BIOGRAPHICAL INFORMATION

Rodney K. Dell
Director, Business Optimization
TAG The | Asset | Group

Specific Responsibilities

Mr. Dell joined TAG The | Asset | Group in January of 2003. He is responsible for providing business optimization consulting services to advise utility executives on recommended changes to business processes, systems, systems integration, organizational roles, training requirements, etc. In addition Mr. Dell directs the architecture for the integrated solutions that TAG pre-configures and pre-integrates to reduce client risk and expedite implementation timeframes.

Past Experience

Mr. Dell has more than a decade of experience in designing business process models and supporting information system and data architectures for utilities. His skills include aligning business capabilities and process objectives with strategic business objectives, developing and designing business process models and system architectures, and defining optimized end-to-end business processes based on standard system functionality. These skills are essential for developing well-architected business solutions to effectively select and integrate systems. Mr. Dell developed his utilities and asset management career during fifteen-year tenure with the South African national electric utility ESKOM

Educational Information

M.S., Engineering, Union College, Schenectady, New York, 1983
B.S., Electrical Engineering, University of Cape Town, South Africa, 1976

Effectively Managing and Analyzing Asset Data

Rodney K. Dell

TAG | The | Asset | Group

380 Interlocken Crescent, Suite 700

Broomfield, CO 80021

ABSTRACT

The ability to access and analyze asset data is a critical aspect of making effective asset management decisions. Different types of asset data, often managed in different applications, need to be brought together for this purpose. An effective application architecture solution brings the required data together into an appropriate environment that enables effective analysis and optimization.

The paper presents sound engineering approaches to effectively manage and analyze asset data that have proven successful in the utility industry. These approaches should be used together to design an optimized utility system that can deliver a quality product. The lessons learned from these engineering disciplines are also appropriate for designing an optimized application architecture that effectively supports the utility's end-to-end business processes.

This paper also presents the problems associated with other approaches such as departmental solutions, over-extending applications, monolithic ERP solutions, and a data-warehouse approach. It also presents a robust approach to selecting and integrating best-of-breed applications to give decision-makers improved access to required data.

INTRODUCTION

The paper presents sound engineering approaches to effectively manage and analyze asset data that have proven successful in the utility industry. These approaches should be used together to design an optimized utility system that can deliver a quality product to customers. The lessons learned from these engineering disciplines are also appropriate for designing an optimized application architecture that effectively supports the utility's end-to-end business processes.

Optimized application architecture provides users the ability to access and analyze asset data to make effective asset management decisions. Different types of asset data, often managed in different applications, should be brought together for this purpose. An effective solution enables the data to be managed as part of the day-to-day business processes and then brings the required data together into an appropriate environment to enable effective analysis.

This paper will discuss four sound engineering principles including design standardization, overall system focus, system master planning, and design iterations. It also presents the problems associated with other approaches and presents a robust approach to application selection and integration. This recommended approach to the implementation of these solutions is not to follow the old traditional waterfall approach but instead to follow an iterative solution gap analysis around a suite of pre-configured and pre-integrated, standard application components.

SOUND ENGINEERING PRINCIPLES

Modern IT methodologies have recently caught up to physical engineering disciplines that have been in place for many decades. This section discusses how the fundamental engineering approaches used in utilities should also be used for designing and implementing an optimized application architecture. This discussion shows that sound principles can be applied across different engineering disciplines. The principles that are presented include design standardization, overall system focus, system master planning, and design iterations.

Design Standardization

One concept used by utilities is standardization – design the same type of facility the same way each time using the same components so that parts and repairs are repeatable. Common standards are generally applied to facility design and materials such as the physical assets of pipes, pumps, and valves. The benefits of standardization to the utility are many, including:

- Increased volume – buying in bulk helps to achieve a reduced purchase price
- Reduced number of tools, procedures, and training required to install, operate and maintain common assets
- Reduced number of spare or custom parts required to support the asset base

Additional benefits can be achieved if utilities work together to agree on common standards across the industry. It is not cost-effective for utilities to design and custom-build pipes and pumps.

