Validating a Commercial Product Architecture

Leandre Adifon, V. A. Lentz, Bruce Lerner, Daryl J. Marvin
Otis Elevator, a UTC Company
5 Farm Springs
Farmington, Connecticut, USA

Abstract. Commercial products in existing industries have an architecture. The functional architecture is implicit from the domain, as envisioned by the founder of the company. The only representation of the functional architecture may be enveloped in the physical architecture. The physical architecture has evolved over time to keep up with innovation and changes in the marketplace. As technology is investigated and sometimes subsequently inserted, the occasion arises to more formally reassess the functional architecture and validate the evolved Physical architecture in regard to expanded functionality and new technology. Today, independent modifications in the various pieces of the product drive substantial changes in the physical architecture. This paper addresses the journey of one commercial company to validate the architecture of the flagship product. The methods and approach will subsequently be used with the other product lines. The goal is a modular architecture with standardized interfaces to enable significant decreases in time to market, and reduction in the cost of production and support. The paper addresses 1) the business drivers for taking the time and resources to do the validation, 2) the methods employed and 3) the plan for evolving the engineering and operations within the company to realize the advantages sought when starting the effort.

BUSINESS DRIVERS

What would drive a company to take the time and apply the resources to document an existing architecture? Otis Elevator found two primary reasons to redirect resources that were critical to on-going Product Development: 1) The need to significantly reduce the time to market for new products and 2) the need to optimize the operations aspects for translation to business results. The Senior VP of Operations noticed that ad hoc interfaces that did not support an operational function of the elevator were costing the company a lot of money in terms of drawings, manufacturing processes and field processes. Since there was not an operational basis for the interface one might ask what was the driver for the connection? The answer is immediacy. The customer wanted a handrail. The most expedient solution was to connect the handrail to an existing structural member which might not have been chosen if a 'clean sheet' design was developed. The solution for a one-time customer request was reused for the next request for the same feature, and this immediate fix was not just for handrails. There were other examples, and the costs were adding up.

The Enterprise Model. Elevators are designed by a global engineering organization and no one center designs all the pieces…. Thus there is a lot of coordination. The overlap in the global time zones for North America, Europe, and Asia provides small slots into which communications among all the team members can be accomplished.

The parts are engineered in x places, manufactured in y places, and need to fit together at one installation site. Installation teams are distributed among z regional companies. When the volume is sufficient, installation teams can focus on a particular product.

The pieces to implement the global designs are procured globally. For strategic pieces, there is one supplier providing pieces around the globe. The local manufacturing or logistic center consolidating shipments to customer sites obtains other pieces. The pieces arrive at the installation site in a compact form as a truck full of pallets containing boxes of elevator pieces. The pieces are installed by local mechanics, according to a global installation process and subsequently maintained by global service and repair processes.

Monitoring of the health of the installation, wherever possible, is remote from the installation. Preventative or corrective maintenance can be scheduled based on the reported status of the elevator prior to a shutdown.
Maintenance and repair personnel are distributed in towns and cities and must be capable of maintaining the installed base without regard to the product specifics.

Product Evolution. Elevators are just one part of the 'vertical transportation system' of a building. Stairs, escalators, moving walkways and even laundry chutes might be other parts of the vertical transportation system. The building is another larger system, and the building system managers want oversight and management of a larger set of subsystems within the building. The building owner wants revenue, and while the space taken by the elevator shafts within the building is necessary for getting people to the higher floors, it does not of itself produce revenue. Thus there is always contention between the performance of the vertical transportation system and the amount of space occupied.

Elevators are typically designed for a 20-year life. Buildings last much longer and thus any architecture for the elevator system needs to support modernization of various pieces at end of life. There are major physical parts in the elevator. It is at this level that innovation typically occurs. The major physical parts operate on independent evolution cycles from the system (the elevator). A new elevator system is the appropriate set of major physical parts, at the latest technology level.

