A PROCESS FOR FUNCTION BASED ARCHITECTURE DEFINITION AND MODELING

A Thesis
Presented to
The Academic Faculty

by

Michael Armstrong

In Partial Fulfillment
of the Requirements for the Degree
Masters of Science in the
School of Aerospace Engineering

Georgia Institute of Technology
April 2008

A PROCESS FOR FUNCTION BASED ARCHITECTURE DEFINITION AND MODELING

Approved by:

Dr. Dimitri Mavris, Advisor
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Elena Garcia
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Danielle Soban
School of Aerospace Engineering
Georgia Institute of Technology

Date Approved: March 27, 2008

For My Family

ACKNOWLEDGEMENTS

I would first like to thank Dr. Dimitri Mavris for the opportunities he has provided me to gain insight to the needs of industry, and for his direction in the conceptualization of this work. Dr. Elena Garcia and Dr. Danielle Soban also are to be recognized for their guidance and support with this work. Their insight and discussion of this topic helped crystallize and solidify the core thesis and motivation for this work and streamline the research performed. I would also like to recognize all the members of the 2006-2007 Optimized Aircraft Power Architecture (OAPA) Grand Challenge team, with special thanks to Cyril de Tenorio, for their help in laying the foundation for this work. My thanks go out to the members of the Energy Optimized Aircraft and Equipment Systems (EOASys) program committee from the AIAA, including Dr. Lester Faleiro, for their guidance and support of this research. Thanks also to Delta Airlines Flight Operations for their assistance in helping me get a feel for the complexity of aircraft systems integration. Finally, special thanks to my wife, Megan, whose encouragement and motivation has offered immeasurable support in the opportunities that I have had here at the Georgia Institute of Technology.

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

LIST OF ABBREVIATIONS

ACARE	Advisory Council for Aerospace Research in Europe
AIAA	American Institute of Aeronautics and Astronautics
ARM	Adaptive Reconfigurable Matrix of alternatives
ASDL	Aerospace Systems Design Laboratory
BF	Boundary Function
COTS	Commercial Off the Shelf
DSM	Design Structure Matrix of Dependency Structure Matrix
ECS	Environment Control System
EOASys	Energy Optimized Aircraft Systems
FMM	Functional Mapping Matrix
FR	Functional Requirement
GFE	Government Furnished Equipment
ICAO	International Civil Aviation Organization
IF	Induced Function
INCOSE	International Council on Systems Engineering
I/O	Input/Output
IPD	Integrated Product Development
IPT	Integrated Product Teams
IRMA	Interactive Reconfigurable Matrix of Alternatives

MADM	Multi-Attribute Decision Making
MEA	More Electric Aircraft
MOET	More Open Electric Technologies
NAP	Noise Abatement Procedure
OAPA	Optimized Aircraft Power Architecture
POA	Power Optimized Aircraft
RAT	Ram Air Turbine
S/W	Software
TEOS	Technologies for Energy Optimized Equipment System

SUMMARY

Developments in electric technologies have the potential to increase the efficiency and performance of commercial aircraft. However, without proper architecture innovation, technology developments at the subsystem level are not sufficient to ensure successful integration. Adaptations to existing architectures work well when trades are made strictly between equivalent systems which fulfill and induce the same functional requirements. However, this approach does not provide the architect with adequate flexibility to integrate technologies with differing functional and physical interfaces. Architecture redefinition is required for proper implementation of non-traditional and innovative architectural elements.

A function-based process for innovative architecture design was developed to provide flexibility in the definition of candidate architectural concepts. Tools and methods were developed which facilitate the definition and exploration of a function-based architectural design space. These include functional decomposition, functional induction, dynamic morphology, adaptive functional mapping, reconfigurable mission definition, and concept level system installation. The Architecture Design Environment (ADEN) was built to integrate these tools and to facilitate the definition of physics-based models in evaluating the performance of candidate architectures.

Using functions as the foundation of this process assists in mitigating assumptions which traditionally govern architecture structures and offers a promising approach to architecting through flexible conceptualization and integration. This toolset provides the framework wherein knowledge from conceptual, preliminary, and detailed design efforts can be linked in the definition of revolutionary architectures.

CHAPTER 1

INTRODUCTION

The current operating environment for commercial airlines imposes increasingly stringent requirements on aircraft performance. The prospective increase in the volume of air travel and in public environmental awareness has lead to demands for quieter, cheaper, and more environmentally friendly flight. These market demands, in concert with ever increasing fuel prices, higher demand for non-propulsive power, and more stringent federal regulations, impose stark demand for lighter, more efficient, and cleaner technologies to be integrated in the aircraft. Aircraft are simply expected to do more, and to do it more efficiently. As a result, aircraft designers are challenged to design innovative airplanes by integrating new, revolutionary technologies. The considerations introduced in this paragraph will be discussed in this section.

Demands on Aircraft Design

The steady increase in yearly revenue passenger miles, as displayed in Figure 1, increases the load on current airports and airspace and the amount of emissions produced by air traffic. This growth represents an increase of nearly 50% in revenue passenger miles in the last decade. As this growth continues, all sectors of the aerospace industry must adjust to meet new demands and challenges.

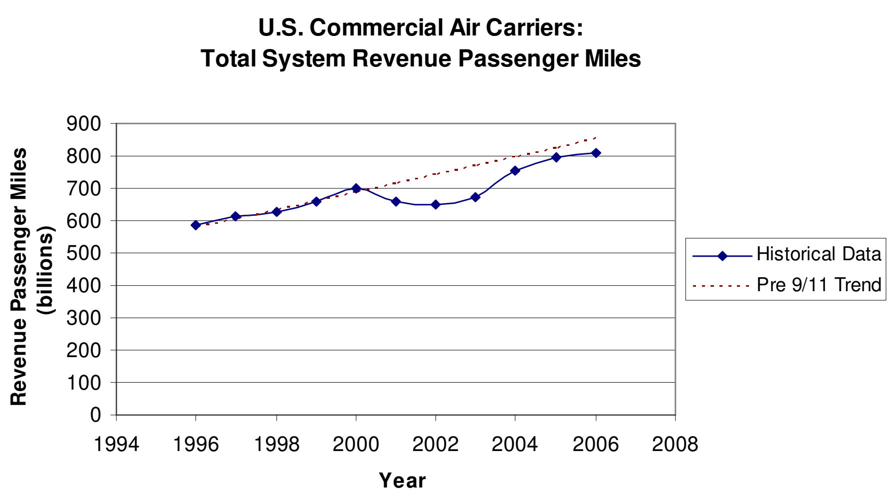


Figure 1: US Commercial Air Traffic: Revenue Passenger Miles [1]

With this increase in the use of air travel, regulations are being developed and enforced to maintain the environment. A growing number of airports have taken measures to limit the amount of noise as shown in Figure 2. These restrictions include noise abatement procedures (NAPs), curfews, fines, specified limits, quotas, and other restrictions imposed by the International Civil Aviation Organization (ICAO) [2]. These restrictions impact the design and use of airports and airspace, the alteration of procedures, and any changes to aircraft design. New technologies and aircraft configurations have emerged with the intent of providing means to reduce aircraft noise.

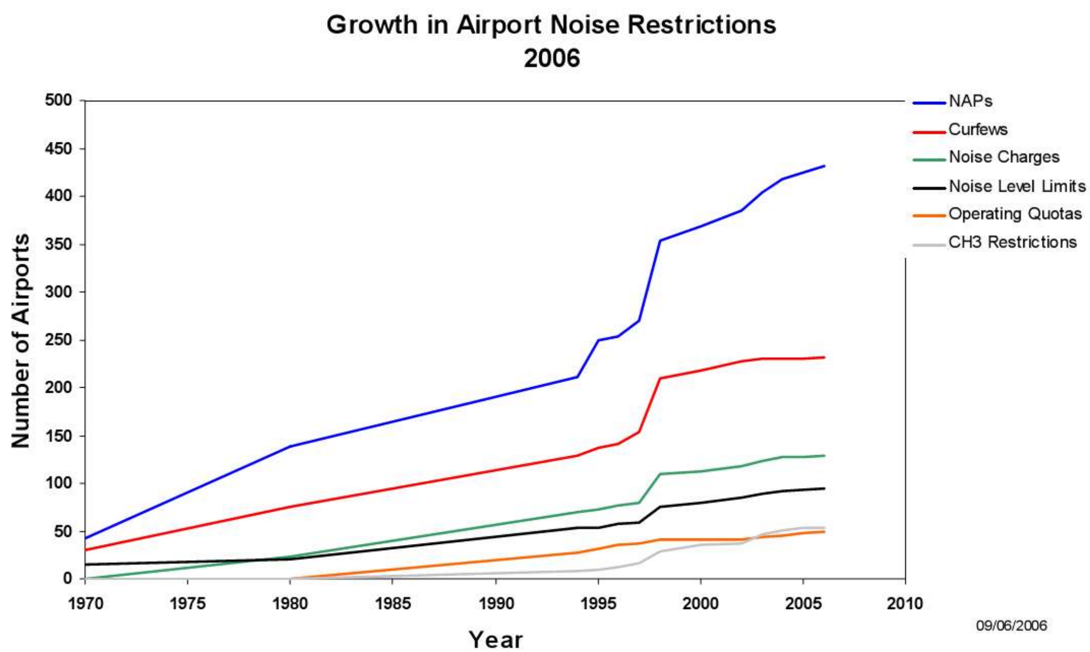


Figure 2: Growing Airport Commercial Noise Restrictions [3]

The air transportation industry makes up only about 2% of the total air pollution produced. However, with an estimated 5% increase in air traffic in the next 10 to 15 years, the impact of air transportation on the environment must be monitored and controlled [4]. The Advisory Council for Aerospace Research in Europe (ACARE), which is a committee consisting of government and industry representatives from across Europe, has set some environmental goals to be accomplished by the aerospace community by the year 2020. These include the reduction of carbon dioxide emission by 50% (equivalent to 50% reduction in fuel consumption) and the reduction of nitrous oxide emissions by 80% [5]. These goals call for significant improvement in the efficiency in performance and integration of new technologies. In the last 30 years the specific fuel consumption has only decreased by 35% [6].

Designers are not only concerned with their product's interaction with the external environment, but also looking to improve the internal environment of the aircraft. The aircraft cabin must be designed to provide a safe and enjoyable experience for the ultimate customers of the airframer (the passengers). In order to provide passenger amenities, power is used for operations other than performing the fundamental functions of the aircraft. In order to increase the marketability of the aircraft within today's air transit system, in-flight entertainment systems, meal services, increased cabin pressure and humidity, and other features are requirements introduced in commercial aircraft design. At the same time as the overall fuel consumption per seat is intended to decrease, as seen in Figure 3, each passenger increases the comfort requirements per seat.

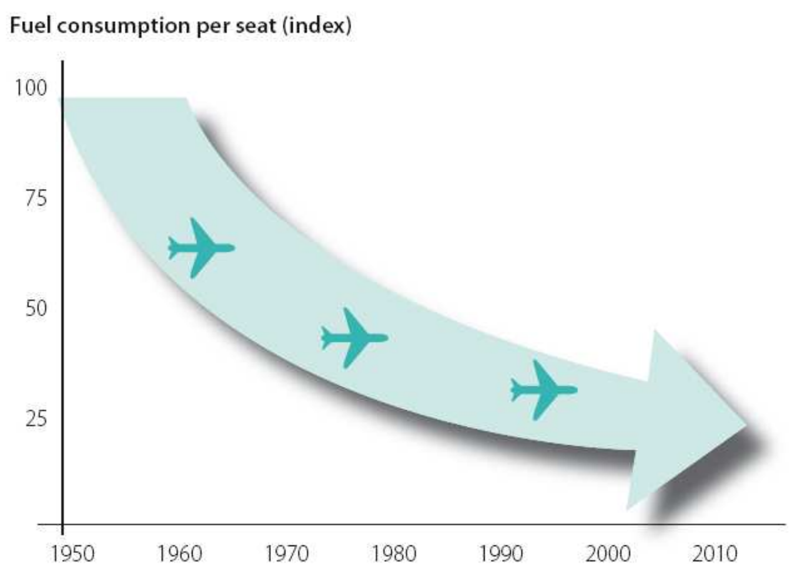


Figure 3: Fuel Consumption per Seat in Long-Range Aircraft [6]

In a report to the Aviation Subcommittee of the Committee on Transportation and Infrastructure of the House of Representatives, John Heimlich, the vice president of the Air Transport Association of America stated that, "At today's consumption rate, every penny increase in the price of a gallon of jet fuel drives an additional $195 million in annual industry operating expenses. In fact, from 2000 to 2005, the industry's fuel tab doubled, from $16.4 billion to an estimated $33 billion, even though it consumed less thanks to increased fuel efficiency [7]." This dramatic increase in jet fuel prices in the last decade can be seen in Figure 4.

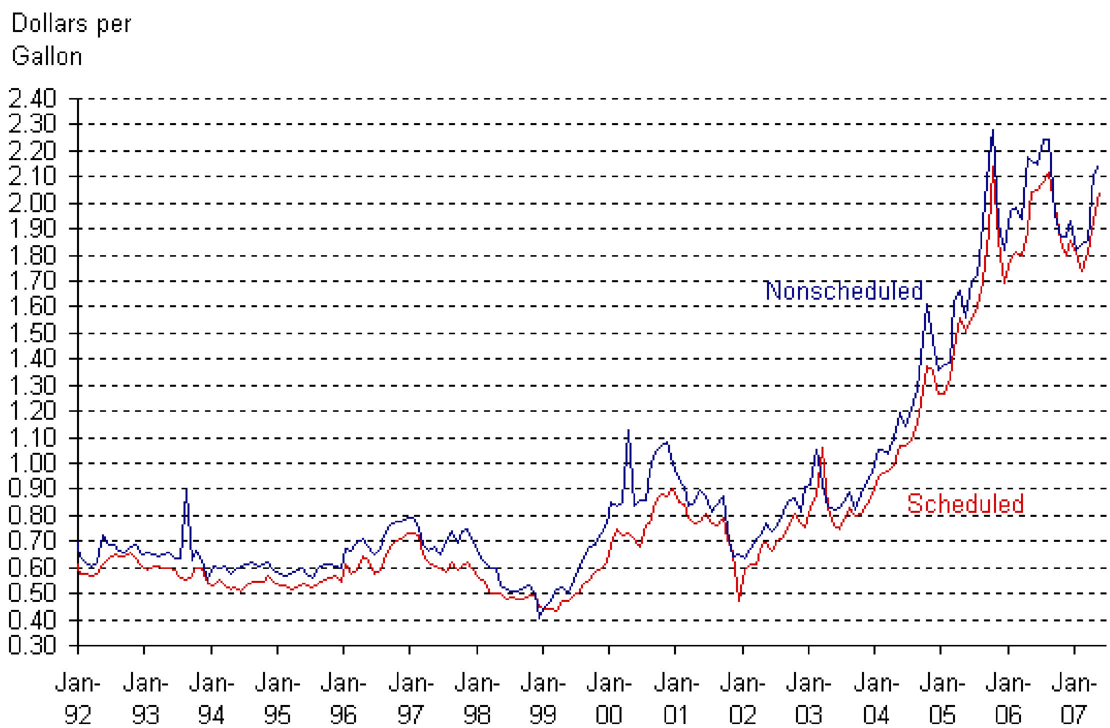


Figure 4: Domestic Airline Jet Fuel Price [1]

Definite trade-offs must be made. The solutions to all these demands come with underlying costs. Reducing the aircraft noise and emissions is paid for by fuel efficiency. Increased demand for non-propulsive power removes energy from the engine aircraft power plants, thereby raising fuel consumption and imposing new requirements into the aircraft architecture. The implementation of a new technology can impose unforeseen side effects which negate all positive impact. The application of new technologies must be considered at the aircraft level to determine the overall benefit of the technology.

In order to meet this growing demand under increasing stiff performance requirements designers must produce a competitive product which can compete in the market by meeting and/or exceeding the regulations while still providing an attractive passenger environment. This is done by the exploration of advanced concepts and technologies which have the promise of providing this advanced capability.

Electric Aircraft Technologies

Historically, non propulsive power was distributed and used by means of hydraulic, mechanical, electrical and pneumatic power. These means reflected the technologies available. Mechanical and hydraulic connections were generally used for flight control operations and pneumatic energy was bled from the engine to provide air for other aircraft functions. Thus the aircraft became a hybrid system, employing many different types of power. This hybrid power structure is entrenched in aircraft design.

With the development of power electronics there has been a recent emphasis on the implementation of electric technologies on new aircraft designs. Advances in electric technologies indicate that increased performance is available with their implementation in commercial aircraft [8]. Conventional technologies are purported to be reaching the point at which increased performance is not available, while electric technologies are at the point in which they promise substantial benefits in performance [9]. This concept is notionally described in Figure 5.

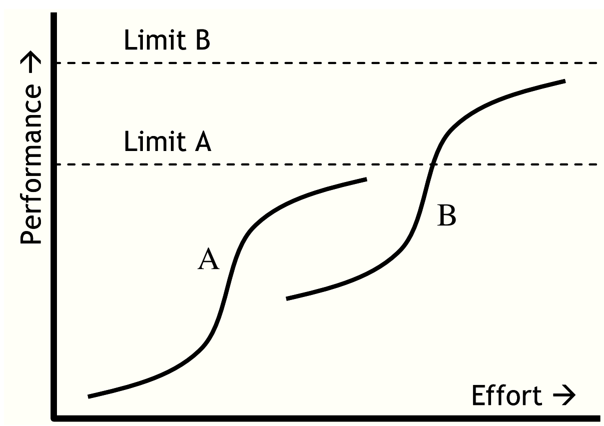


Figure 5: Technology S-curves [10]

In this figure the two curves represent two competing technologies: A and B. An increase in performance in each technology (designated by the y axis) comes with the expenditure of some effort (designated by the x axis). Each technology has a natural performance limit and as the performance nears this limit the expenditure of effort to increase its performance begins to yield less and less benefit. If a new technology is applied which has a higher natural performance limit, there is the potential to exceed the performance of the initial technology with less effort. Although this example is simply notional, it represents the school of thought which supports the exploration of electrical aircraft technologies.

Groups like the More Electric Aircraft (MEA) by the Air Force Research Laboratory, Power Optimized Aircraft (POA) and More Open Electric Technologies (MOET) organized by the European Union, and the Energy Optimized Aircraft Systems (EOASys) program committee from the AIAA have begun to address the issues and benefits of the implementation of more electric technologies in aircraft design. Some of their conclusions from the POA research initiative give insight as to what needs to be done to truly utilize technologies which promise to increase overall aircraft efficiency and reliability.

In June of 2006 a forum (Technologies for Energy Optimized Equipment System, TEOS) was sponsored by the European Union Power Optimized Aircraft Consortium, in which companies discussed technical developments occurring with electric technologies [11]. Some of the conclusions from this meeting were [8]:

1. Electric technologies have potential for superior performance
2. When implemented within conventional architectures, electric technologies yield only a fraction of their potential benefit
3. Many new technologies are at demonstration level
4. Many challenges and issues still need to be addressed
5. Functional thinking is needed for true integration

The scope of work outlined here is to address the 2nd and 5th items of conclusion from the first TEOS forum. These two points address the need to have a means to integrate technologies in aircraft architectures. New architectures are needed, and the integration of new technologies in these architectures must follow a functional rationale. A means to create innovative architectures based on the aircraft functions would allow for the development of better aircraft.

Research Questions

The issues addressed above raise the following questions:

1. What assumptions are made early on during the design process which define the design of aircraft system and unduly constrain the design?
2. What is an architecture?
3. How does architecture play into the effectiveness of the overall design?
4. How does a functional perspective increase the system architect's ability to produce innovative architectures?
5. How do functions need to be formulated in order to allow flexibility in architecture definition?
6. What would be the process of architecture definition? What decisions need to be made and what are the impacts of each decision?
7. What tools are necessary to make these decisions?
8. How can these tools be made to interact appropriately without limiting design freedom?
9. How can a process for architecture definition be tailored to produce architectures in an automated manner?

Thesis Scope

This thesis will review the current understanding and use of systems engineering in architecture design and definition. A functional perspective is proposed to provide the framework which facilitates definition and redefinition of product architecture. Principles of functional architecture definition are explored and tools and methodologies are identified and implemented wherewith an architecture designer can make the decisions necessary to completely define the architecture. This thesis will detail the embodiment of these tools in a software interface which enhances the designer's ability to explore the architecture design space.

It should be noted that, although these principles were developed in the context of commercial aircraft design, these tools and methodologies can be applied to the definition of any complex system.

