Integration Roadmap for Smart Grid:
From Accidental Architecture to Smart Grid Architecture

Tony Giroti

Chairman and Managing Director

BRIDGE Energy Group, Inc.
313 Boston Post Road, Marlboro, MA

TGiroti@BridgeEnergyGroup.com

Keywords: Integration Architecture, Application Integration, Service Oriented Architecture, Data Volume, Smart Grid Reference Architecture.

Abstract

The core business of energy generation, transmission and distribution has been operated and managed by non-IT systems for years. IT has only had modest role in energy operations. With Smart Grid, the paradigm is about to change. The ambitious objectives of Smart Grid, when combined with some early warning signs from those who‘ve embarked on the journey, indicate that the role and complexity of IT is being grossly under-estimated, and that IT is going to play a more prominent, if not dominant, role in making Smart Grid a reality. The Power industry needs to take a careful, hard look at these indicators, do appropriate course correction and reconcile with the role that IT will play in Smart Grid and Demand Response programs. IT will need to develop a Strategic ―Smart Grid Architecture” as opposed to what we call an ―Accidental Architecture”. This paper discusses some of the IT-related challenges that need to be addressed to make Smart Grid programs successful. Gaps in the current approach are identified and an open, vendor independent, product agnostic and technology neutral Smart Grid Reference Architecture (SGRA) is proposed based upon the GridWise® Interoperability Context-Setting Framework. Finally, a roadmap for implementing an Integration Architecture is suggested in order to make Smart Grid and Demand Response a reality.

1. INTRODUCTION

1.1. Overview

The Power industry has traditionally been a laggard in adopting Information Technology, either because of a lack of funding or the absence of business drivers mandating the development of a strategic IT architecture. The motivation for creating a strategic IT architecture (also known as an Enterprise Architecture) has not been compelling to date. To realize the vision of Smart Grid and Demand Response (SG & DR) however, Power companies will have to address, at a minimum, integration-related aspects of strategic enterprise architecture, to address IT challenges related to Interoperability with applications and systems (within and outside their organization), Integration complexity, Data Volume, Real-Time data needs, Event processing, Throughput, Performance, and Security. Furthermore, new business models such as PHEVs, Distributed Renewable Generation and new mandates such as *FERC Order 719* for load curtailment, impose even greater architectural demands around interoperability, application & data integration, IT governance, security and data management. There is enough empirical evidence from organizations implementing SG programs to support the concern that systemic problems within today‘s IT could seriously derail many Smart Grid programs before they get off the ground. Therefore, without a strategic vision, planning, and an architectural approach, Smart Grid and Demand Response programs will pose formidable challenges that cannot be fully solved.

1.2. Relation to GWAC Context-Setting Framework

The GridWise Architecture Council (GWAC) in its Gridwise Interoperability Context-Setting Framework [1] has introduced interoperability groups, categories and a classification of interoperability issues that cut across these layers. GWAC suggests that domain experts ―articulate the detailed nature of each issue areas in separate documents engaging interested experts in their creation”.

The foremost intent of this paper is to raise visibility to IT and Interoperability challenges and propose the importance of addressing them in a proactive and strategic manner in order for companies to successfully deliver SG & DR programs. Based upon GWAC‘s Framework, as illustrated in Figure 1, this paper proposes a *Model* or a Smart Grid Reference Architecture (SGRA) for the Energy Industry that specifically addresses the Integration and Interoperability challenges with respect to the integration issues identified in this document. **The proposed SGRA is independent of any product, technology and/or vendor bias** and could be used by Generation, Transmission, Distribution, Curtailment Service Providers, Coops, Market participants & aggregators, and other stakeholders in the Energy industry to set the foundation and launching of SG & DR programs.

Figure 1 also illustrates the approach to SGRA. At the very top, based upon the GWAC's Context setting framework, this paper identifies 16 major IT related integration challenge areas and 57 sub-areas in detail that have been mapped to the GWAC's three interoperability groups, eight categories and ten cross-cutting issues. A set of metrics and formulas such as the Integration Complexity, the Bridge Index, the Scope calculations due to the Ripple effect and others are proposed that enable stakeholders to quantify the depth of these challenges. An open Smart Grid Reference Architecture (SGRA) with architectural modules and capabilities is derived that is specifically targeted towards solving these challenges. And finally a template for the integration Roadmap is proposed that will enable stakeholders to leverage the SGRA and adopt it to their particular businesses and SG & DR programs.

1.3. Current State of Smart Grid and IT

IT as an afterthought - especially around application integration - continues to be the norm for majority of Smart Grid pilots across North America. For example, most utilities are focused exclusively on deploying smart-meters, communication infrastructure and Meter Data Management (MDM) products in their pilot phases and hardly any have included developing a *strategic* integration architecture that ties MDM data with other enterprise applications such as Outage Management System (OMS), Customer Information System (CIS), Geographical Information System (GIS), Distribution Management System (DMS), and Supervisory Control and Data Acquisition (SCADA). Additionally, the popular approach of connecting MDM with CIS in a point-to-point manner may work for low volume and low transaction pilots but may not scale to production quality volumes and bi-directional communication models. Moreover, if the CIS is ever to be replaced, the MDM integration with CIS will require redesign and rework. In the absence of a strategic IT approach, current integration practices provide little value to achieving the larger SG & DR objectives from an IT perspective. In fact, some early warning signs from those who have embarked on the journey indicate that the integration complexity has been grossly under-estimated and can no longer be ignored. Many projects have been delayed due to the technical challenges brought about by the lack of strategic IT planning.

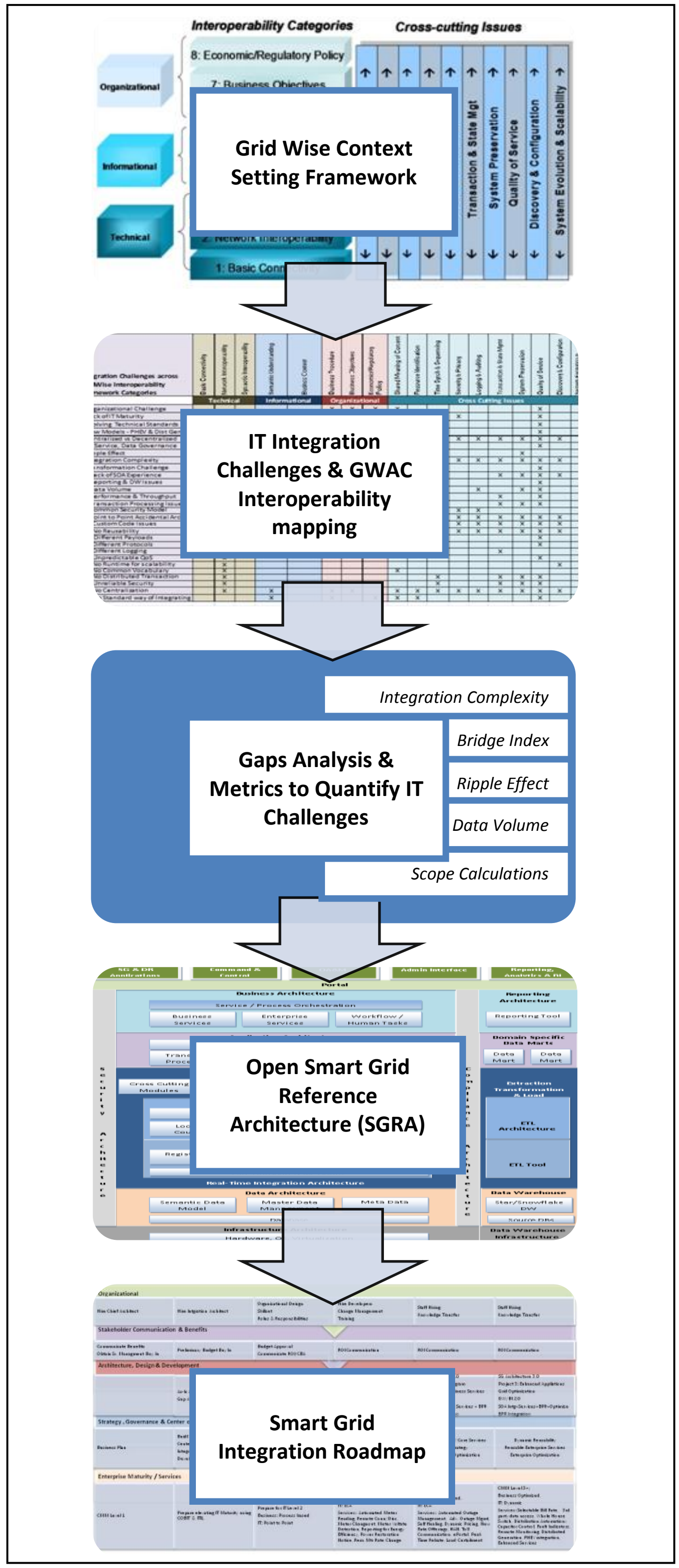


Figure 1: IT Challenges & Smart Grid Reference Architecture per GWAC Context-setting framework

Although the current tactical approach may have served companies well in the past, it will certainly not scale to support the larger vision of Smart Grid and Demand Response. Success or failure could rely squarely on the approach to solving the core IT challenges. Again, success will require a strategic rather than a tactical approach because the stakes are high as the Power industry launches new Smart Grid programs.

1.4. The Choices facing us

The Electric Power industry has two choices: One, to be proactive and have a strategy for managing Grid Operations & IT transformation through a strategic ―Smart Grid integration architecture”, or two, be reactive and tactical in responding to problems as they appear. The latter approach is risky and will prove to be a major impediment to Smart Grid success. Integration has to be one of the top priorities for companies executing SG & DR programs.

2. CURRENT STATE - THE ACCIDENTAL ARCHITECTURE

IT and Power Systems Engineering (OT) applications have typically operated in silos due to the lack of any compelling need for integration – integration between business units, integration between business processes, integration between applications, and even integration of databases. Up until now, the industry has had minimal real-time integration capabilities built into IT systems because applications and data integration needs have been met tactically through a one-off and project-based approach. IT has never had the motivation, the business drivers or the budget to develop a strategic architecture or develop a standardized approach to integration. Application and Data integration requirements have been met through a tactical approach based upon any available technology or middleware offered by the application or system vendor. Available resources have had to develop quick *point-to-point* interfaces between applications to achieve near-term objectives. Each interface is non-standard and custom-coded. Many of these interfaces are batch rather than real-time, with database links and proprietary code that is customized by writing more code within the application.

A point-to-point integration approach is not scalable, precludes future upgrades, and increases risk to the organization, as any change to the application would have a **Ripple Effect** on other downstream applications. Although the custom interface meets short-term needs, it stifles future growth and scalability. The integration gap keeps widening over time with custom code written for each P2P interface. The viral impact of the point-to-point architecture continues to reduce the overall integration capability, making each change riskier than the one prior. Data continues to be locked in silos and sharing becomes a significant challenge over time. This growth over the years has resulted in what

Courtesy: A Utility - name intentionally not disclosed

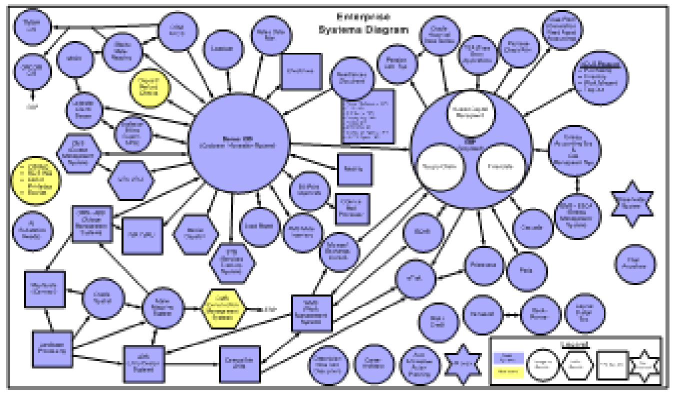


Figure 2: The "Accidental Architecture"

we refer to as an "**Accidental Architecture**". Figure 2 illustrates an example of an Accidental Architecture.

3. IT CHALLENGES

This section discusses Accidental Architecture and a few other IT related challenges that must be addressed as organizations launch SG & DR programs. Successfully tackling these challenges will enable organizations to clearly execute on their vision of developing a Real-Time Integration Architecture that will serve as a foundation for all SG & DR programs. This section describes 16 major IT challenge areas and 57 sub areas. Section 4 provides a summary of these IT challenges for quick reference.

Section 5 maps these challenges to the GWAC context-setting framework.

3.1. Point-to-Point Architecture Challenge

―Accidental Architectures” that have evolved over time are based upon a Point-to-Point (P2P) approach where applications communicate and ―talk” to one another directly through custom-code and without any intermediary. Figure 3 illustrates two such applications (Application A and Application B) that use custom-code to communicate with each other.

