, isolates, and restores power after fault."

There should be no more than about twenty high-level use cases, and many projects may require less than half a dozen. For larger numbers of use cases, it is worthwhile to categorize them by affected area, e.g. customer service, operations, maintenance, etc.

2.2.4 Hold Workshops

Schedule a series of workshops to brainstorm the use cases. These are the "workshops" identified at the top of Figure 2-1. The output of the workshops will be documents capturing at least the following information:

- The **goal** of the use case, which is usually its name. e.g. "Utility remotely connects or disconnects customer".
- The **narrative.** A short English text version of the story.
- The **actors**. An actor is anything in the system that communicates. It may be a person, a device, a piece of software, an organization, or anything else you can think of that acts on its own and can have goals and responsibilities. e.g. a "customer" or a "meter".
- The **assumptions** that the use case is based on. These can constitute requirements in and of themselves.
- The **contracts** and **preconditions** that exist between the actors, e.g. "The customer agrees with the utility to limit demand on selected days in exchange for a lower tariff."
- The **triggering event** that led to the scenario taking place.
- The **steps**. A numbered list of events that tell the story in detail. Each step identifies an actor, what the actor is doing, what information is being passed, and identifies to whom the information is sent. e.g. "7. The operator sends a curtailment command to the meter".

A recommended process for holding the workshops is illustrated in Figure 2-5.

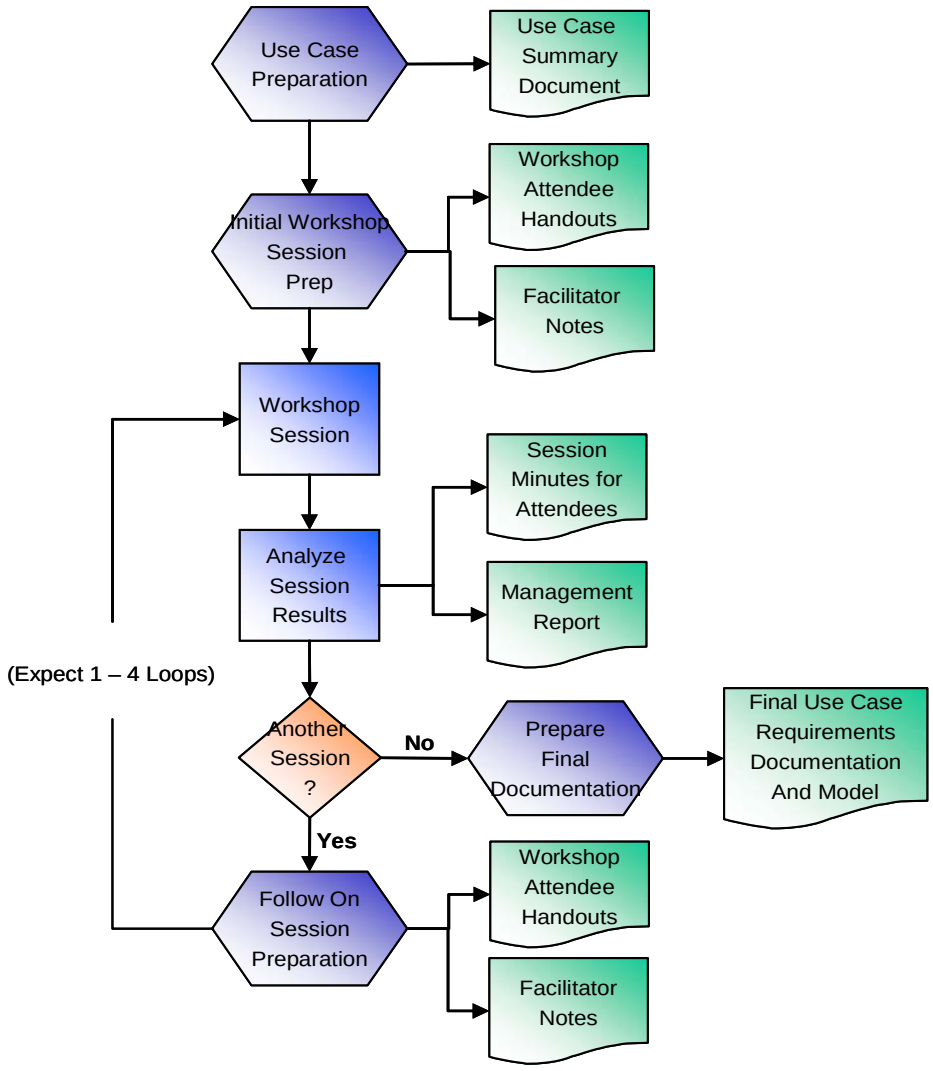

Figure 2-5 Workshop Process

2.2.5 *Identify Requirements and Business Value*

Each requirements team executes the following procedure to define requirements while developing use cases:

1. **Decide on the scope** for this use case. The system will do many different things, and only certain functions will be addressed by any given use case. This will help you **write the narrative**, which may be done now or later in the use case.

2. **List the actors** and what goals they want to accomplish. Select one actor as the primary actor, whose goal will determine when the use case is done.
3. **Identify all stakeholders** and their interests. All actors are stakeholders, but there may be stakeholders who are not actors. The interests of all stakeholders must be satisfied for the use case to be complete.
4. **Identify contracts and preconditions.** What has to happen before the use case can start? Knowing this will help to identify requirements.
5. **Select the main success scenario**, in which the primary actor successfully achieves its goal. There may be a number of other scenarios in which it fails to do so.
6. **Write the steps**. As each step is written, check for the following **requirements** and write them down:
    1. What does this mean the system will have to DO?
    2. How quickly, reliably, safely, compatibly, usably, securely, must it do it?
    3. How might our business process change because of it?
7. **Write the alternate scenarios**. Any place that something can go wrong will be an alternate set of steps. Some of them may halt the main success scenario; others may rejoin it later in the story, i.e. the problem could be fixed.
8. **Check if we're done.** Check to see if all the stakeholders were satisfied, and in particular, if the primary actor reached its goal.

As each requirement is identified, record it in formal requirements language, i.e. subject-verb-object-qualifier. There are "good" requirements and "bad" requirements. A good requirement clearly identifies which part of the organization is responsible for the requirement, and what it must do. It should identify "what", not "how".

Also as each requirement is identified, ***qualitatively*** identify the business value of each requirement. For instance "Will reduce cost by eliminating the following manual steps…" These qualitative business value statements will be used later in the business case analysis.

The scenarios, actors, steps, requirements, and business value statements are the outputs from the use case process, as shown at the top of Figure 2-1. A useful mechanism for capturing use cases is Unified Modeling Language (UML) Activity Diagrams. An example of such a diagram is found in Figure 2-6.

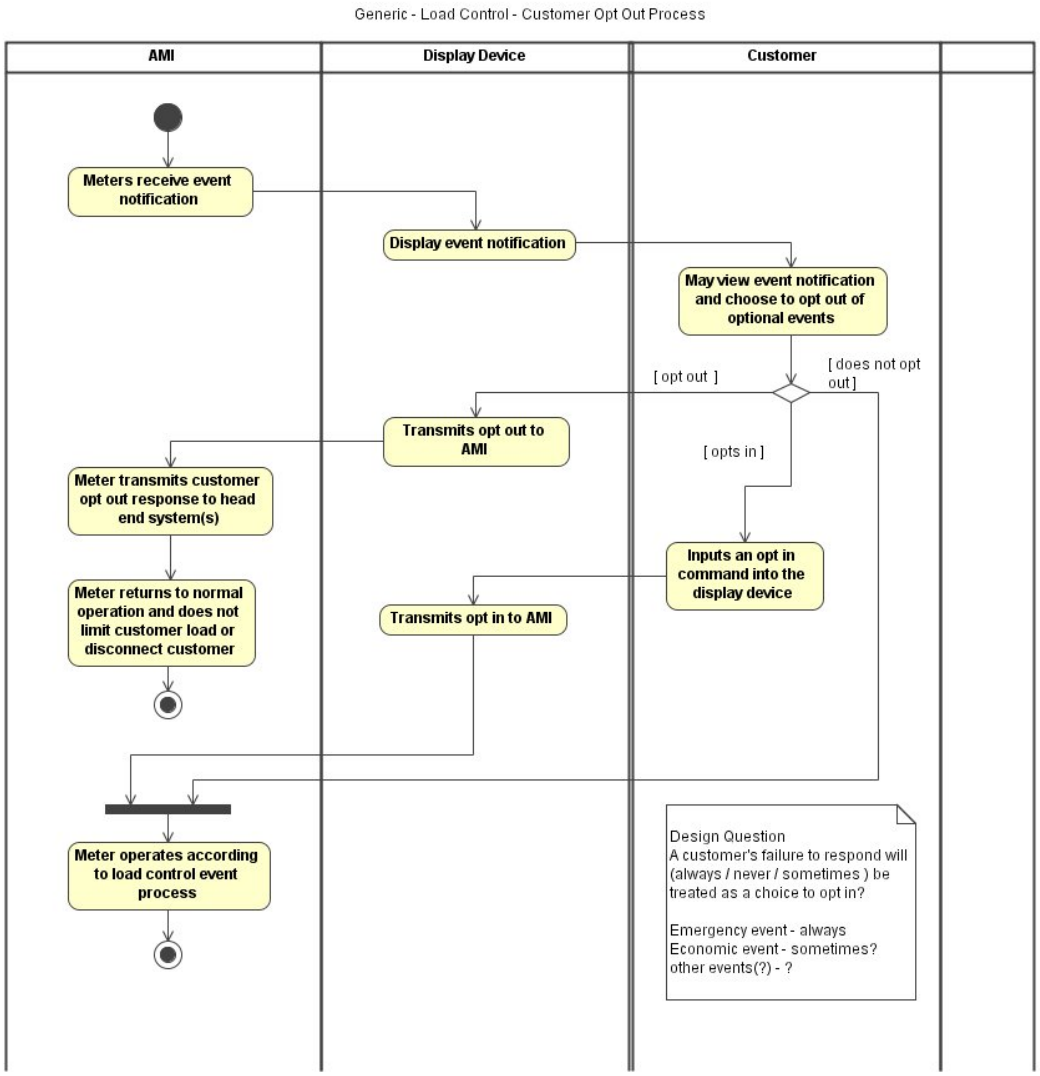

Figure 2-6 Example of an Activity Diagram

2.2.6 Identify Security Risks

Throughout the workshop process, record any potential security risks of each use case. In general, a security analysis will involve the following steps:

1. Identify what assets need to be protected.
2. Identify the threats the assets need to be protected against.

3. Identify any vulnerabilities to these threats found in the existing system.
4. Identify various security measures to protect the assets.
5. Determine which measures are best suited to protecting which assets

This stage begins the security analysis process by identifying assets, threats and vulnerabilities and recording them along with the requirements.

2.2.7 Distill Requirements

The workshop process will identify many requirements, perhaps several hundred. It will be necessary to eliminate duplicates, make wording consistent, and verify business value statements.

2.2.8 Evaluate Requirements vs. Business Case

For each requirement and qualitative business value identified as part of the workshop process, the business case teams begin to assign quantitative benefits and complete lifecycle costs, as shown by the "Trade-Off and Cost Benefit" rectangle in Figure 2-2. This work will further distill requirements and eliminate some entire use cases and scenarios from consideration.

It is recommended that even if the organization chooses at this stage to not initially deploy certain requirements or scenarios, the systems architecture team should continue to consider them. It may be that they will be deployed in some later project, and the architecture should be flexible enough to permit this.

2.2.9 Publish Requirements

For the benefit of the industry and other utilities, as well as vendors who may become involved in the project, you should publish your distilled list of requirements at this stage. As shown in the upper right of Figure 2-2, it may be useful to have your requirements reviewed by an "advisory board" consisting of industry representatives, especially utilities, not immediately involved in the project.

2.3 Design an Architecture

This section describes how to define a systems architecture based on the requirements gathered through the use case process. The deliverables described in this section are identified as "Architecture Documents" in Figure 2-1.

2.3.1 Resolve List of Actors

The list of actors recorded in the use cases typically contains many duplicates and many different names for the same actor. The systems architecture team examines the list, eliminates duplicates, agrees on common names, and by eliminating some actors, and establishes the scope of the architecture being developed. The resulting Global Actor list will be used as a common language for the rest of the architecture development.

2.3.2 Identify Messages Exchanged

For each step identified in the use cases, the systems architecture team identifies what data must be exchanged to complete the step, and estimates how this data might be grouped into messages. The actual type and number of messages exchanged will not be known until after technology selection, but the messages identified here will be useful to determine which technologies are suitable.

As illustrated in Figure 2-1, two useful tools at this stage are Unified Modeling Language (UML) Message Sequence Diagrams, and a Message Matrix. A Message Matrix consists of a spreadsheet or database identifying the relationship between use cases, scenarios, steps, messages, and the source and destination actors of the messages. It could also track which scenarios and/or steps are currently considered to be part of the business case.

Message Sequence Diagrams are reduced forms of Activity diagrams that simply show the messages between actors in a given scenario.

2.3.3 Define Interfaces

From the actors identified in the Global Actor List and the messages from the Message Sequence Diagrams, the systems architecture team can group messages together into interfaces and identify the interfaces between the actors.

There are several useful tools for identifying interfaces. One could use a UML Collaboration Diagram or Activity Diagram, a Yourdon data flow diagram, or simply add another column to the Message Matrix, as illustrated in Figure 2-1.

The set of actors and interfaces is sometimes known as the **conceptual architecture** of the system. It is an idea of how the system should work, without considering actual implementations yet.

2.3.4 Define Security Domains

Once the interfaces between actors have been identified, these interfaces and actors can be grouped into security domains having the same potential threats and technological solutions. Security threats and solutions generally fall into one of four categories: availability, integrity, confidentiality, and non-repudiation.

A specialized security team may be needed in order to define security domains. These domains may fall along organizational, geographic, or technological boundaries.