Derived Drivers. Customer preferences, differences in city standards and construction methods means that contract engineering might be required to tailor the pieces from the catalogue to any particular installation. This contract engineering takes time and requires resources. The contract engineers need to understand the critical performance characteristics of the original design in order to modify it. Teams installing a variety of elevator products with different pieces and different interfaces require more training, tools and more field manuals than teams installing more standardized pieces. Growth by acquisition and the global nature of the enterprise means that not only did we have to accommodate multiple languages but also different sets of terminology for the various parts of the elevator → introducing confusion to the various engineering and operations teams.

The commitment to identify the correct interfaces on which to standardize, define a mechanism for change, and then implement this standardization across the product line was justified with the following;

- the rationalization of the terminology across the engineering centers and with the field;
- the potential for significant improvement in time to market for new equipment, and for individual orders;
- the potential for reductions in the infrastructure required to support the manufacturing, installation and servicing for the product.

THE JOURNEY

The global engineering centers were called to a strategy meeting, and candidates, alternatives, methods and tools were discussed. A smaller, yet still global core team was identified to perform the bulk of the analysis with the review and contribution of the members of the larger team. Two people were assigned full time for about 6 months. The other 5 members of the core team were contributing about 30% of their efforts. The core team met face to face monthly for 6 months as shown on the schedule in the Development Plan.
The core team monthly meetings were augmented with weekly videoconferences.

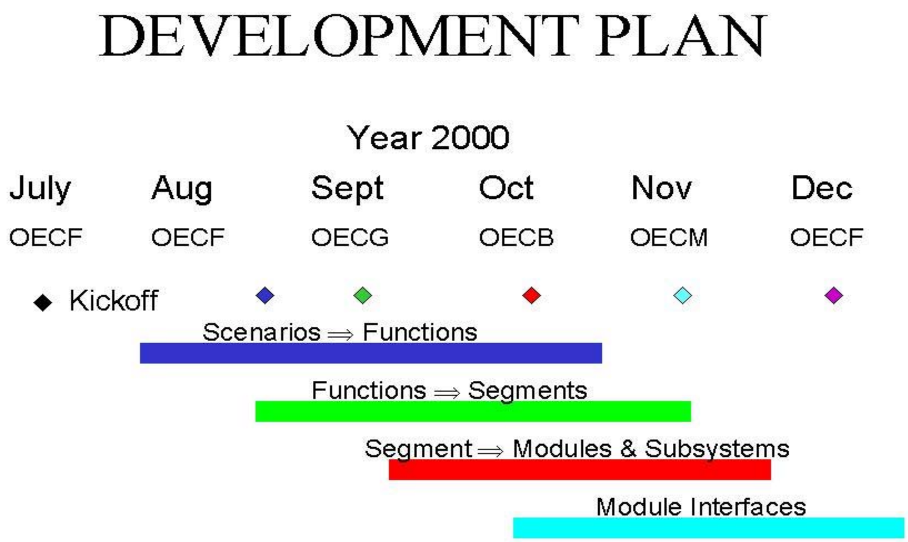

Figure 1. The Development Plan for Validating Architecture

The team applied the standard Systems Engineering methods of operational scenario analysis, functional analysis, physical analysis, and description of the interfaces. The flow of these analyses and the various roles in product development and support are shown in Figure 2 (the arrow chart). The terminology is hybrid based on legacy use of some terms, and introduction of new, globally consistent terminology. As the product passes through the life cycle, the descriptors change, but we develop a consistent entity, which crosses the functional and the physical architectures, that provides the basis for the standard interface; the module.

The modules become the building blocks of the product architectures, and the basis for manufacturing and field support. Cost can be traced from the module to either the segment or the subsystem. The standard interfaces between the modules are defined by Systems Engineering and controlled not only by Systems Engineering, but also by the Engineering, Manufacturing and Field Organizations. Given any one of the three perspectives on the building blocks, there would be fewer building blocks. The aggregation of the perspectives is more difficult to determine, and negotiate, but probably only doubles the number of building blocks.