3.1.1. Purpose of P2P Custom Code

Custom-code is required to handle all aspects of communication between the two applications as they need

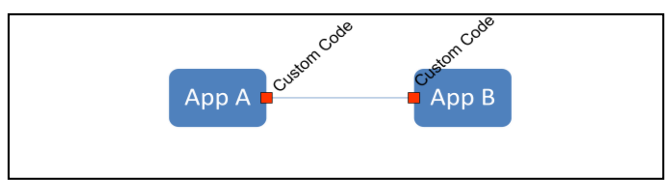


Figure 3: Point to Point Architecture

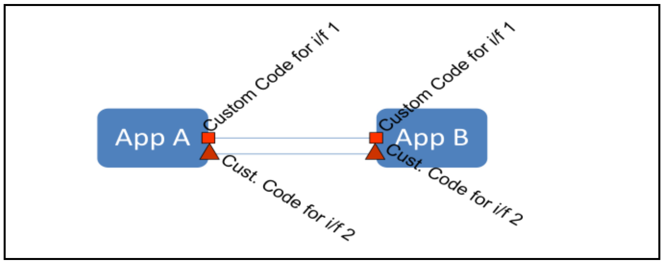


Figure 4: Two P2P Interfaces with Custom Code

to be ―application aware" of each other. Given the tight dependency between two applications participating in a transaction, a P2P interface is also referred to be a ―Tightly Coupled" interface.

Custom code is needed for each interface. For example, if there are two interfaces between two applications, then each application would require two sets of custom code to support two different interfaces. Figure 4 illustrates two custom code modules in each application – one per interface.

3.1.2. Anatomy of P2P Custom Code

P2P interface code for each interface is shown in Figure 5 and includes the following:

(a) The motivation & Business Process

Why the communication is needed. The two applications have custom code that is specific to each interface and how it will be used. Each interface is associated with one business process or one business objective. A different business process requiring similar data may be incapable of using the same interface.

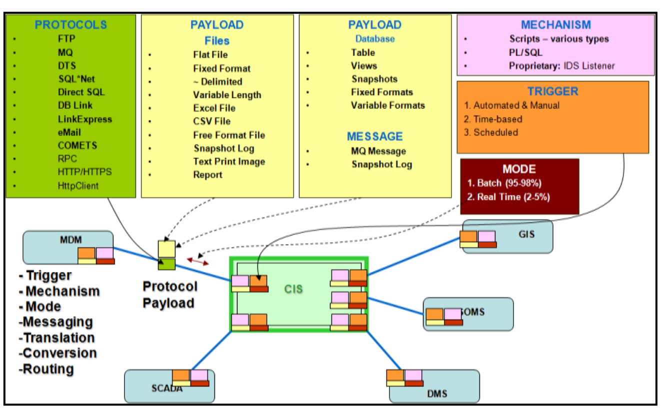


Figure 5: Anatomy of P2P Custom Code

(b) Payload & Message Handling

What data (payload) is being sent and in one of a number of formats. Each application must know what data and format will be used by the other application with which it integrates.

(c) Message Handling:

The sender then creates a message or a file in a particular format that is tailor made to the receiving application's expectations. The receiving application has custom code for each interface that deciphers the message or file and acts upon the data.

(d) Protocol and Communication Handling

How to communicate with the other application, e.g., real-time, batch, over FTP, UDP, TCP/IP, HTTP etc.

This is termed as ‗Communication Handling'. The custom code in each application has information whether it requires a message or a file for a particular interface (and it can be different for two interfaces between two applications). The sender has custom code to manage the sending of the message or file, and the receiving application has reciprocal code to receive the message and processes the inbound message or file. A trigger may invoke a transaction based upon a schedule, a particular time, a manual input or some other level of automation. Similarly, there can be a mode and a particular mechanism to facilitate the message. If a response or some form of acknowledgment is expected for each message sent and received successfully, then the receiver sends an acknowledgement or appropriate response to close the transaction. Additionally the custom code includes code for every interface that makes it ―aware" of the application it is interfacing with.

(e) Frequency

When and how often to communicate: Each of two applications has custom code that is aware of the frequency or timing of message delivery.

(f) Transaction Source and destination

Where is the request coming from and where is it going?

Who is the data intended for and how should the data be consumed?

(g) Error Handling

Each application may also have custom code to handle errors. Often, error handing is a key aspect of integration and it can frequently take more code to handle errors and exception than the actual interface or business logic that is the basis for the interface. Errors can occur during communication, by handling invalid or wrongly formatted data, and through predictable or unpredictable conditions

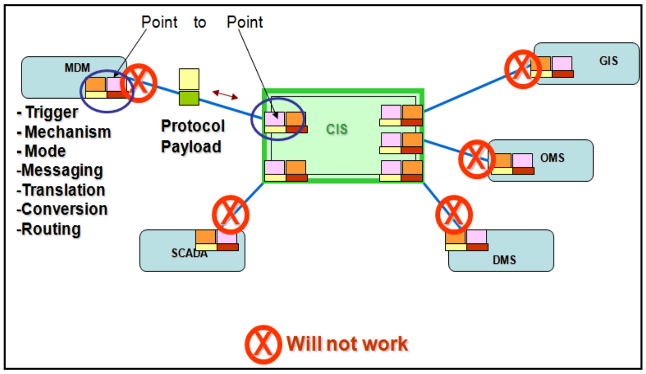


Figure 6: Removing an application will make all P2P custom code inoperable

(i.e. exceptions). Absence or weakness in any one area results in quality issues, making the interface error prone.

Every P2P interface requires this custom-code in both the sending and receiving application. As shown in Figure 6, if any application connected via the P2P custom code is removed, altered or upgraded – in this case a CIS application is to be upgraded - it impacts other ―adjacent" applications, and the custom code will stop working in those applications. If the CIS application is to be upgraded, the P2P interfaces between the CIS and MDM, SCADA, DMS, OMS and GIS will not work.

3.1.3. Anatomy of P2P Custom Code for Real-Time High Transactions Interfaces

Some P2P interfaces requiring real-time integration to support multiple transactions require a higher degree of error handling, reliability, scalability, and durability capabilities. This is because if there are more than two transactions using a particular interface, then each transaction would be made to be ―thread-safe" so that one transaction does not influence or change the context of the other transaction. Some arbitration logic is also developed to prioritize and serialize the transactions. Poorly written custom code can result in blocked transactions, bottlenecks, and latencies due to serialized transactions. Also, databases can get locked or take inordinate amounts of time for simple I/O, and transactions can get ―hang-up" or fail unpredictably.

3.1.4. Why P2P Custom Code is difficult to maintain?

P2P interfaces evolve over time. Most of the P2P interfaces are developed based upon the tactical needs of the project, the integration capabilities of the two applications and by using software and technologies available at the time. Furthermore, design of each interface is one-off and the development languages used may be different depending upon the skills of the developer(s). Over time, lack of standards and a common integration approach, will result in a myriad of custom and proprietary P2P interfaces. Many such interfaces are developed with little-to-no documentation. At times, the code for the interface is also unavailable and each interface may not conform to any standard or integration technology. The proprietary nature of each interface, combined with a lack of documentation adds to its fragility. The P2P interfaces are rarely modified once installed. Over time there is duplicate and redundant code with very little re-usability. Code is written and tested repeatedly for similar capabilities without any motivation for developing a common platform or a set of libraries that can be reused. Over time organizations learn to live with the way a certain interface operates and business processes evolve over time to embrace the idiosyncrasies of application interfaces. Applications in which these custom interfaces live become _un-upgradable', and over time the inertia leads to an **Accidental Architecture**.

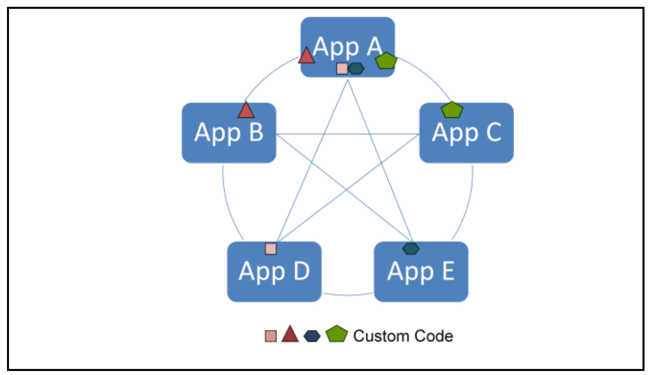


Figure 7: Custom Code per interface in each Application

3.1.5. P2P Custom Code Complexity

Figure 7 illustrates a P2P Architecture between five applications where each application interfaces with the other four applications in a point-to-point manner. In this example, application A integrates with four separate applications in four different ways through four different custom coded modules – one for each application interface. Each of the other applications (B, C, D and E) must also have a custom module to interface with Application A. If application A has two integration points with application B, two with application C and so on, then an IT group will need to write eight modules in application A to support two interfaces per application. Each of the other applications will also need to develop two custom modules per interface to integrate with application A. As is evident, custom code in applications could evolve over time to a point where application upgradability is cost prohibitive, resulting in highly customized applications and systems. If there is ever a need to upgrade such an application, then many downstream applications that are directly connected to that application in a P2P fashion will also be impacted.

3.1.6. Summary of P2P Sub challenges

- Custom code issues;
- No re-usability of any code;
- Different Payloads for integration;
- Different Protocols for integration;
- Different Logging mechanisms;
- Unpredictable QoS;
- No Runtime for Scalability;
- No Common Vocabulary;
- No Distributed Transaction capability;
- Unreliable security;
- No centralization;
- No standard way of integrating

3.2. Integration Complexity Challenge

With P2P architecture, the integration complexity will increase over time. Integration complexity for every organization is different and provides an indication of (a) the scope (investment required) and (b) if a strategic approach to integration is needed. We have described a means of calculating the Integration Complexity for any project, and an integration complexity index – known as the ***Bridge Index,*** will enable organizations to determine if a strategic integration architecture is needed. Armed with that data, organizations can plan their integration strategy as they launch their SG & DR programs. The following may be leveraged to calculate the Integration Complexity and the Bridge Index within an organization.

$$I_c = \sum_{k=1}^{n-1} k \quad [I_d \ . \ I_a]$$

$$N_m = 2.\ I_c$$

BI_c= *Bridge Index* based upon I_c (per Table 1)

Where

- $\boldsymbol{I_c}$ = *Integration Complexity*
- $\boldsymbol{k}$ = *Number of Nodes or Applications*
- $\boldsymbol{I_d}$ = *Integration Density as a Percentage. If every application has an interface with other applications then this is 100% .If every application interfaces with one in two applications then this is 50%. On an average, organizations could use 25% as an average number for their enterprise in general and 40 to 50% for a project with well defined application scope.*
- $\boldsymbol{I_a}$ = *Average Interfaces between Applications. Example, if there are two interfaces between two applications, then this is 2. In general two applications may have more than one integration point between them.*
- $\boldsymbol{N_m}$ = *Number of Integration modules / Custom Code that is required to support the Integration Complexity.*

The following Table illustrates a scale that can be used to measure the Integration Complexity across any organization. First calculate $\boldsymbol{I_c}$ to quantify your organization's Integration Complexity. Then, based upon the table below identify the Bridge Index with complexity as Low, Medium, High or Very High.

Table 1: Integration Complexity & the Bridge Index

Integration Complexity Range	BI_c The Bridge Index
I_c = **1 *to* 10**	Low
I_c = **11 *to* 50**	Medium
I_c = **51 *to* 200**	High
I_c = **201** & *higher*	Very High

For the illustration shown in Figure 7, the Integration Complexity and The Bridge Index can be calculated as:

k = 5 *Nodes or 5 applications*

$$\boldsymbol{I_c = (1 + 2 + 3 + 4)\ X\ 100\%\ \ X\ \ 4 = 40}$$

The Bridge Index with Integration Complexity I_c = 40 is <u>*Medium.*</u>

The Bridge Index = $\boldsymbol{BI_c}$**= Medium**

The higher the I_c the stronger the case for a strategic integration approach. Total number of Custom modules need to support Integration complexity of 40:

$$\boldsymbol{N_m = 2\ x\ 40 = 80}$$

If P2P architecture is leveraged then the total number of custom modules N_m that will be required to be developed is 80 to support the Integration Complexity of 40.

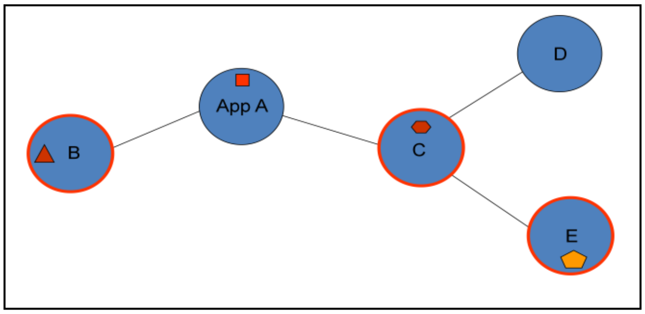


Figure 8: Applications Impacted due to Ripple Effect

3.3. The Ripple Effect Challenge

3.3.1. The Ripple Effect

Tightly coupled applications are at the core of the problem inherent in today‘s IT systems. P2P architecture is inflexible and difficult to undo. As shown in Figure 8, if application A requires an upgrade or needs to be modified, it impacts application B and application C because it is tightly coupled with these applications. If application C is changed, it will require changes to other downstream applications such as application E and perhaps application D. In other words, changes to one application may impact not only the most immediate applications but also the applications that are indirectly connected through second and third degrees of separation. This is termed as the ―Ripple Effect”.