2.3.5 Define Security and Network Management Policies

The organization must decide on a common set of security and network management policies. These policies must define the overall objectives of security measures, the roles of organizations and individuals, and procedures for managing assets and credentials. These policies must also identify how procedures may be modified within each domain while maintaining the overall security objectives.

As illustrated in Figure 2-1, the domains and policies identified in the Security Analysis are included as further requirements to be considered in vendor consultation and in later steps of architecture development.

2.3.6 Break Down Actors into Components

The actors that were defined in the Global Actor List and used to identify the system interfaces will not be the actual physical components of the system. The functions of some actors may be distributed among multiple devices, or a single physical device or computer system may implement several different conceptual actors.

At this stage of the architecture development, consult with vendors regarding the previously published requirements to determine how different suppliers would implement the conceptual architecture. This process may identify some missing actors or interfaces, requiring a change in the architecture.

2.3.7 Assess Candidate Technologies

Examine the candidate technologies identified in the planning stage and any new technologies identified during vendor consultation. Perform a technology assessment including at least the following factors:

- Level of Standardization - Who recognizes it as a standard?
- Level of Openness – How easy/costly is it to obtain and use?
- Level of Adoption – How widely used is it now? In the future?
- Users' Group Support – Does someone promote it? Improve it? Test it?
- Security – Can it be secured? Is it inherently secure?
- Manageability – Can you control, monitor and/or upgrade it remotely?
- Scalability – Will it work when deployed at a large number of sites?
- Object Modeling – Does it group and structure data?
- Self-Description – Can it automatically configure and initialize itself?
- Applicability to the Power Industry – was it intended for use here?
- Applicability to this particular problem domain

Develop a Standards Catalog that captures this analysis, as illustrated in Figure 2-6

Naturally, cost is a factor in assessing the candidate technologies also and must be considered as part of the ongoing business case development.

2.3.8 Map Candidate Technologies to Interfaces

Consider which of the candidate technologies should be used on each interface, and within each security domain. Add this information to the Standards Catalog, or create a separate Standards Mapping document.

It is not necessary at this stage to select a single technology for each interface, any more than one should favor a particular vendors' architecture yet. The architecture at this stage should represent a superset of the best practices in the industry that can meet the requirements of your organization.

2.3.9 Define Integration Interfaces

A key factor at this stage is your organization's existing information technology architecture. You must develop a plan to integrate these existing interfaces to

those of the new project. There may also be other similar projects underway that must be integrated with the current project.

Ideally, the groundwork for this step has already been addressed by ensuring representatives from other parts of the organization was included on the requirements teams and has contributed to the development of the architecture.

2.3.10 Test Architecture against Use Cases

The output from vendor consultation and the technology assessment efforts should result in a ***platform-independent architecture*** that can represent most vendors' products and services. Test this platform-independent architecture against the original use cases, and verifies that all the scenarios to be included in the business case can be implemented using these components and interfaces.

2.4 Select Technologies

This section describes how to select particular technologies to implement the platform-independent architecture that you have developed, and how to evaluate vendor submissions. As illustrated in Figure 2-7, the technology selection process actually begins in the earlier phases with the development of requirements and the initial technology assessment.

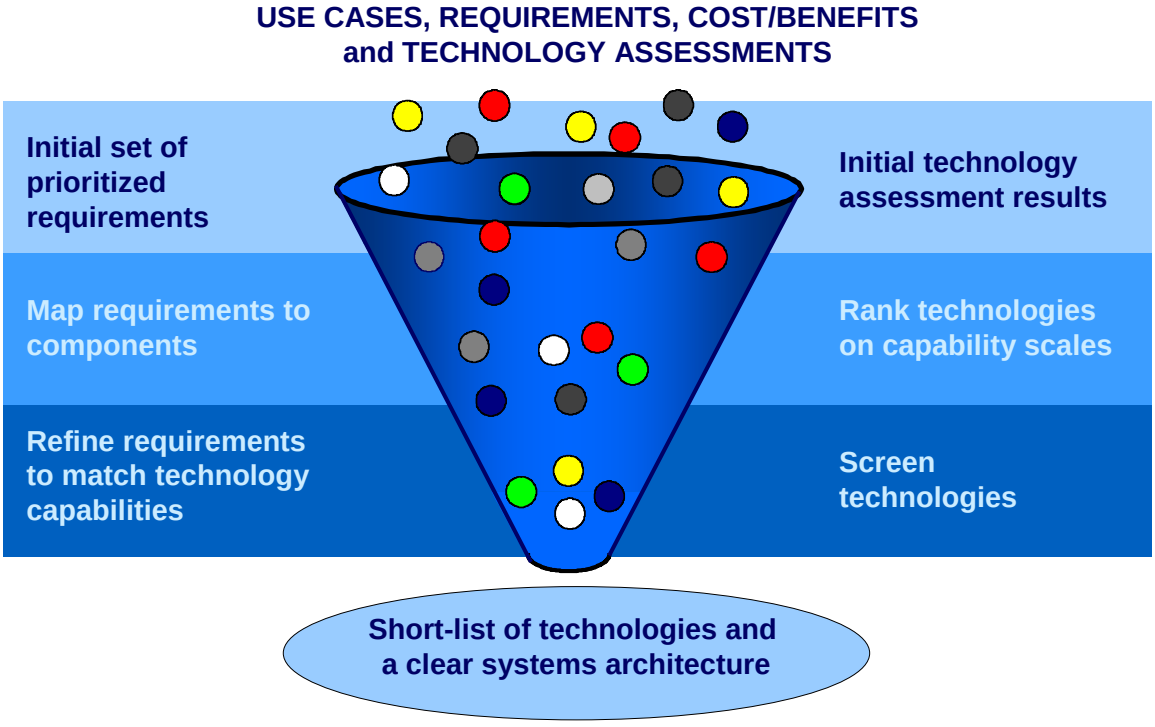

Figure 2-7 Overview of the Technology Selection Process

2.4.1 *Build Technology Capability Scales*

Create a framework of Technology Capability Measurement (TCM) scales[1] from the Distilled Requirements and Security Requirements as illustrated in Figure 2-1. TCM scales are a tool for evaluating any given technology, as one might expect from the name. Perhaps surprisingly, they can and should also be used to evaluate your set of ***requirements***

A framework of TCM scales is essentially a set of ratings from 0 to 5 in a number of categories that are derived by the system architecture team from the use cases and distilled requirements. For communications technologies, suggested categories might be: interoperability, reliability, scalability, security, throughput, and/or latency. For device technologies, some categories might include configurability, programmability, serviceability, power requirements, security, memory, or display. Cost is not considered on the TCM scales; it is a separate item dealt with as part of the trade-off and business case development process.

A rating of 0 represents an unacceptable solution, 1 the minimum solution, and 5 the best possible solution. An example scale is shown in Figure 2-8. The chart illustrates which features are required to receive a particular maturity level rating.

Latching Relay Device (Disconnect Device)

Maturity Level	Customer Reset Options	Commercially available & In use now (>10,000 units)	Current limiting with Flexible Set Point Handling	Current Limiting capabilities	On/Off disconnect	Voltage sensing	200 Amp Rating	Integrated device	Collared Solution
5	X	X	X	X	X	X	X	X	
4		X	X	X	X	X	X	X	
3			X	X	X	X	X	X	
2				X	X	X	X	X	
1					X	X	X	X	
0									X

Figure 2-8 Example of a Technology Capability Measurement Scale

[1] Based on CFTP developed by J. Paap, MIT

2.4.2 *Request Proposals*

At this stage, you have a complete set of requirements, a defined minimum set of components and interfaces, a preferred catalog of technology standards and best practices and a set of scales for performing evaluations. Use this information to request proposals from vendors. As illustrated in Figure 2-2, vendors should be engaged throughout all parts of the design, to give a "reality check" on what is possible with existing technology. When requesting proposals, publish the TCM scales so vendors will know how they will be evaluated.

2.4.3 *Evaluate Requirements and Proposals*

Use the TCM scales to evaluate not only vendors' proposals, but also the distilled list of requirements. The upper end of the TCM scales should represent the best system that could be envisioned, while the actual requirements of the organization, especially in the short term, may be at a much lower level.

2.4.4 *Perform Gap Analysis*

Use the TCM scales to determine the difference between your organization's requirements and the technology offered by the vendors. Since vendor technologies were assessed as part of developing the standards catalog, there should theoretically not be any new or unknown technologies discovered at this point.

The Message Matrix becomes very useful at this stage for assessing communications technologies. If the technology contains messages and constructs that can carry all of the data and transactions identified in the Message Matrix, it is likely to be suitable for the project.

2.4.5 *Trade-Off Requirements*

Based on the gap analysis, select vendors and technologies that are the closest to meeting the requirements. Refine the precision of the cost and benefit estimation process now that the equipment and services to be used is known.

Based on the revised cost and benefit information, determine appropriate thresholds for the cost of each component, and trade off functionality against cost. Revisit the set of requirements and determine which cannot be met for this phase of the project. Document the finalized ***platform-specific architecture***.

2.4.6 *Identify Missing Standards and Technologies*

It is quite possible that some of the requirements of the organization cannot be met by any of the existing solutions at this time, regardless of expenditure. Record each of this missing standards, technologies, or capabilities for inclusion in a Technology Roadmap.

2.4.7 *Create Technology Roadmap*

When technologies or capabilities are missing the organization generally has the following options:

- Offer to work in partnership with a vendor to develop the technology
- Create a work-around solution in-house
- Try to develop the technology in-house
- Relax or refine existing requirements and define interfaces for future expansion when the technology becomes available.

The option chosen for each component or interface will vary depending on benefits, costs, schedule and the business drivers behind the project.

In addition, there may be a number of external factors affecting the availability of the technology, such as:

- Creation of legislation or regulations to encourage development of the technology.
- Creation of an industry organization or forum.
- Completion of an agreed-upon national or international standard.
- Adoption of a technology by key players in the industry.
- Completion of a pilot project by another set of organizations.

Capture all of these events and choices in a Technology Roadmap, indicating your organization's plan for the future. It should clearly indicate dependencies on external factors, actions the organization could take to influence technology development, and the impact on the current project.

2.4.8 *Submit Proposals to Standards Bodies and Industry Groups*

One section of the Technology Roadmap should be a set of recommendations to standards organizations or industry groups that would improve the implementation of this project or similar projects in the future. Submit these

recommendations and participate in the development of the required standards if possible.

2.4.9 Complete Final Business Case

With vendors and technologies selected, the initial benefit/cost analysis can be expanded into a full business case, including a complete financial analysis, as shown in Figure 2-2. Methods may vary per organization and application.

2.5 Deploy Projects

This section describes steps that may be useful prior to performing a complete roll-out of the project, once a platform-specific architecture has been defined.

2.5.1 Demonstrate New Technologies

If the project is deploying new technologies, it is obviously worthwhile to test them ahead of time in lab or field demonstrations. Besides verifying the claims of vendors about the technology's capabilities, these demonstrations can be important signals to the industry of your organization's commitment to new technologies. They can help ensure the success of these technologies by encouraging others to use them.

2.5.2 Invite Participation

Various organizations, including industry groups or other utilities, may wish to participate in the project if it helps to either advance the status of their own projects or if it helps to reduce costs across the industry. Consider the possibility of working with such partners in pilot projects or demonstrations.

2.5.3 Commercialize Advances

As part of the technology roadmap, you may have chosen to develop specifications, adaptors, gateways, converters, or tools that facilitated the deployment of the project. Consider developing such mechanisms as commercial products or publishing them as "open source" implementations so that knowledge gained through the project will not be lost.

2.5.4 Publish a Reference Architecture

In some cases, it may be useful to publish the platform-independent architecture developed for this project as a ***reference architecture*** that could be used by other organizations prior to making technology selections. Such reference architectures can sometimes be endorsed as industry standards, reducing costs for the whole industry.

3 Requirements Capture Methodologies

3.1 Use Case Methodology

To develop requirements based on Use Cases, the utility must organize a process consisting of series of workshops using cross-functional teams, as illustrated in Figure 3-1. Use Cases place particular emphasis on how the system will actually be used when deployed rather than being constrained by the design of existing products. The utility must clearly define the desired requirements, leaving vendors as free as possible to come up with innovative solutions.

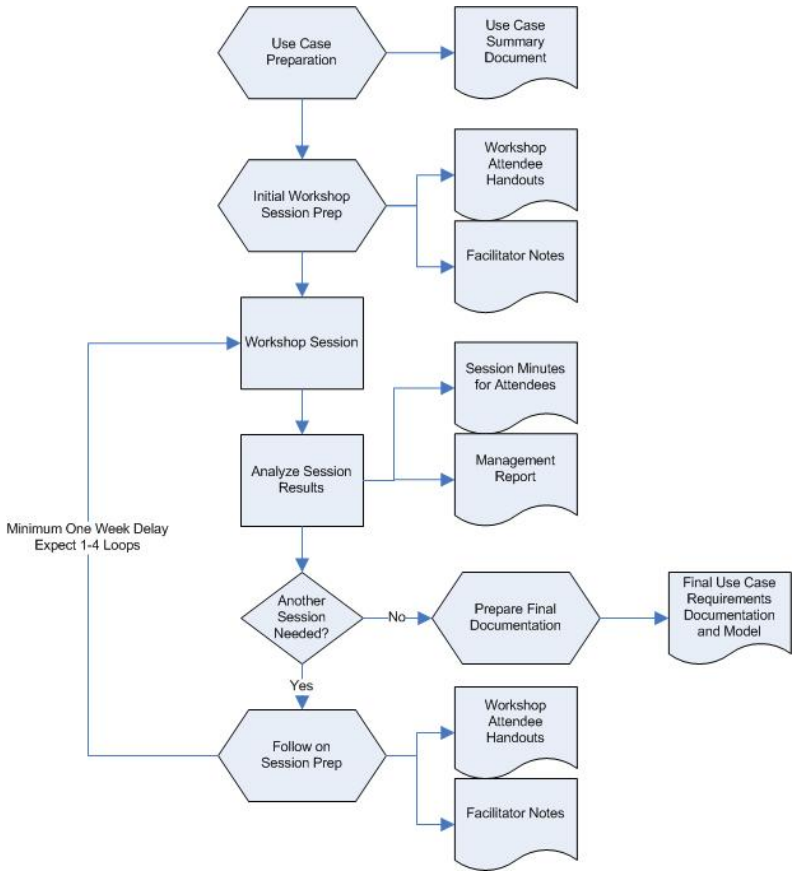

Figure 3-1 The Use Case Workshop Requirements Development Process

3.1.1 Use Case Introduction

A use case is simply a "story" that includes various "actors", and the "path" they take to achieve a particular functional goal. By considering the actions of the actors working to achieve this functional goal, a completed use case results in the documentation of multiple scenarios, each containing a sequence of steps that trace an end-to-end path. These sequential steps describe the functions that the proposed systems and processes must provide, directly leading to the requirements for the given use case.