The general concept of the airplane is extremely mature. The future improvements on aircraft performance capabilities rely on the discovery and implementation of novel technologies or the utilization of the technologies that are already applied on the aircraft to their fullest potential. This involves considering designs which were previously disregarded as being too radical or infeasible by exploring all the possible solutions available. As computer technology increases the speed in which analyses can be performed, designers have the ability to explore the deeper reaches of the design space with more fidelity. They can take details into account that were generally associated with later stages of design and predict performance to a higher level of confidence. This project is to provide the process by which this design space can be defined and explored in search of the optimal aircraft design.

The motivation behind the research and process development detailed in this paper can be simply understood by organizing it into three sections, observation, hypothesis, and approach.

- Observation:
    - Conceptual architectures are defined by assumptions and generalizations regarding the elements within a system and their relationships.
    - These assumptions affect the traditional process of a functional allocation, the detailed description of that physical architecture, and the performance calculations based on traditional systems sizing tools.
- Hypothesis:
    - A functional breakdown which relates functions to physical form can provide a more flexible framework for architectural trade-offs and prevents architectural generalizations.
    - With this functional framework, widely variant physical architectures can be quickly defined and architecturally unique models can be generated.

    - Integrating function based architecture definition tools in a common platform will provide fluidity to knowledge integration.
- Approach
    - Modifications to existing tools and methods will allow them to logically interact to facilitate the definition of a function based architecture development process.

Background chapters 2 and 3 discuss the observations which helped yield this architecture design process. Chapters 4 through 7 address how functions must be formulated to allow them to be flexibly employed in architecture definition and the process, tools, methods used to apply these functions to develop descriptions of physical architectures. A conceptual example is applied in chapters 8 and 9. In this example a design space is defined and 3 conceptual architectures are developed.

Summary

In the face of increasingly stringent performance requirements and constraints, the commercial transport aircraft must provide increased comfort, reliability, and availability in order to compete in today's air traffic network. Efficiency in power consumption is paramount in the design of new aircraft due to the increased demand on performance and the steep increase in conventional fuel prices. In order to operate in this competitive and demanding environment new technologies are emerging with the promise of providing increased capability and efficiency. However, in order to effectively employ new technologies and the framework in which they operate, the aircraft architecture must be adapted to take advantage of all improvement possible from new technologies.

Following the guidance provided by the Power Optimised Aircraft initiative in the European Union, this thesis focuses on architecture definition with a specific focus on how functions can be used as the core defining element. This will lead to a process and toolset design with the intent of driving architecture generation with functional requirements.

CHAPTER 2

BACKGROUND: COMPLEX SYSTEMS DESIGN

Complex Systems

Complex systems design is an intricate and highly constrained decision making process which results in a description of a group of elements forming an object or process [12]. Each decision details specific physical or behavioral attributes of elements within the system. In this manner the system begins to conceptually or physically "take shape." However, in the process of defining this system each decision imposes requirements and limitations which impact other decisions that have to be made. As a result, the freedom of a design team to make any changes to the system becomes extremely difficult. "Hard points" are also imposed on the system. This occurs when elements, interacting with other elements with fixed definitions, are forced into compatibility regardless of detrimental side effects [13]. Something must change in order to make all of the elements work together as illustrated in Figure 6.

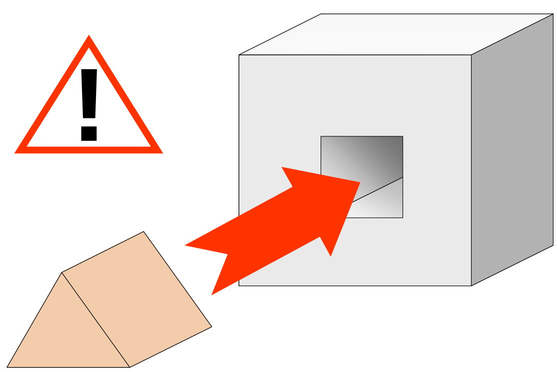


Figure 6: Forced Integration

A designer must be able to make the correct decisions to ensure the success of his/her product. Design decisions are conclusions or judgments based on some body of evidence in order to bring some physical, operational, or functional definition to a product [14]. Therefore, the effectiveness of a complex systems design team depends on its ability to achieve rational solutions to design challenges by effectively formulating the problem and rigorously exploring promising solutions.

The International Council on Systems Engineering defines a system as “a collection of components organized to accomplish a specific function or set of functions [15].” This general definition can be applied to various types of systems. Both an automobile and the internet are classified as systems. However, each has a vastly different form and must be approached in very different ways. Thus, a distinction should be made between a complex system and a system of systems.

A complex system, or monolithic system, is comprised of components which only operate within the context of the entire system. These system elements are not intended to be used independently from the system as a whole [16]. An example of a monolithic system is a personal computer. Many elements of this complex system can be qualified as systems on their own (hard drive, graphics card, keyboard, mouse, etc.) and they all operate to fulfill some task within the overall system. Thus the system (computer) is comprised of system elements (mouse, hard drive, etc.). However, each of these system elements is intended to be used only within the context of the system. The monitor, for example, could be described as a complex system, but it fulfills no function unless it is integrated with the other elements of the PC. The computer is not a system of systems but a monolithic system consisting of exchangeable complex elements. The operation of the system relies on the performance of the individual system elements, and the system elements are intended to perform functions within the framework defined by a system.

A system of systems consists of a group of autonomous elements which can and do operate and fulfill functions independently from the conglomerate system [16]. The elements within a system of systems operate to fulfill specific functions which are not necessarily directly determined by the system of systems. They also have been described as a physically distributed group of elements which interoperate by means of central or distributed management [17] [18], indicating that the elements within the system can be and are generally geographically distributed. The internet, the US Missile Defense

Network, and the World Wide Air Transportation Network are excellent examples of systems of systems. The elements within these systems fulfill functions (often multiple elements fulfilling the same function), independent of the system as a whole, which combine to fulfill an overall task.

To distinguish between a system of systems and a complex system, a complex system is defined as a group of dedicated, interrelated elements which are organized to fulfill specific functions, while a system of systems is defined as a group of independently operating elements whose combined capability fulfill an overall function. For the purpose of this paper the development of process and theory was not directly intended for application to a system of systems. This allows the design of a complex system, like an aircraft, to be approached from a perspective in which elements are dedicated to have capabilities specified by the system as a whole and not as a combination of elements whose combined capabilities are applied to an independent system.

Complex System design

The process of complex system design is made up of multiple steps which are fulfilled by distributed entities within an organization. These steps include pre-design, conceptual design, system level design or preliminary design, detail design, and manufacturing. This research work is directly concerned with the design phases of conceptual and preliminary design in which early physical attributes of the system are defined. This perspective approaches the concept of design after the pre-design activities of problem definition, requirements analysis, and problem specification have occurred. The boundaries between conceptual, preliminary, and detail design are somewhat vague. At some point in conceptual design, when enough alternative designs have been considered and compared and the company feels confident with the potential designs and are willing to invest more resources, a larger group of specialists are assigned to the design to develop the concept further [19].

Conceptual Design

Formulation of the problem occurs primarily within the conceptual design phase in the design process. Although little to no hardware is produced during this phase, it is considered to be the most important phase of the design process [20]. Early decisions impact the overall incurred cost due to the impact they have on later design phases as well as future labor, materials, manufacturing, and overall life cycle of the product. Quality must also be engineered into the product during the early phases of the design and greatly impacts the overall project cost. These early design phases also determine the ability of a company to introduce the product into the market quickly [21].

Conceptual design is not intended to guarantee optimal system performance [22]. Within this design phase, a framework is developed wherein engineers, manufacturers, and customers can operate comfortably, can agree to operate, and can pursue a more detailed definition. Conceptual design considers the overall understanding of the primary functions of the system and investigates to see if the requirements can be met and in which ways this can be accomplished. The deliverables of conceptual design are typically computer or paper-based, in the form of descriptions, reports, and mathematical models [22].

This phase in product development generally includes three steps: problem/project definition, alternative generation, and alternative selection [21] [23] [26]. Scholars differ in their illustration and definition of the exercises involved in each step in product development but all require a process in which the three distinct tasks take place.

Moir and Seabridge illustrate the conceptual design phase as shown in Figure 7.

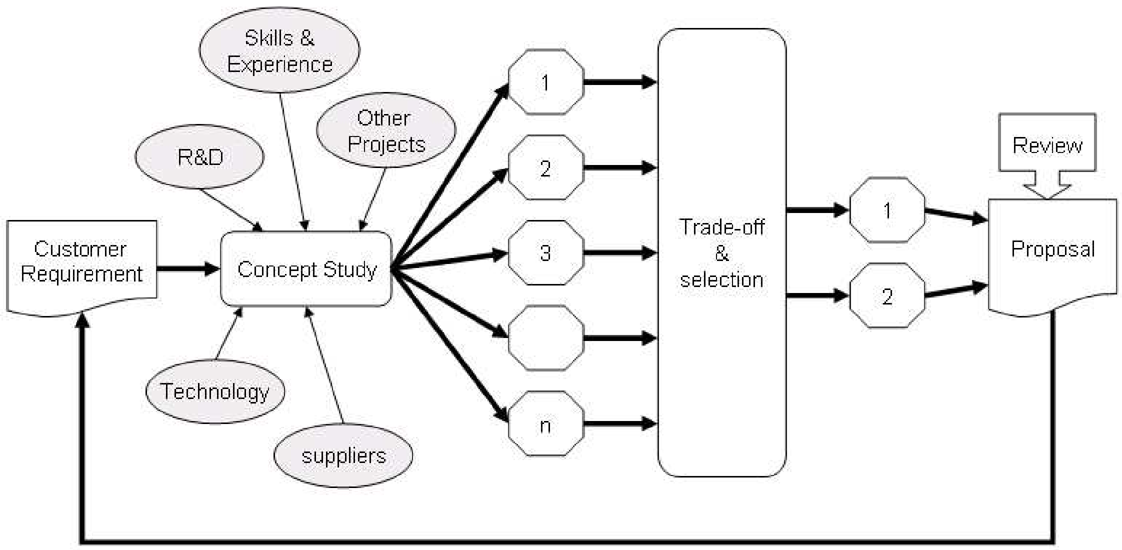


Figure 7: Moir Seabridge's Concept Phase Process

The problem definition portion of the design phase takes place in the blocks entitled customer requirement. Understanding customer needs is extremely important in the conceptual design phase. The eventual success of a project is determined entirely on the designers' ability to ascertain and predict what the customer will need when the product is deployed. As seen in Figure 7, this information is taken into account in order to develop *n* design alternatives. These design alternatives are evaluated and compared in order to determine the most promising solutions which are to be further developed. In order to generate these alternatives, brainstorming tools, morphological studies and functional decomposition can be used following an exhaustive search to gather as much data as possible [21].

The product of the conceptual design process is generally seen as one or more possible solutions. The detail of these solutions depends on the maturity of the basic technologies put into operation and the type of design project [23].

System Level Design

In many conceptual design models there is an intermediary step between the concept development phase and detail design, in which the physical structure, geometries, and tolerances are determined and detailed drawings are drafted. This intermediate step is called the embodiment design [24], preliminary design [21], design definition [22], or system level design [23].

Generally, during the traditional system level design phase the physical elements are arranged into subsystems within the system creating a compartmentalized architecture which carries out system functions [22]. These subsystems are also modeled in detail in order to size the systems and run analyses reflecting on the system level metrics. The analytical design phase is also an appropriate term for system level design because it is the period in which most of the analytical analysis is performed [21].

This segment of the design process determines the relationships of the elements within the system and how they will be interrelated within the fixed conceptual framework determined earlier in the design process. It should be noted that decisions on this level sometimes call for the decisions made earlier, during conceptual design, to be readdressed and altered [21]. However, this reconfiguring on higher levels becomes increasingly difficult as the conceptual physical form begins to take shape. Thus, as an engineer begins to understand the product more fully, the ability to change the structure and form of the design diminishes.

Architecture

Architecture is a fundamental, defining characteristic of every system and has a large impact on its ultimate performance. MIT's Engineering Systems Design Architecture Committee defined system architecture as a description of elements within a system and the interactions between those elements [25]. Other definitions describe system architecture with emphasis on structure and interaction. Maier, Sage, and Lynch define architecture as a grouping of components joined together in a way to fulfill some task that no single element can fulfill individually, or the means by which proper communication and interaction between elements within a system is achieved [16] [26]. Ulrich and Eppinger place more emphasis on conceptualization and standards in architecture design by defining architecture as "the scheme by which the functional elements of the product are arranged into physical [elements] and by which the [elements] interact [23]." Addressing all of these perspectives, architecture denotes entities and their underlying structure whose combined attributes accomplish a task or sets of tasks. All products have a structure and fulfill functions and, in turn, are defined by an underlying architecture [27].

Varying levels of architecture pre-definition cause architectural concepts to emerge during different portions of the design process [23]. Architectures can come forward through dedicated architecture definition exercises or can be altered and adapted from previous concepts. The means by which this architecture definition takes place is generally determined by the maturity of the technologies to be implemented and the level of definition to the project previous to conceptual design. Architecture design is crucial to the success of a project because of its impact on the ability of designers to efficiently and effectively develop new products [23].

This traditional process of creating a new architecture or "architecting" is the means by which a scheme for generalizing elements within a system and their relationships is defined. This scheme dictates the functions which are fulfilled by specific physical elements and the means of interaction with other subsystems [22].

With the definition of a traditional systems architecture, new sets of standards and interfaces are delineated. Functions are grouped together to form tightly linked "chunks"; a chunk being a subset of system elements, or subsystem [28], which represent the physical building blocks of the system [23]. These chunks are defined depending on their roles or functions within the system and in a way which minimizes the interactions between subsystems. This is called clustering [29]. In the definition of the subsets of system elements, system architects define where tight physical relationships will occur. This allows the architect to determine the limit and effect of a change within one subset on another. These subsystems are laid out physically to determine the rough geometric relationships in which interactions are explored [23].

This method of architecture development defines levels of modularity by dividing design responsibility between entities within an organization. The "systems architecture" is seen as a framework in which these systems with predefined boundaries interact and wherein external interdependencies remain independent of the physical definition of the elements within the subsystems [22]. Following this methodology, assumptions must be made regarding the interactions between elements within the system. These defined relationships become standards of component interaction. If these standards cannot be applied the architecture must be expanded to handle new interrelationships.

Perspective and predefinition play major rolls in the use and development of complex systems architectures. Defining a traditional systems architecture for a complex product can also adversely affect the applicability of the architecture. Working within a fixed architecture imposes limitations on the performance of the system. Redesign, evolutionary design, and derivative design are all exercises which generally require definition within a fixed architectural scheme. Applying revolutionary technologies to previously defined architectures can introduce complex interactions which significantly change the interfaces of the system. The modular architecture breaks down when changes within an individual module induce requirements upon a different module along lines not predefined by the intermodule interfaces [28]. A breach of predefined systems definitions can have detrimental impacts on the ability of designers to predict the actual performance of the product. This is significant because most design practices assume a pre-existing architecture framework [30]. Revolutionary systems require creative methods of architecture definition. The implications and limitations of the traditional perspective to architecture definition are discussed in chapter 3.

Optimally, the complex system would be decomposed to a "harmonious state", in which "all elements are divided into unique modules and … all intermodule relationships are … completely described in interface descriptions that also fully describe the emergent system level characteristics [28]." In order to accurately represent the product, architecture models must capture the relationships between the fundamental elements in a way which can describe all attributes and combined performance of the product. Furthermore, these relationships should be described in a way which is not subject to the breaching of constructs or assumptions with the introduction of new elements into the system.

Design Knowledge

For all elements to be adequately captured within this ideal architecture modularization, adequate knowledge is required to fully describe the elements and their interfaces. Each portion of the design process has specific, significant impacts on the success of the resulting design and contributes to the effectiveness of the design. However, the early design phases of the process of product definition, like conceptual design, have the greatest impact on the cost committed for the project as a whole. David G. Ullman displays this axiom in a relationship between length of time spent on the design and the project cost (Figure 8) [31].

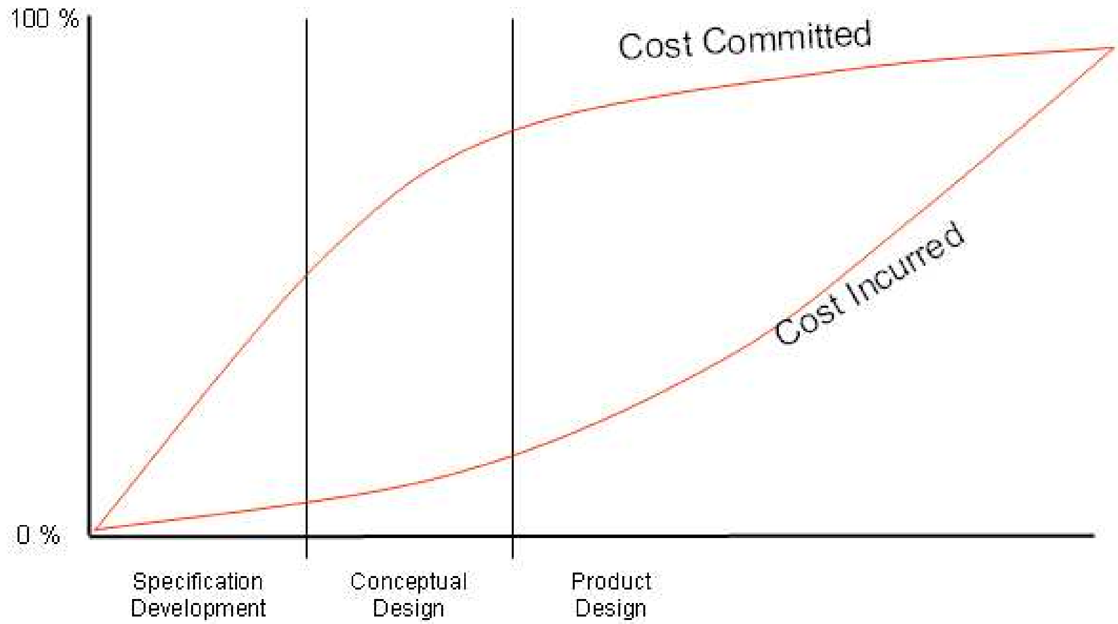


Figure 8: Picture of knowledge cost curve

As seen in Figure 8, the bulk of the cost committed in these two design phases represents upwards of 75% of the cost committed in the project even though the actual cost incurred may not occur until much later. The concept becomes somewhat troubling when additional information is considered. Figure 9 shows a similar cost time curve superimposed with relationships of the design freedom and knowledge [32].

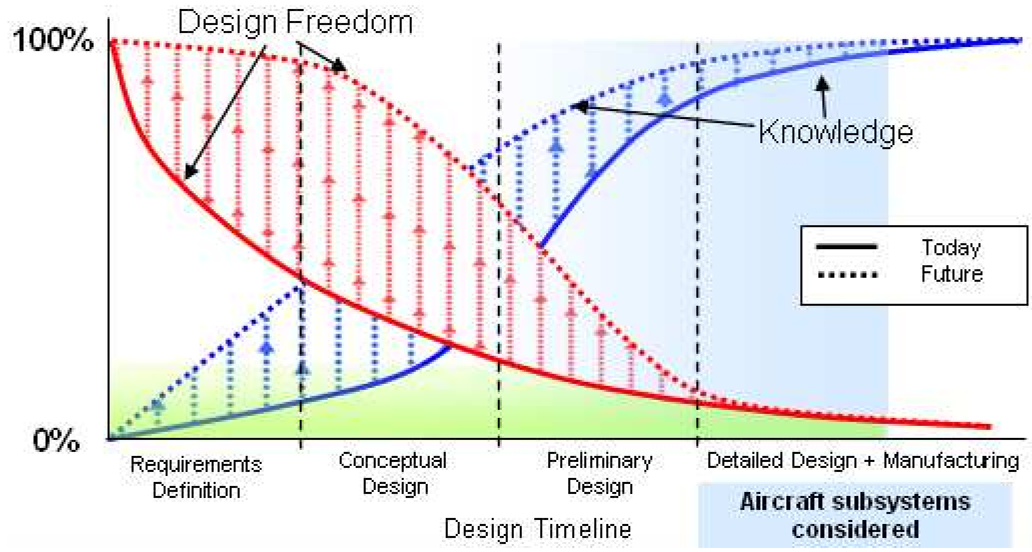


Figure 9: Design freedom, knowledge, and cost committed from Mavris and DeLaurentis [32]

The relationship between design freedom and cost committed is somewhat straight forward. As the design takes shape in the mind of the engineers and designers its new physical or conceptual attributes impose costs on all future activities in the product life cycle [31]. The first two phases of the design process determine the definition of the scope of the project and the conceptual embodiment of the final product. With increased definition, the ability to make changes to the design becomes increasingly difficult thereby reducing the design freedom. These design decisions, however, are made with incomplete quantitative information regarding their impact on the architectural attributes (concepts, requirements, and technologies) [32]. This limited engineering knowledge combined with the economical and time constraints of the problem increase the difficulty of product design. In fact, Ulrich and Eppinger claim that product development involves so much risk that less than half of all products designed could be considered successful [32].