3.3.2. The Ripple Effect Calculation

Applications within the ripple will be candidates for modification. The ripple can cause increase in scope, greater disruption to the business, higher risk, and higher overall cost. A more strategic approach would solve the underlying problems and contain the project scope by reducing the number of applications impacted by the ripple. An approach to containing the Ripple Effect is warranted in order to reduce the impact and scope to reasonable risk and tolerance levels.

Figure 9 shows the Ripple Effect that impacts applications in the first, second, and third ripple.

The applications impacted by the Integration Ripple Effect can be calculated based upon a ―Rule of Thumb” formula as follows:

$$I_R = (n \,.\, A_R) - 1$$

Where

- $\boldsymbol{I_R}$ = *Applications impacted by the Ripple Effect*

- $\boldsymbol{n} = 3$ *for Small IT organizations,* 4 *for medium IT shops and* 5 *for large IT shops*

- $\boldsymbol{A_R}$ = *Applications directly integrated with the scoped application or applications.*

For the illustration shown in Figure 8, considering a small IT shop with $\boldsymbol{A_R = 2}$ and $\boldsymbol{n = 3}$, the Integration Ripple Effect can be calculated to be 5, signifying that about 5 downstream applications can get impacted due to changes in one scoped application.

$$I_R = (3 \,.\, 2) - 1 = 5$$

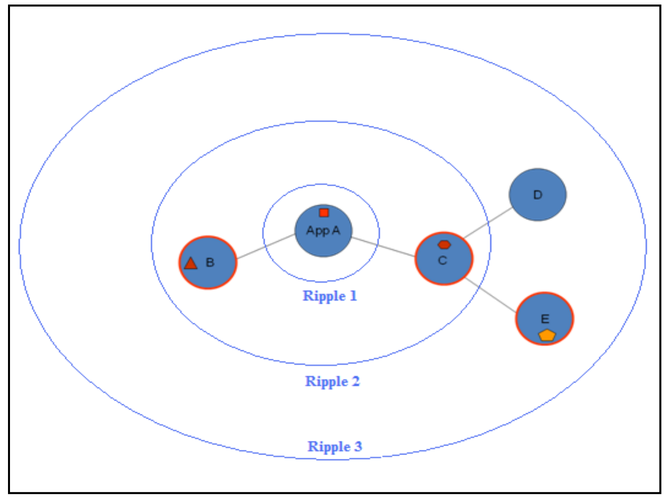


Figure 9: The Ripple Effect

3.3.3. The "indefinable" Scope

Sometimes, the seemingly simple task of defining the Scope of an integration project may be one marred with challenges. It may be difficult to define the applications that are within the scope and those that are not because the impact on downstream applications due to The Ripple Effect may mandate changes to many other applications beyond those originally scoped. Therefore, it may be prudent to (a) exercise caution and not neglect the impact that the Ripple Effect can have to downstream applications beyond those originally scoped and (b) it is important to have a strategy to contain the Ripple Effect in order to manage risk. The higher the Ripple Effect, the greater the risk and higher the business case an organization has for developing a Strategic Architecture.

3.3.4. Summary of Ripple Effect Sub challenges

- Impact on Downstream Applications unknown;
- Defining the Scope Issues.

3.4. The Data Volume Challenge

3.4.1. Data Volume Storage challenge

Some of the initial focus around Smart Grid and Demand Response has been in the areas of deploying Smart Meters and reading these through the Automated Meter Reading (AMR) and Automated Meter Infrastructure (AMI). Meter Data Management products and applications are being employed to collect the data for various Smart Grid initiatives such as Advanced Billing, Real-Time Pricing and managing grid reliability. These programs call for collecting huge volumes of meter data on a fifteen minute interval basis. For a million meters, this data amounts to roughly 1,111 TPS (transactions per second).

Transaction Volume $= \frac{1{,}000{,}000}{15\ X\ 60}\ \frac{transactions}{second} = 1{,}111$ tps

If each transaction is 1,000 bytes (1Kb) then 1Kb x 1111 transactions = 1,111 Kb are required per second. This is equal to 1Mb of data gathering and storage per second. Data Collected Per Hour = 1 MB x 60 x 60 = 3.6GB. This is equal to 85GB per day; 2.6TB per month; and 30 TB per year.

According to some recent data [8] from Austin Energy, their messages are between 4K and16K which means the storage capacity needed for a million meters is equal to 4 to 16 times the above numbers i.e. 340GB to 1.36TB per day; 10.4TB to 41.6 TB per month; and 120 TB to 480 TB per year of storage capacity.

Transactional data collected from customer meters can quickly reach staggering proportions that will require significant storage capacity and an information life cycle management approach to managing the data based upon some strategic approach where the value of data or at least that level of granularity, will gradually diminish over time.

3.4.2. The Data Volume sub-challenges

- Storing Large Volumes of Data;
- No Information Life Cycle Management;
- No Storage Strategy.

3.5. The Performance & Throughput Challenge

3.5.1. Transaction Performance Challenge

Transaction performance is critical to the success of any system. Many SG & DR projects are hitting performance bottlenecks due to architectural constrains. Energy companies might consider the *TPC-APP Benchmark*™ [6] as a way to measure their application performance. TPC-C is a transaction processing benchmark that can be used to do performance related planning that may be required to manage the transaction load and throughput.

Consider an AMI/AMR project that requires collection of data from a million smart-meters at 15 minute interval. Per earlier section:

The Transaction volume $= \frac{1{,}000{,}000}{15\ X\ 60}\ \frac{transactions}{second} = 1{,}111$ tps

This is equal to over 90 million transactions per day. The sheer handling of such transaction load may be a significant challenge and will require careful planning and selection of the appropriate communication technologies and MDM vendors. In addition to collecting the data, an organization will need to manage performance and storage challenges.

3.5.2. Data Transformation Performance Challenge

Each transaction may or may not require Transformation depending upon the meters. If Transformation of data is required for half the transactions then you will have 550 data transformation transactions per second. This includes an additional load of 550 TPS. This is equal to 45 million transactions per day. Additionally, a significant source of bottlenecks is the transformation of XML docs from one format to another if a SOA strategy is employed.

3.5.3. Event Handling Performance Challenge

A Complex Event Processing infrastructure is required to detect system and business events. This infrastructure will need to detect events ‗just-in-time‘. With over 90 million records, the detection of a ―needle in a haystack" must work day after day, month after month, with little to no room for error.

If any of the transactions is a ‗last gasp‘, then such events will require tracking and action. One could assume that there may be 0.01% chance or 1 in every 10,000 meters that may send a last-gasp every day. As a result there may be about 100 last-gasp messages per day that require a business action like automated self-healing or a work order creation and crew-dispatch. Either way, such a transaction needs to be processed when it occurs.

3.5.4. Manual (Human Task) Intervention Challenge

If 0.1% of the transactions ‗error-out‘, then we will have 1 transaction every second (or 60 every minute, or 360 every hour) that will require manual or some type of ―Human

Task Interface". Manual intervention needs to be kept to a minimum.

3.5.5. Database Performance Challenge

A large volume of transactions will need to be written into the database. At the rate of 1,000, about 90 million transactions may be written in a day. In some instances, to narrow down an outage, the last gasp meter data may need to be accessed from the ever growing transactional database (as shown in Data Volume section) resulting in significant performance bottlenecks in database I/O. In this example, if there is 30 days worth of data in the database, then database records that will have to be searched = 90 million x 30 days = 2.7 billion records. This may result in serious database performance issues. Optimizing the database indexes and parallelizing the databases will be a pre-requisite.

3.5.6. System Performance Challenge

A general practice is to add more CPU and hardware to solve a performance problem. Although a short term fix, performance problems will re-surface over and over again until the fundamental architecture is altered.

Storage Architecture and capacity planning around storage devices is required to minimize performance hits and to optimize disk I/O performance.

Transaction accountability requires tracking of each and every transaction all the way from its genesis to termination. A synchronous logging capability may choke a high transactional system. Alternative architectural approaches, such as Asynchronous Logging, may be leveraged in order to optimize performance. If each transaction is 1,000 bytes (1Kb) then 1Kb x 1111 transactions = 1,111 Kb per second are required. This is equal to 1Mb of logging data per second.

3.5.7. The Performance & Throughput challenges summary

- Performance issues related to Retrieving Data / Database Performance;
- Transaction Performance;
- Event Handling performance issues;
- Human Task challenges;
- System Performance.

3.6. Organizational Challenge

3.6.1. CIO, CTO and Budget challenge

In some organizations, the role of Chief Information Officer (CIO) and Chief Technology Officer (CTO) is either missing or its role diminished. Even in organizations with a CIO, IT management is not always represented at the executive level and therefore often does not get its ‗fair share' of budget or resources. This will have to change with SG & DR.

3.6.2. Perception that IT is not Strategic is a challenge

The Electric Power industry typically has been a laggard in investing in the IT organization, leveraging it based upon an ad-hoc and tactical versus a strategic approach. As a result, IT has often been working in a very reactive mode. In most organizations, IT has been relied upon merely to deliver the ―bare minimum" that the business requires. Consequently, many IT departments are missing an organizational structure that is capable of leading a rather large SG & DR initiative which requires strong technical leadership, discipline, standards, processes, methodologies and a governance framework.

3.6.3. Outsourcing challenges

Finally, many in our industry have outsourced the Architecture and Integration to overseas companies or large organizations without appropriate project controls and technical leadership. These projects can only be successful with appropriate project controls, technical leadership and project governance.

3.6.4. Organizational Challenges Summary

a. Missing CIO, CTO and Budget challenge;

b. Perception that IT is not Strategic is a challenge;

c. Outsourcing without controls challenge.

3.7. Service Oriented Architecture Challenge

To do or not to do SOA is the big question? Here are the core challenges based upon history and empirical data that need to be addressed.

3.7.1. SOA is a new paradigm within the Electric Power Industry

SOA is new to the Power industry. There have been numerous false starts in adopting SOA. Many of the early initiatives have failed on first try and are requiring a fresh start – primarily due to lack of SOA vision, strategy and experience. Some early adopters had wrongfully assumed that buying a SOA tool equates to *doing* SOA. This is wrong. Vision, strategy and technical leadership must precede SOA tool procurement.

3.7.2. SOA is being used in a limited way

Many whom have embarked on the SOA journey are using the SOA tool in a very limited capacity without unleashing the true value of SOA – which is developing services and building a loosely coupled architecture. Many have deployed the SOA tool as an Enterprise Service Bus (ESB) but are using only the transport capability of the ESB to merely send and receive messages, primarily through custom code that is based upon a P2P approach. A Loosely Coupled Architecture and appropriate deployment of the tool will alleviate these issues.

3.7.3. Tool Limitation or Missing Strategy?

Thirdly, there are others who have used the tool without success and who have relegated failure to limitations of the tool. Some in the latter group are even considering *another* SOA tool in the hope of generating success. The fact is that most tools have over 80-90% overlap around basic functionality. The SG & DR Program Managers should carefully evaluate (and, in some cases, re-evaluate) their approach, methodology, technical leadership, architecture and project plans. Compared to the Point to Point example shown in Figure 7, here are the overhead reductions with SOA. For a P2P 5-node architecture

$$I_c = 40\ \textit{Interfaces}$$
$$N_m = 80\ \textit{Custom Modules}$$

Compared to P2P architecture, a 5-node SOA, or 5 applications-based SOA architecture will require:

- *5 Application Adapters* (one for each application)
- *5 Integration Points*
- *One type of Protocol and One type of Payload*
- Furthermore the intermediary or the *ESB contains the "Run-time" generic code* to manage the scalability, reliability, maintainability, durability, and flexibility aspects of the interface. None of that exists in the application

3.7.4. SOA Challenges Summary

- SOA new to the industry;
- SOA tool being used in limited way;
- Lack of strategy and vision being perceived as tool limitation.

3.8. Transaction Processing Challenge

3.8.1. Limited experience with OLTP

Transaction Processing systems have rarely surfaced as a requirement in many of today's IT departments within the Electric Power industry. Albeit that Grid Operations and Market Operations have required transactional capabilities at the ISO / RTO level, the Generation, T&D, IOUs and Coops have had minimal business cases causing them to develop transactional systems. Transaction Processing Systems (also referred to as On Line Transaction Processing Systems, or OLTP) have not been pervasive in the Power Systems industry because, by definition, OLTP systems embody capabilities that leverage real-time data to make real-time decisions through real-time interaction with one or more (distributed) applications and one or more (distributed) databases. This has not been a requirement until now. (Just to dispel a misconception, an OLTP system is not purported to be an application or system for taking on line orders – though such an application does qualify as an OLTP system).

Why are transactional or OLTP systems important now?