A use case may have many parts, but the following are the most important:

- The **goal** of the use case, which is usually its name. e.g. "Utility remotely connects or disconnects customer".
- The **narrative.** A short English text version of the story.
- The **actors**. An actor is anything in the system that communicates. It may be a person, a device, a piece of software, an organization, or anything else you can think of that acts on its own and can have goals and responsibilities. e.g. a "customer" or a "meter".
- The **assumptions** that the use case is based on. These can constitute requirements in and of themselves.
- The **contracts** and **preconditions** that exist between the actors, e.g. "The customer agrees with the utility to limit demand on selected days in exchange for a lower tariff."
- The **triggering event** that led to the scenario taking place.
- The **steps**. A numbered list of events that tell the story in detail. Each step identifies an actor, what the actor is doing, what information is being passed, and identifies to whom the information is sent. e.g. "7. The operator sends a curtailment command to the meter".

3.1.2 Use Case Selection

The selection of use cases depends on the high-level project goals and business drivers. A selection of use cases with utility wide scope have been identified by the IntelliGrid Architecture Project. The IntelliGrid Architecture team organized the energy industry into six functional domains:

- Market operations
- Transmission operations
- Distribution operations
- Centralized generation
- Distributed energy resources
- Customer services

A seventh domain, federated systems management, was also identified, which consists of technological functions, such as network management and security that cut across all of the other domains. Among these domains, IntelliGrid has documented eighty high level activities. These activities can be used as a basis for the use cases to consider.

3.2 Use Case Workshops to Develop Requirements

3.2.1 Introduction

A Use Case is a sequence of events that describes one way to use a particular system. It is a story about how a particular user of a system reaches (or fails to reach) a goal. There may be several different scenarios within each use case, each telling a slightly different variation of the story, but all talking about the same goal.

When brainstorming system requirements, there are always a few problems:

- It's difficult to tell whether you are listing real requirements, or just a "wish list".
- It's difficult to know if you thought of everything.
- Once you have your list of requirements, it's not always easy to organize or track it.

The Use Case process of defining a use case forces people to find as many true requirements as possible because:

- It's done from the users' point of view, so it's easier to tell what's really necessary.
- It follows a complete path, so you have more confidence you caught everything.
- Because they're based around user goals, there is a natural way to organize them.

The utility will need to form a number of cross-departmental teams and have assigned them to write Use Cases. As each team develops its story about the system, the team members will discover and make a list of requirements.

The utility assembles the Use Case Workshop members and plans a series of Use Case workshops. There are a number of industry tools that help to facilitate the development of Use Cases. The industry has standardized on a technology suite called Universal Modeling Language (UML) to help facilitate this process. An overview of UML techniques that are used for Use Case development is discussed in the subsequent

sections. Following successful completion of the applicable Use Cases, system architects will need to refine the functional and non-functional requirements process.

3.2.2 Use Case Workshop Membership

The membership of the use case teams should comprise people from internal organizations who are stakeholders or actors in the use case. The teams should contain both subject matter experts as well as decision makers. The subject matter experts provide the technical knowledge needed to ground the use case in reality and the decision makers should be managers with sufficient seniority to choose new policies. It is beneficial to include a workshop facilitator with experience in the use case and requirements gathering process. Each use case team should also have a leader who can guide the team and represent the team in system architecture meetings.

3.2.3 Use Case Workshop Planning

Prior to holding the use case workshop, the team leader(s) and facilitator should create a narrative that describes the high level goal of the use case. The facilitation team can also create straw man content for the workshop including:

- Actors and stakeholders
- Scenario steps
- Discussion topics that are specifically "in" or "out" of scope

These preparatory materials can be presented to the team members at the meeting using slides or handouts. These materials should be reviewed and updated for each additional workshop to reflect the decisions made by the use case team.

3.2.4 Use Case Workshops

A lesson that has been learned from earlier applications of the IntelliGrid use case process is to have a formal indoctrination of the stakeholders in the overall requirements capture process. The size of the project will dictate how best to introduce the team members to the use case process. A focused training session involving all project participants can be used, or the training can occur in the initial use case workshop.

The use case workshops should be structured to engage the team members to participate, keeping in mind that a workshop may have people with widely varying backgrounds and skill sets. A sample workshop presentation is:

- Review and validation of the use case narrative

- Validation of key use case actors and roles
- Discussion of scenarios to be included within the use case
- Discussion of goals for the day's workshop including scenarios to be completed
- General ground rules for the session
- For second and following sessions…a review of outcomes of the prior session(s) and updates on any issues, action items, and parking lot items documented previously

The use case team leader provides overall strategic direction during the course of the workshop sessions. The facilitator provides agenda and process management. The workshop discussion generates proposed scenario steps. Upon general agreement the scenario steps are adopted and the discussion continues until the scenario is completed. The following guidelines can be used to ensure scenario completeness:

- **W**hat, not how. Concentrate on **what** needs to be done, not the technology or network that will make it happen. You don't want to limit the design too early in the process.
- **A**ctor's point of view. Are you looking at the system from the primary actor's (usually the customer's) point of view, or as a designer of the system? The wrong viewpoint can lead to usability problems.
- **V**alue for the actor. Are the steps you're discussing going to help the primary actor accomplish its goal? Are you getting sidetracked?
- **E**ntire scenario. Have you gone far enough back to find the beginning of the scenario, and are you sure you've reached the goal?

The scenario steps should be evaluated by the group to identify functional and non-functional requirements resulting from the step. The requirements also result in the identification of potential quantifiable or non-quantifiable business benefits that can be made possible by the scenarios included in the use case and should be documented.

The results of the use case workshop should be disseminated to the team members and reviewed for accuracy. Questions and issues that arise during the workshop should be addressed by the architecture team and the answers should be included in the post workshop notes. The workshop process is repeated until the use case is complete.

3.2.5 Writing Good Requirements

A requirement is an expression of a ***perceived*** need that something be accomplished or realized. Note that this definition is intended to encompass ***all*** possible requirements for a project. Be aware that in the real world, a "requirement" may merely be something that someone wants – "***desirements".***

The following items help to define "What's a *Good* Requirement?"

Binding

Makes it clear what is optional and what is not

Creates a "contract" with the reader

Shows Responsibility

Identifies what component must take action

Implies whose job it is to ensure it happens

Consistent in Level

May be customer, strategic, functional, design, test, etc.

Should not "jump ahead" to the next level

Measurable

To be usable later in the process

Maybe not by a customer, but by ***someone***

Testable

So you can determine whether the requirement has been met

3.2.5.1 Components of a Good Requirement

Good requirements development utilizes a consistent structure defined as follows:

Requirements are best expressed as complete sentences

Subject-Verb-Object-Qualifier, just like in school

Don't use "and" because that's linking two requirements together that may not actually be linked. Use two sentences.

Subject and Object

Must be well-known parts of the system

Try not to use "the system" – too vague

Define your terms ahead of time

If you involve people, make them ***specific roles*** if possible

GOOD: "operator", "administrator", "maintenance worker"

NOT AS GOOD: "user", or "client"

Verb

Proceeded by a ***binding*** word: "must" is okay, "shall" is best

Can use "may optionally" to identify alternatives

Use "will" only to provide extra info: "this will ensure that…"

Use an ***action*** word

Passive voice is forbidden! It is designed to avoid ownership!

Be as ***precise*** as possible

GOOD: "transmit", "display", "activate", "print", "notify", "connect"

BAD: "is", "be", "have", "contain", "process", "handle", "support"

"permit" is a good action word for user interfaces that puts the responsibility on the system providing the interface

Qualifier

Specifies constraints or performance

Must be ***measurable***

Include a qualifier as often as you can

3.2.5.2 Classes of Requirements

Two types of requirements are developed for the system – 1) Functional and 2) Non-functional:

Functional Requirements

What the system must ***DO***

Actions in response to events, or performed autonomously

Operations and features provided

Non-Functional Requirements

What the system must ***BE***

Also called "constraints", "behavior", "criteria", "performance targets", etc.

Sets limits or controls on how well the system performs the functional requirements.

These are the "..itys" and "…ances": reliability, security, usability, upgradeability, expandability, scalability, compatibility, safety, performance, conformance, etc.

3.3 Use Case Analysis

The goal of the use case analysis is to produce a coherent set of use cases that can be used for subsequent architecture development. The use case analysis process should begin while the workshop process is still underway. The architecture team should review the output of the workshops to evaluate the preliminary requirements and provide feedback. The architecture team can address questions and issues posed by the use case teams and also identify and address scenario gaps or overlaps among different use case teams.

3.3.1 Global Actor List

The workshop process will produce a list of actors and their roles. The names and definitions of these actors should be reviewed to create a standardized list of actor names and roles. The iterative nature of the workshops and the different views that use case teams may hold will cause the creation of duplicate, redundant and conflicting actors. The standardization process should resolve all of these discrepancies.

3.3.2 Activity Diagrams

Activity diagrams are a graphical method to display the events occurring in a use case scenario. The use of this UML tool can provide several benefits. Displaying a use case scenario in a graphical form can allow users to better understand the sequence of events that are occurring and delineate which actors are performing which tasks. The diagram can also reveal points where the textual list of events is insufficient to describe possible outcomes that should be considered in the scenario. The diagram is also useful for indicating interfaces between actors which can also be captured in the UML sequence

diagram. (See Figure 3-2.)

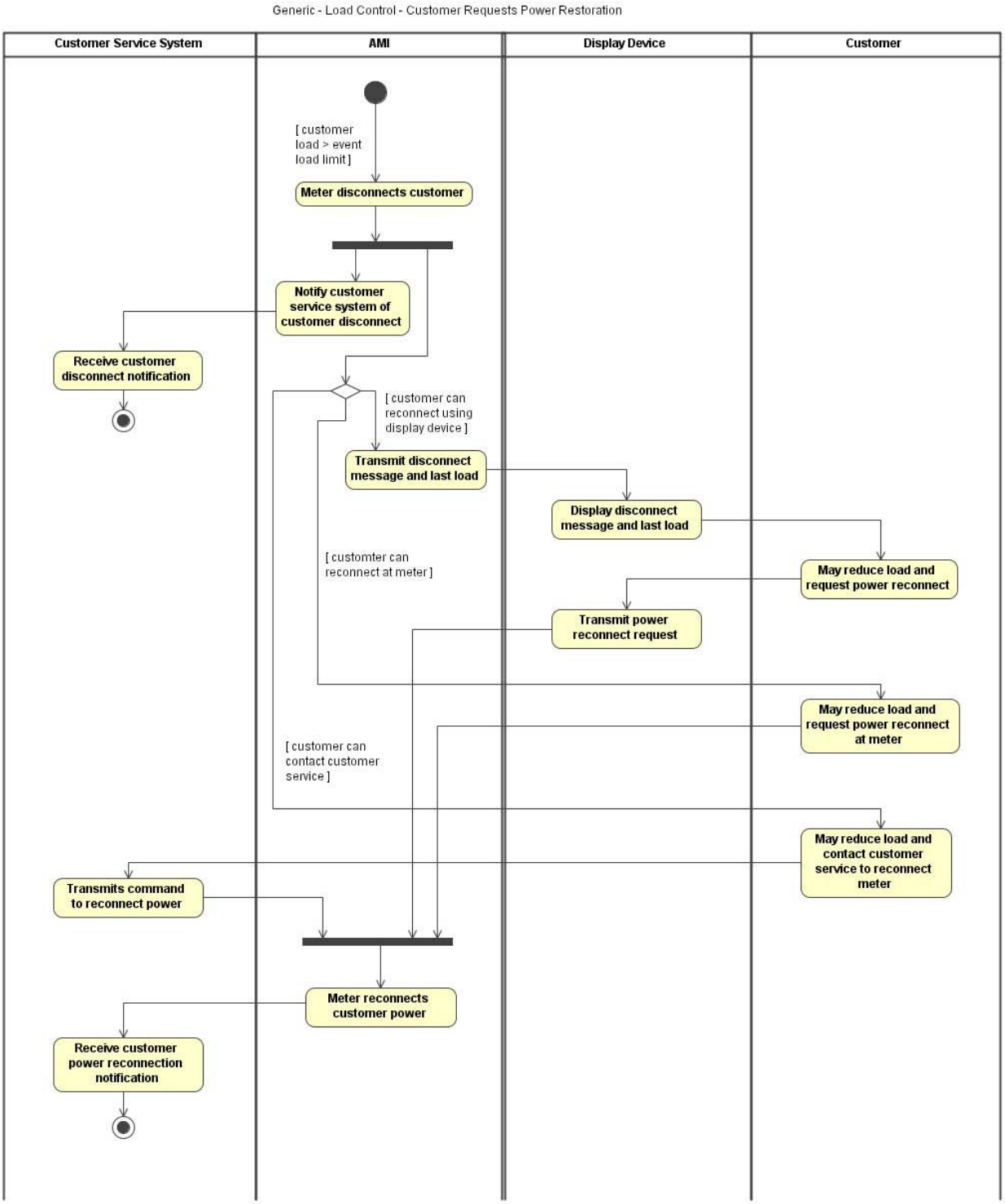

Figure 3-2 Example of an Activity Diagram

3.3.3 *Interface Diagrams*

The purpose of the interface diagram is to provide a single conceptual picture that can be used to express the flow and sequence of data within the system. The interface diagram can be derived from the interactions among the actors indicated by the activity diagram. A preliminary interface diagram can be created during the use case workshops by the architecture team to provide a high level conceptual view. The preliminary diagram should use accepted design patterns and should be responsive to changes required by the use case workshops. There is no defined UML diagram for interface diagrams but the Yourdon dataflow diagram has been used with success in previous projects. (See Figure 3-3.)