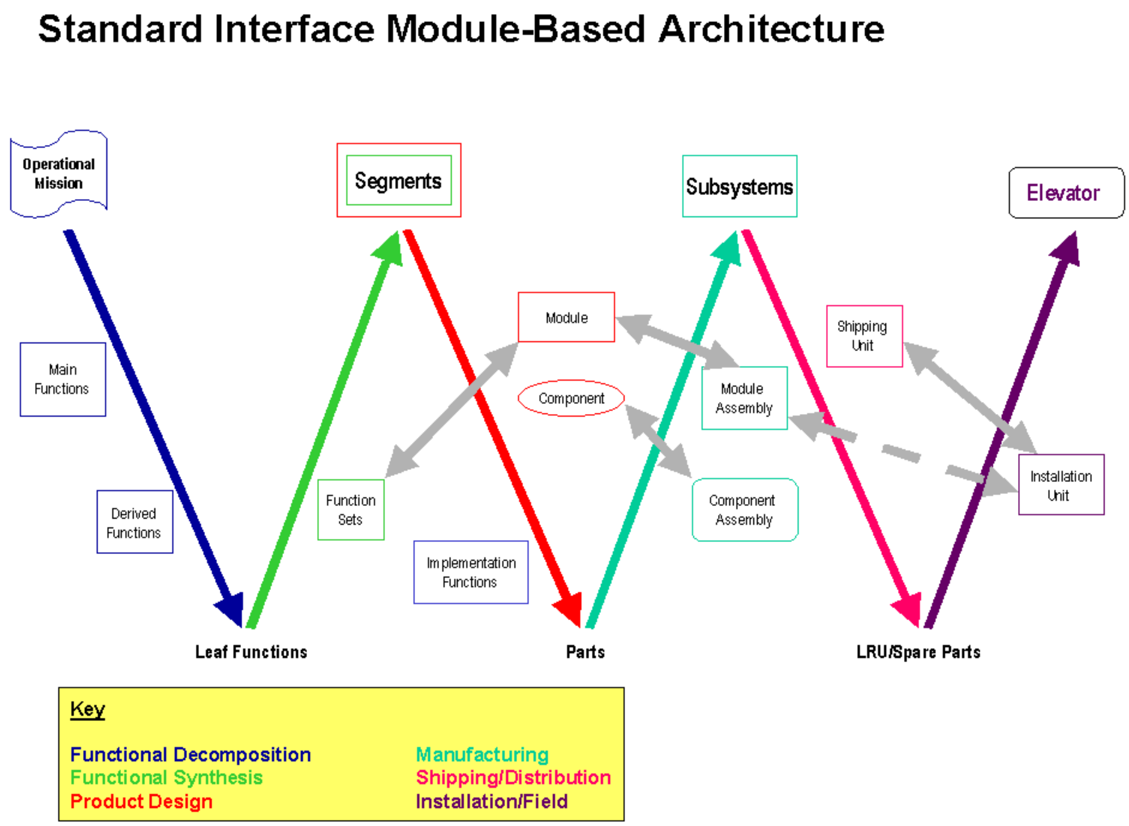

Figure 2. Activities, Process and Roles View of the Architectural Initiatives

Figure 2 touches on three aspects of the initiative. It represents the activities performed in the primary order in which they were performed as shown by the products of each activity. It therefore provides a process overview for the execution of our architectural development. It also identifies the roles within the enterprise that execute these activities. Any standard architecture is required to address the needs of each of the enterprise participants.

Operational and Functional Analyses. Starting with the operational scenarios was a difficult sell. 'We all know them' said the members of the Otis team. In reality, we all knew what we thought they were. Detailing the functions performed in support of the scenarios was also a difficult sell – but critical to start understanding the terminology differences. The functions have also over time taken on the name of the implementation command or part, and the functional description has been lost. During the first meeting there was a great deal of iteration between the functional and the physical analyses. Eventually we accomplished both resulting in the description of functions to a level to support architectural decisions. We split or combined function sets to resolve issues. We have three categories of functions. Main functions directly describe what the product does. Decomposing the main and higher level functions identifies derived functions that do not embody design decisions. The third category includes implementation functions that are required because of design decisions with regard to the physical implementation. The separation of main and derived functions from implementation functions permits us to tightly integrate the functional analysis with the behavior of the product, and separately maintain the implementation functions that are 'prone to changes in technology and design decisions'. The lowest level of each derived and implementation function was 'tagged' as a leaf function. Functional synthesis then became the aggregation of leaf functions to function sets, and eventually to the identification of segments. Segments became the junction of the functional and the physical architectures.