- First, SG & DR programs will require a real-time architecture where disparate applications can interoperate with one another to achieve objectives of SG & DR, such as Self Healing, Load Curtailment, Air Condition Load management and others. These capabilities require applications like DMS, MDM, CIS, OMS, SCADA and others to interoperate without compromising their data integrity or their application performance.
- Secondly, the core focus of Transactional Systems is to minimize latency, maximize throughput and manage technical aspects of a transaction such as managing locking, logging, tracing, and recovery and to guarantee what is termed as the transaction properties – also referred to as the ―ACID" properties. These Transactional capabilities and ACID properties are essential to making Smart Grid and Demand Response a success.

Although sub-second response time is not required to achieve Smart Grid and Demand Response objectives, the tenets of Transactional Processing attributes, like reducing locking, managing large volume of transactions, guaranteeing transaction delivery, logging, recovery and ACID capabilities are required to achieve a Real-Time architecture and SG & DR objectives.

3.8.2. Transaction Processing Challenges Summary

- Distributed Transaction challenges;
- No Two-Phase commit and ―ACID‖ capabilities;
- No Transaction Processing Runtime or TP Monitor.

3.9. Security Challenge

3.9.1. No common security model

Security is a significant challenge in today's interconnected world of Power Systems and IT systems. Key safeguards that have already been mandated by NERC CIP 002-009 requirements and auditing guidelines have ensured safety of critical cyber assets and related technologies. As of this writing, a similar mandate is not yet in place for IT Security. This paper recommends FERC and NERC to provide formal guidelines to protect IT assets or expand the NERC CIP to accommodate the IT-portion of CCA requirements. A cyber-security coordination taskforce has recently released some guideline for the US Smart Grid that could serve as the basis for IT Security [3] [4] [5].

Given the critical role that IT systems will play in concert with SCADA, DMS and MDM applications, many of the decisions for SG & DR initiatives will originate from the programmable business rules and SG & DR applications resident within the IT realm. Transactions such as triggers to connect / disconnect a customer‘s Smart Meter could originate from the CIS application, perhaps based upon a change in customer status, or an outage pattern could be detected based upon consistent ―last-gasp" reads from a localized set of meters. This can result in initiation of self healing of the grid, or a critical peak condition could be detected resulting in automated Load Curtailment instructions to Curtailment Service Providers or a set of smart meters. The bottom line is that IT systems will be integral to increasing the reliability of the grid and empowering customers with new demand response programs. With these capabilities come the need for a higher degree of security and controls. A holistic approach to designing the security architecture will ensure that both threats and breaches can be eliminated.

3.9.2. Security Challenges Summary

- IT security equivalent to NERC CIP 002-009 missing;
- Undefined standards;
- SG & DR business logic to control grid, market, smart meters yet no comprehensive end-to-end Security model.

3.10. Technical Standards – Still evolving

3.10.1. Immature standards

The Electric Power Industry is one of the leaders in leveraging standards in the areas of Power Systems Engineering and more recently there has been significant effort in defining Smart Grid-related Standards. The Electric Power Research Institute (EPRI), in the summer of 2009, delivered its *Smart Grid Interoperability standards Roadmap* to the National Institute of Standards and Technology (NIST) [6].

Other standards, like IEC 61968, IEC 61970, CIM, MultiSpeak, are being developed or released. With so many emerging standards, there is also the issue of which will be adopted and which ones will not. The jury is still out on many of these standards and we recommend caution in leveraging them into your SG & DR programs.

Organizations should keep abreast of the work being done by FERC, which provides the overall policy, direction and market design; the North American Energy Standards Board (NAESB), which is accredited by American National Standards Institute (ANSI) that provides leadership on business practices; the North American Reliability Council (NERC) that offers standards for reliability; Independent Systems Operators (ISOs) and Regional Transmission Operators (RTOs), who are writing operating rules and procedures for software architectures, market interfaces, and security specifications; and Coops that are collaborating amongst themselves on local standards such as MultiSpeak. There are consortia and dozens of other standards bodies currently involved in fostering interoperability. Suffice it to say that there is a lot of activity in these areas and SG & DR program managers should exercise prudence in leveraging the appropriate standards into their programs.

3.10.2. Standard Challenges Summary

a. Standards are a Work in Progress;

b. There are Competing standards.

3.11. Lack of IT maturity

3.11.1. Motivation for IT maturity missing

IT in the Energy industry has been a laggard in adopting processes & methodologies for the various reasons mentioned earlier. SG & DR programs cannot succeed unless the maturity of IT is elevated in various disciplines. The corollary is also true in that the lack of such methodologies and artifacts is surely the cause of project failure and may hinder SG & DR success. This paper suggests that IT should consider leveraging industry standards, frameworks, methodologies, and best practices as they embark on the SG & DR journey. The industry should invest to elevate the maturity of IT processes to CMM level 2 or higher by adopting industry frameworks such as COBIT, ITIL or SEI CMM. For SG & DR, Level 2 should be the absolute minimum target. The range is as follows: Level 0: Non-existent; Level 1: Initial/*ad hoc*; Level 2: Repeatable but Intuitive; Level 3: Defined Process; Level 4: Managed and Measurable; Level 5: Optimized.

3.11.2. IT maturity Challenges Summary

a. Mostly at CMM Level 1 – not optimal for SG success;

b. Elevating to CMM Level 2+ will require serious work.

3.12. Centralized versus Federated Approach Challenge

Architectural design decisions require a careful analysis of evaluating the pros and cons of Centralized versus Federated approach to data management, development, testing, control and governance. In a centralized approach, major responsibility for a specific task resides within a team that is centralized. In a federated approach various distributed parties are responsible for their respective tasks with guidance from a centralized body. One approach is not better than the other and may differ from task to task. Every organization needs to determine the model that works for it. In some instances, a combination of a centralized and a federated approach may be applied. For example, to develop a data warehouse for analytics and compliance reporting, a centralized approach to developing an Enterprise Data Warehouse may be applied where data is sourced and consolidated from disparate systems such as meter readings from MDM, operational data from SCADA, customer data from CIS and outage data from OMS. However, a federated model could allow different business units in Customer Service and Operations to access their own dataset through Data-Marts.

With new business models such as the ones mandated by *FERC Order 719*, Curtailment Service Providers can bid into the open market for load curtailment services. In this model, each CSP would retain control of their own customers for one-on-one retail billing amongst their customers and yet do settlement at the wholesale level with the local ISO. In this model, a federated approach will enable each stakeholder such as the ISO, CSP, IOU and Coops to manage their own customer data and yet participate in wholesale curtailment settlement with the ISO.

3.12.1. Centralized vs Federated Challenges Summary

a. Holistic end to end business models need to be reviewed for the right approach;

b. Data management & ownership controls lacking and will be required for SG & DR;

c. New federated architecture is required.

3.13. PHEV, Distributed Generation and new Business Model Challenges

New business models are emerging that will tax the underlying IT systems over time. Plug-In Hybrid Electric Vehicles will need to be serviced, charged and finally billed for their usage. Distributed Generation capacity that is sold back to the utility will require C&I settlement and retail billing. Evolving customer expectations will warrant newer capabilities to be delivered over time. New IT applications will need to be integrated into an existing IT application portfolio and new business processes will be developed to accommodate consumer‘s growing needs and the changing landscape of Utilities‘ business models. Preparing for these challenges will enable Utilities to deliver new services gradually over time without making wholesale changes to their IT. A strategic architecture is required that accommodates the business case of delivering yet-to-be-known business applications and services.

3.13.1. Summary

- New ways of Settlement required that have not been envisioned;
- Lack of agility to achieve to new business models.

3.14. Reporting & DW Challenge

Corporate or Enterprise Data Warehouses have not been a norm in the Power industry. Reporting needs have been met traditionally through the use of operational reports taken directly from the transactional systems. Going forward, the status quo is not the recommended approach.

First, because enterprise IT will be taxed to its limits and reporting off of transactional systems may reduce application performance and impact other critical systems. Secondly, organizations will be able to mine volumes of usage, outage data, peak load and other market and operational data that will be collected from Smart Meters and other applications. Letting this data go unused is unacceptable, because this data can be used to do historic reporting, trend analysis, ad-hoc reporting, ―what-if" analytics, better planning, and forecasting. Such data can also be used to improve customer service, lower cost of operation, increase grid reliability, and improve market operations.

3.14.1. DW/BI Challenges Summary

- Lack of awareness to have a Data Warehouse and Business Intelligence strategy;
- Source of current reporting is operational databases. Such is undesirable when real SG and DR programs are launched as IT will have transactional database requiring high throughput and large data volumes.

3.15. IT & Service Governance Challenge

IT Governance programs enable management to oversee the effectiveness of IT. As Smart Grid and Demand Response projects gain momentum, IT will be front and center of the business of Power delivery. IT's role in keeping aligned with the business, participating in Regulatory compliance, and assisting in risk mitigation will become key drivers to increasing stakeholder value. In the short term, IT Governance programs should therefore be put in place for the following:

- Timely project delivery;
- Developing a centralized function through a Center of Excellence to centralize decision making;
- To develop applications in a decentralized manner;

In the medium-to-long term, IT Governance programs should focus on:

- Alignment with Business strategy;
- Providing a mechanism for controlling IT operations and holding IT accountable to the business;
- Providing value to the business, for example, implementing predictable SLAs to guarantee quality of service;
- Delivering expectations per regulatory requirements and corporate compliance expectations.

3.15.1. IT Governance Challenges Summary

a. Weak governance in place;

b. IT discipline is weak;

c. No Data Governance standards.

3.16. Transformation Challenges

Transformation is often overlooked as a challenge. Whether it's an application upgrade, replacement of a legacy application with a COTS solution, or installation of a new program, a well thought through transformation strategy will reduce down-time and provide a better quality of service to the business. The key challenges during the Transformation from the old system to the new system are: The capability to support old and new Interfaces; Validation of the functionality in Production; Change Management – across technology, business and resources.

3.16.1. Transformation Challenges Summary

- Issues with upgrading highly customized legacy applications with COTS;
- Challenge to support old and new interfaces;
- Validation of functionality in production;
- Change Management.

4. SUMMARY OF IT CHALLENGES

This section gives a summary of aforementioned challenges. There are a total of 16 IT Challenge categories and 57 sub-challenges.

(1) Organizational Challenge

- Missing CIO, CTO and Budget challenge;
- Perception that IT is not Strategic is a challenge;
- Outsourcing without controls challenge.

(2) Lack of IT Maturity

- Mostly at CMM Level 1 – not optimal for SG success;
- Elevating to CMM Level 2+ will require serious work.

(3) Evolving Technical Standards

- Standards are a Work in Progress;
- There are Competing standards.

(4) New Models – PHEV and Distributed Generation

- New ways of Settlement required that have not been envisioned;
- Lack of agility to achieve to new business models.

(5) Centralized versus Decentralized

- Holistic end to end business models need to be reviewed for the right approach;
- Data ownership and management controls lacking and will be required;
- New federated architecture is required.

(6) IT, Service and Data Governance

- Weak governance in place;
- IT discipline is weak;
- No Data Governance standards.

(7) Ripple Effect

- Impact on Downstream Applications unknown;
- Currently plausible approach to defining the Scope.

(8) Integration Complexity

- Number of Applications requiring modification;

- Number of Integration Points;
- End to End Integration Testing issues;
- Interoperability Testing challenges;

(9) Transformation Challenge

- Issues with upgrading highly customized legacy applications with COTS;
- Challenge to support old and new interfaces;
- Validation of functionality in production;
- Change Management.

(10) Lack of SOA experience

- SOA new to the industry;
- SOA tool being used in limited way;
- Lack of strategy and vision being perceived as tool limitation.

(11) Reporting and Data Warehousing Challenge

- Lack of awareness to have a Data Warehouse and Business Intelligence strategy;
- Source of current reporting is operational databases. Such is undesirable when real SG and DR programs are launched as IT will have transactional database requiring high throughput and large data volumes;

(12) Data Volume

- Storing Large Volumes of Data;
- No Information Life Cycle Management;
- No Storage Strategy.

(13) Performance and Throughput

- Performance issues related to Retrieving Data / Database Performance;
- Transaction Performance;
- Event Handling performance issues;
- Human Task challenges;
- System Performance.

(14) Transaction Processing Issues

- Distributed Transaction challenges;
- No Two-Phase commit and ―ACID‖ capabilities;
- No Transaction Processing Runtime or TP Monitor.

(15) Common Security Model

- IT security equivalent to NERC CIP 002-009 missing;
- Undefined standards;
- SG & DR business logic to control grid, market, smart meters yet no comprehensive end-to-end Security model.

(16) Point to Point Accidental Architecture

- Custom code issues;
- No re-usability of any code;
- Different Payloads for integration;
- Different Protocols for integration;
- Different Logging mechanisms;
- Unpredictable QoS;
- No Runtime for Scalability;
- No Common Vocabulary;
- No Distributed Transaction capability;
- Unreliable security;
- No centralization;
- No standard way of integrating.