The development complex system is wholly based on the assumption that the abstractions developed are valid and applicable to the given design. Levis and Wagenhals state, “The customer and the architect assume that these components will work properly because they will be constructed and installed in accordance with established codes and guidelines" [33].

Lack of knowledge regarding complex monolithic systems early in the design phase indicates that established codes and guidelines may not yet exist or are non-applicable to a given design. Therefore, applying these assumptions and principles to new aircraft design can lead to major hurdles in the implementation of new technologies. In order to provide a competitive product, designers must make the right assumptions early in the design process. These assumptions must be accurate and verifiable during the process from beginning to end.

Michael Sinnett, chief engineer of systems development for the Boeing 787 Dreamliner, spoke about the decision to change the cabin air pressurization method from engine bled to electrically compressed. This single change to the aircraft architecture imposed dramatic changes to the predefined or assumed relationships within the system. Sinnett said, "When we decided on electric pressurization, it lowered aircraft empty weight 1,000-2,000 lb. and fuel burn was down several percent, but the numbers got muddied as the 787 got integrated. It's hard to say where the weight has gone [9]." The initial performance estimates for the technology infused systems did not take into account the multiple system level changes which needed to occur within the architecture to facilitate component integration.

Architecting is generally performed in the embodiment design phase. The definition and selection of an architecture depends on the ability of the designers to describe the architecture. These descriptions are then related and compared to determine which architecture is the most desirable [34]. Complex systems are typically defined by large, multidisciplinary architectures. Therefore, it is difficult to capture all of the detail associated with an architecture with the limited design knowledge available during conceptual design.

Although it may not be explicitly defined by the design team during conceptual design, the concept is envisaged with some form of architecture. Assumptions are generally made as to the architectural definition. These assumption form an abstraction used to capture the general detail of architecture during conceptual design [35] which can become a constraining element to further product development. Considering the relationship between knowledge, cost, and design freedom, a more developed understanding of the architecture during the initial phase of design can give critical guidance to the design team [30].

Thus, there is a necessity to make valid architectural decisions early in the design process. Steps must be taken early in the design process to develop and use critical architectural knowledge. Figure 10 displays the traditional relationships between information within engineering design and marketing [36]. All of the groupings of information represent decisions that must be made in the product design process, each requiring information and feedback from other decision making processes. This figure does not necessarily imply any order, but simply indicates the relationships between design information.

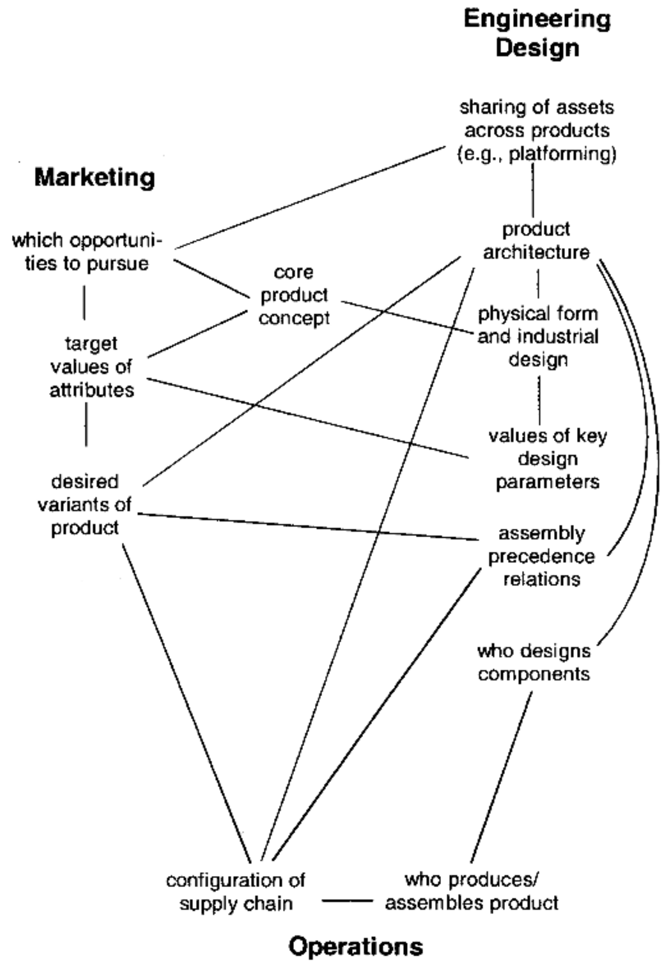


Figure 10: Relationships in Engineering Design and Marketing from Gerwin and Barrowman [36]

The concept architecture as defined in this paper is a combination of both the physical form of the product and the relationship between the composing entities. This is represented in the red box in Figure 11. In order to increase the fidelity the concept generated during conceptual design, knowledge regarding the product architecture and physical form must be infused into the core concept. As this is done, designers can more directly evaluate the effect of the architecture on the desired performance.

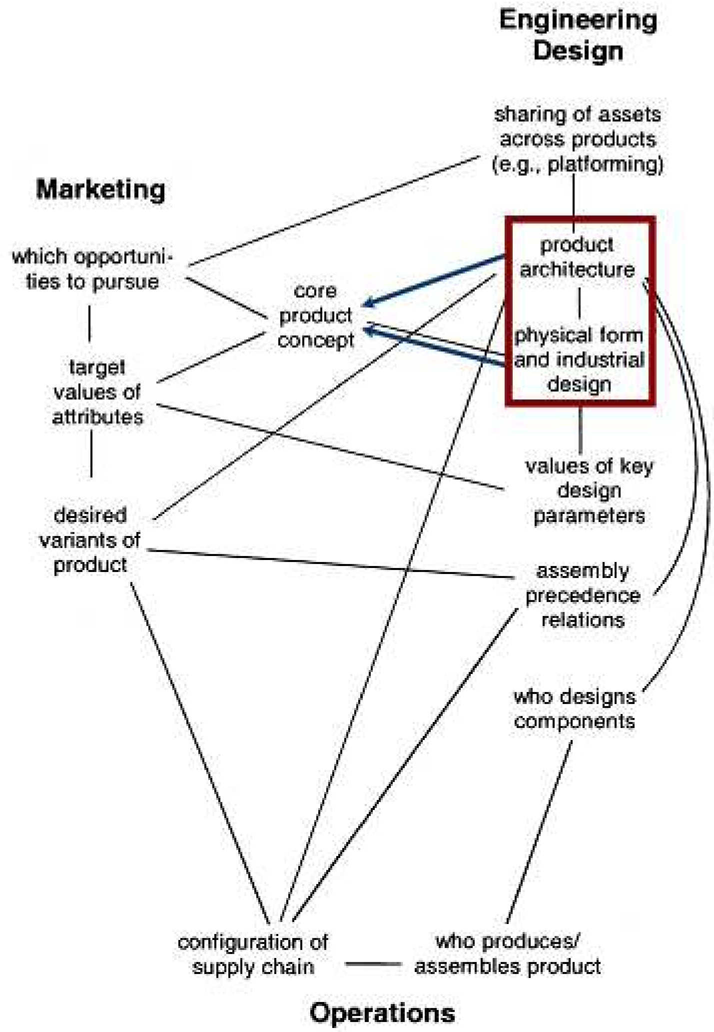


Figure 11: Relationships within engineering design and marketing with increased architectural focus

The National Academy Press published the document, "Design in the New Millennium," which identifies long-term goals to enable "extremely advanced product and process design." As stated in this document, the ideal for the future would be to develop the capability of combining the two steps outlined previously (conceptual and embodiment design) with the process of detail design into one design step. This would be enabled through a process which generates alternatives, determines element details and performance, and facilitates trade-offs [19]. In order for this to occur a well-organized and efficient process for the definition of architectural alternatives must be developed.

Information and Understanding

Every decision is based on some sort of model [37]. The existence of data or information does not translate directly to design success. The value of information is dependant on its level of definition and accuracy. Key to the development of knowledge is the processing of raw data in order to form an understanding upon which decisions can be made. This process of formulation requires some form of model [38]. The information hierarchy is shown below in Figure 12.

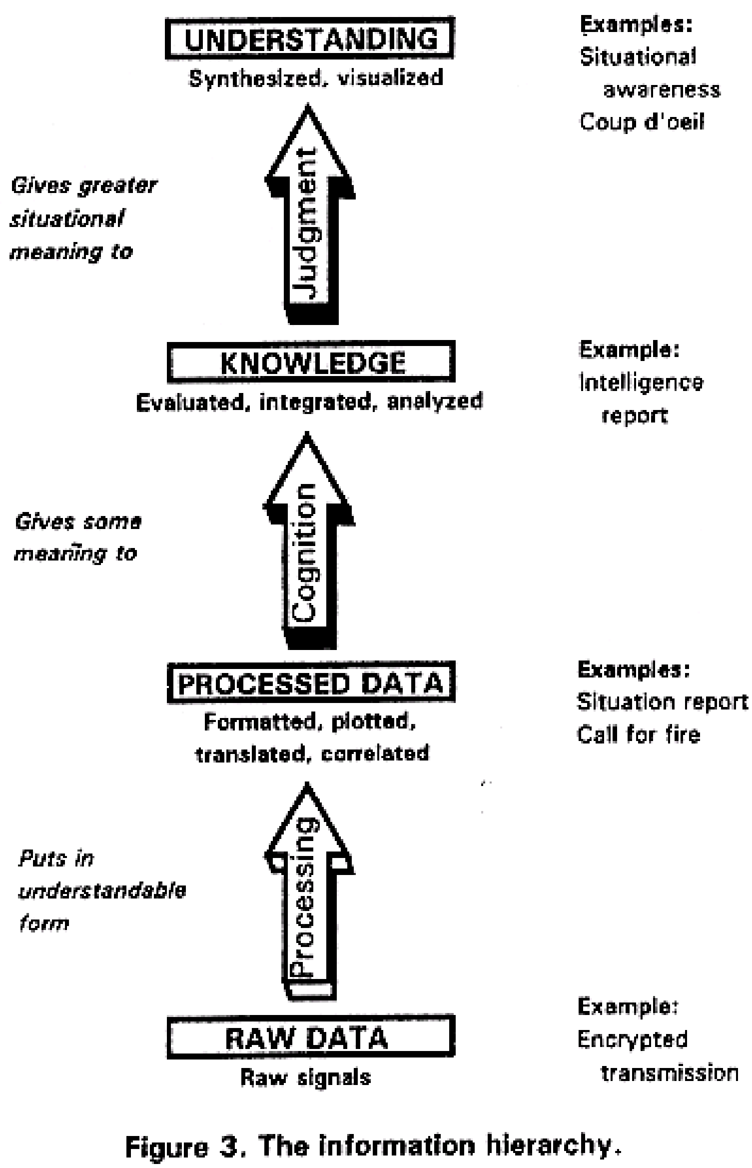


Figure 12: Information Hierarchy from the Department of the Navy [38]

Understanding is defined as a synthesized and visualized form of data. This understanding is knowledge applied to a particular situation yielding a comprehension and awareness of correlations and details about the given situation [39]. In order to be properly understood the data must be processed and organized to allow it to be interpreted. Once the data is processed the results need to be evaluated and analyzed. Finally this knowledge must be applied to the underlying issues in order to aid the decision maker in deriving a solution [38].

Increasing the amount of engineering knowledge and design freedom during the conceptual level requires the definition of robust and reconfigurable processes which can be used to quickly construct and assess complex systems during the conceptual design phase. In order for this to occur, accurate models must be easily defined and integrated early in the design process in order to generate and process data appropriate to the product design task.

The process of working with data in order to generate knowledge is central to engineering decision making. Engineers employ science, mathematics, physics, computing, economics, probabilistics and many other fields as means to process raw data into a form of information that can be successfully weighed and compared.

Modeling

For any product and especially in the case of complex systems, it is irrational to expect to assemble the complete product without first making an evaluation of the concepts and technologies involved. Tools like prototypes, simulation, and modeling provide insight into the behavior of individual or groups of system elements [22].

Models can be lumped into two basic groups: descriptive or predictive. Descriptive models are intended to communicate or convey some principle or physical relationship. This type of model aids in the understanding of the problem but does not characterize the behavior of a system or element [21]. Predictive models are used to calculate the behavior of some element under specific conditions. Both varieties of models have their place in the conceptual design process. Descriptive models aid in problem formulation and concept description during alternative generation while predictive models provide a means to gauge the performance of the design.

Companies incur great risk when undertaking the design production of large, complex systems. Therefore, a detailed understanding of the performance of elements within an architecture is valuable knowledge to ensure adequate performance and ensure good return on investment. This places pressure on system designers to increase the capability to accurately and efficiently predict system performance.

<u>Historical Regression</u>

The application of historical information to predict future performance is prevalent in well-established fields, like aerospace. Estimates for the attributes of system elements are often statistically determined by comparing known system level requirements or attributes to historical trends related to that element [40]. These statistical relationships can perform well for traditional systems but require augmentation with the implementation of innovative technologies or configurations [41].

John D. Sterman, Management Director of the Systems Design Group as MIT, states regarding the work of Donella Meadows and Jennifer Robinson [42]:

> "Models rarely fail because we used the wrong regression technique or because the model didn't fit the historical data well enough. Models fail because more basic questions about the suitability of the model to the purpose weren't asked, because a narrow boundary cut critical feedbacks, because we kept the assumptions hidden from the clients, or because we failed to include important stakeholders in the process" [37].

Historical data is biased on the basis of the underlying assumptions implemented when the data was obtained regarding the boundary and robustness of the relevant systems under consideration and scope in which the data was obtained.

Weight is of large concern during the design of an aircraft. The weight of this complex system has a dramatic influence on its performance. The extensive source of data concerning the weight of aircraft components and system level aircraft attributes allows designers to develop complex statistical relationships used to estimate the weights of aircraft systems [40]. Scaling factors are required when the assumptions and element definitions are breached [35] [41]. These types of models are simply estimates to represent the cumulative attributes on the system level. They do not take into account the interactions between elements and do not develop understanding regarding the driving relationships and requirement interactions which yield differences in element attributes.

Historical techniques can also be implemented with knowledge regarding a well-defined baseline model. With evolutionary changes to this baseline through the introduction of new technologies or components, assumptions must be made about changes of the behavior and performance of the baseline [35]. These assumptions follow anecdotal evidence and are made using engineering reasoning.

<u>Anecdotal decision making</u>

Due to the abstract and intangible nature of early design many decisions are based on predictions or beliefs about future design requirements and sometimes unreliable data. During the conceptual design phase point problems are very amorphous and, in turn, complex, leading to difficulty in applying analytical assessment methods [27]. In these circumstances judgments stem from historical information but are subjective, relying heavily on heuristic rules and the understanding and experience of individuals or groups of designers [43].

Heuristics methodology, or anecdotal decision making, is based on rules of thumb or the common sense, or contextual sense, and reflect the strategy of a design to produce an agreeable solution to avoid catastrophic failure. Heuristics can be described as "trick of the trade" or "engineering reasoning" [27]. This anecdotal reasoning can provide direction and insight into a design with vague definition and is useful, but it sometimes induces systematic and critical mistakes because it is subject to the biases and stereotypes of the parties involved [43]. With the application of heuristics, design is sometimes described as more of an art than a science [27].

Integrated Product Design

Due to the multidisciplinary nature of conceptual design, as shown in Figure 11, companies often employ a means of coordinating and integrating existing corporate knowledge. Integrated Product Teams (IPT) are composed of experts from diverse disciplinary, organizational, and systems backgrounds and are employed during the conceptual, preliminary, and detail design phases. This team is tasked with what has been termed Integrated Product Development (IPD). The IPT is simply a means of facilitating human resources to develop and process technical and market knowledge in order to formulate design solutions [36] [44] [45]. IPTs are important tools to engage in multiple forms of engineering knowledge processing. IPTs must work within these processes to formulate the design solutions.

Physics Based Design

Physics-based models are analogues of physical reality and stand in proxy of elements, groups of elements, or processes and convey information regarding response to stimuli and performance in the real world. For complex systems design, groups of models are generally required to provide the information to be used by the designer [46]. The level of fidelity of the process in which data is transformed to knowledge determines the validity, accuracy, and extent to which the knowledge can be applied to aid understanding. Physics-based models provide a way of analyzing and representing the behavior and attributes of system elements based on their environment and provide numerical, quantitative results regarding component or system performance.

Predictive models exist in different forms which vary in fidelity and usefulness. George Box, prominent statistician in the 20^{th} century, stated that “All models are wrong, but some are useful [47].” A model’s accuracy is dependent on its ability to describe the relationship between stimuli and behavior in all reasonable operating conditions. Precise models also require that the underlying assumptions are accurate and that the structure of the model does not limit the application of the model to all desired scenarios [48].

(Example: viscous vs. compressible flow equations, or Bernoulli equations, both model fluid dynamics). Physics-based performance analysis techniques are crucial to the development of revolutionary or "first-of-a-kind" designs [19].

Summary

Organizing a vast group of physical elements into a form which can fulfill functions requires foresight on the part of the designers. Complex systems design is inherently a process of making a large number of decisions, each of which defines or refines the product structure. Often in the process of design, decisions made early in the design process limit the flexibility of a design and impose limitations on the decisions that can be made. This is due to assumptions made in this decision making process. Assumptions can be used to give conceptual structure to the product but may also limit the flexibility of the design during the infusion of new technologies.

For commercial aircraft, technology infusion often occurs at the systems architecture level. At this point in design, architects are severely limited as to the amount of change that is allowed to propagate through the system. In order to limit the number of challenges that arise in the late phases of design process, knowledge is required to predict, understand and avoid possible issues that may arise. This generation of knowledge can occur through many forms of modeling: historical regression, anecdotal reasoning, integrated product design, and physics-based modeling. Models are essential in the definition of the architecture of a complex system.

Justifying and proving a design requires an implementation of tools which can accurately predict the performance of a given system and its interactions. Because of the size and complexity of a complex system analytical models are necessary in validating and predicting the performance of the architecture elements. However, every model is subject to the assumptions made in its definition. Therefore, perspective becomes a driving force behind the use of function based architecture definition.

CHAPTER 3
BACKGROUND: DESIGN PERSPECTIVE

In order to facilitate the development of architecture knowledge, an appropriate perspective must be taken to allow the generic definition of a product through which alternative architectures can be defined and architecture trade-offs can be considered. The process of architecture design must be formulated so as to allow the architect to efficiently define and alter the complex system and facilitate the process of modeling that architecture. It must also be able to capture and indicate all changes that must be made to the architecture as a result of architectural decisions. A transition from a focus on physical elements as the central design premise during the early design phases to a focus on functional based architecture definition would allow for the definition of widely varying physical architectures within the same functional description and requirements [19]. The tasks that a design is intended to fulfill remain constant regardless of the physical implementation of the function.

Including more dependable information regarding architecture definition during the conceptual design process requires that a design team employ some type of modeling technique. This model must be capable of allowing the definition and objective comparison of various architectural configurations. Moir and Seabridge express the necessity of developing "'soft' representations of a system that can be modeled and remodeled without incurring excessive cost." In this context Moir and Seabridge refer to "soft" as a qualifier meaning physics based simulation models which verify design characteristics and performance without the creation of the physical element [22]. However, the "softness" of the model as stated above relies on the underlying assumptions imposed in the modeling structure. The structure of the model can be such that the model does not readily allow the redefinition of model constructs.

During the conceptual phase of traditional aircraft architecture design, computer models represent each of the predefined systems within the aircraft architecture (engine, ECS, electrical system, hydraulic system). These systems models generate attributes and relationships along standard architectural interfaces [40]. The attributes of the system are defined by a general description of other system attributes. To enable trade-offs to be performed on the architecture (both elements and relationships) the routines representing the systems elements as well as what information being passed between elements must be reconfigurable. This invasive dissection and redefinition of the architectural model requires a means to modify the required routines within the model and to reroute the information between these routines.