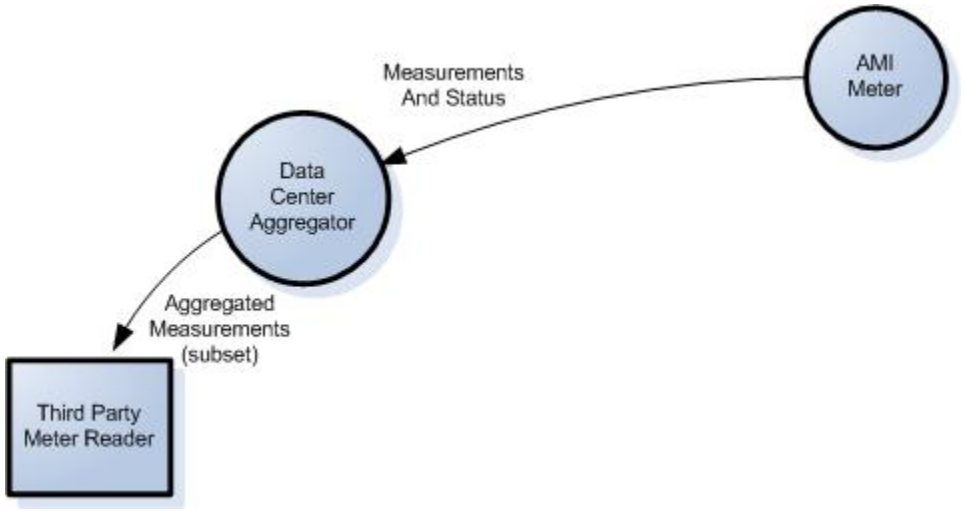

Figure 3-3 Interface Diagram Example

3.3.4 *Message Sequence Diagrams*

The message sequence diagram is a standard UML sequence diagram that illustrates the step-by-step interactions between the actors in a scenario. The message exchanges indicate where interfaces between actors exist and the types and frequency of data that are exchanged. The message sequence diagram can be derived from the use case scenarios or from activity diagrams. Checking that the scenarios, activity diagrams and message sequence diagrams are consistent ensures that the use case data has been correctly captured by the diagrams. The messages should also be entered into a spreadsheet or database to facilitate the entry of additional information about the

message exchanges and to aid the architecture team in the system analysis. (See Figure 3-4.)

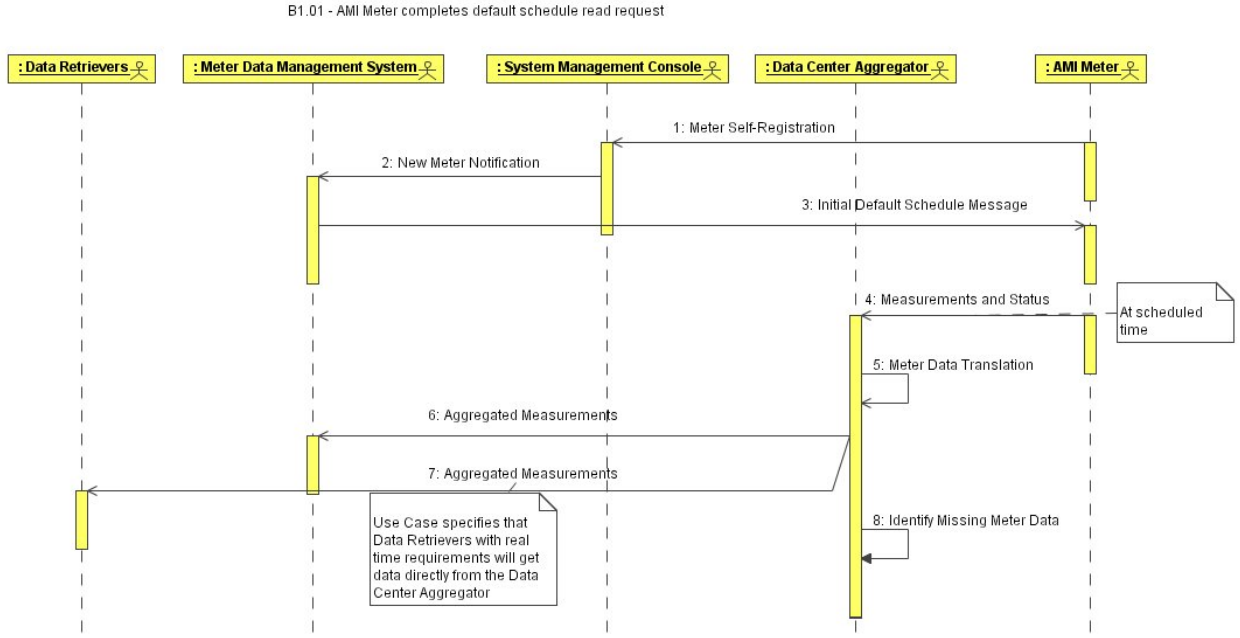

Figure 3-4 Message Sequence Diagram Example

3.3.5 Use Case Interaction Diagrams

Use case interaction diagrams are versions of the interface diagram for each use case which include only the actors and interfaces involved in the use case. These diagrams are useful to illustrate the core parts of a particular use case and to make comparisons between use cases.

3.3.6 Refining Requirements

Refining requirements is the process of reviewing, classifying and consolidating the requirements. The review process should begin during the use case workshop by the architecture team. The architecture team can provide feedback to the use case teams that the requirements that they are generating are of sufficient depth to satisfy the needs of the project. When the use case workshops are complete the final set of requirements can by classified into groups so that similar requirements can be compared for consistency or removed if deemed redundant. The classification can be done by system component or by interface. Non-functional requirements generated by different use cases may also specify the same metric to differing levels of performance. Evaluation of the associated costs and values will determine which level of performance will be

specified as a requirement. The documentation of assumptions made by use case teams as requirements should also occur during this phase so that the assumptions are documented and validated against other system requirements.

4 Conceptual Architecture & Design Process

The Conceptual Architecture is a highly generic and generalized view of the system that is used to communicate the boundaries of the scope of the project and the major architectural elements. The purpose of the conceptual architecture is to convey an understanding of the high-level structure of the intended system to the stakeholders and project team.

Figure 2-1 shows the components of the conceptual architecture and summarizes the methodology used to develop them. Each rectangle represents a component document of the conceptual architecture; documents produced earlier in the project are shown closer to the top of the diagram. Arrows between the components indicate how portions of one component were used to create other components. Larger arrows at the bottom of the diagram indicate how these components are likely to be used in later stages of the project.

The diagram represents an extremely simplified version of the process. The analyses performed in each stage of the development are always used as feedback to improve the results of the previous stages, and there are many interim documents not shown in this diagram. All deliverables, up to and including the original use case documents, are to be considered "living documents" throughout the process.

4.1 Use Case Documents

Use Cases are the vehicle to capture requirements for the system. (Use Cases are covered in detail above in Section 3.) The Use Cases also serve as the basis for allocating functionality to components and architecture elements within the system solution. They help the utility communicate its vision of the system implementation to the outside world.

Each Use Case Documents describes a "story" about one function of the system, in which some "actor" accomplishes a particular goal. The Use Case Documents are produced through brainstorming by cross-functional teams. Each use case consists of a narrative, a list of the involved actors, several scenarios consisting of multiple steps, and the messages and requirements arising from these scenarios.

4.2 Message Sequence Diagrams

The engineering team should develop a standard Unified Modeling Language (UML) Message Sequence Diagram to represent each of the scenarios defined in the Use Case Documents. Each Message Sequence Diagram graphically shows how messages must be exchanged over the system network in order to perform the corresponding scenario.

4.3 Global Actor List

The lists of actors identified in each Use Case Document will be compiled into a Global Actor List. The list defines a coherent set of roles (actors) that users or components of the system may play when interacting with it. This list ensures that actors have consistent names across all use cases. To fully understand the system's purpose a person must know who will be using the system and how the system responds to a particular actor. The team then reviews the list to eliminate ambiguities and duplications so each actor represents a logical component of the system.

4.4 Requirements Database

All the requirements identified in the Use Case Documents by the cross-departmental teams should be compiled in an online database as they were written by the teams. A number of commercial software development toolsets used to manage use cases and requirements are available including IBM's Rational Requisite Pro. A Requirements Management Plan describes the database and requirements types as well as the process for managing system requirements. The database can produce a number of different reports, including lists of functional requirements (describing what the system does) and non-functional requirements (describing how and when it does them).

4.5 Security Analysis

Because many of the requirements teams did not address security issues in depth, the engineering team undertook a separate Security Analysis. The result of this analysis was a spreadsheet identifying the data requiring protection within the system, cross-referenced with a variety of technologies and methodologies that might be able to protect that data.

4.6 Interface Diagram

The Interface Diagram shows in a single Yourdon-style dataflow diagram all the primary actors identified in the Global Actor List and the main interfaces between those actors. All the messages identified in the Message Sequence Diagrams fall into one of the interfaces depicted in the Interface Diagram. The purpose of the interface diagram is to provide a high-level view of the conceptual architecture in which all members of

the development team and the vendors contributing to the project can recognize the functions for which they are responsible. It also provides a framework for classifying messages and beginning the design of the system communications network.

4.7 Message Matrix

Each of the messages described in the Message Sequence Diagrams should be listed in a spreadsheet identifying the source, destination, sequence, periodicity, and type of the message, as well as the Interface from the Interface Diagram that the message belongs to. Eventually this Message Matrix will be used to estimate the required performance of the system network.

4.8 Use Case Interaction Diagrams

A Use Case Interaction Diagram represents the subset of the Interface Diagram that applies to a particular Use Case. There is one Interaction Diagram for each of the Use Cases. Interaction Diagrams show the relationships among actors and use cases within a system. Interaction Diagrams are used by the architecture team to determine how use cases are related and what additional integration requirements the solution must meet.

4.9 Standards Catalog and Mapping

The Standards Catalog is a listing of candidate standards that may be applicable to the system. The engineering team shall evaluate candidate standards to determine fit with the system objectives and design principles. The Standards Mapping spreadsheet indicates which of the candidate standards are applicable to each of the interfaces identified in the Interface Diagram.

4.10 Component Architecture

The logical components identified in the Global Actor List are broken down into more physical representations in the Component Architecture diagrams. The Component architectures help the utility manage the complexity of the proposed system design by breaking the design into smaller segments of the system solution. The Component Architecture shall be used to facilitate communication of the utility's requirements to vendors and other stakeholders.

The component architecture includes the Requirements to Component Mapping, which allows the architecture team to understand the required behavior of each component necessary to realize the use cases. The map also allows the architecture and engineering teams to understand component interface boundaries and requirements and how

components must function together within the proposed solution to fulfill particular use case scenarios.

4.11 Distilled Requirements

The raw requirements gathered from the Use Case Documents and stored in the Requirements Database have been filtered to eliminate duplicates and any requirements that do not apply directly to components of the infrastructure. The distilled requirements are also classified in a number of different categories in preparation for releasing them to vendors.

4.12 Capability Frameworks

The Technology Capability Measurement frameworks are a series of scales consisting of numbers from 0 to 5 used to evaluate the capability of a system in a number of different technology categories. The TCM scales represent a superset of the requirements identified in the Requirements Database and are used to indicate to vendors the direction that the utility would like to see vendor product development evolve.

4.13 Architectural Decisions Document

The Architectural Decisions document is not shown in Figure 1 because it is written based on the development of *all* of the other components of the conceptual architecture. This document captures important decisions about any aspect of the architecture including the structure of the system, the provision and allocation of functions, the contextual fitness of the system and adherence to standards. Architecture is understood partly through the record of the important decisions made during its development. The justification and evaluation criteria are recorded in this document alongside the decision, along with any references to more generally applicable principles, policies and guidelines, which are found in other work products or in external references.

5 Security Best Practices

5.1 Introduction

Security plays a vital role any information system today. Availability, integrity, confidentiality and non-repudiation are the categories for security threats. Once the interfaces between actors have been identified in the Conceptual Architecture, these interfaces and actors can be grouped into security domains having the same potential threats and technological solutions. A specialized security team should be assembled in order to define security domains. The security organization must decide on a common set of security and network management policies. These policies must define the overall objectives of security measures, the roles of organizations and individuals, and procedures for managing assets and credentials. These policies must also identify how procedures may be modified within each domain while maintaining the overall security objectives.

5.2 Information Security Fundamentals

5.2.1 Landscape and Adversaries

Fundamentally, security is about managing risk that stems from the malicious actions of an adversary. Adversaries and the technological landscape are not static, and their rates of change vary widely – from the overnight paradigm shift to the subtle erosion of engineering assumptions. Security therefore becomes a living, ever-changing process requiring constant monitoring and action. Just like any other aspect of systems engineering, controls must be built to facilitate manipulation of this dynamic system. The controls may then be used to keep the system within a desired set of parameters. A security framework may be viewed as a means to implement security controls on a system in the face of a continuously evolving landscape and adversary profile.

5.2.2 Confidentiality, Availability, and Integrity

A key aspect of Information Security is to preserve the Confidentiality, Integrity and Availability of an organization's information. It is only with information that it can engage in commercial activities. Loss of one or more of these attributes, can threaten the continued existence of even the largest corporate entities. Confidentiality is defined

as assurances that information is shared only among authorized persons or organizations. The term Integrity means assurance that the information is authentic and complete and ensuring that information can be relied upon to be sufficiently accurate for its purpose. Assurance that the systems responsible for delivering, storing, and processing information are accessible when needed by those who need them defines the term Availability.