As there are advantages to using standard components and including them into the overall utility system to meet specific needs, the same is true for information systems. To cost-effectively

support a utility's end-to-end business processes best-of-breed applications with the following characteristics are necessary:

- Use of Commercial-Off-The-Shelf (COTS) software
- Employ modern technology (such as web-based and object design)
- Provide intuitive user interfaces that align with the workflow
- Integrated workflow
- Support a 'plug and play' systems integration bus approach

Custom asset design efforts should be focused on the few instances where configuring and integrating standard material items cannot satisfy specific utility requirements. Similarly, application design effort should be focused on designing, building, and integrating COTS application components.

Overall System Focus

The overall utility system generally consists of obtaining the raw product from several sources, then transporting, treating, transmitting, storing, conditioning, and distributing the product. In the case of a wastewater system, the value chain is completed by collecting, treating, transmitting, and disposing of the wastewater. Utilities usually manage the entire or specific portions of the overall utility system or value chain.

Since the utility system needs to work as one entire end-to-end process, it would be ineffective to focus all investments on only one or two components. The resulting quality of the product is only as good as the weakest link in the chain of processes. Therefore, utilities need to start by defining the service levels that different customers require and then work backward to define the system capacity and reliability needs for each portion of the system. This needs assessment, compared to the current asset capabilities and risks, defines the improvements required for each group of assets. Based on the criticality of each asset group, the required improvement projects can be prioritized depending on the available budget.

Similarly, utilities have a set of core end-to-end business processes that they need to perform effectively and efficiently. Operational excellence should be the objective of most utilities since their customers generally do not want their utility to increase their costs to pay for operational inefficiencies. Utility management needs to clearly define the business drivers and objectives.

Based on performance areas that need improvement, the value-adding end-to-end business processes need to be assessed. This assessment identifies the business capability improvements needed for each process. Solution components then need to be developed that fit into the overall application architecture. The solution improvement projects can be prioritized depending on the criticality of the process chain improved, the level of improvement, and the available budget.

System Master Plan

All utilities develop a system master plan to provide a strategic level, long-term forecast of the system capacity and reliability improvement projects. This is based on forecasting the customer capacity and reliability requirements and optimizing which improvement projects are needed to

meet this future scenario. This system master plan is updated on a regular basis to refine the forecast and optimize improvement projects.

Prior to the design and construction of any major customer project, these needs are reflected against the optimized system master plan to see if this need was forecast and merely needs the proposed solution to be implemented. If the need was not anticipated, a rework of this area of the master plan may be required. This approach prevents infrastructure from being built that does not fit properly into the overall system. Planning engineers use major customer projects to advance relevant portions of desired capacity that is often more than required for the individual customer, but will be needed in the near future. Growth should be planned for that accommodates the whole system for the long-term.

Similarly, all IT application projects should be aligned with overall application architecture. So, for example, if a work and maintenance solution is required for the T&D portion of the system, the longer-term requirements of the treatment plants need to be taken into account. This prevents the utility from ending up with multiple applications that do essentially the same thing, which is expensive to purchase, integrate, implement and maintain.

The recommended way to cost-effectively support the a utility's end-to-end business processes is to develop an overall application architecture that consists of standardized best-of-breed applications, that preferably use a modern web-based technology, with integrated workflow, that are effectively integrated using a robust systems integration bus approach. The few resulting capability gaps, generally areas of competitive advantage, should be the only focus of any application development effort.

Design Iterations

Another concept effectively used by utilities is to have a design project developed iteratively, in layers of detail. Utilities do not usually provide high-level requirements and expect a detailed design for the solution. More often, they follow an iterative design process starting with a conceptual design, then a preliminary design, and increasing level of design details before agreeing on the final detailed design. Generally it is beneficial to invest up to 20% of a projects cost in optimizing the various levels of detail on a solution's planning and design.

This approach enables the utility to envision what the final solution will look like and considers many perspectives and potential impacts of the proposed design. Each level of design detail develops representative 'artifacts' of the solution that enable the users and stakeholders to visualize the proposed solution design and identify and agree on required changes before moving to the next level of design detail. 3-D animated designs are useful representations of the final product and help users consider different perspectives to identify required changes. The requested changes are sometimes due to a miscommunication of initial requirements and are also sometimes due to the user realizing an improved requirement based on what they see as the solution unfolds. This approach avoids major and expensive rework at the detailed design level.

Similarly, the recommended approach to IT application design and implementation should follow an iterative approach where an application is configured and tested at successive levels of detail to see how well it supports the end-to-end business processes and identify required

changes. Experience has shown that at least three iterations are ideal to identify and eliminate major risks and problems.