A set of tools must also be identified and developed with which the designer interacts to make all of the decisions necessary to embody the architecture. The scope of this work is to introduce and develop function based frameworks, tools, and methodologies that enable the designer to quickly define and assess architectural performance and provide an interface in which these architectural decisions can be made.

Product Decomposition

The process of architecture definition follows the same process as conceptual design: problem/project definition, alternative generation, and alternative selection. The work presented here addresses the perspective required during the project definition phase and the means and interface in which alternatives can be generated. Methods and means to select the architectural alternative are not directly addressed in this work. However, with the theory and tools in place alternative selection techniques can be applied. Cursory discussion of these techniques will be given later as related to the first two phases and to illustrate areas of potential future work.

Tyson R. Browning, Senior Project Manager in the Enterprise Productivity Strategy Group, Lockheed Martin Aeronautics Company, described the traditional systems engineering approach (the processes of project definition and alternative generation) as including 3 steps [49]:

1) Decomposition of system into system elements
2) Determining the relationships between the system elements
3) Reorganize the relationships between the system elements

The third step defined by Browning can be seen as a reformulation and comparative process. The first formulation of relationships will not necessarily meet all performance criteria or meet all the functional requirements. Therefore, reorganization may be necessary to improve performance. Equally, a step could be included which readdresses the decomposition to reformulate the system elements. The foundation of these three steps reveals a process that includes the decomposition of the system conceptually and a definition in a quasi-physical manner wherein element relationships are established.

Forsberg, Mooz, and Cotterman visualize this process in the basic Vee model. This is shown in Figure 13. In this context design is simply a process of decomposing defining the problem in some logical sequence and then integrating the decomposed system while verifying that specifications are met [50].

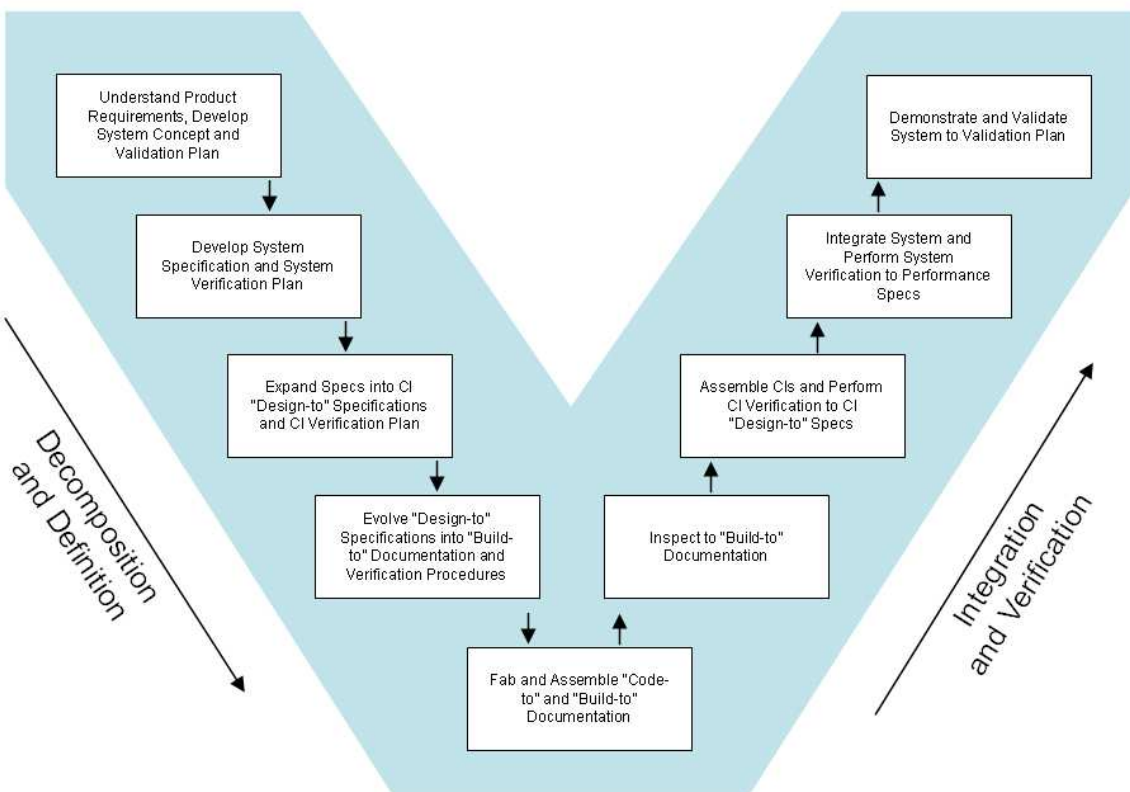


Figure 13: Basic Vee Model

In this Vee model decomposition and definition exercises are performed with foresight as to the manner in which these will be implemented and how success will be later determined during verification. Then, during integration and verification the designer revisits the decomposition to determine if the requirements are being met and if the product decomposition and definition is appropriate [50].

In the process of problem definition the engineer is tasked with understanding and organizing the design problem. Decomposition is the process of compartmentalizing a system into smaller more manageable elements [26]. This decomposition identifies these decomposed elements as the required building blocks of the system. These building blocks can be combined to form the complete system [51]. Therefore, the manner in which the system is decomposed determines the framework in which a designer can construct a variety of architectures. Any combination of alternative technology choices can be made and implemented, as long as they can fit within this designated framework. This is in essence the modular description of the system.

Modularity

Architecting is, in essence, the modularizing of the overall product by means of defining the fundamental elements with regards to anticipated functional interactions [28]. In defining an architecture, the fundamental elements must be defined and characterized in a way which allows them to logically interact with other systems elements. By defining these fundamental elements modularly, alternative configurations can be defined.

Modularity is based on relationships between structural elements of an architecture [12]. This is due to the fact that functions can take very different physical forms, with varying elements and features. McClelland and Rumelhart define modularity in terms of relationships between modules. They state that “a module is a unit whose structural elements are powerfully connected among themselves and relatively weakly connected to elements in other units [52].” Modularity becomes the means in which the complexity of individual modules is hidden from the whole, and architecture acts as the framework in which these modules interact [12].

Modularization becomes the means in which various teams or companies can adopt responsibility for the design and development of a portion of a design. Because the details of each module are hidden from the view of all other entities, often including the integrators, companies maintain ownership of their engineering knowledge and experience assets.

Difficulties arise with the modularization of an architecture. The intermodule interfaces must be able to change without affecting the internal workings of the module. Conversely, changes to the internal workings must not interfere with the intermodule interfaces [34]. As details are hidden within each module lack of transparency and interplay can limit the effectiveness of the design. If a function or attribute of one module could assist in the performance of a function contained within another module and this relationship is not captured by the interface definition, the architecture definition becomes a limiting factor to the performance of the product.

Complex systems are often defined with some intended form of modularity. Ulrich and Eppinger define three types of modularity: slot-modular, bus-modular, and sectional-modular [23]. Pictorial representations of these architectures are displayed in Figure 14. In this figure the semi-circle, square, and triangle represent the system modules of the systems and the light blue foundation is the architectures framework.

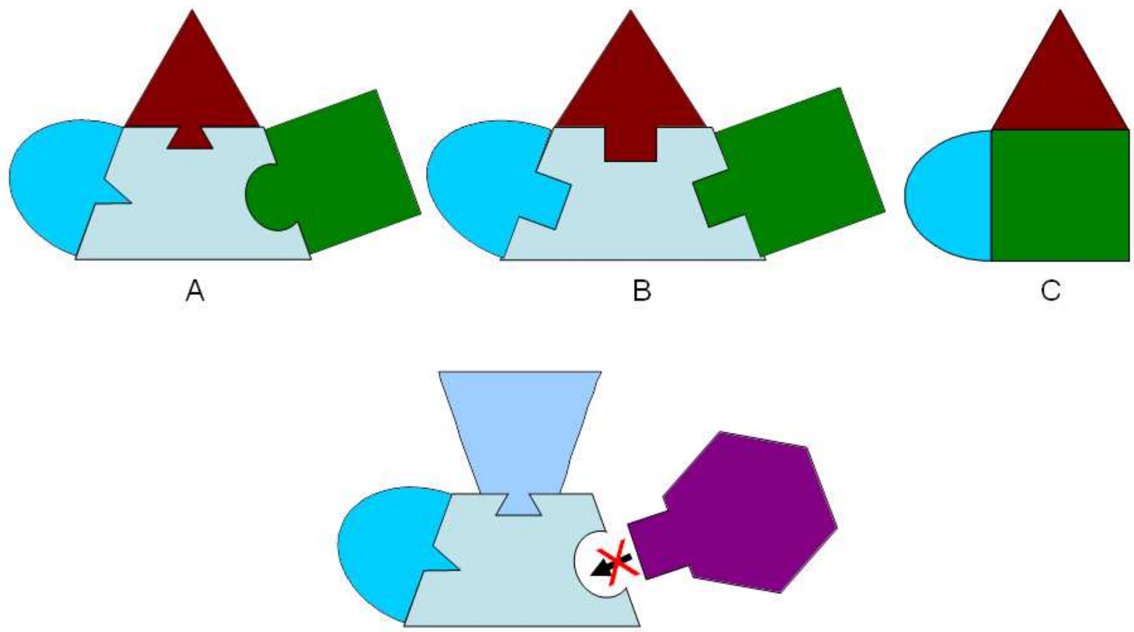


Figure 14: Modular architectures A) Slot-modular, B) Bus-modular, C) Sectional-modular

The first architecture, slot-modular, is the most common in multidisciplinary complex systems because each module is defined with a unique interface. Complex systems often include vastly varying physical interfaces. These modules must interface in a very specific means. The traditional definition of this type of architecture was described when explaining architecture definition and will be looked at later.

A bus-modular architecture indicates that the interface of each module is defined with common interfaces. These are often seen in the form of cards for computer systems, electrical bus systems, or standard gauges for mechanical interfaces.

The last typical architecture is sectional-modular. This architecture is characterized by identical interfaces which can be attached to any other module by means of this interface (e.g. standard piping systems) [23].

Implied by the slot and bus modular architectures is framework within which the system modules operate. This framework becomes the means by which information and data is exchanged between the systems or the physical device which structurally supports the modules. Decomposition defines the manner in which the building blocks of the system are defined. Therefore, the perspective taken during the decomposition process defines the type of modularity which the product will have.

Modularizing the concept architecture allows flexibility in the definition and redefinition of architecture concepts. With higher levels of modularity, many different architecture concepts can be generated which can utilize vastly different physical means in the fulfillment of product requirements. This modularity can also extend to the definition of modeling and simulation tools for justifying architecture concepts. Modularity in architecture element definition can improve the ability to model widely variant architecture concepts.

The perception adopted during the process of decomposition greatly affects the flexibility of the architecture design. Three general standpoints can be taken during the process of decomposition: physical, functional, or disciplinary.

Physical Perspective

Physical product decomposition looks at the system in terms of physical elements and their common physical relationships (spatial organization, energy type and flow, material, form). When the system is well-defined, all of the attributes are readily apparent both visually and substantially. This decomposition is the most intuitive because it can be easily observed. It also provides a very clear boundary between system elements. Decomposing a system physically yields a catalogue of physical parts or groupings of parts depending on common physical or spatial relationships.

An example of a physical breakdown would be the decomposition of an internal combustion engine into its elements. Figure 15 displays a physical decomposition of a typical pushrod V6 internal combustion engine. Physical decompositions can be easily described by images and charts. Engineering drawings are examples of very detailed and extensive physical breakdowns.

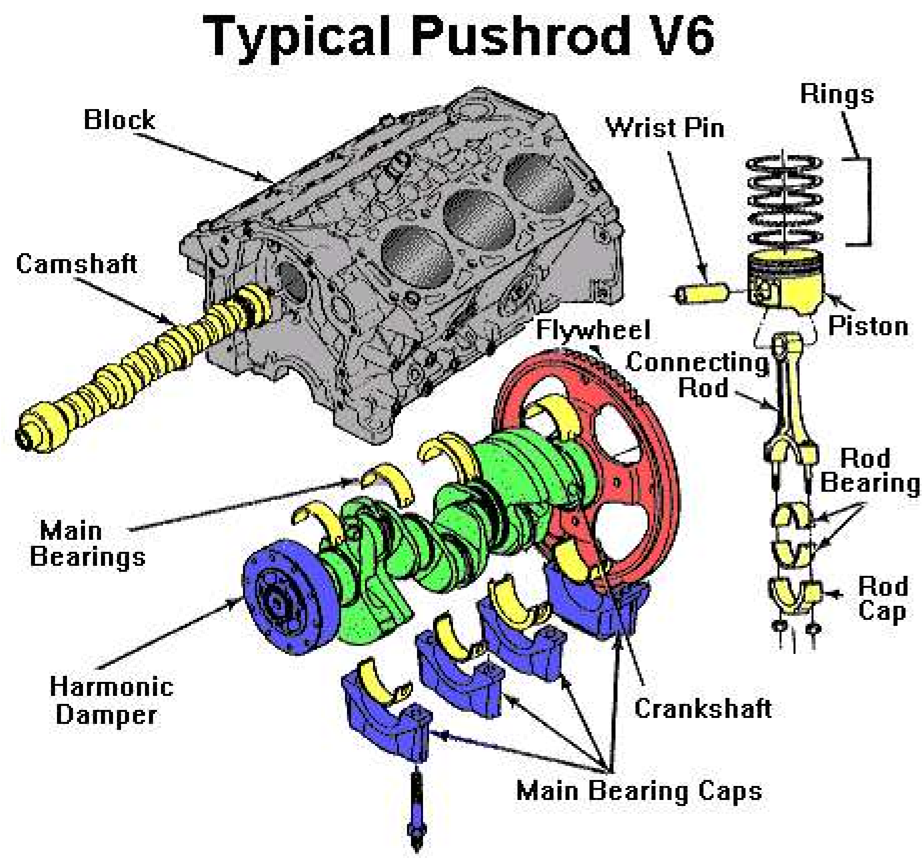


Figure 15: Pushrod V6 Internal Combustion engine physical Decomposition

As displayed in Figure 15 a physical breakdown is the description of the physical elements composing the system and their direct physical relationship. When decomposing the automobile, the engine has been defined as a module within that system. The classification of this engine also distinguishes it as a physical entity. V6 indicates that the engine has 6 cylinders and the pushrod or overhead valve designation indicates that the camshaft is located physically within the engine block. References to specific engines or types of engines are typically done with regards to physical definition (displacement: big block, 50 cc/valve orientation: L-head, F-head, I-head/cycle: 2 stroke, 4 stroke).

In the case of the engine displayed, all relationships between the elements are defined mechanically. For other physically defined groupings of element within a decomposition, there may not be a direct spatial relationship. These elements are arranged due to other physical relationships. Systems like the lubrication system can be classified as dealing with the same material (oil), while others can be decomposed and grouped because of integrated physical behavior, like the suspension system.

Although the physical breakdown is very clear and visually understandable, it requires the reformulation of the decomposition with every physical architectural change. Candidate architectures may have widely varying physical structure and relationships. Similar decomposed modules may have extremely different roles or may be excluded entirely. Assumptions made during the conceptual design process regarding physical implementation of architectural concepts severely limit the designer's ability to explore revolutionary product concepts.

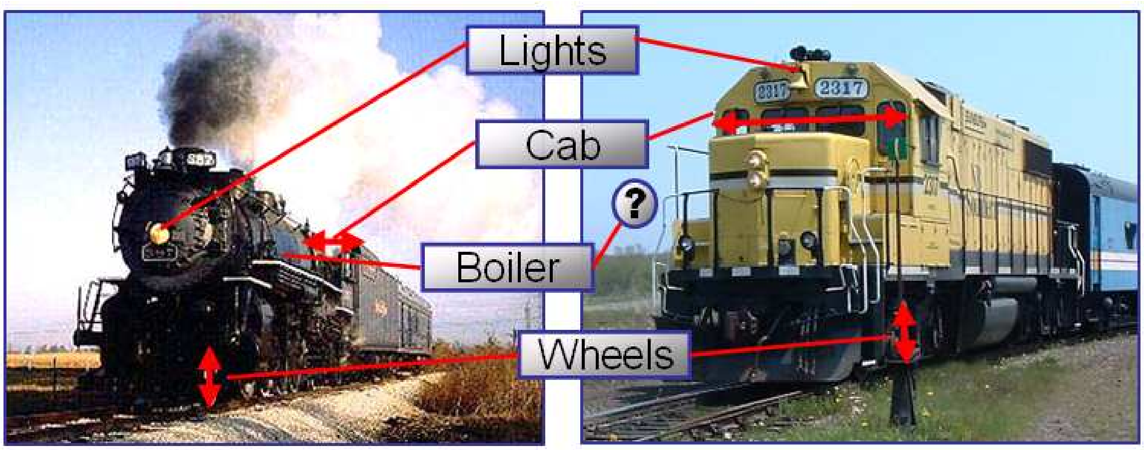


Figure 16: Physical Breakdown of two train engine candidates

Consider the two complex systems displayed in Figure 16: the steam engine and the diesel engine. These two candidate systems are intended to fulfill the same functions. Both are intended to provide the force required to pull freight or passenger train cars. They both are intended to operate in the same environment and have the same mission definition.

During conceptual design it is the responsibility of the designer to produce the optimal design. Therefore, he/she must have the capability of defining various candidate products for the given design. A physical decomposition is not adequate to allow the necessary generality to explore the two alternative architectures in Figure 16. The physical decompositions that characterize each of these alternative architectures are different. Analogous components between the two architectures do not always exist. For example, the boiler is a critical component of the steam engine's architecture while no comparable component exists in the diesel engine.

Other components may be physically similar but are illogically compared. Both the diesel engine and the steam engine have a cab, but the size of the cab is not necessarily a reasonable comparative metric without further information. Considerations regarding the purpose of the physical component are needed for adequate comparison. The activities that occur in the cab of the steam engine include activities like shoveling coal or loading wood and stoking the fire. These activities are not required in the diesel engine. The number of operators and operating environment (increased temperature from the boiler) may be different for each alternative also.

Physical decomposition does not provide an adequate framework within which architecture designers can make innovations and explore new designs. More generic building blocks are necessary to allow for innovation within conceptual design.

Functional Perspective

The overarching purpose of a design is to fulfill some customer requirement. These requirements constitute tasks which must be fulfilled by the system in order to be successful. In terms of the architectural definition of a product, the customer requirements are the same regardless of the product architecture. However, product design can apply various physical means of providing the same set of customer requirements. A functional framework provides a consistent platform upon which the designer makes the decisions to define alternative architectures.

A function is an action. Gerhard Pahl describes a function as the effects of an element on its surroundings through means of material, energy, or information. In other words, the function describes what a product does [53]. Nam Suh defines functions as a description of goals that a product must achieve [54]. Thus, combining these two definitions a function is the objective physical performance of a given element.

The overall actions that a product must perform are independent of the method implementation. Thus, these functions cannot be visualized or represented by an object. Therefore the conceptual embodiment of the function must be understood in terms of the result of the function being fulfilled or a representative physical embodiment. A simple example is the function to “provide light.” This function represents an action that can be fulfilled through multiple means. This function may be fulfilled by a LED, incandescent bulb, fire, bio luminescent material, etc. The function itself does not indicate which means is most effective, efficient, or preferable. It simply defines the core element of the product and remains generic enough to allow any physical embodiment to be considered.

Generic product definition comes in the form of functional analysis. Systems Engineering Fundamentals (SEF) describes systems engineering as the means of translating all the requirements developed through pre-design activities into a functional description of the system. Functional analysis is the process of logically arranging and decomposing functions in order to create a functional architecture. This functional architecture is simply a description of the system in terms of functions [55]. In the SEF the functional architecture refers to the classification of the system by functions only, not the arrangement of functions within systems at the physical level [26] or clustering. INCOSE Systems Engineering Handbook (SEH) defines functional analysis as simply the process which “determines what the system must do” [15].

Entities within the aircraft industry have recognized the need for a deeper understanding and implementation of functions in their complex system. The Power Optimised Aircraft (POA) project, commissioned through the European Commission’s 5th Framework program for Research and Technology Development, was a task to explore the way forward for aircraft equipment systems [56]. One of the findings of POA was that the way forward for aircraft architecture development was the application of a function oriented viewpoint [8].

Consider the two trains again. The tasks that the trains are to perform are the same regardless of the technology set upon which they are based. The train must still provide forward momentum, contain passengers and cargo, follow the tracks, and provide means by which the speed is controlled. In regards to the product level functions, these two architectures are identical. They simply have different performance in fulfilling the requirements. Within the framework of systems tasks, or functions, any architecture can be applied.