5. IT CHALLENGES MAPPED TO INTEROPERABILITY FRAMEWORK

This section describes how IT challenges relate to the GridWise Context-Setting Interoperability Framework. Table 2 maps 16 IT related integration challenges to the GWAC‘s three layers of interoperability groups, eight interoperability categories and across a classification of ten interoperability issues that cut across these layers as follows. For the purposes of simplicity, only high level IT challenge categories have been mapped except P2P where the sub-challenges have been mapped as well.

Rest of this page intentionally left blank.

Integration Challenges across GridWise Interoperability Framework Categories	Basic Connectivity	Network Interoperability	Syntactic Interoperability	Semantic Understanding	Business Context	Business Procedure	Business Objectives	Economic/ Regulatory Policy	Shared Meaning of Content	Resource Identification	Time Synch & Sequencing	Security & Privacy	Logging & Auditing	Transaction & State Mgmt	System Preservation	Quality of Service	Discovery & Configuration	System Evolution & Scalability
	Technical			Informational		Organizational			Cross Cutting Issues									
1 Organizational Challenge						X	X	X	X							X		
2 Lack of IT Maturity						X	X		X			X				X		X
3 Evolving Technical Standards						X										X		X
4 New Models - PHEV & Dist Gen.						X										X		X
5 Centralized vs Decentralized				X	X	X			X	X		X	X	X	X	X	X	X
6 IT, Service, Data Governance				X	X	X			X							X		X
7 Ripple Effect		X	X	X						X					X			X
8 Integration Complexity		X	X	X					X	X	X	X	X	X	X	X	X	X
9 Transformation Challenge					X	X										X		X
10 Lack of SOA Experience			X	X	X	X			X					X	X	X	X	X
11 Reporting & DW Issues				X	X				X							X		
12 Data Volume		X											X		X	X		X
13 Performance & Throughput		X												X		X		X
14 Transaction Processing Issues		X	X								X			X	X	X		X
15 Common Security Model			X	X	X	X	X					X	X					
16 Point to Point Accidental Arch.		X	X	X	X	X			X	X	X	X	X	X	X	X	X	X
Custom Code Issues		X							X	X	X	X	X	X	X	X	X	X
No Reusability		X	X						X	X		X	X	X	X	X	X	X
Different Payloads		X								X						X		
Different Protocols		X								X						X		
Different Logging		X												X				
Unpredictable QoS		X														X		
No Runtime for scalability		X															X	X
No Common Vocabulary		X							X									
No Distributed Transaction		X									X			X	X	X		X
Unreliable Security		X									X			X	X	X		X
No Centralization		X		X		X	X		X	X	X	X	X	X	X	X	X	
No Standard way of Integrating				X		X		X	X	X						X		X

Table 2: Integration Challenges mapped to GWAC Context-setting Interoperability Framework

6. FUTURE STATE – THE SMART GRID REFERENCE ARCHITECTURE (SGRA)

Smart Grid and Demand Response initiatives will require real-time integration of applications and systems to enable real-time communication and timely sharing of data to make informed decisions.

6.1. Interoperability Goals

According to GridWise Architecture Council the GWAC Context-setting Framework is designed to make ―no architectural or technical recommendations but establishes a context to discuss alternatives and complementary approaches. The framework is a high level operational view common to the electricity community used to communicate within the electricity system to compare, align, and harmonize solutions and processes as well as with the management other critical infrastructure.” Additionally, ―Architectures are blueprints for solutions addressing the issues identified in the framework.” This section proposes such an Architecture that is designed to address the core issues of integration challenges presented earlier.

6.2. SGRA Overview

Figure 10 illustrates a *vendor, product and technology neutral Smart Grid Reference Architecture*. Smart Grid Program Managers should not confuse a tool (such as a SOA tool) with Smart Grid Architecture. A tool is neither the architecture, nor the solution, but a way to realize the Smart Grid architecture and the solution. To be successful at SG & DR, we suggest an architecture and operating environment that must include the following:

1. **An Open Smart Grid Reference Architecture**: Ten key attributes of the Reference Architecture are described in this section.
2. **The Smart Grid Governance**: The Smart Grid Architecture requires an operational environment that includes a Center of Excellence that centralizes all common functions, sound technical and project leadership that works with the IT management/CIO/CTO and processes & methodologies that brings discipline (section 7).
3. **The Tools and Infrastructure**: A discussion about various tools is not within the scope of this document. However, tools and infrastructure that are required to realize the Smart Grid architecture and vision will vary from project to project and may include: A SOA toolset, development tools, configuration management tools, a source control tool, infrastructure for development, testing and production etc.

There are ten key aspects to the Smart Grid Architecture (SGRA) include the following:

1. **Infrastructure Architecture**
   - Hardware, OS and Virtualization.
2. **Data Architecture**
   - Relational Database;
   - Semantic Data Model;
   - Master Data Management;
   - Meta Data.
3. **Real-Time Integration Architecture**
   - SOA Tool or capability that includes:
     - ESB;
     - Transport;
     - Messaging;
     - Registry;
     - Routing;
     - Transformation;
     - Complex Event Processing.
   - Service Oriented Architecture (SOA):
     - Event Driven Architecture;
     - Loosely coupled;
     - Canonical Data Model;
     - Technical Services.
4. **Application Architecture**
   - Transaction Processing;
   - Common Services;
   - Real-Time Architecture;
   - Development SDK.
5. **Business Architecture**
   - Business Services;
   - Enterprise Services;
   - Workflow;
   - Service & Process Orchestration.
6. **Security Architecture**
   - End to End security.
7. **Compliance Architecture**
   - End to End compliance.

8. **Portal**
   - Top Tier for all access: customers, employees, partners.
9. **Applications**
   - SG & DR Applications;
   - Command and Control;
   - OA&M;
   - Reporting & OLAP;
   - Enterprise Services.
10. **Smart Grid & Demand Response Applications**
    - Remote Connect / Disconnect;
    - Load Curtailment;
    - Self Healing;
    - Automated Outage detection;
    - Customer usage ePortal;
    - Others.

Key Architectural Capabilities:

(1) Loosely Coupled integration architecture (LCA), capabilities which enable application upgrades and application replacement with limited disruption to other IT applications.

(2) Real-Time Integration Architecture and Real-Time Enterprise (RTE) Integration capabilities so that applications can integrate in real-time and share data in real-time.

(3) Event Driven Architecture (EDA) to handle business and technical events. This should include Management by Exception (MBE) to process any abnormal predicted or unpredicted event in a high transaction/ high throughput transactional environment.

(4) [On Line] Transaction Processing (OLTP) capabilities to facilitate distributed transactions.

(5) Robust Complex Event Processing, Error handling and Exceptions management capability (CEP).

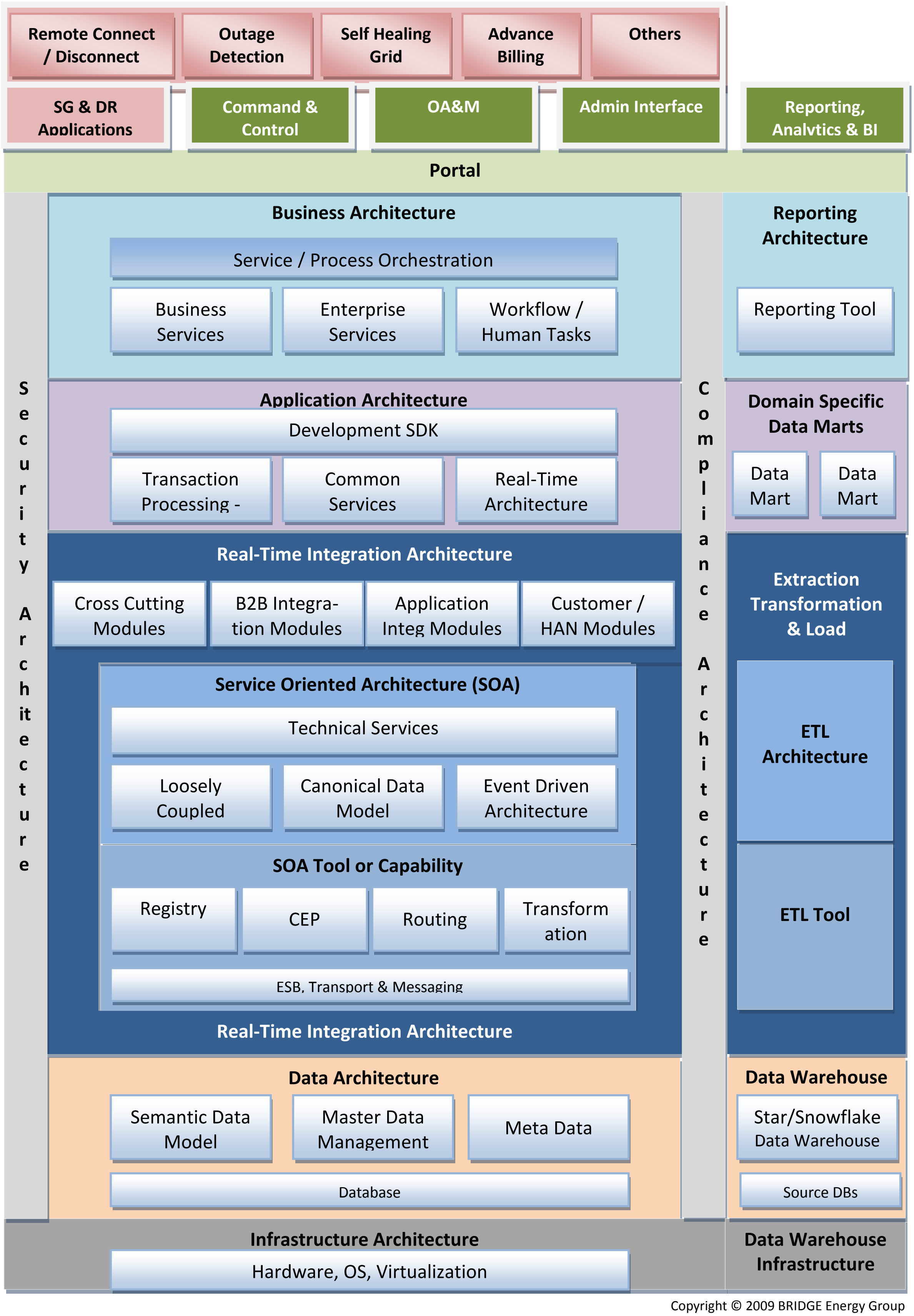
(6) Security and Compliance Architecture (S&C).

(7) On Line Analytical Processing (OLAP), Data Warehousing (DW) and Reporting Capabilities.

(8) Re-usable Modules, Libraries and Frameworks with re-usable code that provides cross-cutting capabilities (FRA of many other services).

Rest of this page intentionally left blank.

Figure 10: Open Smart Grid Reference Architecture (SGRA)



6.3. SGRA - Leveraging Loosely Couple Architecture & Service Oriented Architecture

Assuming that the Integration Complexity is enough to warrant SOA, the question then is not *should* a company do SOA, but what is required in order *to implement* SOA.

There is a widely held perception among SG & DR Program Managers that SOA is prohibitively difficult to do. First, it must be understood that SOA is not a technology or a product, but a Strategy and an Approach. SOA is *the* way to create an Integration Architecture for the enterprise that is based upon "Loose Coupling" and Services. Other alternatives – i.e. P2P or EAI – are inflexible, un-scalable, complex and more expensive overall. These alternatives can result in increased risk to the project and higher total cost of ownership.

Figure 11 illustrates the total cost of ownership comparison between SOA and P2P. The higher the number of integration points the higher the development and maintenance cost in general. However, the overall cost of both development and maintenance by leveraging the SOA approach will radically reduce as the number of integration points increase. Every organization will have an inflection point where the total cost of doing the "SOA-way" will be lower than the total cost of doing "P2P-way". In general, for Integration Complexity (I_c) of 20 or higher - The Bridge Index of Medium - the SOA approach will provide an overall lower cost of ownership (refer to section 3.2). The chart indicates that the inflection point is attained at around 15 to 20 integration points.

Figure 12 illustrates the cost of implementing a single P2P versus a single SOA interface. As shown in section 3.1, each P2P interface requires custom code in each of the participating applications. The cost of implementing this custom code is somewhat constant for every P2P interface, assuming each integration point is of same complexity. The cost for a SOA integration point, by comparison, if done right, would gradually become lower for each interface due to reusability aspects of the SOA approach.

Besides cost, the P2P and SOA differ in their approach. The SOA is strategic, business driven and top-down. The P2P is purely a tactical, technical and bottom-up approach. SOA is based upon the premise that the Business drives the architecture. Based upon that premise, an IT organization that delivers services is equipped to handle the needs of the Business as, and when, needed. SG & DR requires that kind of agility. In other words, as opposed to IT offering monolithic applications, IT builds a set of services that it threads together for the purposes of the Business. This gives agility to the organization, as these services can be threaded together as needed without engaging in unduly expensive projects. IT, then, becomes a true service organization that brings agility to the business of SG & DR.