The traditional IT waterfall approach uses linear steps in a single-pass process to define business needs, select and/or design applications and integration, configure and/or build applications and integration, test and implement the solution. There are several problems with this risky approach. The process is long and uses language to convey difficult concepts, which results in solutions seldom looking like the original design. Another limitation of this approach is not being able to respond to changes in requirements later in the process without major cost and schedule impacts.

Examples of modern IT development methodologies are the Rational Unified Process (RUP) and the Agile development approach. The latter approach breaks down the development into small chunks, which is similar to the design package approach used on design-build construction projects. The recommended methodology for implementing COTS-based solution components is the EPIC methodology (developed by Carnegie Mellon University), which has modified the RUP methodology especially designed to support COTS-based solution component implementations.

ALTERNATIVE SOLUTION SRATEGIES

This section presents some of the common problems associated with alternative solution strategies that are often sold as a "silver bullet". The solution strategies discussed below include departmental solutions, over-extending applications, monolithic ERP solution, and a data-warehouse approach.

Departmental Solutions

Often, a department within a utility will purchase and implement applications that solve localized problems. The problem with taking a departmental/ functional approach is that most end-to-end business processes span many departments. This results in situations within the process where information is merely thrown "over the wall" to the next department. This practice causes disconnects in the process, which causes unnecessary time delays and duplicate data entry, which automatically results in data integrity problems.

In addition, for many applications within a single department to work, they need specific criteria in terms of their data model. Some of the data these applications need should be managed from within a different application. Often these applications are not effectively integrated to share this data, which again results in duplicate data capture and resulting data integrity problems.

This approach also results in duplicate application functionality, such as utilities that have two or three work and maintenance management systems or siloed document management systems. Often, the common reason for attachment to multiple systems is that each department believes it is different and unique and has special requirements. While this may be true in terms of the asset details, it is not in terms of the fundamental asset management processes. Without appropriate rationalization for the application architecture and implementation of an effective systems integration solution, these disparate data sources impede the utility's ability to effectively manage, share, and analyze data required to make optimized decisions.

Over-extending Applications

Software vendors often think their products are a cure-all, which often leads some to believe that these products can support processes that are not appropriate for them. While the software can always be forced to fit, it can be an inelegant and ineffective solutions. As the saying goes, “to a hammer, everything looks like a nail”.

An example of this approach is when the utility GIS is extended to perform the role of many standard COTS application components. This sometimes works in the short-term, however, this soon has the result of providing sub-optimal business solutions, for example, building the GIS to be a work and maintenance management application or a content management application. Some GIS applications have become so unwieldy they had to be redesigned from scratch.

In the same way, certain applications should be used for their strengths. For example, a GIS is the recommended environment for much of the asset performance analysis, since it provides the ability to highlight assets thematically on maps according to flexibly driven data queries. This application environment provides an intuitive geo-spatial orientation for presentation of this data that provides insights usually lost in textual reports. Therefore, the recommended solution is to use the GIS to access the wealth of asset data that is managed in various best-of-breed applications through a systems integration bus and enable the decision-makers to perform easy analysis.

By careful selection of appropriate COTS applications, most desired capabilities can be provided at a fraction of the lifecycle cost, more quickly, and at lower risk. When it comes to application architectural design, it is just as important to define what business processes an application will be used to support, as which ones it will not. The scope of the various applications should be defined in business rules that complement the application architecture.

Monolithic ERP Solutions

Some of the major ERP software vendors sell their products as an ‘enterprise solution’ that can supposedly solve virtually all of a utility’s application needs. This approach is unrealistic since there are many specialized needs a utility has that an ERP will not solve such as providing solutions for specialized components such as a GIS, SCADA, system analysis, content management, and CRM. ERP applications are most effective when they are focused on supporting business areas that have proven successful such as General Ledger, Accounts Payable, Purchasing, Payroll and HR. Several other areas of ERP products are not well-supported such as Capital Program Optimization and Management, Project and Contract Work Management, Maintenance Optimization, Work and Maintenance Management, and Field Service Management. In addition, the design of the ERP modules and the associated systems integration often do not suit utility best practices.

A major drawback is the paradigm that ERP products have functional development and are not focused on streamlining the end-to-end business processes. This makes the user experience non-intuitive and the business processes labor intensive – people feel like they spend a lot of resources, time, and energy on ERP products but receive very little in return.