This section should help you understand the basic ideas behind confidentiality, Integrity, and Availability used in the risk assessment and security compliance process by IT security policies. These classifications represent the key security requirements of any system and are extremely important for risk analysis and security requirements captured during the discovery phase of the security risk assessment process.

Confidentiality is the privacy of an asset. Specifically, confidentiality can be defined as which people, under what conditions are authorized to access an asset. Confidentiality refers to limiting information access and disclosure to authorized users -- "the right people" -- and preventing access by or disclosure to unauthorized ones -- "the wrong people." Underpinning the goal of confidentiality are authentication methods like user-IDs and passwords that uniquely identify a data system's users, and supporting control methods that limit each identified user's access to the data system's resources. Also critical to Confidentiality -- and data integrity and availability as well -- are protections against malicious software (malware), spyware, spam and phishing attacks.

Integrity is more difficult to define than confidentiality as there are two primary properties to consider when evaluating it. First, there is the notion that an asset should be trusted; that is, there is an expectation that an asset will only be modified in appropriate ways by appropriate people rather than from an imposter. The second part of integrity is that in the event that data is damaged, or incorrectly altered by authorized or unauthorized personnel, you must consider how important it is that the data be restored to a trustworthy state with minimum loss. While the relative risks associated with these categories depend on the particular context, the general rule is that humans are the weakest link. (That's why each user's ability and willingness to use a data system securely are critical.)

Availability represents the requirement that an asset be accessible to authorized person, entity, or device. Availability refers, unsurprisingly, to the availability of information resources. Almost all modern organizations are highly dependent on functioning information systems. Many literally could not operate without them. Availability, like other aspects of security, may be affected by purely technical issues (e.g., a malfunctioning part of a computer or communications device), natural phenomena

(e.g., wind or water), or human causes (accidental or deliberate). As a general rule, the more critical a component is, the higher its availability will be.

Security efforts to assure confidentiality, integrity and availability can be divided into those oriented to prevention and those focused on detection. The latter aims to rapidly discover and correct for lapses that could not be -- or at least were not -- prevented.

The balance between prevention and detection for depends on the circumstances, and the available security technologies. Most information systems employ a range of intrusion prevention methods, of which user-IDs and passwords are only one part. They also employ detection methods like audit trails to pick up suspicious activity that may signal an intrusion.

5.2.3 Systemic Risk

A particularly holistic method of incorporating information security into the work processes is to accept information security risk as systemic and account for it using a risk management approach. The risk management process is no different for a utility or even for a SCADA/Control System than it is for any other large business. The difference lies in the priorities and constraints for an electric utility which are driven by the business model and become process parameters.

For any organization, an information security framework may be effectively modeled / described by reflecting the types of data on hand and the environments in which they may be found. For an electric power utility, this can typically be broken down into three basic categories of data and three environmental groups. From a logical standpoint, information can be categorized into non-electric system data (such as payroll, customer info, or public relations), engineering data (such as configuration, analysis, maintenance, and testing), and operational data (SCADA). The environmental realm may be broken down into the corporate network, the control center, and remote facilities. At a high level, guidance may then be provided at the various intersections of information and environment.

5.2.4 Security Domains

The security concept used by IntelliGrid focuses on Security Domains (SD) that are defined by their security boundaries.

A Security Domain encompasses a common security method applied to a collection of computer and network systems used to provide integrity, availability, authentication, and confidentiality.

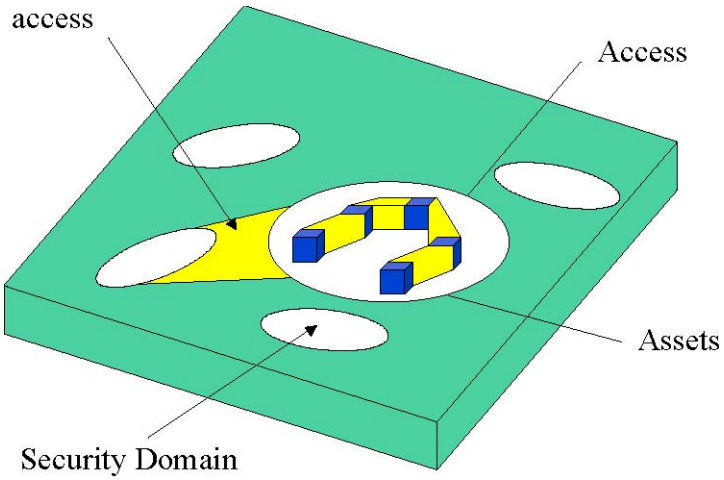

Figure 5-1 Representation of the Security Domain Concept

A Security Domain (SD) represents a set of resources (e.g. network, computational, and physical) that is governed/secured and managed through a consistent set of security policies and processes. Thus each Security Domain is responsible for its own general security process (e.g. Assessment, Policy, Deployment, Monitoring, and Training). In addition to the general security process, a Security Domain provides a well-known set of security functions that are used to secure transactions and information within that domain.

Security Management covers a set of functions that:

(a) protects computer systems and communications networks from unauthorized access by persons, acts, or influences

(b) creates, deletes, and controls security services and mechanisms

(c) distributes security-relevant information

(d) reports security-relevant events

(e) controls the distribution of cryptographic keying material

(f) authorizes subscriber access, rights, and privileges

Based upon this definition, it is the Security Management of a Security Domain that is responsible for the risk assessment, developing security policies and strategies, and implementing those policies and strategies.

A successful Security Domain will define and implement the following security services:

Access Control: prevent unauthorized use of a resource, including the prevention of use of a resource in an unauthorized manner. There are generally three (3) categories of Access Control that need to be addressed within a Security Domain: Physical; Resource; and Information.

Ensuring Trust: An application of security methods allowing one entity to assume that a second entity will behave exactly as the first entity expects. Trust may apply only for some specific function. The critical role of Trust in the authentication framework is to describe the relationship between an authenticating entity and a certification authority. An authenticating entity must be certain that it can trust the certification authority to create only valid and reliable certificates. There are two methods of authentication that are prevalent in today's electronic systems: Role Based Authentication and Individual Authentication.

Ensuring Confidentiality: The security method that ensures information is not made available or disclosed to unauthorized individuals, entities, or process. There are two (2) categories of Confidentiality that need to be addressed within a Security Domain: Protection from un-intentional disclosure and overall protection of information.

Ensuring Integrity: The security method that keeps information from being modified or otherwise corrupted either maliciously or accidentally.

Creation of Security Policy: A set of rules and practices that regulate how an organization manages, protects, and distributes sensitive equipment and information. Security Policy also covers training on security procedures.

Security Management Infrastructure (SMI): System elements and activities that support security policy by monitoring, controlling, and auditing security services. SMI also handles mechanisms distributing security information and reporting security events.

Security services implemented in a Secure Domain must be specified for inter-domain and intra-domain exchanges.

Table 5-1: Services needed for Intra/Inter Domain Security

m: mandatory, o: optional

Security Service	**Intra-Domain**	**Inter-Domain**	**Comments**
Audit	m	m	
Authorization for Access Control	m	m	
Confidentiality	o	m	
Credential Conversion	o	m	
Credential Renewal	m	m	
Delegation	o	m	
Firewall Transversal	o	m	
Identity Establishment	m	m	
Identity Mapping	o	m	
Information Integrity	m	m	
Inter-Domain Security	Not Applicable	m	
Non-Repudiation	m	m	
Path Routing and QOS	o	o	
Policy	m	m	
Privacy	o	o	
Profile	m	m	
Quality of Identity	See comment	m	In order to provide this service for inter-domain, it must be available for intra-domain applications to make use of.
Security Against Denial of Service	o	m	
Security Assurance	m	m	
Security Protocol Mapping	o	m	
Security Service Availability Discovery	m	m	
Setting and Verifying User Authorization	m	m	
Single Sign-On	m	Not Applicable	
Trust Establishment	m	m	
User and Group Management	m	m	

5.3 Scope and Integration

5.3.1 Data Classification

Data can be effectively measured along the axes of sensitivity and value. From a sensitivity standpoint, data should be examined in each of the separate categories of confidentiality, availability, and integrity. How sensitive is the data to compromise in each of the three areas?

For a utility, the issues of availability and integrity tend to be most dominant, and it is important to note that this runs counter to the majority of other industries. The result is that the vast majority of technical solutions available address confidentiality first and foremost. Encryption is a prime example. Depending on the implementation, it typically addresses confidentiality to a high degree, may have some collateral benefits for integrity, and has little or possibly even a negative affect on availability. Is it possible that you really don't care if an intruder sees the data; you just don't want them to be able to tinker with it? Does the data get stale and irrelevant? How long does it take? Longer than it takes to use a strong encryption algorithm without spending a fortune on hardware? Longer than it takes anyone shy of a nation-state to break a "weak" encryption algorithm?

Once you have determined the sensitivity of your application data, you need to go back and assign it a relative value. How important is the data? Will loss or corruption cause a blackout? Loss of business efficiency or revenue? Loss of marketplace advantage? How much engineering effort will be involved in recovery from compromise of the data? You may even decide that you need to assign a separate value for each of the aspects of confidentiality, availability, and integrity.

5.3.2 Requirements

The utility must identify what its functional and non-functional requirements are with regard to security. Functional requirements are things that the solution must either do or not do, e.g.: "the solution must support authentication, authorization, and accounting." Functional requirements do not have any kind of measurement in them. Non-functional requirements are used to describe a measure of how well the solution must do something, e.g.: "the solution must support at least 10Mb/sec peak transfer rate." The utility should define functional and non-functional requirements from both the business and data perspectives.

Be sure that you examine the technical implications of the solutions you are considering and map them to the specific requirements that you generate. In such a complex and varied territory, it is easy to stumble across tough constraints late in the game if you are not careful. For example, if you decide to run a VPN across a satellite solution, you need to understand the impact it will have on the overall latency. In short, you need to do a thorough job of identifying the interdependencies of your requirements as they map to any potential solution. Failure to do so can cause significant headache and re-engineering time down the road.

5.3.3 Risk Management

If the goal of security is to protect designated assets from malicious action, the challenge is that in the real-world security is limited by resource allocation. This means that the security domain is predominantly about minimizing, but not eliminating risk. The most effective security usually revolves around risk management coupled with business continuity planning.

The following tasks comprise the core philosophy of approach:

- Asset Identification – What is being protected and why?
- Threat Identification – What events present danger to identified assets?
- Vulnerability Analysis – What are the system weaknesses, how severe are they, and how often are they exposed?
- Risk Evaluation – What level of concern is assigned to a threat exploiting a vulnerability to compromise an asset?
- Risk Treatment – In what ways can risk be avoided, reduced (mitigated), and/or transferred (e.g.: insurance)?
- Cost / Benefit Analysis – What is the balance point between the value of a risk and the expense of treating it?

As mentioned above, not all risk is eliminated in real-world security. Some risk is accepted, or "retained." Therefore by definition, sometimes the worst happens. This is the foundation for business continuity planning – "Plan B." A business continuity plan provides instructions for maintaining operations when retained risk is realized.

At its fundamental core, Risk (R) is calculated as being the product of the Likelihood (L) of a cost event occurring and the magnitude or value (V) of that cost.

R = L * V (Equation 5-1)

There are many ways to represent the Value of cost, and at its simplest, it can be a sheer monetary value. Also a utility can use a qualitative ranking for Value. Value for this scenario could be assigned on a scale of 1 to 10. While this is an arbitrary measure, it at least provides the utility with a metric by which they can evaluate relative risk and form decisions about the implementation of mitigation techniques. The utility can then designate a threshold of acceptable risk, above which all risks should be mitigated to the satisfaction of the utility.

5.4 Technologies

5.4.1 Open Standards

As a general rule, equipment that follows open standards tends to be more scalable, efficient, robust, and interoperable. Open systems can receive significant peer scrutiny and often benefit from such technical review. Closed, proprietary algorithms and protocols are viewed as unknown (and therefore untrusted) variables in the security world, and are usually found to be far less secure than those having been exposed to the scrutiny of the broader community. That being said, not all standards are equal, even if they are openly published. Do your homework to find out how long a standard has been in use, how rigorously it has been tested, any issues that are known and documented, and what its status is in the security community.

5.4.2 Open Security Technologies

IntelliGrid has assembled an extensive reference list of security technologies and security documents to assist the IT security team in its efforts. These are located in the IntelliGrid *IECSA Volume 4 Technical Analysis; Appendix D Technologies, Services, and Best Practices*. A copy of the Table of Contents is provided in this Guide as Appendix B. The sections that cite security related information include:

1.3 Security Technologies

2.1 Security Services

3.2 Security Best Practices

4.1 Security Technology Documents

Topics covered in detail in these documents are:

- Policy and Framework Related Technologies
- General Security Technologies
- Media and Network Layer Technologies
- Transport Layer Security Technologies
- Application Layer Security Technologies
- XML Related Technologies
- Common Security Services
- Security Policy
- ISO/IEC Documents on Security Technologies
- Federal Documents on Security Technologies
- IETF Internet RFCs on Security Technologies
- IEEE Standards for Security

6 Technology Capability Assessment and Selection

The development of the IntelliGrid Architecture included an overview of all the technologies in use at the time. New technologies are constantly being developed. IntelliGrid continues to evaluate available technologies for inclusion in the IntelliGrid Architecture. Some technologies will fall out of market favor and others may be too proprietary. This is an ongoing process which is captured and updated by IntelliGrid.

The IntelliGrid Architecture defines a set of operating environments inside the utility. These are defined globally as:

Customer Metering

Inside the Substation,

Between Substations

Substation-to-Control Center

Inside the Control Center

Between Control Centers

Distribution system and Feeders

Between Utility and Marketplace

For each IntelliGrid project a detailed technology capability assessment is performed. Two technology assessment methods are illustrated in this section – 1) Technology Capability Methodology and 2) Telecommunication Technology Assessment using Weighting Scales.