This benefit of a functional decomposition also becomes a limiting factor in its application. It is generic enough to classify any candidate architecture. However, because of this generic nature it becomes somewhat non-intuitive in definition and application. Difficulties arise in linking the nonspecific functional architecture of a product to the desired tangible physical form.

Functions are often visualized by their physical structure. However, the functional decomposition should not reflect specific physical implementations, in turn limiting the exploration of radical concepts. Functional product decomposition requires an appropriate level of granularity. The scope of this decomposition must provide appropriate detail but be fashioned so as to be non-constrictive to physical alternatives.

The level to which the architecture is functionally defined depends on the extent to which the concept needs to be developed. The level of granularity determines the usefulness of the functional breakdown in product comparison.

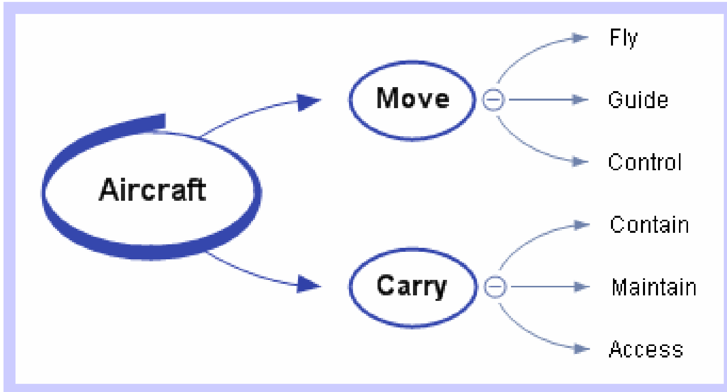


Figure 17: General functional decomposition

For example, the functional breakdown in Figure 17 is an accurate description of the functional upper level of an aircraft. However, the level to which the decomposition is carried out does not provide adequate insight into the alternative architectures which have the capability of fulfilling these functions. The decomposition in Figure 17 is an appropriate beginning to a detailed functional breakdown.

A too detailed functional breakdown can also be limiting in its usefulness during concept development. Multiple architectural decisions must be made in the implementation of a given architecture. Thus, every additional function included in the functional breakdown impacts the order of magnitude of alternative architecture candidates. The level of abstraction adopted for every function within the functional breakdown is driven by the scope of the trade-offs that the architect wants to consider for that function. For example, if a designer wants to design a new military ground transport vehicle, but is primarily considering changes in the electrical system architecture, the granularity used for functions that are relevant to the electrical elements will be fine. However, the decomposition of functions relating to impact resistance capabilities may be relatively coarse. The scope of this design may be considered independent of the armor configurations. Both the electrical system and the armor adopted for the design are important and will need to be considered in sizing the systems, but the design freedom applied to impact resistance capabilities may be very limited. Conceptual scope is very important in functional decomposition

While decomposing a product into Function/Solution Chains the designer asks the question, “What does the product have to do?” while avoiding the question, “How is this to be done?” [55]. Thus, approaching the decomposition above, the designer would further investigate the requirements of each of the functions described in the lowest level of aircraft functional breakdown. “What does the product have to do in order to fly?” would be answered with a lower level functional structure. “Produce lift” and “produce thrust” are appropriate sub-functions for “fly” because they can be considered unequivocal product requirements. For an aircraft to be an aircraft it must have the capability of producing lift and producing thrust. The boundary functions of an architecture are determined and defined through understanding how the product will interface with its users and operating environment. These requirements can be explicitly defined by an RFP, customer requirements or imposed standards, or can be derived through knowledge of company processes and engineering experience [22].

The question “what if …” becomes useful in functional decomposition. Once a function is fully decomposed conceptually testing the validity of this product description can be performed by considering various physical implementations and identifying limiting functional descriptions. If relevant physical alternatives cannot be appropriately described by the function decomposition, it must be altered to be generally applicable.

Disciplinary

A disciplinary decomposition is the grouping of elements within a complex system depending on their physical relevancy to fit within defined analysis groupings. These disciplinary groups represent branches of expertise that must be applied to define product performance. This is described through an example of the commercial aircraft disciplinary definition. An aircraft can be decomposed into multiple disciplines: aerodynamics, structures, propulsion, electricity, pneumatics, hydraulics, flight control, etc. Each one of these disciplines will be taken into consideration in the design of the aircraft. In breaking down the product in terms of functions, the designer begins to understand the areas of expertise that will be required in architecture definition.

Disciplinary decomposition is very useful in determining architecture behavior and attributes during detailed analysis and optimization. However, in organizing, defining, and sizing an architecture disciplinary decomposition does very little to define the system and must be used in correlation with other architecting approaches.

Figure 18 shows an example of the three decompositions for an internal combustion engine in tree view.

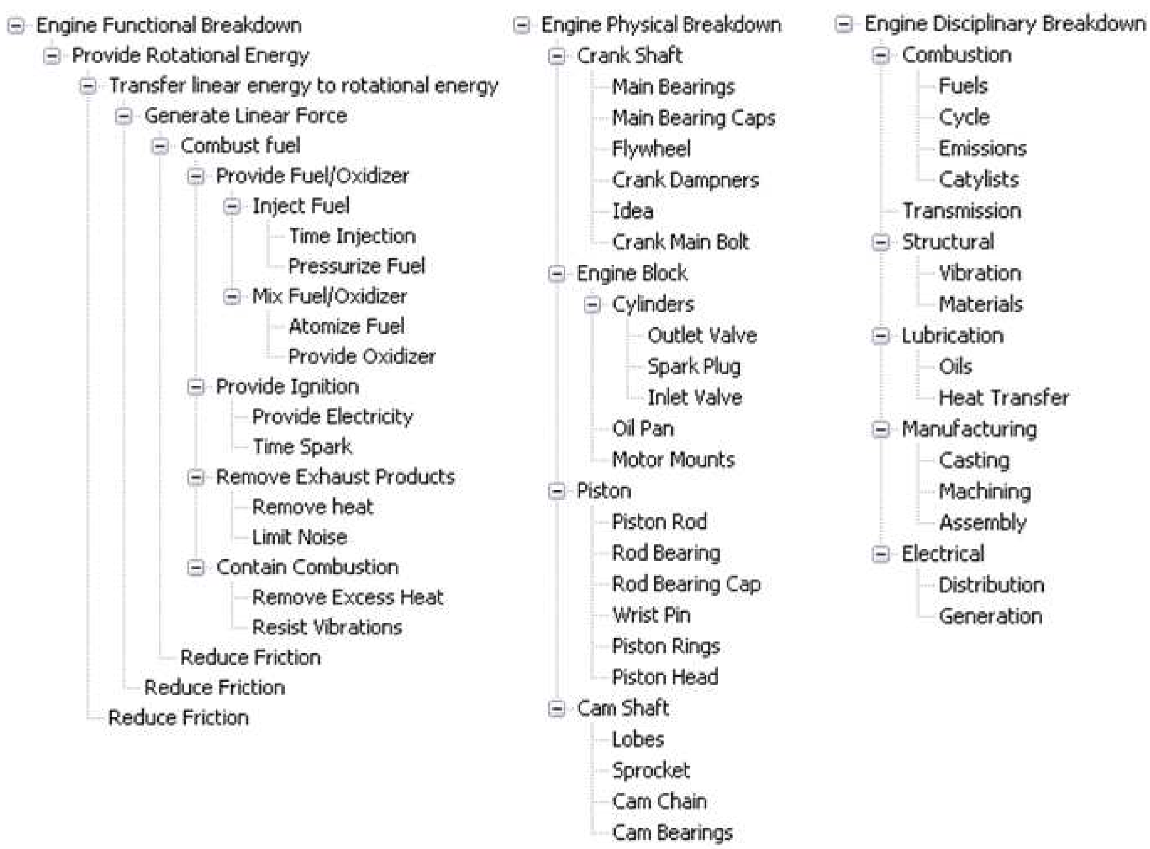


Figure 18: Engine Decompositions

Understanding the functional, physical, and disciplinary implications of candidate architecture is critical to developing a successful product design. From Figure 18 we see each breakdown gives insight to the function, form, and required knowledge that define an engine. Functional decomposition gives insight to the tasks that must be fulfilled for a product to function properly and indicates how these tasks are conceptually related. Physical decomposition becomes an embodiment of the defined function and describes physical relationships that must be defined and examined. Finally, disciplines are the means by which these physical relationships are understood and describe the types of expertise needed to determine product performance.

Traditional Perspective

Multiple factors are taken into account during the definition of these systems and subsystems. Physical, functional, and disciplinary knowledge is used in the division and allocation of functions to specific generalized systems. Engineering knowledge, experience, and organizational structures become the facilitators by which this decomposition and integration take place. However, in the process of traditional architecture definition, there exists a disconnect between the conceptual definition of the functional and physical structure and the process of actually embodying and assessing the attributes and performance of the developed architecture. The functional description of the system can be subject to assumed physical relationships, thus causing the functions to lose their generality. In addition, legacy models, based on the analysis and performance of previous products, do not fully capture distinguishing attributes of the new architecture. Historical regressions and tools based on systems generalizations, which are often used in conceptual design, cannot physically capture the true performance of a revolutionary architecture [32] [35] [37].

Systems Engineering Fundamentals describes functional analysis and allocation as the linkage between requirements analysis and product synthesis. Following the definition of the system requirements, functional analysis and allocation is the means by which the basic actions of the product are specified and the functional architecture is defined. This is done by specifying system states and modes and the functional relationships. INCOSE's Systems Engineering Handbook states that "functional analysis and decomposition can be performed independently of system architecture, but functional allocation obviously requires a system architectural structure" [57].

Functional allocation is defined as the means by which functions with similar assumed attributes, location, performance requirements, physical embodiments, or other relationships are lumped together in subsystems. The Department of Defense Systems

Engineering Fundamentals [58] document mirrors the sentiment that these function and physical architecture generalizations are somewhat independent. In Figure 19 the functional architecture is mapped against the physical architecture indicating their relationships.

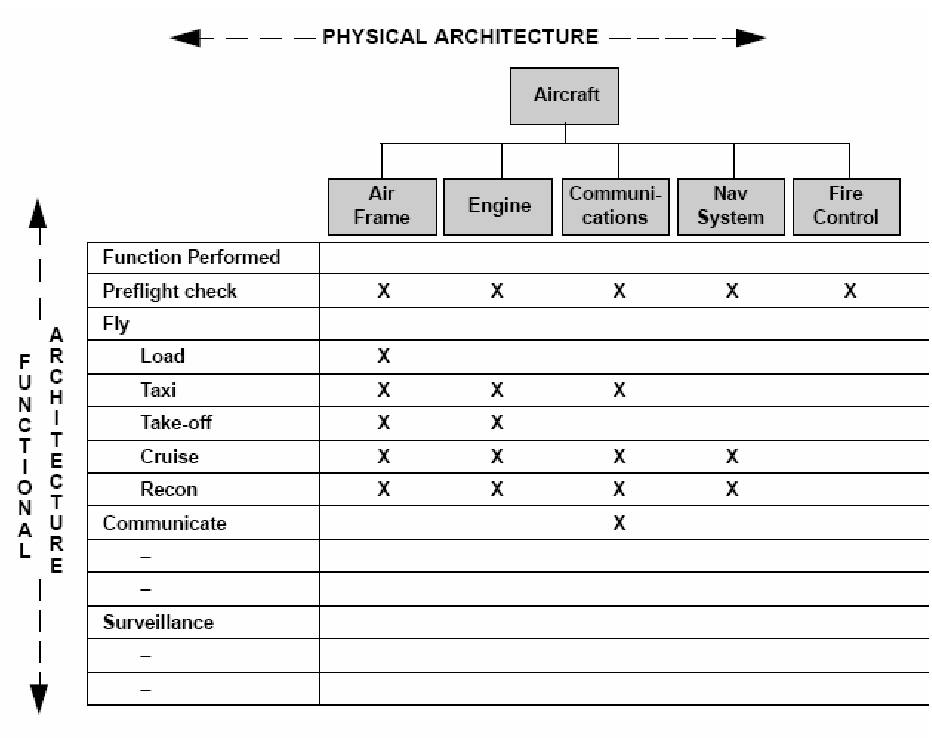


Figure 19: DOD, SEF Function/Physical Matrix

Systems architectures are developed by "allocating" the functions to a physical architectural concept. Thus, functions and their generalized relationships are used to guide the definition of an architecture. The concept of grouping functions within systems is often achieved through "clustering" these conceptual relationships by means of tools like the design structure matrix. In this context the systems architecture becomes a generic description of functional implementation or product concept. These generalizations of lumped functions become the defining element of the architecture.

Systems

Once the process of functional allocation is complete, the product level functional interactions are no longer the focus in architecture development. Each system is defined as a critical building block of the architecture. These systems are groupings of physical elements which are grouped together in disciplinary groups as shown in Figure 20 and fulfill specific architecture level functions (an example of systems groupings are the ATA chapters).

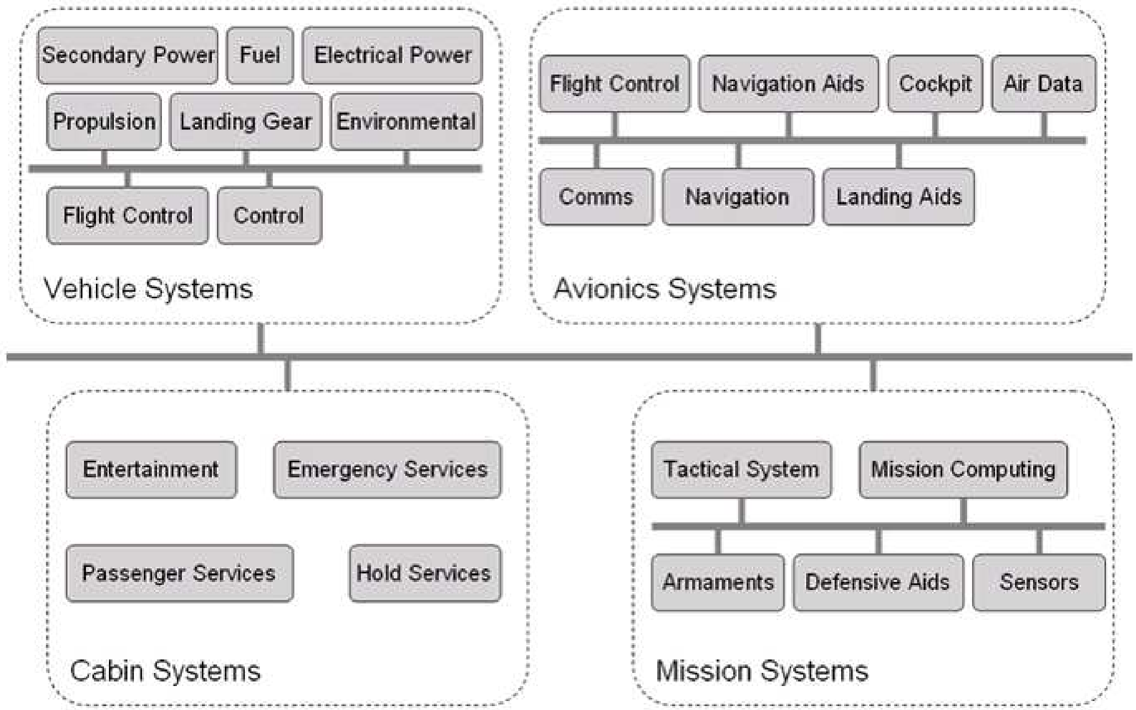


Figure 20: Aircraft Systems [22]

Each of these systems represents packages which must be developed in order for the system to be designed. These packets of work represent the fulfillment of the functions allocated to this specific system in terms of physical components and disciplinary sub-elements.

An example of the typical aircraft framework was developed by the ATA in the 1940s and is called the ATA chapters. The ATA chapters are currently used as a framework for aircraft decomposition [59]. ATA specification 100 provides guidelines for classifying the aircraft in terms through numbering schemes and grouping components into standard systems [60]. The ATA Chapters classify groupings of elements based on physical and disciplinary similarity and are broken down further into segments, or lower level groupings of similar components. A listing of the ATA chapters which group the aircraft into systems is available in Appendix B.

Traditionally, the generalized subsystem becomes the critical building block of the architecture. These subsystems are physical systems intended to fulfill defined functions. Trade-offs are made between systems on the upper conceptual level and subsystems on the system level. These subsystems are either physical elements or functions which are grouped together in disciplinary groups. Each of these systems groups represents packages of work that must be accomplished in order for the system to be designed. Once the functional allocation is completed the high level functions no longer serve as defining elements within the architecture but remain as guidelines to future developments.

This focus on bounded groupings and simplified relationship between complex modules drives the concept of architecting towards a definition of robust standards and conventions which regulate information and physical relationships between systems and the troubleshooting of changes which propagate through the system. If the perimeter of the modules changes, or if new interrelationships improve aircraft performance, the boundaries between the modules shift. With these shifting boundaries it is difficult to truly predict the performance and attributes of integrated systems.

Architecture definition in this context has a segregated structure for several reasons. The first reason is to minimize the diversity of knowledge required by entities responsible for designing one module. Limited information is necessary across module boundaries. It also maintains higher levels of interaction between entities existing within a given module [61]. Thereby, the scope of individual system modules becomes well-defined. Some have even suggested decomposing the architecture simply along lines of company division to alleviate risk of faulty technical interfaces [62].

The division of disciplines between engineering teams, industry partners, and academia creates an environment in which incomplete understanding and knowledge is used to integrate portions of a complex systems interface to specific, specialized entities. Often the decomposition of a system reflects risk-sharing relationships, in which specific entities are responsible for large portions of the overall architecture. Thereby, the system integrator outsources to companies with specific resources and training. These entities buy into the design and assume a portion of the financial risk associated with the design thereby reducing the amount of risk incurred in taking on the whole complex system design task. In the framework of generalized systems architecture, the role of the integrator is limited to the management of interfaces between the functionally allocated systems. Integration, in this context, occurs on the perimeter of modules which have been assigned to different contractors

Many new products tend to emerge through evolutionary processes [25]. Evolutionary design intends that the underlying design concept and structure remain unchanged, but new technologies are implemented within a given concept architecture to increase performance [32]. As promising technologies are developed, complex systems adapt to implement and envelop this new technology. This evolutionary approach generally leads to architectural innovation.

The addition of technologies promises much improvement to the system performance. However, these benefits are not seen without introduction of revolutionary architectures. Changes to the standard interfaces require redefinition of the architecture relationships. Seemingly simple design changes to one component in the system can induce changes throughout the system which are difficult to quantify. Evolutionary architecture design and definition becomes a problem of tracking and encapsulating the propagation of a change to a fixed framework [63].

Typical sizing of these systems during conceptual design is performed through applying previously defined codes, which, being based on the performance of previously defined products and architectures, are augmented to estimate the performance impacts of a new architecture containing revolutionary technologies. These performance “deltas” are used to alter the existing code to estimate impacts of new methods and technologies.

Summary

The perspective taken during architecture definition impacts the adaptability of the conceptual architecture. Decomposition is the means by which this perspective plays a role in the definition of the structure of the system. The fundamental elements of the architecture can be conceptualized in many different ways: functional, physical, or disciplinary. Physical and disciplinary decomposition approaches require the designer to assume relationships between elements of the architecture. These assumptions simplify the structure of the system and put all the elements in a reasonable place. However, with the implementation of revolutionary technologies, these assumptions are no longer valid. Traditional architectures are formulated using a systems approach to architecture definition and are subsequently subject to the assumptions which fix these relationships. The only breakdown which does not unduly constrict the design space through assumed physical relationships is a modularity based on the functions. This is due to the fact that functions are completely independent of the physical architecture. However, functions must be formulated in a manner which maintains their independence and provides a link to physical architecture. This formulation of functions is discussed in the next chapter.

CHAPTER 4
FUNCTION BASED ARCHITECTURE DESIGN

In order to address the constrictions imposed by assumptions made regarding the physical description of the architecture and internal functional relationships during functional allocation a flexible process of architecture definition is required. Because functions provide a uniform framework upon which any physical implementation can be applied, they were adopted as the central element of the process of architecting described by this paper. Basing the architecture definition of a complex system entirely on functions is difficult. Carliss Y. Baldwin and Kim B. Clark state, "After some analysis, we concluded that it is difficult to base a definition of modularity on functions, which are inherently manifold and nonstatinoary" [12]. The ability to decompose and define an architecture on the basis of functions requires an appropriate definition of concepts and theory which enable this adapting, non-stationary framework.