The recommended approach to designing an optimized application architecture is to select a comprehensive set of best-of-breed standard component applications that are intuitive to use, have integrated workflow engines, and can easily plug-and-play into the utility's integration bus. This approach enables applications to be upgraded or replaced without the massive impact, cost and risk to the business that implementing monolithic applications involves. Even the large software vendors are moving to component-based architectures that enable them to implement or upgrade modules without having to change all modules at the same time. ERP vendors pre-integrate their modules to suit their business processes and rules. Although these work well for standard back-office modules, often the asset management related modules are not very mature, and are often not appropriate for utilities.

Finally, having control on the business triggers within smaller application components and using systems integration messages enables the utility to configure the application architecture to provide an overall solution that effectively supports their core business processes.

Data Warehouse Solutions

Data warehouses that use major databases are great tools for consolidating data into an off-line environment where the data can be "sliced and diced" without fear of collapsing the operational applications that support the day-to-day operations of the utility's end-to-end business processes.

However, in this role it is merely another application that supports tactical and strategic business processes and it also needs to plug into the systems integration bus. It is not realistic to assume that a data warehouse will consolidate and thereby solve any data inconsistencies. A data warehouse simply enables the data from disparate sources to be brought together which clearly reveals all the data inconsistencies that exist – not that one should spend huge amounts of money to discover the obvious. Implementing complicated extract transform and load (ETL) processes to populate the data warehouse, which are not part of normal business processes, adds an unnecessary burden to the organization.

RECOMMENDED SOLUTION APPROACH

Define the Enterprise Information Architecture

The recommended set of strategic items a utility need to define as part of an enterprise information architecture are presented in the Figure 1 below. Well-defined information architecture will enable it to effectively manage its data and applications. The components that need to be defined are the following:

1. **Business Drivers and Objectives**
    - Business drivers – what regulatory, customer and other pressures do they experience
    - Business objectives – this includes internal objectives as well as service levels that define the levels of service their customers and regulators requiring

2. Business Processes

- What are the core end-to-end business processes – these enable the utility to add value to their customers, meet their regulatory requirements and business objectives

3. Business Requirements

- Capability requirements – defines the abilities the utility needs to have to enable it to perform the core end-to-end business processes well
- Business information model – defines the major entities that are important for the utility to access or manage to support the core end-to-end business processes well
- Logical application components – the application components that enable the required business capability