6.1 Technologies Capability Methodology

Based on continuing engagement with suppliers of power system environment technologies, it is apparent that there is considerable diversity in not only the capabilities, but in the basic approach and stated strategic directions of the technology vendors. In order to objectively understand the capabilities of various solution components as well as communicate desired features to support system requirements, utilities should implement an abbreviated form of the Technology Capabilities Maturity

methodology (TCM)[2] which results in a simple capability maturity framework using a 0 to 5 scale for several technology categories.

The TCM scales described in this document serve as a tool to help a utility evaluate and communicate their perspective on the current state of the art, as well as define an evolutionary path for capabilities that the utility considers central to the system. Utilities should create a set of required technologies based on an initial analysis of the output from requirements workshops. These will be a collection of related features that the utility believes will significantly affect the benefit to be derived from the deployed system. The utility should use the TCM scales to gauge the relative maturity of various communications technology available from suppliers now and in the near future. This will enable the utility and suppliers to discuss product availability timelines and capabilities in the context of the TCM scales and assist in the refinement of the utility's ultimate requirements.

These scales identify maturity levels for specific capabilities, with the low end of a scale generally based on what are commonly available today and the middle to upper range of the scale associated with what the utility believes it needs to capture the most value from a proposed system solution. As a result, the scales are optimized to differentiate capabilities in the range of interest to the utility, and are not intended to be linear. Overall, the utility should expect the ultimate requirements to fall somewhere in the middle to upper range for most of the technology elements but will continue to evolve upward. This process allows a utility to qualitatively perform an assessment of the industry's offerings and develop a technology gap analysis that vendors may want to address in the future.

6.1.1 TCM Approach

The TCM methodology as described here is based upon breaking the technology evaluation problem down into several technology categories and within each of those categories, identifying specific capabilities that are desirable and necessary to meet the requirements captured for the system deployment. Typical (but not exclusive) technology capability categories are:

Device Platform Flexibility (Micro-controller unit, storage, and memory)
Availability
Interoperability (Meters to Communications)
Reliability

[2] Based on CFTP developed by J. Paap, MIT

Scalability
Security
Serviceability / Maintainability
Throughput (Daily)
Throughput (on demand poll)
Wide Area Network (WAN)
Home Area Network (HAN)
Local Area Network (LAN)

The capabilities associated with each category are presented in a table as columns ordered from most technologically advanced on the left to least technologically advanced on the right. The rows of the table indicate a capability maturity level on a scale from 0 to 5 that represents an overall index of the relative level of technology capability where 0 represents the least capable and 5 represents the most capable and advanced from the utility's perspective. For a given capability level, X's are placed in cells under the respective technological capabilities that must be implemented to be considered to be at that level. The result allows the utility to refer to a capability level by a single number for a specific technology category – the table for that category clearly defines which specific capabilities are necessary to achieve that level.

6.1.2 TCM Example: Field Device Availability

In this section, an example is provided to illustrate the TCM development. This category describes the ability of the platform to support graceful failure and recovery mechanisms.

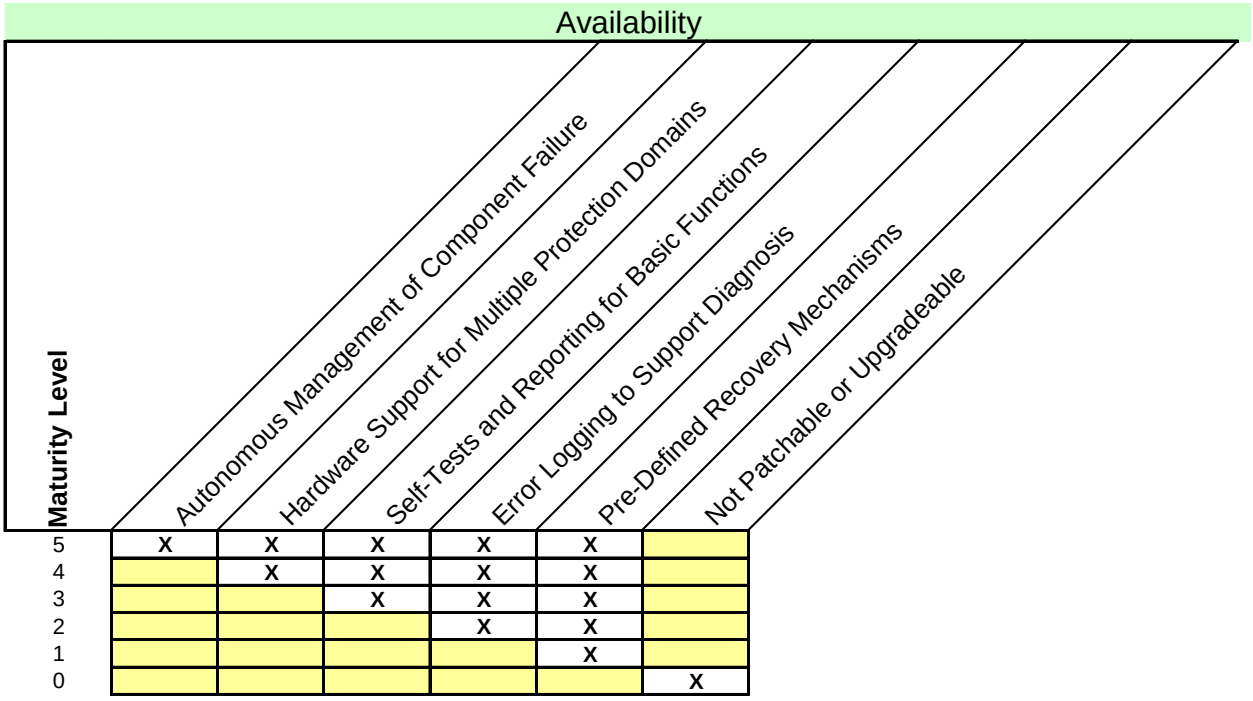

Figure 6-1 Example of Field Device Availability

Detailed Scale Definitions

Autonomous Management of Component Failure

Capable of identifying and automatically disabling system components with persistent failures, i.e. locking out bad memory addresses, applications, device adapters etc. For non-metering devices, this level of capability would include redundancy and the ability to select among multiple alternatives to accomplish the task at hand.

Hardware Support for Multiple Protection Domains

Hardware support of multiple protection domains, i.e. ring levels in 80386 terminology. Hardware can differentiate and provide different capabilities, authorization etc. to core, supervisory, and application code. The communications device supports exception - based and/or interrupt-driven, error detection and logging routines. Failed software components would be automatically restarted.

Self-Tests and Reporting for Basic Functions

The communications hardware has self test and reporting mechanisms for core capabilities sufficient to ascertain their basic functionality i.e. whether the processor is able to communicate with HAN Network Interface Card (NIC), whether there are signals present from current and voltage sensors etc., and whether it is able to detect and or initiate message traffic with other devices or components.

Error Logging to Support Diagnosis

Communications module supports error logging with sufficient information to diagnose or recreate the error condition. The device has the capable of asynchronously notifying other devices or components of an error condition (i.e. outside of regular reporting).

Pre-Defined Recovery Mechanisms

The solution supports error logging with sufficient information to diagnose or recreate the error condition. The communications device is capable of asynchronously notifying other devices or components of an error condition (i.e. outside of regular reporting).

Not Patchable or Upgradeable

The communications device cannot be altered without returning the unit to a repair center.

As shown in the above example, each functional requirement identified under the heading of "Availability" has been assigned an "X" that identifies its current maturity level.

6.2 Telecommunication Technology Survey

6.2.1 Scope of Telecommunication Technology Assessment

The purpose of this section is to assess and recommend a set of "technologies" used in information exchange, where the term "technology" refers generically to any of the following:

Individual communications protocols

Suites or profiles consisting of several protocols

Media (i.e. physical links) used for communications

Classes of networks, protocols, or devices

Communications services

Standards that enable communications, such as standard file formats

The reader should note the following additional restrictions on the scope of this section: This is a *telecommunications* technology assessment only. It does not evaluate other types of technologies that might be used for information systems, e.g. displays,

processors, etc. This section does not recommend *the* final list of suitable technologies for a power system environment, but simply *a* list from which specific choices for a power system environment reference design may be made. In some cases particular technology choices will *never* be specified, because one of the goals of an IntelliGrid Reference Design is to be independent of the lowest layers of communications technologies.

6.2.2 Open Standards

Project engineers can see the list of IntelliGrid standards, technologies, and best practices associated with a power system environment by referencing Appendix C which lists the *IntelliGrid IECSA Vol 4: Technical Analysis; Appendix D Technologies, Services, and Best Practices; Table of Contents* or on the IntelliGrid website. At the same time, project engineers will need the flexibility to use alternative technologies for specific implementations, due to existing legacy systems, existing vendor products, time constraints for implementation, financial constraints, current company policies on technology choices, and a variety of other factors.

6.2.3 Assessment Organization and Approach

This section concerns itself with developing:

A list of technology ***evaluation criteria*** that would be use to utilities interested in implementing power system environment oriented projects. These criteria are derived from the IntelliGrid principles and recommendations.

Technology Assessment: Each technology can be described and assessed with respect to the evaluation criteria. In addition, each assessment should address the following five areas:

> Application to the selected power system environment – how this technology would be used in the portal environment.
>
> Strengths – what the technology does well.
>
> Concerns – reasons why the technology may be less suitable.
>
> References – a list of web sites where the specifications for the technology and more information can be found.
>
> Layers – the OSI layer or layers that the technology implements.

The technology assessments are organized according to ***service groups*** illustrated in Figure 6-2.

The concentric rings in the figure indicate more generic, shared or common technologies toward the center, and more specialized, project-specific or application-specific technologies toward the outside.

When selecting technologies for a particular power system environment project, a system engineer should start at the center and work outwards.

Figure 6-2 Communications Service Groups

6.2.4 Communication Service Groups

The following is a list of the communication service groups. It is important that **all** of these service groups be represented in any particular implementation.

Core Networking – a common suite of protocols is needed to provide interoperable connectivity in a network that may vary greatly in topology and bandwidth.

Security – Because consumer portals must deal directly with personal consumer information and billing, it is critical to build security measures into consumer portal communications from the very beginning.

Network Management – Because of the huge numbers of devices that can be involved in a portal network, it is vital that standard technologies be used for managing the network, i.e. collecting statistics, alarms and status information on the communications network itself.

Presentation and Configuration – A key principle of the IntelliGrid Architecture is the use of "meta-data" for formally describing and exchanging how devices are configured, and how they report data. Consumer portals need this capability to manage the large number of devices involved.

Wide Area Network (WAN) Technologies – The problem of how to reach the consumer site represents the most rapidly-changing area of portal technology, and the one that will have the most impact on its commercial viability.

Local Area Network (LAN) Technologies – The technology that makes a portal distinct from being just a "smart meter" or "smart thermostat" is its ability to network with other devices locally.

Power System Operations – Several of the key applications for portals involve integration with distribution system operations, such as outage detection, power quality monitoring, and emergency load shedding.

Consumer Applications – Of course, electrical metering is the application that most people associate with a consumer portal. However, various aspects of building automation are also fundamental to the consumer portal concept, and are therefore discussed here.

6.2.5 General Observations

It is important that any consumer portal reference design or implementation include technologies from *each* of the service groups. Individual projects may choose to emphasize one service group over another, but for a good design, all service groups are necessary.

In each service group, there are typically two or three candidate technologies, each of which is nearly equally suited for use in the power system environment compared to the other. For example, IPSec and TLS, SNMP and CMIP, or ANSI C12 and DLMS/COSEM. The success of a reference design may rely on how they can be made to work together, rather than choosing one over the other.

There are liable to be overlaps even between technologies in different service groups. Therefore a reference design will need to clearly identify roles and options that will permit them to work together properly, and not just define a "shopping list" of

technologies. It is vital that any reference design be completely independent of the local-area or wide-area networking technologies available.

Many of the consumer application technologies support mechanisms that would allow their data to be "tunneled" through an IP network. One possible strategy for harmonization of these various technologies would be for the portal to simply act as a gateway for such tunnels, "wrapping and unwrapping" messages in IP "envelopes" from clients at the utility site to equipment on the customer premises and vice versa.

6.2.6 Technology Ratings

The table below shows ratings given to each of the technologies described in this document, using the evaluation criteria described below.

The purpose of this table is not to suggest that any particular technology is "the best", even within its service group. This set of ratings assumes an equal weight for all criteria, but a particular project may choose to perform a weighted average. The purpose of providing these ratings is rather to give an overview of the capabilities of the technologies and to show how the evaluation criteria could be used in the future.

The intent of this document is to provide a "short list", not a comprehensive discussion of all technologies. For a more comprehensive (but less detailed) list of technologies, please refer to the IntelliGrid Architecture documentation.

Lastly, it should be noted that one criterion that is missing from this table is how well the protocol does its job. This is not shown in the table, because it is extremely difficult to find a non-subjective metric for such a quality. For that kind of information, the reader should look at the Strengths and Concerns sections for each technology. The Level of Adoption, Applicability to the Power Industry, and Applicability to the Consumer Domain metrics may also provide clues in this regard.

IntelliGrid recommends the criteria described in this section be used for evaluating technologies in consumer portal projects. These criteria are interrelated and may affect each other. For instance, open standards are preferred, but a solution that is partly proprietary and solves a problem of scalability or security may end up being the best choice if it provides well-defined, published interfaces.

Each of the following sections explains the need for a particular criterion, describes how the criterion might be evaluated in qualitative terms, and then provides a table explaining how the technologies in this document were rated on this criterion. Although each criterion is arguably a multi-dimensional measurement, it was necessary for simplicity's sake to normalize ratings to a number from one to five.