Long discussions during the face to face weeks were needed to assign functions to segments and gain agreement on specific modules. Early in the series of face to face meetings, we established criteria for determining the boundaries between the functional and physical entities. In subsequent meetings with the manufacturing and field folk, we established criteria for changing the boundaries for modules and subsystems according to manufacturing and field drivers. Did the criteria change over time? Not substantially. They were not established in a vacuum, but with attention to the internal needs of the elevator system, and with attention to the external factors such as the impact of technology, the impact of regulatory changes, and changes in the market. Changes in the market might be architectural drivers, or customer preferences for Aesthetics.

Functional to Physical Transition. The segment organizations are responsible for providing the functionality associated with the function sets that have been allocated to them. The implementation entity, which delivers the functionality of one or more function sets, is a module. Modules are designed using components, sub-components and parts. As we leave the engineering world and enter the industrial world, the modules are fabricated, then integrated into subsystems.

Segments and Subsystems. From a physical view, both Segments and Subsystems are composed of (the same) Modules. The Segments offer a transition between the functional and physical worlds, meeting the need for independence from change with coherence of functionality. The Subsystems provide the transition from manufacturing to the field, rationalizing manufacturing needs with the physical and temporal constraints of installation and service.

Physical Analyses. As we reviewed the physical architecture, we decided to retain established manufacturing and sourcing subsystems when it made sense. Thus the new physical architecture is not dramatically different from the old in terms of modules which are the common element between Segments and Subsystems. Primarily, boundaries have been adjusted. We were not really surprised that the result was realignment, not revolution. However, we now have a better understanding of some of difficulties experienced with prior product developments caused by overlapping product functionality and development responsibility. This also helps to sell the revisions, in that the changes can be described as contributing to the solution of real problems.

Requirements. Where are the requirements? The functions identify what the system must do. Design Parameters, the how well and under what conditions, added to the statement of the function, yield the requirements, and vary by the product families. We design for a low rise, a mid-rise, and a high rise product family. For the product description, non-functional requirements are also identified for the standard items such as cost and reliability.

Specific instantiations of the modules, with specific requirements for the product families, will result in module assemblies that have the same interfaces with perhaps different internal characteristics. These assemblies get a part number and a home in the drawing tree for a catalog item. There will be a generic set of spare parts, as well as specific sets. The modules can be packaged independently or together for shipping. The drivers for shipping units are what make sense for the shipping container and the order of installation at the job site. Packaging of installation units needs to be driven by the installation procedures, as the equipment needs to stay in the container as long as possible to avoid damage. Thus the shipping packages should contain installation units used in adjacent installation steps.

Interface Definitions. 'Interface' was a word that meant a variety of things to many of our associates. In our practice, an Interface Control Document captured the physical embodiments of a connection between two design entities. The goal was to lock down this definition early in the development cycle to allow independent design. Often substantial rework was required when the independent designs had evolved separately without an understanding of the underlying functionality. Thus the state of the practice is that interfaces are to be communicated, then not referred to.

The architects decided that the terminology for 'interface' needed to be separated from the terminology for 'link' as shown in the extract from the 'Concepts and Terminology Document'. In the definition for a link the functional and the physical links, link ends were to be identified. Thus as the technology changes, the physical interface can change while the functional interface remains stable.

Element	An entity that is a part of the product.
Encoded Item	An Item that is physically transferred from one element to another element and can be interpreted by a defined code by the receiving element.
Encoding	The definition by which an encoded item can be interpreted. An encoding will have an associated specification.
Entity	A generic, non specific term for anything described in the architecture
Functional Interface	The Functional Links that are input to or output from an element.
Functional Link	A Functional Link carries item(s) from one element to another. It is directional.
Item	An item is input to or output from a Function. An item is carried by a Functional Link and crosses a physical link end.
Physical Interface	All of the physical link ends that are associated with external physical links of an element.
Physical Link	A Physical Link attaches two elements. It is composed of two physical link ends. Items associated with functional links are carried by the physical link. It is generally not directional.
Physical Link End	A Physical Link End is described by its characteristic attributes. These attributes vary based on the type of physical link end (informational, electrical, or mechanical).