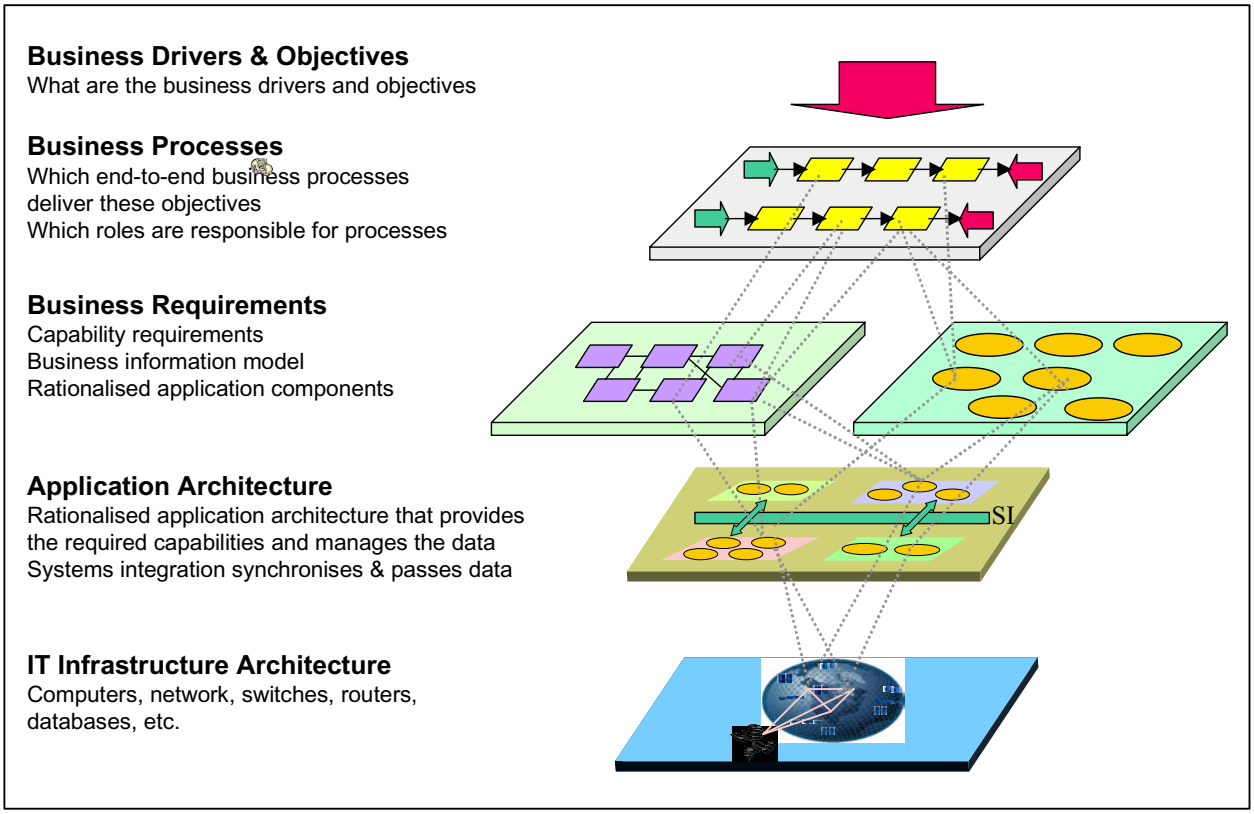

Figure 1: Components of Enterprise Information Architecture

4. Application Architecture

- Rationalized application architecture – limited suite of effectively integrated applications required to support the end-to-end business processes. Recommend the use of Commercial-Off-The-Shelf (COTS) software, that employs modern technology (such as web-based and object design), provide intuitive user interfaces that align with the

workflow, has an integrated workflow and supports a 'plug and play' systems integration bus approach

- Systems integration – used to synchronize data that is required in various applications, and passing data to support the workflow or provide data that needs to be presented in one application and is managed in other applications. Use of an information bus that helps the utility move towards plug-and-play applications architecture is recommended

5. IT Infrastructure Architecture

- Design and effective, reliable and secure IT infrastructure that supports the applications
- Standardize the IT infrastructure components

Data Improvement Approach

The recommended approach to define the how the utility data, and specifically the asset data, can be managed more effectively is given in the following steps that leverages the enterprise information architecture defined above:

1. Business Performance and Data Gaps

- Identify which important business objectives are not being achieved
- Identify the end-to-end business processes that support the achievement of these objectives and those that manage the data that is critical to these processes
- Assess the end-to-end business processes to identify why the objectives are not being achieved – identify end-to-end business process improvement opportunities such as changes in policies, business processes and responsible roles, organizational structures, applications, systems integration, data validation, and data custodianship
- Develop the solution projects - these will address the majority of the important improvement opportunities including the data integrity problems, using increased data validation at point of capture, data synchronization between applications, and improved data responsibilities and audit checking
- Prioritize and phase the solution projects – based on the benefit/cost ratio, strategic importance and dependencies

This approach will enable a utility to define its required business capability improvement projects to enable its end-to-end business processes to deliver the required service levels and business objectives, which will eliminate important data integrity issues.

CONCLUSION

The ability to access and analyze asset data is a critical aspect of making effective asset management decisions. Different types of asset data, often managed in different applications, need to be brought together for this purpose. An effective application architecture solution brings

the required data together into an appropriate environment that enables effective analysis and optimization.

Four sound engineering principles including design standardization, overall system focus, system master planning, and design iterations have been used by utilities for many years and can be applied to developing an effective application architecture that supports a utility's end-to-end business processes. The recommended approach uses these principles during the design and implementation of these solutions, does not follow the old traditional waterfall approach, but instead follows an iterative solution gap analysis around a suite of pre-configured and pre-integrated, standard application components.

A robust approach was presented for selecting and integrating best-of-breed applications to enable decision-makers improved access to data required to make optimized asset management decisions. In addition this integrated solution will enable important data to be managed as part of the day-to-day business processes. There are several problems associated with alternative solution strategies that are often sold as a "silver bullet". These failed solution strategies include departmental solutions, over-extending applications, monolithic ERP solution, and a data-warehouse approach.