	Standardization	Openness	Adoption	User's Group	Security	Manageability	Scaleability	Object Modeling	Self-Description	Power Industry	Consumer	**TOTAL**	Bar Graph
Core Networking													
IPv4	3	5	5	4	2	4	4	1	2	3	1	**34**	**********
IPv6	3	5	2	4	4	4	5	1	5	2	1	**36**	**********
TCP	3	5	5	4	2	4	4	1	2	3	1	**34**	**********
UDP	3	5	5	4	1	4	4	1	1	3	1	**32**	**********
HTTP	3	5	5	5	2	3	4	1	4	3	1	**36**	**********
Security													
TLS	3	5	5	4	5	3	4	1	3	3	2	**38**	**********
IPSec	3	5	5	5	5	3	5	1	3	3	2	**40**	**********
HTTPS	3	5	5	5	4	2	4	1	4	3	2	**38**	**********
SSH	2	4	4	3	5	2	2	1	2	2	2	**29**	**********
X.509	5	4	4	1	5	5	3	1	4	3	2	**37**	**********
IEEE 802.11i	5	3	2	5	5	4	2	1	2	2	1	**32**	**********
Management													
Basic IP	3	5	5	4	1	5	4	1	3	3	3	**37**	**********
SNMP	3	5	5	4	2	5	3	4	2	2	2	**37**	**********
CMIP	5	3	2	1	3	5	3	4	2	1	1	**30**	**********
NTP/SNTP	3	5	5	4	1	5	4	1	2	3	3	**36**	**********
IEEE 1588 (PTP)	5	3	2	4	1	5	3	1	2	4	4	**34**	**********
Presentation													
HTML	3	5	5	4	2	5	5	4	4	3	3	**43**	**********
XML	3	5	5	4	2	5	5	5	4	2	2	**42**	**********
BNF	2	5	3	1	1	1	5	5	3	3	3	**32**	**********
ASN.1	5	5	5	1	1	1	5	5	3	3	2	**36**	**********
IEC 61850-6 (SCL)	5	5	2	5	2	5	2	4	4	4	2	**40**	**********
SOAP and Web Service	3	5	4	4	2	5	2	4	5	2	2	**38**	**********
ebXML	5	5	2	4	2	5	2	4	5	2	2	**38**	**********
LANs													
Ethernet	5	5	5	1	3	4	3	1	5	3	2	**37**	**********
Wi-Fi	5	4	4	5	3	4	2	1	5	2	2	**37**	**********
ZigBee	5	4	1	5	4	4	2	1	5	4	4	**39**	**********
Bluetooth	5	4	4	5	2	4	2	1	5	1	1	**34**	**********
HomePlug	3	3	2	5	3	3	3	1	5	2	2	**32**	**********
X10	1	4	5	2	1	1	1	1	5	4	4	**29**	**********
WANs													
DSL	5	4	5	5	4	4	4	3	3	2	2	**41**	**********
Cable	5	5	5	5	4	4	4	3	3	2	2	**42**	**********
WiMAX	5	4	2	5	4	3	3	3	5	1	1	**36**	**********
Access BPL	1	2	2	2	3	2	4	1	2	4	4	**27**	**********
Narrowband PLC	5	3	4	1	1	2	3	1	1	5	4	**30**	**********
Paging	3	2	5	1	1	2	4	1	5	3	3	**30**	**********
Satellite	2	2	2	1	4	4	3	1	1	3	3	**26**	**********
Cellular	5	1	2	2	3	4	4	3	5	3	3	**35**	**********
FTTH	5	3	2	2	4	4	4	3	3	1	1	**32**	**********
Power System Operations													
DNP3	5	4	5	5	2	1	3	2	3	4	2	**36**	**********
IEC 60870-5-104	5	4	5	4	2	1	3	2	3	5	2	**36**	**********
IEC 61850	5	3	2	5	3	1	3	4	5	5	1	**37**	**********
IEC 61968/61970	5	3	2	4	2	1	2	5	5	5	1	**35**	**********
IEC 60870-6 TASE.2	5	4	5	4	2	1	2	2	2	5	1	**33**	**********
Consumer Application													
ANSI/IEEE C12	5	4	3	2	2	1	4	4	3	5	5	**38**	**********
DLMS/COSEM	5	4	3	5	3	1	4	4	3	5	5	**42**	**********
BACnet	4	4	4	5	2	1	2	2	2	4	5	**35**	**********
EIA 709 (LONWorks)	4	2	3	5	2	2	3	2	3	3	5	**34**	**********
Konnex (EN 50090)	4	4	3	5	1	4	2	3	5	3	5	**39**	**********

Figure 6-3 Example: Customer Portal Technology Ratings

6.2.7 Criteria for Evaluation

Level of Standardization

Technologies used for consumer portals should be both open and standardized. Standardization refers to how well-defined the technology is, and how well it is recognized by its potential user community as a viable alternative.

Level of Openness

The term openness indicates a measure of how easy it is to obtain and use the technology. It may be well-defined (high level of standardization) and widely used (high level of adoption), but still not be open, if it can only be used through a license agreement with a particular vendor.

Openness is important for consumer portal technologies because it will reduce barriers for new vendors to enter the market, and therefore help to create economies of scale.

Level of Adoption

Regardless of how well-defined, recognized or open a standard is, the primary measure of its success is how widely it is used. The Internet standards, for instance, are not International Standards recognized by ISO or the IEC; however, they are some of the most widely used communications protocols in the world.

It is important that technology chosen for use in a consumer portal be widely used, because a large user base ensures:

- **Usefulness.** Problems with the technology will be more quickly identified and corrected.
- **Stability.** It will be more likely to evolve rather than become obsolete (e.g. Ethernet)
- **Longevity.** More people will have a stake in its continued use.

These factors will make it less likely that a wholesale upgrade of consumer portals will be necessary in the future.

Level of Users' Group Support

A contributing factor to the acceptance of any technology is whether a users' group exists to help maintain it. Some standards end up being abandoned after they have been published because the standards organizations that created them are not designed

to help them thrive. A users' group usually provides one or more of the following services that help the promotion and continuation of the technology:

Web Site Up-to-Date List of Members/Vendors/Users Frequently-Asked-Questions (FAQ) List Access to the Specifications Discussion Forum Mailing List Trade Show Booths	Help Line Newsletter Conformance Test Procedures Conformance Test Labs or Lab Certification Program Quality Program Vendor Advertisements

All of these measures help prevent the technology from becoming obsolete and reduce the cost of implementation. Technologies with existing users' groups are therefore preferred for consumer portals.

Security

Security is vital to consumer portals, for two main reasons:

- **Confidentiality.** Consumer portals by definition must deal with customer-specific data, perhaps including personal information, billing costs, and behavior patterns. It is important this information be kept confidential.
- **Access Control and Authorization.** Consumer portals may also control customer equipment, and it is important that such control be performed only by those authorized to do so.
- **Non-Repudiation.** It is important that certain transactions, such as a consumer choosing to opt in to a particular tariff, be recorded so they cannot be denied later.

These are only a few of the major security threats to consumer portals. All of these concerns are complicated by the fact that many of the candidate technologies are wireless and therefore easy for attackers to access without being traced.

All technologies used in consumer portals should therefore be easy to integrate with common security technologies.

Security is part of a larger aspect of systems known as **trust**. The measure of trust in a system includes, in addition to the quality of its security, its level of integrity (reliability) and its performance.

The security industry is unique in that it is constantly "fighting a war" in which "locks" are continually being upgraded to protect against new types of "lock picks". In addition to the other criteria discussed in this chapter, portal security technologies should therefore also be:

- Easy to upgrade with new algorithms, key sizes, and credentials
- Well-reviewed, accepted, and monitored by cryptographic experts
- Able to negotiate alternate parameters and choices
- Configurable to match users' security policies

Manageability

To be cost-effective, a network of consumer portals must be remotely managed. In this case, "managed" means being able to perform the following operations on any portal from a central site:

- Enable or disable the device
- Enable or disable particular communications links
- Enable or disable spontaneous alarm reporting
- Change communications configuration parameters, such as addresses, routing choices, buffer sizes, window sizes, transmission rates, and security credentials
- Gather operational statistics
- Upload or download software or firmware
- Synchronize the time at the portal

These are management functions only; there may be many other similar functions a portal must support in order to perform its duties, such as downloading new metering tariffs.

A key factor here is the **upgradeability** of each consumer portal. In a consumer portal environment, performing any of the functions listed above either locally or manually will be prohibitive. It must be possible to upgrade consumer portals remotely because of:

- The sheer number of devices that must be accessed
- The distance they will be located from the utility operations center

- Changing technology that would otherwise cause "stranded assets".

All technologies used in consumer portals should therefore be easy to co-exist with, and integrate with, common network management techniques.

Scalability

To be successful, consumer portals must be deployed in millions of consumer sites. Therefore, any technology chosen must be scaleable and cost-effective for a large number of devices.

Scalability is a particular concern in the case of the wide-area network technologies discussed. The number of addresses required to specify individual portals can be a challenge for some technologies.

Another important factor is the scalability of network management and security. Software, firmware, passwords and other security credentials may need to be downloaded to millions of devices. Technologies that support mass management should be encouraged for use with portals.

Use of Object Modeling

It is a key principle of the IntelliGrid Architecture that all utility application-layer technologies should be object-oriented. In other words, all the data transferred in utility networks should be:

- Organized into standard logical groupings, usually called objects. Objects are abstract representations of real-world functions and processes – in this case, the power system.
- Accessed using a standard (ideally human-readable) naming convention
- Arranged in a hierarchy that permits clients to perform operations on subsets of the data
- Associated as a group with standard functions or services (often called methods)
- Expandable to include vendor-specific or proprietary data
- Reducible to a standard minimum subset
- Object modeling makes it possible to configure and manage the enormous amount of data provided by a network of consumer portals.

Use of Self-Description and Meta-data

Another two related IntelliGrid Architecture principles that must be applied to consumer portals are the use of self-description and meta-data.

Self-description is the ability for a server device, such as a portal, to describe itself to a client, such as a master station or data concentrator. The server informs the client what data it has available, what format the data is in, and how to access the data. Self-description is fairly common in commercial computing applications, and makes possible what is generically known as "plug and play". In the utility industry, this information has typically been manually entered.

Self-description reduces the cost of deploying consumer portals by:

- Reducing labor costs during installation and configuration by automating a human process.
- Reducing errors in configuration due to memory errors or mistyping.
- Reducing the amount of testing that must be performed on communications paths to correct human errors.
- Self-description may be performed either online or offline. When it is done online, the information is transferred within the protocol stream, usually at initialization time. When it is performed offline, the vendor or user of the server device provides a file in a standard format that describes the device. Offline self-description tends to be preferred because of the delay that online self-description may cause at start-up.
- "Meta-data" is a term literally meaning "information about data". Meta-data includes self-description as well as other ways to organize information so it is easily identifiable and human readable, such as document markup languages (e.g. HTML, XML). This type of technology is necessary to achieve the scale of deployment needed for consumer portals.

Applicability to the Power Industry

Many technologies were eliminated from consideration for use in portals because they were too specific to particular industries. There may be several protocols, for instance, that could work technically but use object models and functions that are too specific to industrial automation.

Other technologies were too generic depending on where they were applied. XML, for instance, is an excellent generic presentation technology, but would have to be adapted to the power industry with a specific schema, as in IEC 61850-6 Substation Configuration Language.

Applicability to the Consumer Domain

Most of these technologies were selected because they addressed at least some part of the consumer domain. The wide area networks are obviously included because there is a need to reach remote customer sites.

Several of the technologies discussed here are applicable to the power industry but are not specific to the consumer domain. These technologies should be considered for consumer portals because of:

- Their ability to seamlessly link consumer portals to other parts of the power system, like transmission or distribution substations, or energy management systems.
- Their ability to be tailored to the consumer environment. For instance, IEC 61850 and IEC 61970 permit additions to their object models.

APPENDICES

Appendix A IntelliGrid Case Studies and Examples

The IntelliGrid Architecture is a process that utilities can use when designing communications-based systems on the transmission and distribution systems as well as on the customer side. There are several high-level concepts that are promoted in the IntelliGrid Architecture:

- Integration of systems
- Use of standards-based open systems that interoperate
- Definition of applications and requirements
- Mapping technology solutions to requirements

The IntelliGrid Architecture has developed several tools that can used when designing communications-related systems:

- Templates for capturing and defining requirements
- Recommendations of standards and technologies to use
- Strategies for building security into systems
- Strategies for migrating to open, standards-based systems and integrating new open, standards-based systems with existing systems
- Strategies for developing "layered" solutions that minimize the impact of changing technologies in the future

An application of the IntelliGrid Architecture is the use of any or all of the above tools and high-level concepts. Several IntelliGrid Partners have started projects that implement IntelliGrid Architecture's tools and concepts. These projects include:

- **TXU:** Advanced Metering Infrastructure Development
- **SRP:** Substation LAN Deployment and Equipment Monitoring
- **LIPA:** Utility and Consumer Device SCADA via BPL and Wireless Communications
- **CEC:** Demand Responsive Infrastructure Reference Design
- **SCE:** Advanced Metering Infrastructure Development and Deployment
- **PPGC-O:** Field Device Communications, Phasor Measurement, and FSM
- **Alliant Energy:** Distribution Monitoring System Replacement

These projects were chosen for initial application of IntelliGrid because they share many of the following attributes:

- Clear problem definition
- Not too simplistic, not too complex
- Implement one or more IntelliGrid principles and/or technologies
- Well defined project timeline
- Buy in from staff across the enterprise
- Plausible value story
- Measurable results during and at end of project
- Ability to clearly transfer the results to other IntelliGrid partners and industry

The IntelliGrid project team has the following roles and responsibilities for these implementation projects:

- Act as coaches and teach the IntelliGrid fundamentals
- Empower IntelliGrid partner staff to implement IntelliGrid on their own to the greatest extent possible
- Review and critique IntelliGrid partner deliverables at various steps throughout the process
- Document the entire process
- Sanitize and transform into case study format
- Provide technology transfer to other IntelliGrid partners and industry

It is important to note that these early implementations serve as a laboratory for testing and improving the IntelliGrid Architecture and IntelliGrid Technologies. They allow us to learn how to apply the IntelliGrid requirements gathering and use case development process. These projects also allow us to develop practical methods of mapping requirements to technologies and to develop refinements to the architecture itself. Another important aspect of these projects is that they facilitate the development of case studies to improve our ability to tell the IntelliGrid story.