Table 1. An Extract of Definitions Related to Interfaces

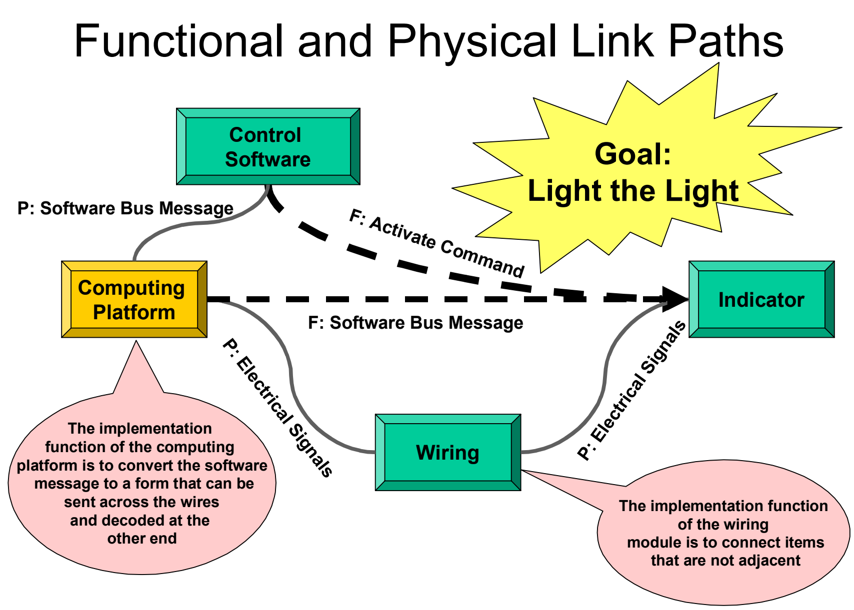

Figure 3. An Example of Functional and Physical Links

Methods and Tools. We used a number of methods and tools to support out analysis and design activities. Our most basic tools were Microsoft Word and PowerPoint, which we used primarily for communication as they are widely available and can integrate the output from a

variety of specialized tools. For the definition of our schema we used UML notation as embodied in Rational Rose. For scenario capture, functional analysis and synthesis and module definition (including interfaces), we used CORE by VITECH.
We originally thought to use the Design Structure Matrices as taught at MIT to analyze the goodness of the resulting architectures. This would satisfy a thesis requirement for one of the members of the support team. Further discussion indicated this to be premature, and we instead used the "The Principles of Design" as advocated by Nam P. Suh (Suh, 1990) also of MIT. For every decision we weighted the solution in favor of Suh's Independence and Information Axioms.

A method that we employed to gain understanding and consistency quickly during our face-to-face meetings was the establishment of criteria for determining the boundaries between the functional entities and physical entities and the mapping between them. This was actually a part of a Structured Decision & Risk Management method by Systems Process, Inc. which we used to make and document decisions. Having the criteria did not keep the discussions from happening. But after the various points of view were tabled, using the criteria brought us to a conclusion – and kept us from repeatedly revisiting the same decision. The Decision Link tool by Telelogic, which implements the SDMRM methodology, was used to document the decisions we made against specific criteria.

The Tests. The architecture has to be validated as applicable not only to new equipment, but also to support specific contracts (with tailoring) and to the modernization side of the business. The architecture had to be validated with use. The application of the standardized interfaces was piloted on a couple of pieces of the elevator that were undergoing significant change during the time this architecting activity was in high gear. The resulting architecture had to be evaluated for impact on the business systems of the organization. On one hand this was an opportunity to rationalize and improve the business systems, but it did introduce change

The Result. The concept of a standard architecture and standardized interfaces is proven for this community. Some of the interfaces have been identified according to the principles of this effort. The terminology changes are beginning to be heard around the globe. Disciplines outside of engineering are actively participating in the finalization of the building blocks.