Figure 11: P2P vs SOA Total Cost of Ownership Comparison

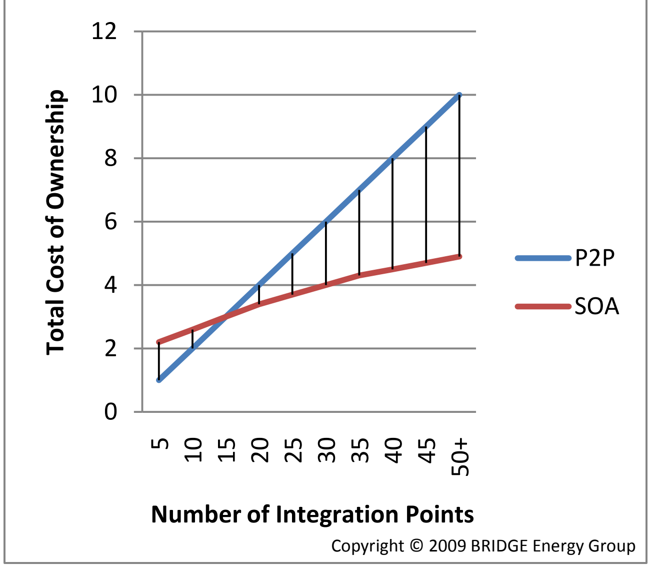
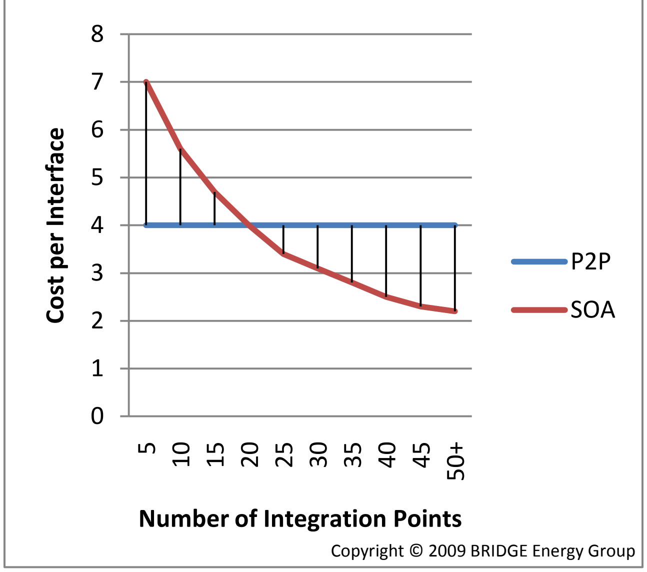


Figure 12: Development Cost of each Interface



This approach will enable Smart Grid and Demand Response programs to be launched over time in a predictable fashion. As an example, consider the new mandates in some States/Provinces to provision Distributed Renewable Generation capacity. With distributed generation comes the need to do Settlements. An agile organization will have the capability to provision new Settlement procedures that have not existed in the past. A SOA-based

IT architecture provides the ability and agility of the back-office to adapt and deliver upon these mandates.

What is different about SOA? With SOA, rather than seeing your IT as a set of applications, databases, storage and other resources, it becomes viewed as a set of value-added services. This is different from the other distributed computing paradigms of the past (DCE, CORBA, J2EE and .NET). For example, a function – like checking the status of a meter – within an application can be offered as a separate service or services that can be invoked to create a business process. From an enterprise perspective, as opposed to deploying a monolithic application, the organization deploys a set of Meter services like *Check Meter Status* and *Check Meter Tampering, Remote Connect, Remote Disconnect* etc. These services can then be tied [*orchestrated* in technical parlance] to form a business process.

The major components of SOA that should be incorporated in the strategy are as follows:

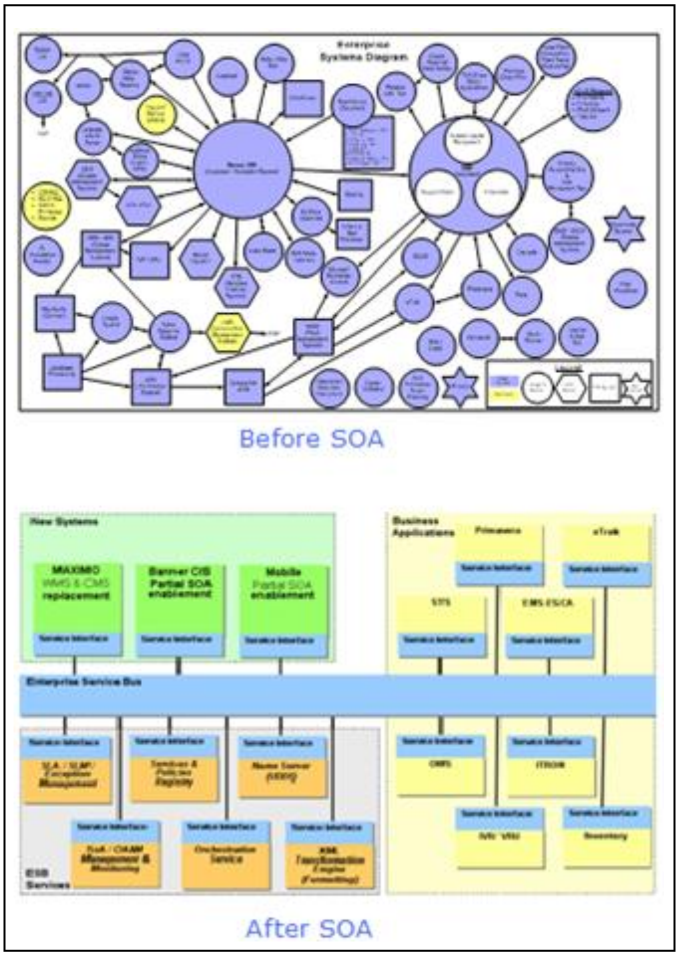


Figure 13: Before and After SOA Architecture

- **Services:** SOA is the next generation of distributed computing paradigms that professes an architecture which is based upon the concept of –Services” which are driven by the Business Architecture. Services may be built from scratch or encapsulated services can be built over old legacy systems.

- **Service Orchestration:** Services can be –tied” together not through programmatic code but through an orchestration capability known as Business Process Execution Language (BPEL) – which is the industry standard for orchestrating the services. The ability to orchestrate these services on an as-needed basis to achieve a business process provides the business agility necessary for SG & DR programs.

- **Web Services:** SOA fosters the development of web services that are built to operate in a connectionless world that is based upon XML (Extensible Markup Language) and WSDL (Web Service Definition Language). Developing a portfolio of SG & DR services provides an unprecedented integration capability to launch various SG and DR programs without the need to launch expensive large budget projects.

- **Enterprise Service Bus:** Although not a pre-requisite to SOA, the ESBs have become integral to SOA architecture as it provides mediation services for reliable communication, delivery and management of data and messages. ESB provides value added communication capabilities such as the ability for one application to publish data that can be subscribed by other applications and to create an Event-driven architecture enabling applications to detect and act on certain business events. Event Monitoring, Logging, Tracing, Data Transformation and Service Discovery are examples of re-usable services that can be easily leveraged through the ESB. An ESB‘s event management capability enables the development of the –Real-Time Enterprise”. Ironically, although ESBs are not a pre-requisite to SOA, many organizations see –ESB” based integration as the starting point for embracing SOA.

The aging infrastructure and homegrown applications that are often past their prime, require replacement and upgrades. Unfortunately, the tight coupling amongst these applications within an IT ecosystem precludes such upgrades and poses significant risk to SG & DR programs. Because of the P2P architecture and the impact of the Ripple Effect on other downstream applications, this is not a trivial exercise. SOA does provide the promise of developing a Loosely Coupled Architecture that has the capability to undo tight coupling amongst applications. Secondly, SOA provides a capability to develop services that can be developed once and leveraged across various SG & DR programs many times over. These are core SOA capabilities that can make SG & DR a success. But SOA is a good tool and must be used carefully. Improper use of a good tool can result in failure. Given that the Power industry has been a laggard in adopting information technologies, SG / DR Program Managers should exercise caution and be prudent about leveraging SOA. Not using SOA, however, is a greater risk. Figure 12 illustrates the *Before* and *After* SOA Architecture.

6.4. SGRA - Real-time Enterprise Architecture

The second aspect of the Smart Grid Architecture is to provision Real-Time decision making which is possible only if data can be harnessed as it is generated (without much latency) and is applied towards a specific objective that requires data as it happens. Such capabilities are possible only with a *Real-Time Enterprise Integration Architecture (RTE),* where immediacy of data is critical and data flows seamlessly between applications and systems (with appropriate governance and security controls). This real-time or ―active” data has significantly more value than the static and old data as it can be harnessed to make just-in-time decisions, such as automated outage detection through the last-gasp meter data for proactive customer service and proactive self healing of the grid; detection of current load and critical peak conditions to initiate automated load curtailment programs to curtail power at participating C&I customer premises, or to perform air conditioning load curtailment at participating retail households. Non real-time integration requirements via batch-data or ―passive” flow of data can be leveraged appropriately for non real-time decision making. Data Warehouse and Business Intelligence systems are an example in which the passive data can be leveraged to do trend reporting, ―what-if” analysis and in understanding historic load conditions. Both active and passive data has value and can be used strategically to make real-time and non real-time decisions.

6.5. SGRA - Event Driven Architecture

The third aspect of the Smart Grid Architecture is its ability to manage hundreds, or thousands, or even millions of transactions in such a way that events are generated, detected, and processed with pre-defined business logic and predictable conditions. An event can be considered as any notable condition that happens inside or outside your IT or your business. Usually, an event is detected as data and message flows between applications. An event in general could be a business event – such as detection of an outage condition or a system event such as failure of the MDM application to collect meter data. An event may also signify a problem, an exception, a predictable error, an impending problem, an opportunity, a threshold, or a deviation from the norm.

Given the transaction volume generated by Smart Meters, Smart Grid Architecture would also require a *Management by Exception (MBE)* capability where any error related to the integration of data and messages between systems and applications is captured, a trend identified and eventually addressed within a meaningful timeframe. In this case, MBE alludes to the capability where an abnormal condition, such as an exception, or an error requires special attention without any significant overhead or management on the rest of the system.

The Smart Grid Architecture should include an *Event Driven Architecture (EDA)* capability to process events as and when they occur with minimal human intervention.

6.6. SGRA - Complex Event Processing

The fourth aspect of the Smart Grid Architecture is its ability to process complex events. CEP is a capability that will be required to handle hundreds of events that will be generated due to a real-time architecture where millions of transactions can flow in and out of the architecture on a daily basis. As opposed to handling every event one at a time, a CEP capability would enable a complex event to be generated by consolidating a bunch of single events. In other words, a pattern of a collection of events could be combined to form a complex event. As an example, a last-gasp from a meter could be an event in isolation, but a number of last-gasps may signify a pattern that could be recognized, consolidated and a complex event could be generated to create a state of outage in a particular area and flagged with the Outage Management System and the Customer Information System.

Within the Smart Gird environment, there are two types of transactions one can envision: first, the transactions that will require real-time processing – such as a message to OMS to register an outage condition based upon the last gasps of Smart Meters; and secondly, those that are brought in and saved; for example, the Smart Meter data that is collected and processed separately. The latter is the ―historical” data which will require some historical processing of events after the fact.

A CEP capability therefore will enable the Smart Grid Architecture to identify and handle complex events and perform necessary actions—all without human intervention with appropriate security and controls.

6.7. SGRA - Transactional Capabilities (OLTP Architecture)

The fifth aspect the Smart Grid Architecture is to provide an OLTP capability to handle distributed transactions across multiple Energy applications and databases. Leveraging the transactional or “ACID” properties of OLTP systems will enable robust deployment of SG & DR programs. The following transactional capabilities must be part of the Smart Grid Architecture:

Atomicity: The Atomicity capability offered by OLTP systems guarantees that a transaction within the auspices of OLTP is either completed successfully or not completed at all. In other words, a transaction inserting a record in one database and updating another database either guarantees that both sub-transactions will be completed or that neither will be done. In fact, if the first one was done successfully and the second one was unsuccessful then the first one should be undone. What Atomicity guarantees is that there can never be a state when one sub-transaction is “committed” and the other one is uncommitted, because that will impact the referential integrity of the underlying database and system. Take for example a transaction where a new work order is to be created in a Workforce Management System, or an Asset Management system (like Maximo) that eventually updates three applications - the company‘s Human Resource Application (such as SAP HR), the Project Management and Timesheet Application (such as Primavera), and the Customer Service and Billing System (such as Banner CIS). If this transaction was enveloped within a Transactional OLTP system, then the system will guarantee that either the entire transaction will be successful so that all participating applications (in this case, SAP HR, Primavera and Banner CIS) are aware of the new work order, or the transaction will be cancelled if one of the participants in the transaction errors-out. Imagine a scenario without OLTP where hundreds of such transactions are executing and there is little to no guarantee of atomicity. The chances are that within a few weeks, if not days, the various applications and systems will have inconsistent databases and systems. By the time this mistake is realized, undoing the system and developing a new system will be extremely expensive and time consuming.