The facilitating capabilities used for this process are adopted from the work done by the Optimized Aircraft Power Architecture (OAPA) grand challenge team from the Aerospace Systems Design Lab (ASDL) at the Georgia Institute of Technology. The OAPA team was commissioned by the Energy Optimized Aircraft Systems (EOASys) Program Committee from the American Institute of Aeronautics and Astronautics (AIAA) to consider the system level impacts of electric technologies on aircraft architecture definition and integration. This method of functional decomposition is also described in Mehdi Hashemian's thesis from the University of Saskatchewan Saskatoon, "Design for Adaptability."

Adopting this methodology allows the designer to define the level of modularity on the basis of functions. In so doing, any physical orientation which can be applied to the fulfillment of the product's functions can be considered as a valid alternative. Therefore, the relationships between the functions of a product and the physical implementation of these functions must be understood and defined by the designer. In order for this to occur, functions must be organized in a way as to provide a means to facilitate physical definition.

Decomposition

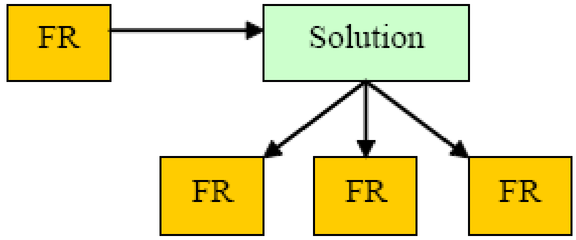
Boundary and Induced Functions

Every physical element within a system is implemented to provide some functionality. In turn these physical elements impose new tasks that must be fulfilled. This linking of functions required to accomplish some product level task is referred to as the functional chain. There are two different kinds of functions making up the functional chain: Boundary and Induced.

Boundary functions are functions defined by the product requirements, which are non-architectural specific. These must occur regardless of the physical description of the product. Induced Functions are imposed by choices regarding the physical fulfillment of other functions. Therefore, functions take the form of new requirements imposed by a physical system or grouping of physical systems.

Nam Suh describes these relationships by referring to this as a hierarchy of functional requirements. He asserts that the functional requirements at a certain hierarchical level of the functional definition cannot be defined until the means of physical fulfillment to the functions within the previous level have been developed [54].

Mehdi Hashemian describes this as the concept of recursive decomposition. He states that the functions and sub-functions have a direct causal relationship between each other by means of the physical implementation of the function. This is displayed in Figure 21. In this figure functional requirement (FR) is fulfilled by a given solution. This solution in turn induces additional functional requirements [64].



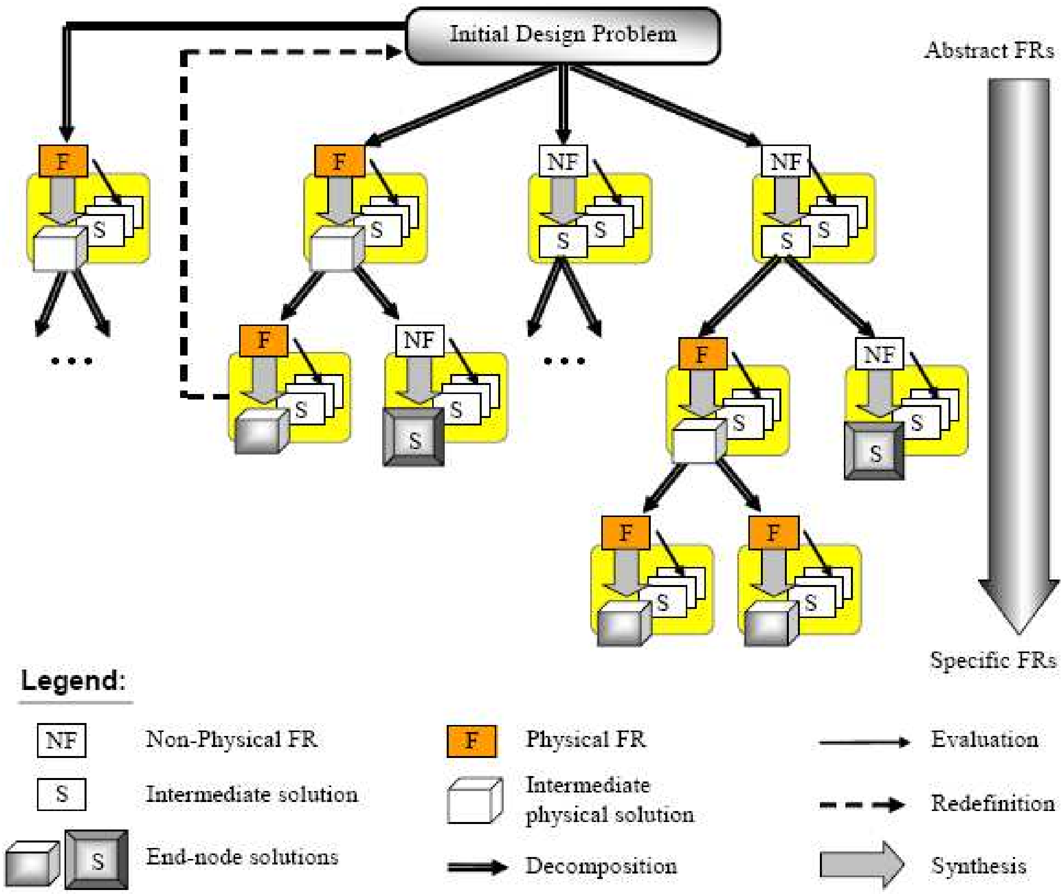


Figure 21: Functional decomposition from Mehdi Hashemian thesis

Suh gives an example of this functional hierarchy, seen in Figure 23. Here, a lathe is the overall concept which can be decomposed into constituting elements (in the boxes). These boxes are needed as a fulfillment of a given function (indicated by the arrows). Once an element is defined new functions are defined. In this case the use of a gear box to fulfill some product level function induces the need to fulfill new functions which are embodied by the spindle assembly, feed screw, and frame.

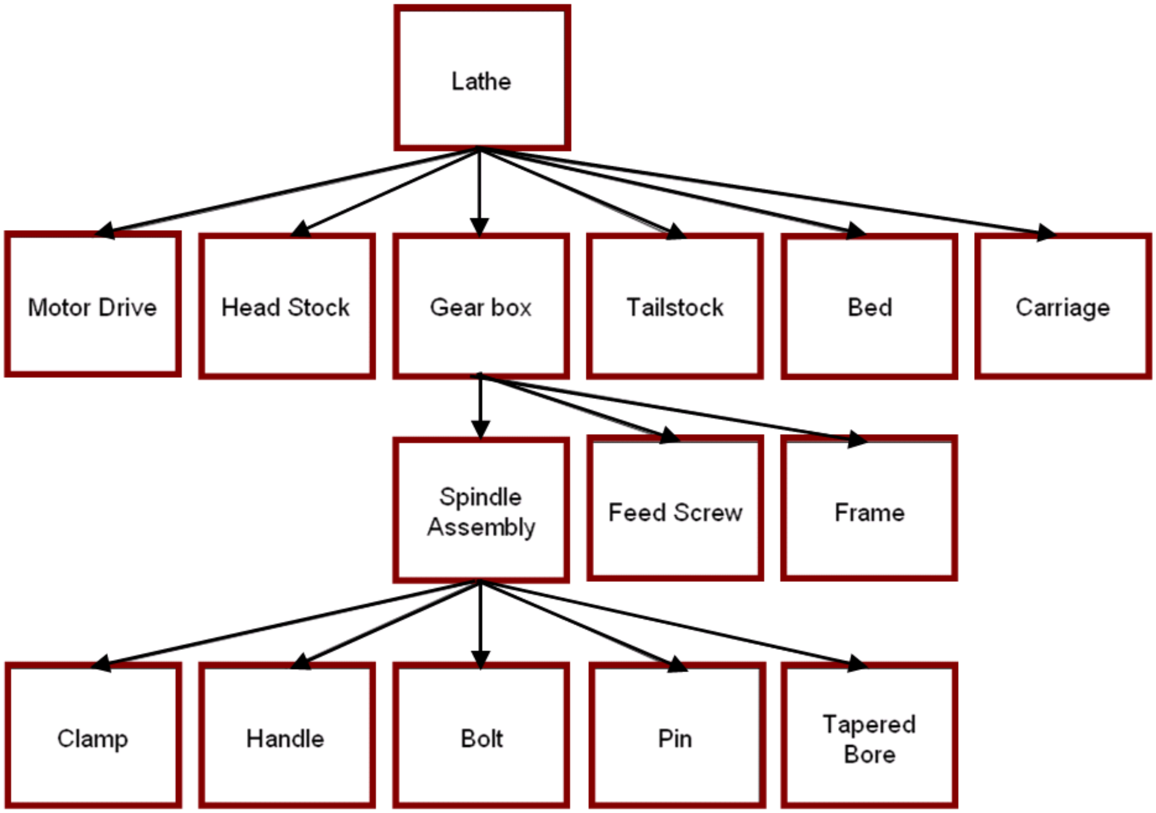


Figure 22: Hierarchy of lathe design in physical domain from Suh [54]

Another simple example for a functional chain is one defined by the fulfillment of functions accomplished by a flashlight. This is displayed in Figure 7. In this case, the flashlight's main function is to generate light. This function is the boundary function because it must be unequivocally fulfilled by the design. Many alternatives can be chosen to fulfill this function. Many elements have the capability to produce light. These may include alternatives like light bulbs, fire, bioluminescence, etc. Some alternatives are the more logical choices because each will induce another set of functions that need to be fulfilled to enable this physical element to work.

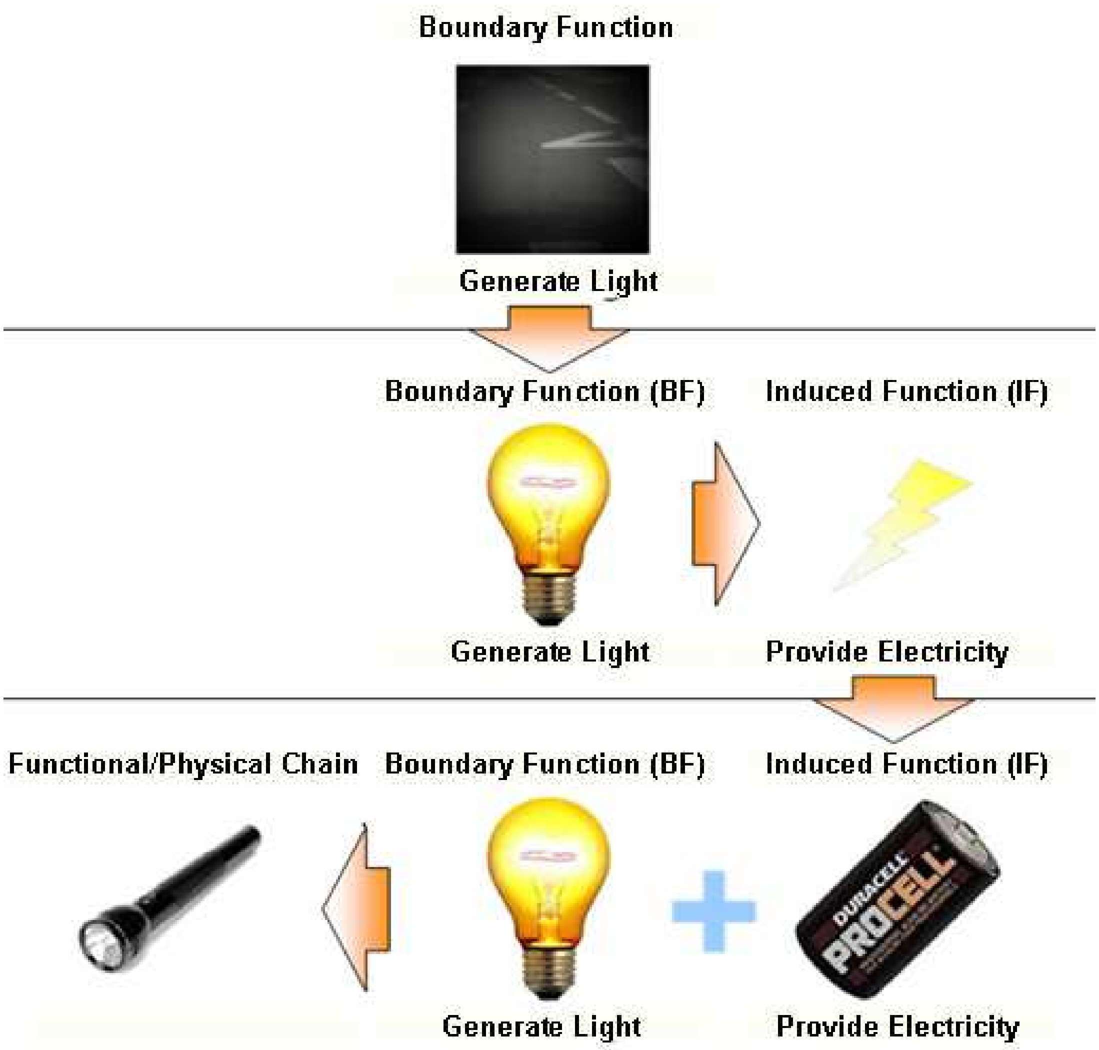


Figure 23: Boundary, Induced, and Function/Solution Chain Generation

Suppose a light bulb is chosen to fulfill the boundary function to generate light. In order for this function to be fulfilled, another function is induced. This induced function is to provide electricity. If other alternatives had been chosen, the induced functions may have been very different. The initial functional breakdown of the product must occur at the boundary function level. These functions are intended to be independent of the physical implementation and must remain fixed for all architectural concepts. Induced functions are explored after this generic breakdown is achieved and actual physical elements are identified which can fulfill these boundary functions.

Following this approach, alternative definition is a portion of the decomposition process. The alternative elements are the essential building blocks of a complex system. The functions describe actions, and the elements themselves indicate behavior.

Alternative definition and functional analysis are also closely correlated because induced functions are directly related to which alternatives are included in the architecture design space. Functional analysis and alternative definition need to be performed iteratively as shown in Figure 24.

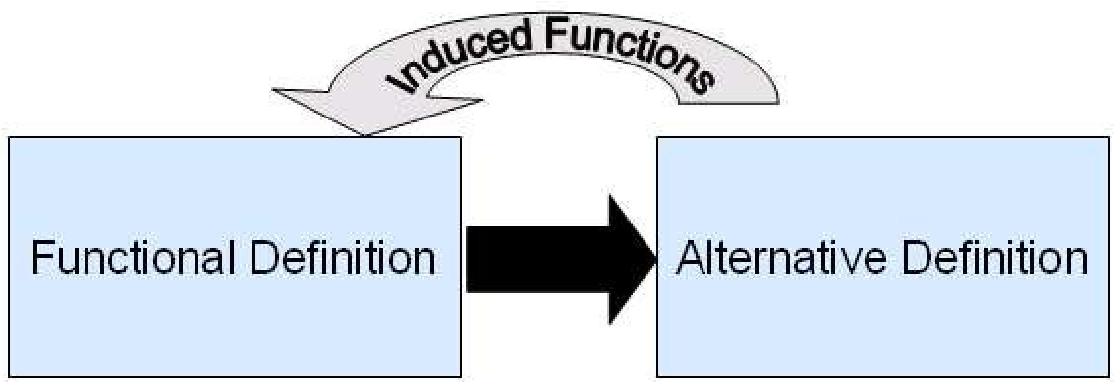


Figure 24: Functional and Alternative Definition

The alternatives and the functional interrelationships defined for each alternative constitute the architecture design space for the complex system. With these in place the designer can proceed to explore combinations of alternatives within this space with varying interrelationships. This exercise of building the design space for the architecture requires tools which characterize the relationships between Function/Solution Chains and the physical system elements.

Implementing induced functions into the conceptualization of the architecture allows the functional description of the architecture to changes as decisions are made. These functional relationships, which are traditionally assumed or defaulted, govern the structure of the system, determine the requirements on each of the elements in the system, and affect the overall performance of the architecture. Characterizing these induced functions provides for flexibility in the structure of the architecture. By categorizing physical elements by their impact on the function requirements on the architecture, the architecture becomes modularized based on the functions which the architecture is fulfilling. Typical architecting "schemes" are reflections of well-understood functional relationships. However, the effect of revolutionary technologies on an architecture requires flexibility which can be captured through the induction of new functional relationships.

Functional/Solution Chains

Once the boundary and all of the induced functions are identified, this combination of tasks needed to fulfill the boundary function is described as the functional/solution chain FSC or aggregate function. The attributes of the physical embodiment of Function/Solution Chains become product level physical descriptions which can be compared between architectures. The boundary functions are consistent across architectures. Therefore, comparisons of architecture alternatives could take place on the basis of functions.

With these classifications of functions, a different form of modularity emerges. This modularity is based on the fulfillment of functions in a functional framework. In this functional framework the designer explores the fulfillment of all of the functions by the definition of alternative functional chains. This allows the architecture to take on widely varying physical forms.

Figure 25 and Figure 26 display the difference between a traditional convention for systems modularity and the concept of functional modularity.

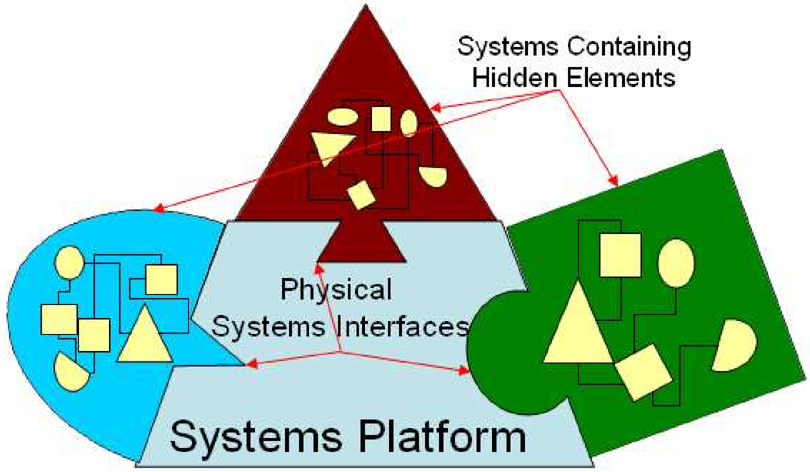


Figure 25: Traditional Systems Modularity

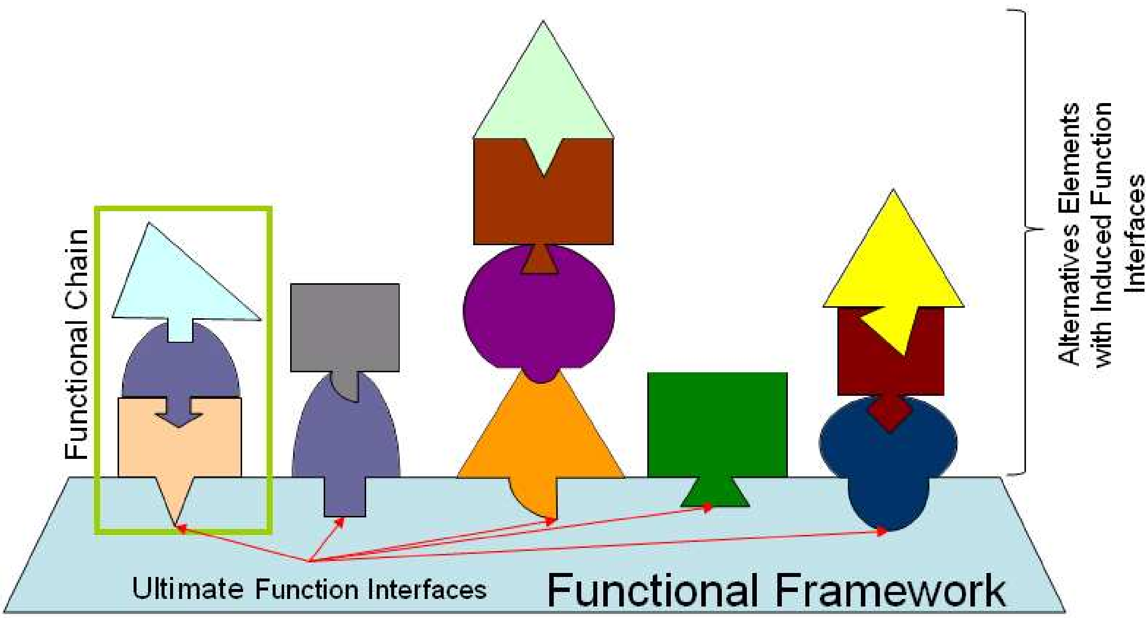


Figure 26: Functional Modularity

Both methods of defining a modular architecture require the definition of all of the physical elements represented by geometric shapes in Figure 25 and Figure 26. However, the traditional approach adopts predefined systems interfaces, while the functional modularity approach allows a flexible definition of elements within the system.