As we work through these implementations, the IntelliGrid team will be documenting the lessons learned and finding effective ways to transfer the knowledge gained to the

rest of the IntelliGrid partners and industry. The following sections provide some additional detail for each project.

A.1 TXU: Advanced Metering Infrastructure Development

A.1.1 Problem Statements:

- TXU is implementing an AMR system which will entail replacing 3 million meters
- TXU wants to utilize the data for several enterprise applications in addition to the metering related functions
- TXU needs to choose a system integrator, middleware provider, and application provider to facilitate implementing these functions

A.1.2 Project Goals and Objectives:

- Develop a way to integrate the data from one or more AMR provider's proprietary system
- Initiate a migration to integrate future equipment and applications through open systems
- Complete the core system integration implementation before the end of 2005

A.1.3 Benefits to IntelliGrid Partners:

- Develop a vendor evaluation methodology that addresses the vendors support of open systems and applicable standards
- Develop the means to integrate common proprietary AMR systems into IEC 61970 (CIM) compatible containers for sharing over an information bus
- Use IntelliGrid Templates for initial and ongoing requirements elicitation
- Transfer this process to TXU system integrators

A.1.4 Approach:

- Task 1: High level assessment of the TXU process to implement an AMR system
- Task 2: Review of Vendor Proposals / Presentations
- Task 3: Facilitate proprietary AMR system interface specification and design
- Task 4: Review overall AMR information architecture design
- Task 5: Training

- Task 6: Documentation
- Task 7: Provide telephone and e-mail support

A.1.5 Status:

Tasks 1 and 2 complete

A.1.6 Next Steps:

Determine if task plan needs to be modified based on selection of vendor

A.2 SRP: Substation LAN Deployment and Equipment Monitoring

A.2.1 Problem Statements:

- A cost efficient LAN network is needed to provide near real time information for improved decision making to various users corporate wide about these important substation assets.
- Need to increase reliability indices (CAIDI, SAIFI, CAIFI) and improve maintenance planning and utilization of assets
- Need to improve response time to outages by obtaining quicker access to more intelligent data.

A.2.2 Project Goals and Objectives:

- Develop a way for SRP to integrate field data and make it available for sharing across the enterprise
- Improve protection monitoring of system faults, condition monitoring of major substation assets and supervision of IEDs.
- Initiate a migration to integrate future field equipment through open systems

A.2.3 Project Scope:

The scope of this demonstration is limited to a test implementation in the SRP Browning Substation

A.2.4 Benefits to IntelliGrid Partners:

- Develop the means to map DNP3 and other existing field device protocols into IEC 61850 object models.
- Develop the means to map field measurements in 61850 object format into IEC 61968/61970 (CIM) containers from SRP field devices for sharing over an information bus
- Develop the approach to applying the IEC 61850-6 Substation Configuration Language (SCL) for documenting and integrating these systems
- Use IntelliGrid Templates for requirements elicitation
- Possible mapping of SCL to UML model

A.2.5 Approach:

- SRP will develop requirements for usage of new substation LAN and field device data
- IntelliGrid team will assist SRP in mapping field device attributes to 61850 object model and document the mapping using SCL

A.2.6 Status:

Requirements development (use cases) in progress

A.2.7 Next Steps:

IntelliGrid team to review draft use cases

A.3 LIPA: Utility and Consumer Device SCADA via BPL and Wireless Communications

A.3.1 Project Purpose:

- This project is designed to demonstrate several of the essential components for the development of the self healing grid of the future.
- It will deploy devices and demonstrate communications technologies that are essential milestones for the future of the electric utility grid.
- Data coming back to the company systems (DO, billing, etc.) will come be connected through the UIB (Utility Information Bus).

A.3.2 Project Goals and Objectives:

- Install and test operations devices on the distribution system connected via BPL (Broadband over Power Line) and/or wireless communications to the Distribution Operator (DO).
- Test AMR (Automated Meter Reading) and other customer devices connected via BPL and/or wireless communications to the appropriate company system.
- Demonstrate internet and VoIP (Voice over Internet Protocol) telephony via BPL and/or wireless.

A.3.3 Project Scope:

The scope of this demonstration is limited to a test implementation within the Hauppauge Industrial park and an adjoining residential area.

A.3.4 Benefits to IntelliGrid Partners:

- Experiment with two core customer and device communication technologies and determine their applicability
- Determine challenges to mapping field device communications into enterprise information systems
- Develop methods to determine how to practically apply the IntelliGrid requirements gathering and use case development process
- Develop template text for inclusion in RFQ's that indicate expectations of vendors relative to IntelliGrid Architecture compatibility

A.3.5 Approach:

- Four parallel initiatives
    - Substation / Industrial Park Feeders
    - Industrial Park Customers
    - Residential Feeders
    - Residential Customers
- Compare and contrast BPL and WiMAX
- Implement AMR, VoIP, condition monitoring, remote cut-on/cut-off, general control, and related applications

A.3.6 Status:

IED procurement specification for IntelliGrid Compatibility completed and will be included in RFPs

A.3.7 Next Steps:

IntelliGrid team to meet with LIPA team to help develop and review use cases

A.4 California Energy Commission (CEC): Demand Responsive Infrastructure Reference Design

A.4.1 Problem Statements:

- The CEC noticed that the California IOU's were proposing multiple, proprietary, non-interoperable systems to facilitate implementing demand responsive systems
- The CEC would like the state to avoid implementing systems that quickly become stranded assets as has occurred in past attempts at implementing demand response systems

A.4.2 Project Goals and Objectives:

- Develop a high level reference design that the IOU's and vendors can follow that will foster interoperability and sustainable system development and evolution
- Initiate a migration to integrate future equipment and applications through open systems
- Begin the technology transfer process through workshops and other forums

A.4.3 Benefits to IntelliGrid Partners:

- Determine how the IntelliGrid Architecture can be used to create a more concrete reference design
- Develop a set of core principles applicable to AMI and demand responsive infrastructure
- IntelliGrid Architecture Methods Development Goals
- Use IntelliGrid stakeholder engagement process for requirements elicitation
- Develop tech transfer methodology

A.4.4 Approach:

- Develop a high level reference design based on IntelliGrid Architecture concepts

- Do this through stakeholder engagement
- Rely on industry to work out the details

A.4.5 Status:

Project complete

A.4.6 Next Steps:

- Facilitating technology transfer to OpenAMI
- Begin scoping out thermostat reference design in the context of the IntelliGrid Portal requirements and overarching reference design

A.5 Southern California Edison: Advanced Metering Infrastructure (AMI) Project

This project is the largest and most far reaching application of the IntelliGrid Architecture.

A.5.1 Problem Statements:

- Develop the Business Case for deploy an Advanced Metering Infrastructure (AMI) program, replacing 5 million customer electric and gas meters. Identify benefits throughout the SCE organization.

A.5.2 Project Goals and Objectives:

- Develop a AMI conceptual architecture and design that the SCE and their vendors to install 5 million meters
- Develop internal process and systems supporting of the new AMI systems
- Utilize the IntelliGrid Architecture requirements capture process and a systems engineering approach based on object modeling and the application of open standards for the design, specification, procurement, development
- Foster open systems interoperability and sustainable system development and evolution through OpenAMI and UtilityAMI

A.5.3 Benefits to IntelliGrid Partners:

- Determine how the IntelliGrid Architecture can be used to create a more concrete reference design

- Develop a set of core principles applicable to AMI and demand responsive infrastructure
- IntelliGrid Architecture Methods Development Goals
- Use IntelliGrid stakeholder engagement process for requirements elicitation
- Develop tech transfer methodology

A.5.4 Approach:

- Develop a high level reference design based on IntelliGrid Architecture concepts
- Do this through stakeholder engagement
- Open documentation process @ www.sce.com/ami

A.5.5 Status:

- This project is the largest and most far reaching application of the IntelliGrid Architecture.
- Developed 18 separate Use Cases and supporting functional and non-functional requirements
- Created IntelliGrid based AMI Conceptual Architecture and Design
- Released IntelliGrid based Request for Information for status of vendor proposal solicitation

A.5.6 Next Steps:

- Track vendor proposals for best fit to IntelliGrid generated requirements and design
- Begin scoping out implementation and rollout of AMI system inside SCE

A.6 PPGC-O: Field Device Communications, Phasor Measurement, and Fast Simulation & Modeling

This project is in the formative stages. The project involves interacting with three other projects – Phasor Measurement, PQ Assessment, and 61850 field device communications. Core goals and objectives include:

- Re-engage the IntelliGrid Architecture process with the T-FSM project
- Capture the relevant use cases
- Perform the appropriate analysis to identify potential technologies

- Test out candidate technologies virtually using super computer based simulation
- Deploy candidate solutions at PPGC-O that have high potential value, can be concretely validated for effectiveness, have a high probability of success, and can be implemented in a reasonable time frame and budget.

A.7 Alliant Energy: Distribution Monitoring System

This project utilized IntelliGrid to evaluate replacement of their distribution monitoring and control system. This project produced a technology assessment and mapping of equipment offerings capable of meeting the Alliant distribution monitoring system requirements using commercially available offerings that comply with IntelliGrid design principles.

Appendix B IntelliGrid Architecture Resources

IntelliGrid Architecture Systems Overview*

The IntelliGrid Architecture Final Report
Guidelines for Assisting the Understanding and Use of IntelliGrid Architecture
Recommendations - Distribution and Transmission Operations, Volume II
What Constitutes an IntelliGrid Application?

How to Implement IntelliGrid Projects

Template SOW for IntelliGrid demonstration projects
IntelliGrid Use Case Template
AMI Vendor Selection Score Sheets
IntelliGrid UML model in XMI format

IntelliGrid Requirements Capture and Use Cases

IntelliGrid Use Case Preparation Guidelines
Hitchhiker's Guide to Finding Requirements with Use Cases

IntelliGrid Projects and Examples

Early Applications of the IntelliGrid Architecture
What Constitutes an IntelliGrid Application?
SCE's Advanced Metering Infrastructure (AMI) System Engineering Approach
SCE Advanced Metering Infrastructure (AMI) Systems Engineering Overview, PowerPoint Presentation
SCE Advanced Metering Infrastructure (AMI) Conceptual Architecture
The Case for Use Cases - Salt River Project, Smart Grid Newsletter Case Study
http://www.smartgridnews.com/artman/publish/article_176.html
Distribution System Monitoring Replacement at Alliant Energy, Smart Grid Newsletter Case Study
http://www.smartgridnews.com/artman/publish/article_162.html

*** NOTE: All documents, presentations, and publications can be found at –**

http://publish.intelligrid.info/

Publications

An Integrated Electric Communications System Architecture – Power & Energy Continuity
Toward a Smart Electric Grid – Electric Light and Power
Advanced Metering at Southern California Edison – Implementing the Vision Using A Modern Systems Engineering Approach, T & D World

IntelliGrid Technology Assessments

A Guide to Existing Communications, Information, and Systems IntelliGrid Architecture Technologies
IntelliGrid Customer Portal Telecom Assessment
IntelliGrid Telecommunication Technology Assessment for Alliant Energy
IntelliGrid Compliance from an IED Procurement Point of View White Paper
Energy Service Portal Development – Assessment and Recommendations, EPRI/CEIDS December 2003
The Integrated Energy and Communications Systems Architecture, Volume IV: Technical Analysis, EPRI/CEIDS July 2004
A Strawman Reference Design for Demand Response Information Exchange, EnerNex Corporation for California Energy Commission, October 2004

Web site / services

http://www.intelligrid.info/
http://publish.intelligrid.info/
http://www.epri.com/intelligrid/
http://www.californiademandresponse.org/doc/ReferenceDesign.pdf
http://www.sce.com/PowerandEnvironment/ami/

Appendix C IntelliGrid IECSA Vol 4: Technical Analysis; Appendix D Technologies, Services, and Best Practices; Table of Contents

---

Table of Contents

Export Control Restrictions

Access to and use of EPRI Intellectual Property is granted with the specific understanding and requirement that responsibility for ensuring full compliance with all applicable U.S. and foreign export laws and regulations is being undertaken by you and your company. This includes an obligation to ensure that any individual receiving access hereunder who is not a U.S. citizen or permanent U.S. resident is permitted access under applicable U.S. and foreign export laws and regulations. In the event you are uncertain whether you or your company may lawfully obtain access to this EPRI Intellectual Property, you acknowledge that it is your obligation to consult with your company's legal counsel to determine whether this access is lawful. Although EPRI may make available on a case-by-case basis an informal assessment of the applicable U.S. export classification for specific EPRI Intellectual Property, you and your company acknowledge that this assessment is solely for informational purposes and not for reliance purposes. You and your company acknowledge that it is still the obligation of you and your company to make your own assessment of the applicable U.S. export classification and ensure compliance accordingly. You and your company understand and acknowledge your obligations to make a prompt report to EPRI and the appropriate authorities regarding any access to or use of EPRI Intellectual Property hereunder that may be in violation of applicable U.S. or foreign export laws or regulations.

The Electric Power Research Institute (EPRI)

The Electric Power Research Institute (EPRI), with major locations in Palo Alto, California, and Charlotte, North Carolina, was established in 1973 as an independent, nonprofit center for public interest energy and environmental research. EPRI brings together members, participants, the Institute's scientists and engineers, and other leading experts to work collaboratively on solutions to the challenges of electric power. These solutions span nearly every area of electricity generation, delivery, and use, including health, safety, and environment. EPRI's members represent over 90% of the electricity generated in the United States. International participation represents nearly 15% of EPRI's total research, development, and demonstration program.

Together…Shaping the Future of Electricity

Printed on recycled paper in the United States of America

1013610

ELECTRIC POWER RESEARCH INSTITUTE
3420 Hillview Avenue, Palo Alto, California 94304-1388 • PO Box 10412, Palo Alto, California 94303-0813 • USA
800.313.3774 • 650.855.2121 • askepri@epri.com • www.epri.com