Consistency: When a transaction is executed, the system will transform from the current state to a new state and not an unknown state. For example, if a last gasp is detected from a few meters, then that data can be processed and a condition of “outage” be created that may create a trouble ticket in the Outage Management System, and a work-order created in a WMS, followed by an immediate dispatch of field crew if required. In this example, the state of a specific customer, a feeder line, a transformer, a substation or a zone is either in the state of “Outage” or not. It‘s not “Outage” in one and “No Outage” in another.

Isolation: Isolation refers to the capability where each transaction is independent of each other or is isolated from other transactions. Isolation is a perquisite in a Smart Grid and Demand Response program where dozens of work orders are being created to dispatch work crews, hundreds of “last gasps” need to be processed during a likely outage, or when thousands of remote thermostats are to be curtailed to manage peak load, and possibly millions of Meter reads need to be processed every day. Without transaction isolation, serious flaws in the transaction management can result in wrong work orders, missed “last gasps”, wrong thermostat curtailments and meter misreads. IT will need to spend significant resources to manage these risks and without the proper architecture the problem may be irreconcilable.

Durability: Durability refers to the condition of the system where all committed changes survive any system failures. IT has often been a weak link in the industry with little investment in the areas of transaction and data “survivability”. This is very important in a highly regulated, customer oriented and security sensitive environment. Typically, at the database level, each database (Oracle, SQL Server, and DB2 etc.) guarantees database durability in that, if the transaction is written in the transaction log, then the database guarantees that it will be durable. Durability becomes a concern if a business transaction spawns multiple applications and databases resulting in the need for end-to-end transaction durability, as opposed to durability within a subset of a larger transaction. Smart Grid & Demand Response requires many scenarios related to Customer Services, Grid reliability, Market operations, Load Curtailment and many others, all of which may spawn multiple applications and databases. The objective is for these end-to-end transactions to provide holistic end-to-end durability across all the systems, including legacy systems participating in a business transaction.

6.8. SGRA - Security & Compliance Architecture

The sixth aspect of the Smart Grid Architecture is the capability to provide end to end holistic security and compliance capability through a Security and Compliance Architecture (S&C). This paper recommends a few guidelines that must be considered as organizations launch their SG & DR initiatives.

- **IT Security Architecture**
    - An end-to-end Security Architecture should be developed within IT that spawns the Business Architecture, Application Architecture, Data Architecture and Infrastructure Architecture;
    - Data flowing between applications must be secure;
    - Web Services must be secure and governed under strict security;
    - A Firewall must exist between Power Engineering functions and IT functions per NERC CIP Security.

- **Consumer Security**
    - Every customer should be able to see their own meter data and bills, not others';
    - Hackers should not be able to access consumer premise Smart Meters and determine if the consumer is home or away;
    - Consumer data may not be shared with appliance, meter and tool manufacturers unless the Consumer has agreed to it;
    - Developing a Privacy Policy should be prioritized.
- **Transaction Security**
    - Every transaction must be secure from hacking, pilfering or spoofing all the way from the communication network to inside the IT applications;
    - WS-Security could be used for Web service security.
- **Data Security**
    - Data must be made available on a need to know basis;
    - Ownership, Control, Management and Control of data should be a major component of the IT strategy and architecture.
- **Compliance**
    - The Architecture provides capability within the tolerance of various legislation requirements and compliance restrictions.

6.9. SGRA – Data Warehouse & Reporting Architecture

The seventh aspect of the Smart Grid architecture is not related to transactions but reporting. It is suggested that transactional systems should not be used to perform reporting. Instead an enterprise data warehouse be developed that leverages data from the transactional system that might include customer usage data, outage data, peak load and other market and operational data that will be collected from Smart Meters and other applications. The transactional system may however be used to light operational reporting that does not impact transactional performance. This can then be used to create static and ad-hoc reports, historic reporting, trend analysis and ―what-if" analytics. Such data can also be used to improve customer service, lower cost of operation, increase grid reliability, and improve market operations.

6.10. SGRA – Framework of Reusable Libraries

The final aspect of the Smart Grid architecture should be a Framework of reusable libraries and code that every organization needs to develop to foster standardization and reusability to increase reliability, consistency, predictability and reduce development and testing time. The GWAC refers to this as Cross-cutting issues. We propose that in addition to the GWAC cross cutting issues, there are numerous others that must be addressed. Once developed and tested, these libraries and services can be used repeatedly reducing time to market and lower cost.

The Smart Grid Architecture can be described as:

SGA = SOA + RTE + EDA + CEP + OLTP + S&C + FRA

7. SMART GRID ARCHITECTURE GOVERNANCE

7.1. Center of Excellence

The Smart Grid Architecture should not be built in a vacuum and will only be successful if it is developed with a collaborative yet centralized approach, with appropriate controls to monitor and measure program success. We recommend developing a Center of Excellence to centralize the development of the Smart Grid Architecture, to create standards, and foster the use of common capabilities to launch Smart Grid and Demand Response programs.

A Center of Excellence (COE) for Integration will enable all integration functions to be centralized with the COE group. The COE is responsible for defining standards that should be leveraged by various application teams or different project teams. Functions of COE:

- Defines standards and processes;
- Selects tools and products;
- Develops reusable code and libraries that can be used as APIs for Application teams;
- Promotes collaboration and best practices;
- Acts as an Advance R&D group;
- Evangelizes technology and capabilities to a broader group including business;

Questions to ask in setting up the COE:

- Does your company have a culture that can support a centralized group that defines best practices and would dictate its use company-wide?
- If the COE identifies standards would other teams use them?
- Does the company recognize the need for centralizing some of the activities?
- Does your company support a collaborative approach?

- What type of organization model does your organization have? Do you support direct reporting, an influence model or a matrix model?
- Are people effective in a matrix model?

Answers to these questions should be leveraged with experience from previous projects to assess if the COE can be successful. One would also need management buy-in to initiate the COE group.

7.2. Organizational Structure

Two resources are key to the success of Smart Grid Programs. In most companies, these key resource will work directly for the office of the CIO.

1. Technical Leadership: Hire a CTO with SOA experience. Alternatively, a company can hire a consultant (or a team of consultants) who have the SOA experience in the Energy industry.

   Caution: (a) The candidate must have Successful SOA project experience; (b) the hired party or consultant must be neutral and not aligned with any vendor or product and (c) Experience of the individual or individuals is more important than the ―company".

2. Project Management: The IT program manager for the SG & DR programs should be tactically involved in managing the program – and not be a ―figurehead". This PM works directly for the office of the CIO or the SG & DR leader, and is responsible for managing the IT related SG & DR initiative. Alternatively, the SOA Architect or Lead may serve as a Project Manager (if he or she has the skills) until team size dictates otherwise.

   Important: The Day-to-Day tactical PM role should *not* be outsourced to the company performing the implementation, or a product or a tool vendor whose product or tool is being used. These roles should be either managed internally by an employee/consultant or by another 3rd party. Vendors should not be dissuaded against having their own PM but they should report to the program PM.

3. Outsourcing and Off-shoring: Caution is recommended when it comes to outsourcing and off-shoring integration. Integration activities require active participation of and collaboration with various stakeholders. It will require workshops, collaborative sessions, frequent design reviews and related activities. Off-shoring integration activities may not be as effective as expected on other projects. Outsourcing to a near shore company and team is a reasonable option as long as the contract is well defined, artifacts of delivery are well defined, and the team is engaged and understands their responsibilities. The Tactical PM and Technical Leadership must work actively with the Outsourcing vendor on a daily basis to achieve the expected results. Without proper controls, an outsourced model may become the weakest link in SG & DR program chain -- so extreme caution is recommended.

7.3. Methodology & Artifacts

Architecture is incomplete without a methodology and artifacts. It is important to identify the SDLC Methodology that will be leveraged and the actual artifacts that will be produced. Each artifact must have a template and assigned party based upon the responsibility assignment criteria: Responsibility, Accountability, Consultation, and Informing. The parties could be: the Steering Committee, Project Sponsor, Project Lead & Project Manager, Business Analyst, Technical Architect, Developers etc.

A SDLC methodology, such as the Waterfall model, Joint Application Development (JAD), Rapid Application Development (RAD) Iterative Development, and others should be leveraged to bring discipline to the systems and software development.

Artifacts – Clear understanding of the software artifacts that will be required and delivered over the course of a SG & DR project or program. Lack of artifacts will cause delays, mismatched expectations and un-necessary risk of communication and collaboration. These risks can be easily avoided. This paper suggests that the project manager define and maintain a detailed portfolio of these artifacts that need to be produced over the life of the project, including entry and exit criteria for the success of each phase.

	Responsibility	Accountability	Consult	Inform
Project Charter	Steering Committee	Steering Committee		
Business Requirements	Business Analyst	Project Manager	Business Users	Biz Unit Manager
Functional Requirements	Business Analyst	Project / Lead Proj. Manager	Technical Architect	Business
High Level Design	Technical Architect	Project Lead/ Proj. Manager	Biz Analyst, CIO/CTO	Business Users
Low Level Design	Technical Architect, Developers,	Technical Architect, PM	CIO/CTO, Business Analyst	Business

7.4. Interoperability Testing

Key to the success of Smart Grid programs is to establish the capability to perform end to end interoperability testing with applications and systems both within the organization and outside the organization. The capability should prove:

- Interoperability testing is 100% working or else it is to be considered failed. There should not be any ―grey area".
- Participating applications and parties are delivering data per design criteria and expectations;
- End to end functionality is proven per design;
- Performance expectations are being met;
- Security is meeting organizational and regulatory expectations;
- Audit and Tracing data provides context for easy diagnostics;
- Failure conditions are working as designed;
- Distributed transactions should meet two-phase commit criteria and OLTP ―ACID‖ properties;

8. ROADMAP & STRATEGY

This section provides a template for the Smart Grid Integration Roadmap.

First a technology adoption roadmap – specifically the Service oriented architecture approach to performing the integration. Figure 14 illustrates a simple Roadmap of how an IT organization can transform from a P2P Accidental Architecture to a SOA-based architecture.

Second, a Smart Grid business and IT transformation roadmap as shown in Figure 15 that illustrates how IT can be aligned with the Business to enable successful deployment of various Smart Grid & Demand Response Programs.

8.1. P2P to SOA IT Transformation

Based upon Figure 14, an organization could follow an evolutionary approach rather than a revolutionary approach to transforming their IT. The Roadmap is as follows:

- Year 1: In the first year of IT Transformation, an organization would initiate the program and go through some early learning and discovery process to understand the value of SOA to the organization. A Center of Excellence is established. The organization may go through an RFP process to procure an SOA tool and leverage that into building a Proof of Concept. A proof of concept will provide the data to build a business plan with the ROI to justify investment for the next phases. A Reference Architecture is established. A project would be identified that will fund the investment in IT transformation. The value proposition to the sponsor will be a higher return on investment after 2-3 years. Some early Standards are put in place during this time. Additionally, a Reference Architecture established early in the year would be tailored to build a solution that meets the needs of the organization.
- Year 2: In the second year, the SOA tool (perhaps the ESB part) will be used as an Integration Layer to integrate applications. The Reference Architecture is exercised to build an integration specific solution. If organizations were to leverage SOA in an evolutionary way, then we recommend using a Loosely-Coupled Architecture as the Integration platform and some early development of services.
- Year 3: In the third year, the organization starts to use the Services as a practice and leverages the Services for more projects. The organization continues to write reusable code and modules and leverages that into building services. The services can be orchestrated to build business processes.
- Year 4: In the fourth year the Business Processes are Re-engineered to optimize the organization. Rather than one project, the team views various projects holistically and prepares to optimize the business processes through the BPR approach.
- Year 5: In the fifth year, the services are developed and continue to be developed, the business processes have been optimized and the organization is ready for optimization. At the completion of this phase, the IT organization is working in lock step with the business and delivering the value to all stakeholders in an effective and efficient manner.

8.2. Smart Grid Transformation Roadmap

The Smart Grid Architecture roadmap includes:

A) Five key tracks:

- Organizational
- Stakeholder Communication and Benefits
- Architecture, Design and Development
- Strategy, Governance and Center of Excellence
- Enterprise and Service Maturity

B) Six Phases:

- Initiation & Planning
- Preparation
- Pilot

- Smart Grid 1.0
- Smart Grid 2.0
- Smart Grid 3.0

Unlike the previous figure, the timeline in Figure 15 has been intentionally not defined because each company has unique plans and timeline expectations for Smart Grid and Demand Response programs. However, for the purposes of reference the six phases could be spread over a period of 4 to 8 years – an average timeframe based upon various Smart Grid programs underway throughout North America.

It is important to realize that this is a generic roadmap and may not directly fit the needs of every organization. ―Customizing" without proper context may not lead to expected results. Care and prudence must be exercised as planners launch their SG & DR programs. We firmly believe that every company‘s business drivers, motivation, plans and IT environment is different. As a result, the Roadmap for every company will be different.

As opposed to describing every item in the roadmap, this paper describes only the key focus of each phase.