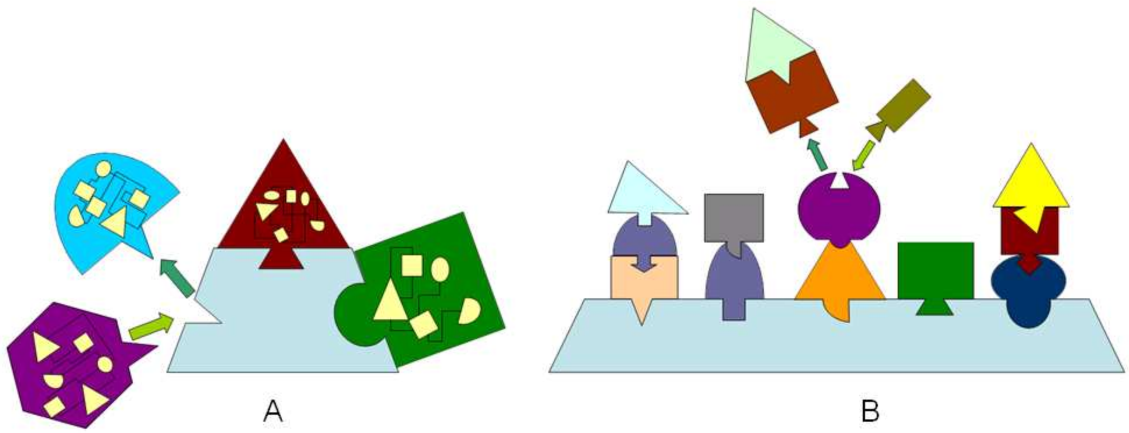


Figure 27: Product Trade-offs A) Traditional Systems Modularity, B) Functional Modularity

Figure 27 is an illustration representing the redefinition of the product architecture. Using the traditional systems modularity approach systems and technologies may be exchanged easily as long as interfaces are preserved. With functional modularity any technology can be applied when called upon by the function. No assumption is made regarding organization. Again, it should be mentioned that when interfaces defined by the traditional approach are altered significant alteration to the architecture definition is required. Interfaces defined using functional modularity are not constricted by physical definition. However, there may be intricate interrelationships developed when physical systems fulfill functions in multiple functional chains.

These functional chains can be compared to the function flow block diagram, which is a flow diagram which shows the relationships between the fulfillments of functions. In the context of a function flow block diagram the functions are generally arranged in a sequential order, designating which tasks must be fulfilled in which order [15]. The idea of functional modularity does not specify that the tasks must be sequential. The functional relationship defines some physical or logical interface, through which information or power is transmitted.

In contrast to traditional methods of characterizing the functional chains, the process of inducing functional requirements introduced in this text does not assume functional relationships and chains before physical elements are included in the system. Other processes begin with the definition of a function flow block diagram [65] and assume relationships between functions, while this process begins with the classification of functions and physical elements and allows these chains to be built as decisions are made.

With a functional flow block diagram or this method of functional decomposition there can be highly intricate interactions between elements within different functional chains. Hashemian assumes in his work that the functional structure is ideal, meaning that there are no relationships between functions except between parent function and sub-function. This allows the functional chains to be considered independently. However, it does not take possible interactions into account. For example, the electrical system on an aircraft fulfills the function to provide electricity. This function is required by multiple elements within the functional structure. The requirement for electricity is imposed with the implementation of electrical anti-icing systems which would fulfill the function to protect from ice and the requirement to provide light to the cabin. Thus one physical element is dedicated to multiple functions.

Considering products on the basis of induced functions allows generalized grouping of induced functions to be made. Some induced functions appear often in complex systems. For the processes and tools defined in this text, these general groupings are related to the use of energy throughout the architecture. Four main groups appear: providing, transforming, storing, and distributing some type of material, energy, or information. These functions are entitled power functions because they typically govern the overall efficiency of the system and track the use of power throughout the architecture. Power elements within a complex system provide functionality for many different elements within the functional chain. Thus, a single physical element can fulfill the functional requirements at any level of the hierarchy. This adds the necessity to interrelate the requirements of each of the functional chains.

Summary

The functional breakdown developed for this process of architecture definition requires the classification of functions between boundary and induced functions. Not all functions that must occur in the architecture are independent of the physical nature of the architecture. Only those which interact with the environment in which the product is used can be defined as fundamental elements of the architecture. Induced functions also play a role in fulfilling the overall product functions in order to support other physical elements in the architecture. Thus the relationships between functions and physical elements create functional chains. In these chains, functions require physical elements and in turn the physical elements induce new functions. These chains can be highly interrelated. Elements in one chain may fulfill multiple induced functions. In order to formulate these functional chains in a directed and logical manner, tools and processes must be developed to manage all architecture decisions that must be made, all physical elements in the system, and all relationships that can exist between these elements.

CHAPTER 5

FUNCTION BASED ARCHITECTURE DESIGN PROCESS

In review of the topics discussed in the previous three chapters it has been established that aircraft design is a process which is subject to many conflicting requirements imposed by the market, regulations, and operating environment. Designing a complex system, like an aircraft, requires the integration of multiple elements within the system. The process of defining the elements within the system and the relationships between these elements is termed architecture design. Architecture considerations are generally addressed in later portions of the design process and are constrained by inappropriate assumptions made during concept definition. In order to infuse knowledge forward in the design process some sort of model must be applied which captures more detailed information regarding the architecture. Although all models can be useful, physics based models are the only types of models which do not rely entirely on assumptions and tacit knowledge.

Traditional approaches to architecture definition and analysis are subject to limitations because of a hybrid approach to decomposition (physical, functional, disciplinary). Also, proper conceptualization and methods have been briefly introduced in the previous chapters. In order to focus the architecture of functions boundary and induced functions must be organized in a manner which maintains the relationship between function and physical definition and guides the organization and implementation of analytical models.

In order to overcome issues associated with the traditional approach and to maintain the information necessary to build this architecture, a function based architecture design process was defined which is based on functional decomposition and a systematic and flexible process for defining the physical nature of the architecture. This chapter introduces the process and the Architecture Design Environment toolset developed for function based architecture definition and modeling. The remaining chapters of this document will detail the specific tools and methods implemented in the steps of this process.

Process for Architecture Design

Architecture definition, like all complex design tasks, is a process of decomposing the product into fundamental elements or concepts and determining the means of fulfilling each one through a synthesis or physical definition process. This process for function based architecture definition is displayed in the Vee diagram in Figure 28. The fist side of the Vee represents the conceptual decomposition, and the right side represents the process of defining an architecture alternative. This process of architecture design is meant to link the requirements of a product to its physical form, providing means by which this architecture instance can be analyzed and compared to other candidates. Thus, the initial step is the process of understanding the product requirements and the system concept, and the final step on the left is using the information regarding the concept to determine the performance of the defined architecture and comparing it to the requirements.

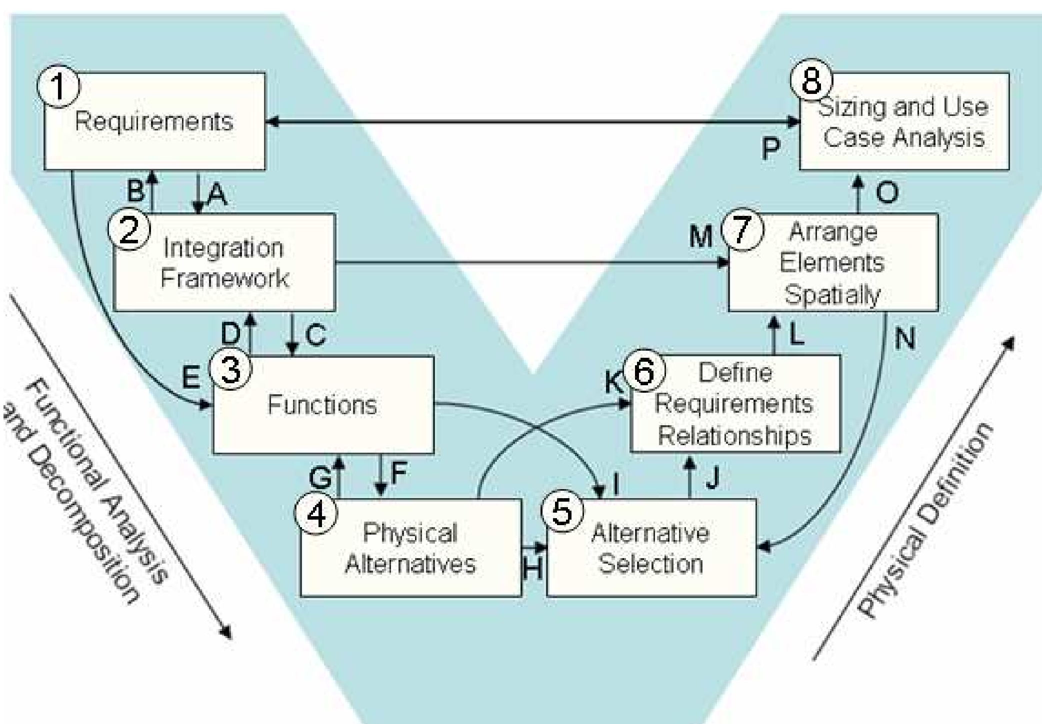


Figure 28: Function Based Architecture Definition Vee Diagram

A brief overview of the process is given here. More information regarding the tools and processes used in this process is given in the following chapters.

The designer must first understand the scope of the design, the product requirements, and the governing issues of the design task. This task includes the definition of values of merit by which architectures are compared (arrow P, Figure 28) once an architecture is defined. With these values of merit and the defined scope (arrow A, Figure 28) the rules which determine system level relationships for the project must be defined. These rules regard the physical installation of elements within the system and the information which will be needed from the functions and physical components. Examples of scope include the relationship between physical zones within the system and the environment. These defined installation relationships determine the information which is generated when specific alternatives are defined to zones. This is a definition of what happens to the overall system based on installation decisions (heat transfer relationships, geometric relationships, drag impact, new induced functions, etc.). This topic will be further discussed when installation is introduced.

The scope and requirements of the project and the installation considerations lead to the definition of the boundary functions of the product (arrows C and E in Figure 28). As discussed in the section entitled "Boundary and Induced Functions." There is a tight relationship between the definition of physical alternatives and the definitions of the functions of the product, hence, the feed forward and backward relationships between these two exercises (arrows F and G in Figure 28). The other upward flowing arrows B and D indicate a relationship between this function and alternative definition and the relationships governing the attributes and effectiveness of the overall system.

The first step of the definition process is the selection of physical alternatives to fulfill the functions. These decisions are based on the boundary functions, the alternatives available to fulfill the functions, and the functions induced by alternative definition (arrows H and I). The relationships between these alternatives must then be designated based on the alternatives selected and their defined attributes. As the alternatives and relationships are defined they must be physically placed within the system. This is done during installation. As this is done the rules governing installation (arrow M, Figure 28) determine if new alternatives must be designated to fulfill new induced functions (arrow N). Finally, once the elements are chosen, networked, and placed within the system, the attributes of these elements and the performance of the system architecture must be determined based on the figures of merit defined initially.

This process was fashioned to utilize the relationships between functions and physical elements on the basis of functional induction and to integrate decisions made about elements into the definition of an architecture. It also allows for the modular management and grouping of all architecture knowledge in a way which facilitates the modeling and simulation of the architecture.

Architecture Design Environment (ADEN)

Complex relationships developed in the definition of boundary and induced functions motivate the definition of tools and processes to fulfill steps of architecture design displayed in Figure 28. These tools utilize a flexible functional framework and tightly integrated process of defining the fundamental physical elements and functional relationships as well as the definition of the alternative concepts. These tools are superimposed on Figure 28 in Figure 29 below and will be discussed in the next two chapters.

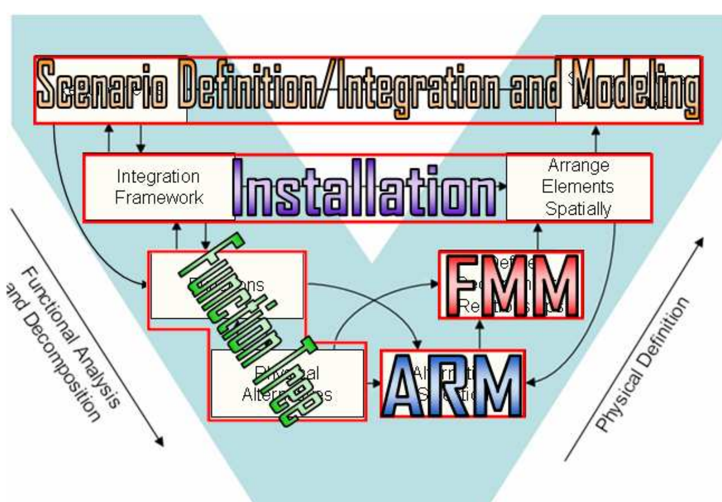


Figure 29: Architecture Design Tools

The complex relationships and interactions between steps of the architecture design process induce the need for tool interplay which provides fluidity and management of all the decisions made. An object oriented program called the Architecture Design Environment (ADEN) was developed to facilitate this process of architecture design and to address the third portion of the hypothesis of process integration. This toolset focuses primarily on the functional to physical definition of the architecture represented in steps 3-5 in Figure 28 but was built with the intent of interfacing and utilizing information generated in other portions of the design.

The scope of this research was to develop the process, tools, and interface with which a complex systems designer will be able to define an architecture design space and easily identify candidate architectures. This interface includes functional and alternative definition, alternative selection, configuration definition, an interface for installation definition, and a method to defining the operating space for the architecture. This tool is intended to be the method by which architecture definition tools can be integrated to assess the performance and practicality of multiple designs. Therefore, the description of the architecture will be defined as a means which will easily be accessible to existing integration software. Installation considerations (steps 2 and 7, Figure 28) and overall architecture evaluation (steps 1 and 8) are to be handled as external tools integrated into the ADEN framework. The ADEN information flow diagram is displayed in Figure 30.

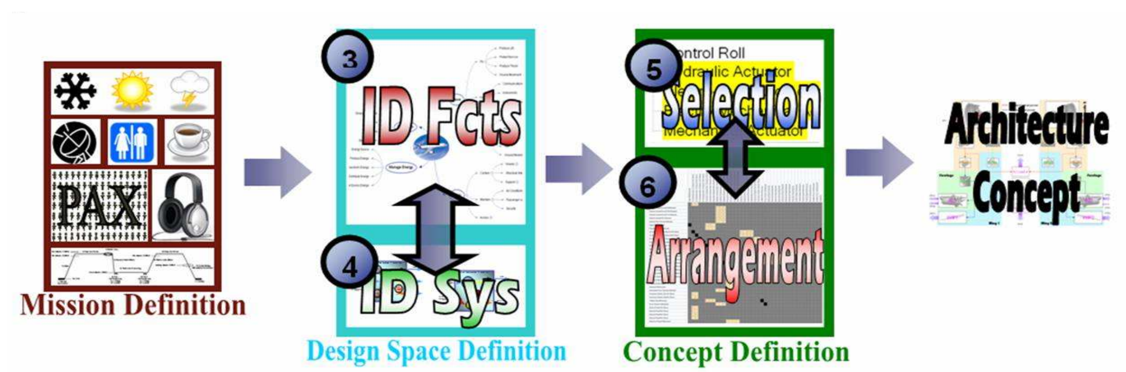


Figure 30: Architecture Definition and Analysis Process

This process utilizes functional and physical definition tools described in chapter 4. The design space definition process follows the principles of functional induction and utilizes a functional tree to manage and organize both functions and alternatives. The concept definition uses the functional tree as the means to select alternatives in the Adaptive Reconfigurable Matrix of Alternatives (ARM) and to indicate the axis of the Functional Mapping Matrix (FMM), which then is used to configure the relationships. The ARM and the FMM will be discussed in chapter 7.

To validate the flexibility and applicability of the tools, methods, and theory, this interface will be applied to the architecture design of a commercial transport aircraft. In this case study, the aircraft will be decomposed and candidate architectures will be developed. In definition of these alternative concepts, limitations and assumptions of this process will be discussed as well as future work that would be required to refine the process and its implementation. This would include future work regarding the theory associated with the concepts discussed in this paper and the requirements of tools and methods which would be used to manage the information generated by this process to physically size and assess the performance of the defined architectures.

Architecture Design Environment

In order to provide the flexibility of design space and concept definition the ADEN tool was designed in an object oriented environment. As a Visual Basic tool, the structure of the functional breakdown and the instantiation of physical elements can be easily manipulated. The tools and principles used in both the design space and concept definition processes are reviewed in this section. More details about the ADEN tools are available in Appendix C.

The tools developed to embody this process were created with two main interfaces. These interfaces are displayed in Figure 31. The first interface is used to define the design space of the architecture in terms of the functions and possible physical alternatives to fulfill those functions. The second interface is intended for concept definition. With this interface, the architect defines which elements are used to fulfill the functions, how these elements are interrelated, where the elements are placed in the architecture, and how the mission of the aircraft will be configured. The specific elements of these tools will be discussed within the next chapter.

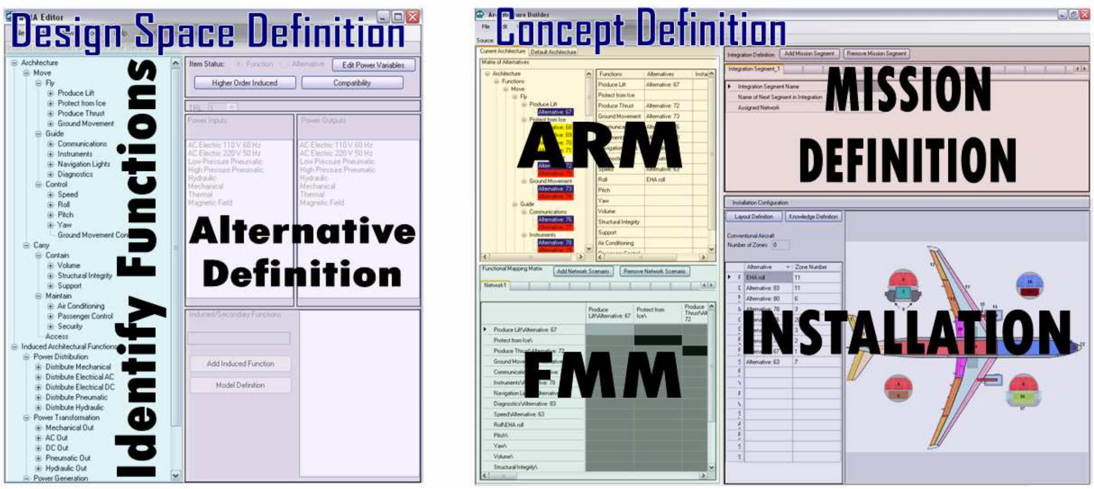


Figure 31: ADEN Interfaces

These interfaces utilize Visual Basic command and control tools, such as tree and gridviews, as well as data lists and images. The VB interface allows for the organizing of the data on the screen in a manner which facilitates the definition of the architecture. Rules defined during design space definition are used during concept definition to manage feasible choices and cause functions to be induced in the ARM based on decisions made in other portions of the design process.

The output of this tool is a script based description of the architecture which lists all of the relationships occurring between models used in the performance analysis of the architecture. These relationships can occur amongst system element models, closely spaced groupings of elements in zones, and the mission analysis. These relationships are shown in Figure 32.