Metrics to evaluate the effectiveness / progress of insertion of the standard architecture into the enterprise will be tracked for at least the next 3 product cycles. These include times to market, number of standardized interfaces changes, number of top level requirements changes. This initiative has signed up for a portion of the planned reduction in time to market.

NEXT STEPS

The next step is the transition of product designs to the new architecture. Followed by the population of the infrastructure to make it easier for engineering to use standards, than to regenerate the material to represent the pieces, the interfaces, and the requirements for same.

As the product development activities continue over the next few years, interfaces will be standardized, as there is a need to bring a new piece to market. Some of the interfaces will be tightly controlled by Systems Engineering, others will be identified, but interface and link definitions will be delegated to the segment level. It is expected that the functional analysis, and interface analysis will flow down within the segments.

As the modules and interfaces are standardized, 'standard requirements' will be documented for reuse across products. This will eventually lead to increased availability of standard tests. The availability of these reusable standards will contribute to the reduction in time to market, and the increase in the quality of the products in the field.

We have been working to standardize the enterprise information architecture for the last few years, this architecting activity will provide the basis for completing that effort. The next set of changes will be driven by the terminology and information flow to support the various parts of the enterprise.

WHY WILL IT WORK?

This journey satisfied a business need. The architecting activity was performed in response to a pair of business needs: minimizing time to market, and reducing the cost of manufacturing and field support.

The architecting activity supported the broader enterprise community. The results were tested to satisfy a variety of Otis Communities. Thus more than new product engineering had a benefit from the results.

The activity had executive commitment: The Senior VP of Operations initiated the journey, tracked the progress of the team, and publicly supports the interim results of the team.

The analyses had consistency. Individual decisions were made to a set of published criteria, and the rationale was captured. The information is available for subsequent engineers and architects to use in assessing the impact of changes whether they are involved in new product, or contract engineering or trying to save money in the factory.

****

REFERENCES

Suh, Nam P., *The Principles of Design*, Oxford University Press, 1990

BIOGRAPHIES

Leandre Adifon is Senior Manager Systems Engineering for Otis Elevator Company. Previous responsibilities include Manager of Advanced Systems Research, and R&D Manager for Otis in Italy (for seven years). For two years he was a member of European Elevator Code Committee. Prior to Otis, Leandre was a Consultant in Robotics and Machine-tools. He holds a BSc. and an MSc. degrees in Mechanical Engineering. He has also earned an MSc. in Finance Technology and an MBA, both from Rensslaer Polytechnic Institute. He is a member of the Society of Engineers in Italy and an INCOSE member. He speaks fluently five European languages.

V. A. Lentz is responsible for Systems Process and Infrastructure for Otis Elevator Company. She has been with United Technologies since 1996. Her focus is the use of Systems Engineering in commercial enterprises. Prior to UTC, she spent 30 years at IBM Federal Systems, LORAL, Lockheed Martin building large, unprecedented computer-based systems such as Global Positioning System Control Segment. She was also responsible for Systems Engineering Technology, Process and Training. Ms. Lentz was President of INCOSE in 1996, a recipient of the Founders Award in 1999, and represents UTC on the INCOSE Corporate Advisory Board.

Bruce Lerner is a Principal Systems Engineer at Otis Elevator, a United Technologies Company. He has 14 years experience developing embedded Software for communications and control systems and facilitating Software Process Improvement activities. His last five years have been applied to Systems Analysis and Architecting and the Systems Engineering process. Bruce is a member of INCOSE and the IEEE Computer Society.

Daryl J. Marvin is a Principal Engineer at Otis Elevator, a United Technologies Company. He has 10 years of experience in industry with Superior Electric and Otis Elevator. At Superior Electric, he worked on variable reluctance motor control and variable reluctance resolver design and signal processing. At Otis Elevator, he first worked on drive control algorithms, progressing into drive architecture, elevator control architecture, and finally to overall elevator system architecture.