8.2.1. Initiation & Planning Phase

The key focus of this phase is to hire the Lead Architect, Project Manager and to initiate the project. Some early evangelism is warranted to get the project off to a start. An early roadmap will be established, a set of presentations are created that will be leveraged to engage appropriate stakeholders and management. Organizational planning prior to embarking on the SG & DR journey is the key focus in this phase.

8.2.2. Preparation Phase

The key focus of this phase is:

(a) For the Chief Architect to work with the CIO and IT management to translate the vision into Strategy and Roadmap. In order to develop the strategy, it is imperative to do an ―As-Is" Assessment of the current IT systems and perform a gap analysis to identify the enterprise architecture in the current IT. Such analysis will provide details to build-out the IT strategy and roadmap.

(b) Prepare to Elevate the Maturity of IT: Elevate the maturity of IT processes to CMM level 2 or higher, by adopting industry frameworks such as COBIT, ITIL or SEI CMM. These frameworks provide guidelines for process performance improvement and can be applied by business and IT for various purposes such as planning and organization functions, delivery and support capabilities, application procurement, acquisition and implementation of software and systems, project management, and various other IT tasks and services. For those familiar with Six Sigma, this framework could be used only after foundation maturity Level 3 or higher is attained i.e., processes are defined and used repeatedly. As a reference, there are six levels of maturity levels. For SG & DR Level 2 should be the absolute minimum target.

(c) The third key focus is to Prepare Preliminary Budget and get early Management Buy-in.

(d) And lastly, in this phase we calculate the *Integration complexity*, *The Bridge Index* and *Ripple Effect* to scope applications for the pilot. Use the quantitative analysis to build the Business Plan that justifies investment in new tools and technologies.

8.2.3. Pilot Phase

This key focus in this phase is to

(a) Leverage a real pilot to start developing the Smart Grid Architecture. Generally an MDM application integrating with the CIS application is piloted. There can be other pilots as well. If a SOA tool is required based upon Integration complexity, then a RFP is typically required to procure the SOA tool.

(b) A ‗deep-dive‘ evaluation of SOA technologies and architecture is performed. Perhaps a RFP is issued.

(c) A Methodology and a set of Artifacts is selected or identified.

8.2.4. SG 1.0 Phase

This is the phase where one project is identified and will be implemented from start to finish. The Smart Grid Architecture will be developed in this phase. Key Focus in this phase:

- **Develop Services**: Example: Automated Meter Reading, Remote Connect & Disconnect, Meter Change-out, Meter Voltage Detection, Reporting for Energy Efficiency, Power Restoration Notice, Remote Meter Rate Change. There is a Service Rollout strategy.
- **IT Maturity**: CMM Level 2 with Business: Enterprise based and IT with new Loosely coupled Architecture, First Phase of Enterprise / Integration Strategy with Building of Reusable Modules.
- **SOA**: Data and Some Service Level Integration

8.2.5. SG 2.0 Phase

This is the phase where more than one project is being implemented. Various Services are being developed with synergies among the developed code.

- **Develop Services**: Example: Automated Outage Management, Advance Outage Management, Self Healing, Dynamic Pricing, New Rate Offerings, HAN, ToU Communication, ePortal, Peak CMM Level 2.5:

- **IT Maturity**: CMM Level 2.5 with Business Optimized and IT with LCA.

- **SOA**: Data and Service Level Integration

8.2.6. SG 3.0 Phase

This is the phase where many projects are being implemented. Various Services are being developed with synergies among the developed code. Smart Grid Programs are gaining momentum and success.

- **Develop Services**: Example: Selectable Bill Rate, 3rd party data access, Whole House Switch, Distribution Automation: Capacitor Control, Fault Indicators, Remote Monitoring, Distributed Generation, PHEV integration, Enhanced Services.
- **IT Maturity**: CMM Level 3+, with Business Optimized and IT is Dynamic

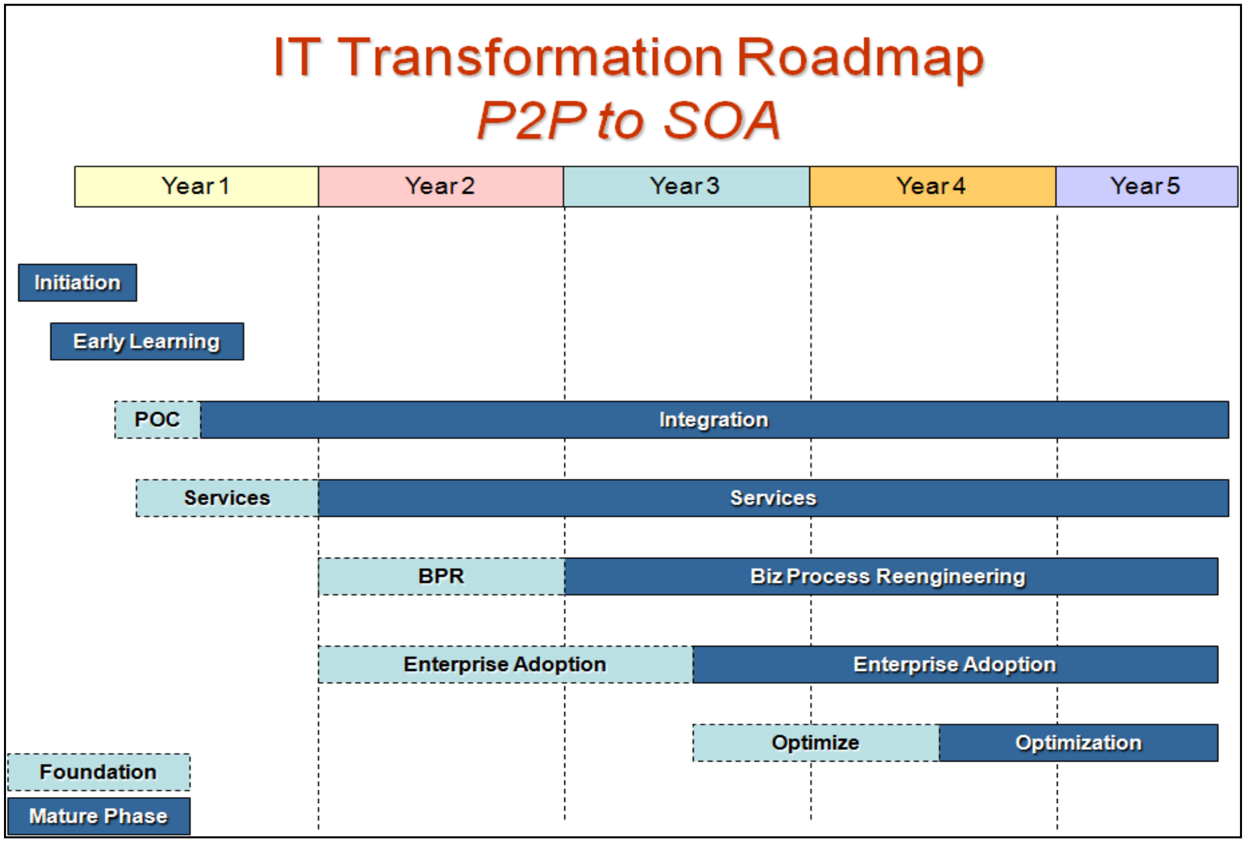
- **SOA**: Data and Service Level Integration. Plus Business Process Re Engineering and Optimization starts.

8.2.7. SG1.0 to SG3.0

In every phase, change management, configuration management, release management and testing should get special consideration. Appropriate resources should be allocated for the following:

- Training;
- Knowledge Transfer;
- Define appropriate controls around Change management, configuration management and release management;
- Hiring appropriate resources;
- RE-training internal employees and
- Collaboration amongst stakeholders;
- Interoperability testing.

Figure 14: IT Transformation Roadmap - From P2P to SOA (template)



Smart Grid Integration Roadmap (Template)

INITIATION & PLANNING	PREPRATION	PILOT	SG 1.0	SG 2.0	SG 3.0
Organizational					
Hire Chief Architect	Hire Intgration Architect	Organizational Deisgn Skillset Roles & Responsibilities	Hire Developers Change Management Training	Staff Hiring Knowledge Transfer	Staff Hiring Knowledge Transfer
Stakeholder Communication & Benefits					
Communicate Benefits Obtain Sr. Managment Buy In	Preliminary Budget Buy In	Budget Approval Communicate ROI/CBA	ROI Communication	ROI Communication	ROI Communication
Architecture, Design & Development					
	As-Is Assessment Gap Analysis	SG Architecture Proof of Concept MDM + CIS Pilot or Equivalent SOA Connectivity / Integration **Data Integration**	SG Architecture 1.0 Project 1: Using SOA MDM + CIS + GIS + Other SOA Integration + Services **Services Integration**	SG Architecture 2.0 Project 2 : DR Program Integration & Business Services DW/BI 1.0 SOA Integration + Services + BPR **Process Integration**	SG Architecture 3.0 Project 3: Enhanced Applisations Grid Optimization DW/BI 2.0 SOA Intg+Services+BPR+Optimize **BPR Integration**
Strategy , Governance & Center of Excellence					
Initiate Program Business Plan	Buidl Vision, Roadmap & Strategy Center of Excellence Integration Complexity Development Approach	Prepare ROI & CBA SOA Tool Evaluation, RFP Core Tech. Initiatives IT Governance, Security Standards & Methodology	Enterprise / Integration Strategy Build Reusable Modules Service Rollout Strategy Transformation Strategy	Reusable Data / Core Services BPR Strategy Integration Optimization	Dynamic Reusability Reusable Enterprise Services Enterprise Optimization
Enterprise Maturity / Services					
CMM Level 1	Prepare elevating IT Maturity using COBIT & ITIL	Prepare for IT Level 2 Business: Process based IT: Point to Point	CMM Level 2, Business: Enterprise based IT: LCA **Services**: Automated Meter Reading, Remote Conn/Disc, Meter Changeout, Meter Voltate Detection, Reporting for Energy Efficiency, Power Restoration Notice, Rem. Mtr Rate Change	CMM Lvel 2.5:, Business: Optimized, IT: LCA **Services**: Automated Outage Management, Adv. Outage Mgmt, Self Healing, Dynamic Pricing, New Rate Offerings, HAN, ToU Communication, ePortal, Peak Time Rebate, Load Curtailment	CMM Level 3+; Business Optimized, IT: Dynamic **Services**: Selectable Bill Rate, 3rd party data access, Whole House Switch, Distribution Automation: Capacitor Control, Fault Indicators, Remote Monitoring, Distributed Generation, PHEV integration, Enhanced Services

8.3. References

[1] Gridwise® Interoperability-Context Setting Framework, The GridWise Architecture Council, March 2008.

[2] Report to NIST on the Smart Grid Interoperability Standards Roadmap—Post Comment Period Version, *Contract No. SB1341-09-CN-0031—Deliverable 10)* August 10, 2009, EPRI

[3] Smart Grid Policy, March 19, 2009, 126 FERC ¶ 61,253.

[4] Energy Independence and Security Act of 2007, Pub. L. No. 110-140, 121 Stat. 1492 (2007) (EISA)

[5] NIST Framework and Roadmap for Smart Grid Interoperability Standards Release 1.0, Sept 2009.

[6] TPC-C Benchmark http://www.tpc.org

[7] COBIT 4.1, IT Governance Institute

[8] Smart Grid News, Austin Energy Numbers on Data Volume,

http://www.smartgridnews.com/artman/publish/Technologies_Security_News/That-Smart-Grid-Data-Surge-We-Mentioned-Earlier-You-Can-t-Ignore-It-1351.html

Author Biography

Tony Giroti is the Chairman and Managing Director of BRIDGE Energy Group which is focused on Grid Operation, Market Operation and Smart Grid. He has expertise in Enterprise Architecture, IT Transformation, Application Integration, Service Oriented Architecture and Business Intelligence. After completing his Bachelor of Engineering in Electrical Engineering, Tony trained at Crompton Greaves Ltd. in Power Systems division designing large transformers. Tony also holds a Master of Science in Electrical and Computer Engineering from The University of Massachusetts. He has over 20 years of experience in managing Information Technology products, platforms and applications. Most recently, he has been leading various Smart Grid and Demand Response initiatives with IOUs and T&D operators assisting IT and Management in developing Smart Grid roadmap, business plan, Integration strategy and launching Center of Excellence for Smart Grid, Enterprise and Application Integration. Earlier, Tony also worked in the Telecom and Financial services industry as CIO and CTO developing enterprise platforms and running large Business and IT transformation programs. Tony is an active speaker at conferences, is teaching courses in Compliance, Security and Governance, has written numerous white papers and has been granted 4 patents by US Patent and Trademark office in the areas of SOA/XML/IT platform - Patent #7,492,873; #7,369,540, #7,061,928 and a fourth awaiting a Patent Number.

Tony had also started two venture capital-backed global companies in the areas of Data Warehouse and Unified Communication. He took the latter one public. He is currently the Executive director of an Australian public company, former Chairman of the IEEE, President of Power Engineering Society, Chairman ISACANE and former President, CEO and Chairman of two technology companies. Tony is CISA certified. Tony's Contact: Email: TGiroti@BridgeEnergyGroup.com and mobile 617.480.6550.