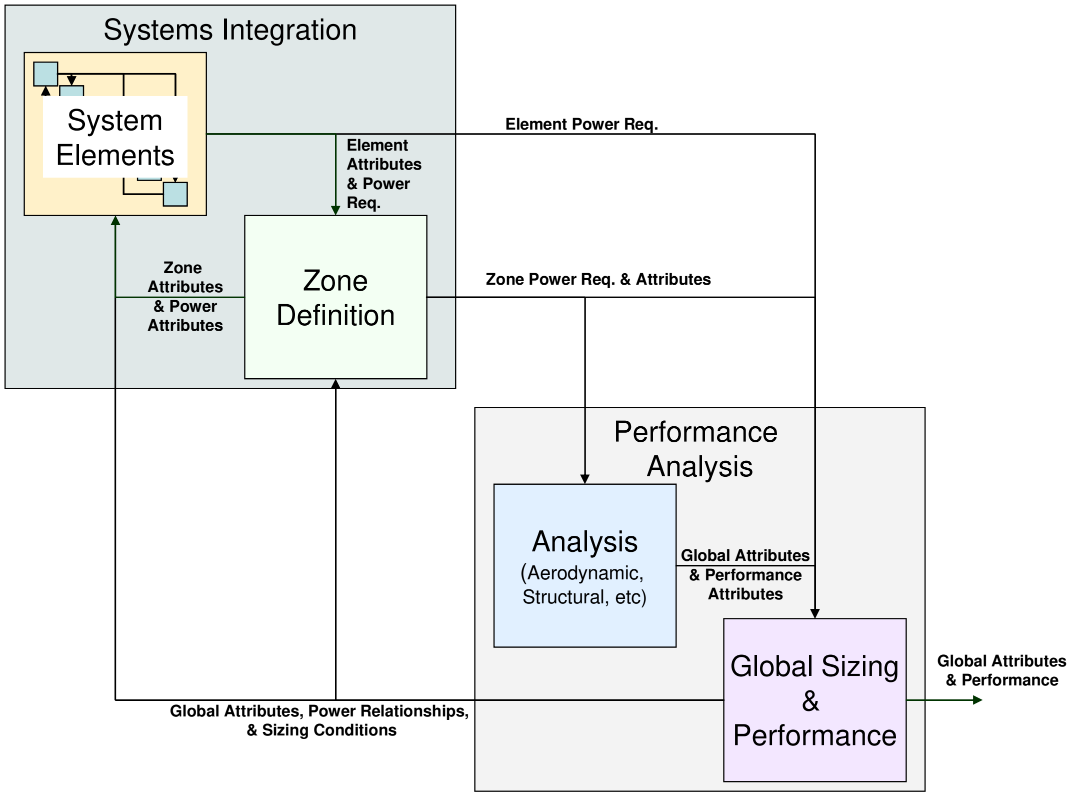


Figure 32: Model Relationships DSM

The output file consists of a listing of every model that will be needed for simulation, a mapping of all the relationships between the models which exist regardless of the network connections (attributes relationships, zone location relationships), and a listing of each power relationship governed by the FMM. The relationships between the zone definition and the performance analysis models and between the analysis codes and the global sizing and performance models are static for all architectures defined. All other relationships between systems elements and the other modeling and simulation elements can change for different architectures defined. The relationships within the system elements can change during modeling and simulation and must be allowed to adapt for different sizing scenarios. This will be discussed further in the next chapter.

Each one of the relationships between these models is defined in the output file, completely defining the modeling and simulation environment. The ADEN toolset acts as the interface in which architecture decisions drive modeling and simulation, which occurs in the background.

Summary

This functions based process for architecture definition requires appropriate decomposition and a flexible means of alternative definition. The decomposition process involves understanding and translating requirements to appropriate mission definition and functions, identifying the terms by which the attributes of architectural elements will be used to define the overall critical attributes of the architecture, decomposing the requirement to boundary and induced functions, and identifying alternatives which can be used to fulfill these functions. Physical definition of this architecture involves selecting technologies and components to be used in the fulfillment of these functions, defining the relationships between these elements, organizing these elements in some spatial layout, and assessing their performance in fulfilling the requirements. Tools were used or adapted to address each step of this process for architecture design and will be discussed in chapters 6 and 7. The Architecture Design Environment (ADEN) integrates these tools and provides an interface in which a designer can make the decisions necessary to define the architecture. A benefit of this process of decomposition and definition is that each physical element must be characterized by all information necessary to determine its performance. In so doing, the I/O for modeling and simulation are readily available with the conceptual definition of the architecture. If each element in the architecture is represented by a physical model, this process defines all relationships between these models and defines the use-cases of these models for simulation purposes. By defining the means by which all architecture elements should be interrelated, the ADEN tool has the capability to bridge some of the gaps between architecture conceptualization and physics based sizing and analysis.

CHAPTER 6

DESIGN SPACE DEFINITION

In order to develop architectural concepts, it is important to identify the extent to which trades will be made and the level to which this architecture will be defined. The designer must identify what decisions must be made and all the possible means by which a solution to these issues can be found. All combinations of possible solutions constitute the design space for the architecture. In this process, the design space for the architecture is made up of all functions which characterize the architecture and all possible physical elements which can be employed to fulfill these functions. This chapter addresses development of the functional breakdown, including boundary and induced functions, general groupings of induced functions, physical element characterization, and the means by which element attributes are defined and integrated to the product level (mission scenarios and zones).

Requirements

Requirements analysis is the means of generating a valid description of desired product attributes or goals which are logically organized to guide product development. It should be noted that requirements analysis can be done without considering the technologies that will be implemented in the product. Requirements analysis considers what needs to be done by a product and is not troubled with how these are to be accomplished. These needs include the product's purpose, the critical players, the performance requirements, the operational and time constraints, and the metrics for success. David Hays states in his book "Requirements Analysis" [66],

> "It is important not to confuse requirements analysis with system design. Analysis is concerned solely with what some call the problem space or the universe of discourse … Design, in the solution space, is the specific application of particular technology to address that enterprise … There is a common tendency for designers, when they are analyzing requirements, to construct the analysis results in terms of a particular technology … They

go into the effort with preconceptions of what the solution space is going to look like, so they seek out problems they already know how to solve."

Requirements analysis precedes all definition of the product functions or physical attributes. Identifying appropriate boundary functions for a product begins with identifying the environment in which it is to operate throughout its lifecycle regardless of the physical structure of the product and the relationships of this environment and the product itself. Moir and Seabridge discuss typical design drivers that are present in the requirements analysis of an aircraft: safety, cost, environmental conditions, performance, quality, human/machine interface, structure, crew and passengers, stores and cargo, functional performance, and standards and regulations [22].

These external influences can be categorized into coherent groupings. The DoD recommends grouping these requirements in a database which lumps these design drivers into project requirements, mission requirements, customer specified requirements, and interface, environmental, and non-functional requirements [55]. This organization of the desired attributes can be considered as a concept of operations (Con Ops) as described by the INCOSE [15]. This document gives a complete description of all product requirements and performance metrics.

This description of requirements in the concept of operations must then be translated into inputs to this function based architecting process. The requirements generated during requirements analysis impact the functional design space by providing the boundary functions to be fulfilled, defining the sizing scenarios for each technology, configuring the constraints and input attributes to the sizing models, and comparing independently designed systems performance attributes. Each of these groups of attributes is derived from the different categories of requirements.

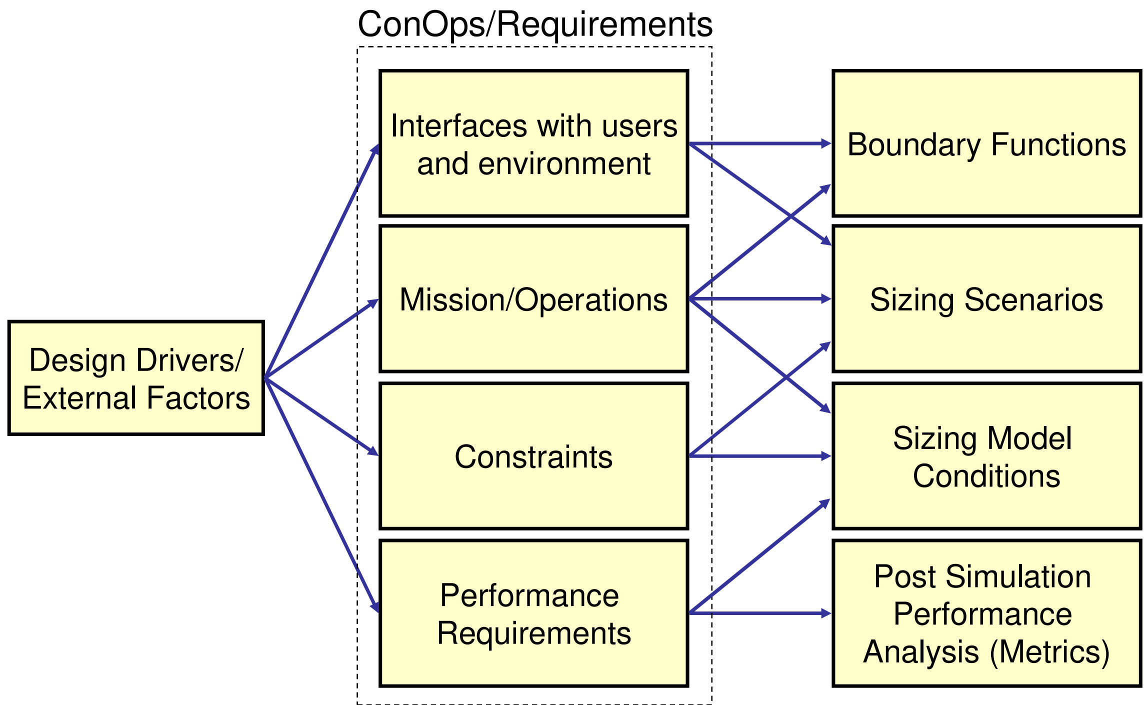


Figure 33: Mapping of Design Drivers to Design Space Definition

The inputs to this process and their relationships with the requirements described in requirements analysis are described in this section.

Boundary Functions

This process of architecting is facilitated by means of boundary functions. These functions are the actions that must be achieved by the architecture, defined by the architecture's interfaces with the environment and users and the mission description. The desired actions of the product can be described in or inferred from the requirements document. These can be stated outright ("the aircraft must …") or can be contingent on other information in the requirements documents. For example, the function to protect the wing from ice or to provide grounding during a lighting strike may not be directly stated in the requirements document. However, both are necessary functions that must be fulfilled to design an effective commercial aircraft. These requirements are inferred from the environmental conditions and the interactions that the product will have with the environment.

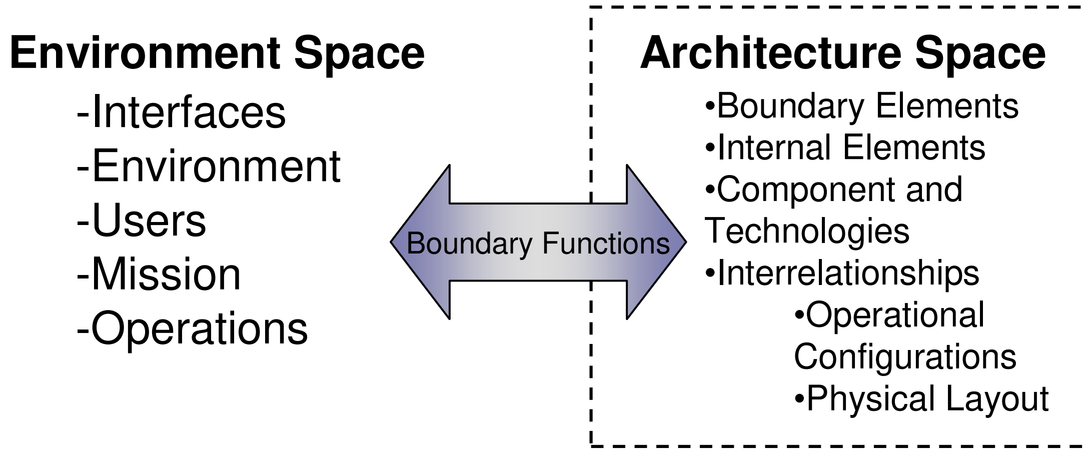


Figure 34: Boundary Functions- Interactions between Architecture and Environment

As indicated by Figure 34 a boundary function is the description of the interface between the architecture and all environmental conditions. In order to allow the architecture to take any general form, these relationships must be generated from the requirements alone and independent of the physical solution [66].

Boundary functions are actions which, when fulfilled, directly accomplish the requirements of the aircraft. Much of the complexity in deriving standard terminology and taxonomy for functional definition stems from the missing of boundary and induced functions. As discussed in chapter 3, an appropriate level of detail is necessary to completely define the function and understand the alternatives available to fulfill the function later in the process. This boundary function must be stated in a way which facilitates the conceptualization of a potential fulfillment of the action. Many "typical" functions for a product do not fit the description of a boundary function but are induced by other architecture decisions. Appendix D shows a comparison of typical function defied by multiple authors. Induced functions will be discussed later in this chapter.

Sizing Scenarios

The sizing scenarios are determined by a description of the mission and objectives, the interfaces, and any constraints imposed on the product. In aircraft conceptual design, sizing is typically performed with a mission profile and fuel fraction calculation. The mission is segmented into phases, each representing a portion of flight which imposes different environment conditions, functional requirements, and constraints on the aircraft. Examples of four different mission sizing profiles are shown in Figure 35 [41].

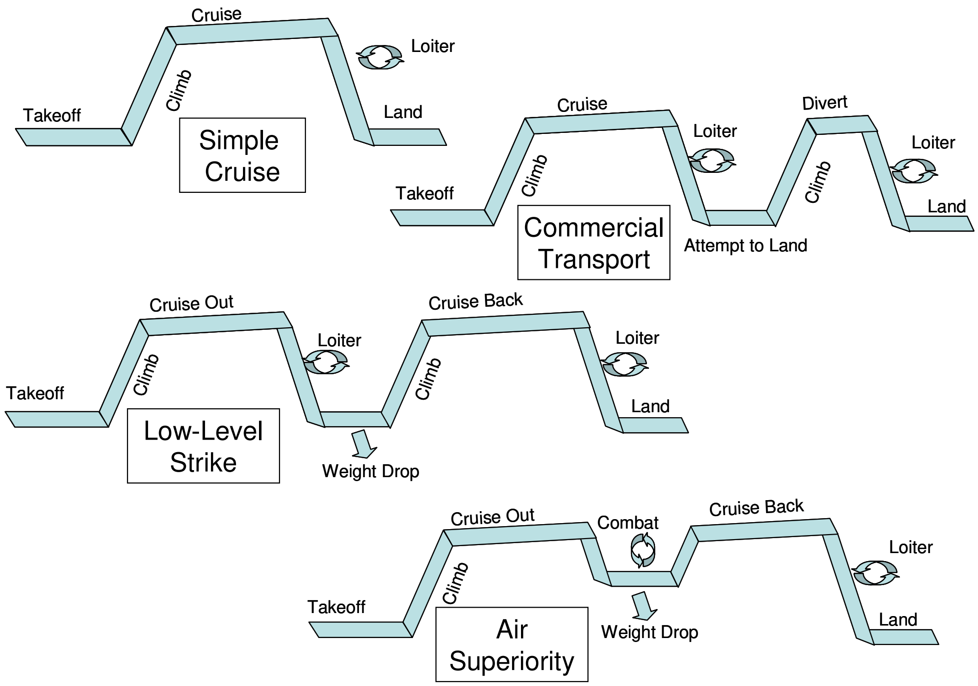


Figure 35: Typical mission profiles for sizing (Raymer)

These mission scenarios illustrate the mission requirements of four very different aircraft and highlight different functions that are to be performed during different portions of the mission. These mission phases are defined by the boundary interfaces during each mission phase. For example, during the cruise portion of the commercial transport mission, food service may be provided to passengers in the cabin. This indicates a new relationship with the environment space (passengers in the cabin). Thus, new boundary requirements and attributes (energy requirements, cg shift etc) must be captured in the sizing scenarios. This may not have a direct effect on the overall geometry of the aircraft; however, combinations of requirements and internal effects can impact the overall performance of the aircraft. Each combination of interactions with the environment which the architecture will see in its operation must be used to size the architecture.

Sizing Models

Each function must be fulfilled by some physical element or combination of physical elements. Thus, as functions are defined, physical alternatives must be identified as part of the architecture design space. These alternatives must be defined and characterized by their I/O. Each element must be sized for all sizing conditions, thus imposing its own attributes and requirements on the other architecture elements differently at each condition.

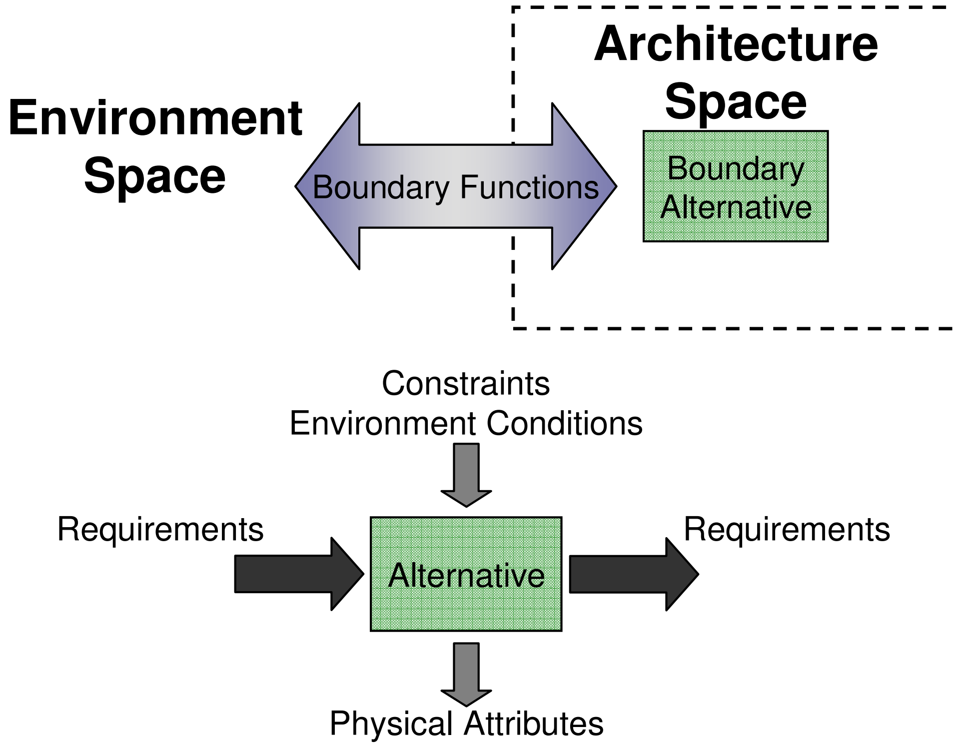


Figure 36: Alternative Characterization

The model representing these physical elements allows the generation of new requirements implying new induced functions. Induced functions will be discussed further in this section. The attributes and requirements of the alternative are dependant on the constraints and conditions under which it performs and the measure of the requirement that it must fulfill. These conditions are determined by the sizing scenarios which size the architecture, the constraints governing the physical attributes, and the performance requirements. Characterization of the elements of the architecture allows these models to be configured and linked in order to size them concurrently in an iterative manner.

Post Simulation Performance

The architecture generation process will be discussed in the concept definition section. Once an architecture concept is defined and the sizing and synthesis is performed, candidate architectures are compared and the best architecture is selected. This comparison is performed on the basis of the performance attributes (metrics) of each architecture.

These metrics of comparison must also be described by or inferred from the requirements description and are directly related to the performance requirements. Decisions must be made as to whether the requirement will be handled as a means of comparing two complete architecture concepts or as constraints n the modeling process. For example, reliability and safety can be handled through probability calculations which can provide qualitative comparison of complete systems. However, specific safety hazards can be addressed with specific functions (provide fire suppression, prevent disk perforation). Safety criteria must be handled as guides to which elements can be selected, how these should be interrelated, and where these would be appropriately positioned in the layout. These performance requirements can be handled as conditions for architecture generation or as points of comparison between architectures.

Typically in aircraft design, weight and total fuel burn are often the attributes which are used to compare aircraft designs. Other qualitative comparisons between products can also drive the choice between two candidates. Metrics like overall look, comfort, and the way in which the product is perceived are more difficult to compare but can be addressed by non-qualitative means (e.g. focus groups, surveys, etc).

Induced Function Definition

The concept of functional induction was described in chapter 5. However, here we discuss grouping of induced functions to facilitate model development. In order to develop the structure of the model representing this architecture concurrently with the definition of the concept, rules must govern the relationships and transfer of information between physical entities [67]. The induced functions must be formulated in a way which facilitates model definition. The categories of induced functions utilized in this process are power functions [68] (power distribution, transformation, generation, and storage), secondary/tertiary functions, and installation induced functions.

Power Functions

The use of energy has become one of the highest concerns in commercial aviation. Energy efficiency by means of fuel consumption per revenue passenger mile is one of the most critical measures of effectiveness for commercial aircraft performance. Every element used in the aircraft architecture affects the energy performance of the aircraft. This occurs directly through energy requirements to operate these elements or through the means of supporting the physical elements itself (weight/lift, volume, and drag/thrust) [68]. Physical attributes are provided by the sizing model of the element itself and are needed to generate attributes at the overall system level, while power related information is directly needed by other system elements. This is illustrated in Figure 37.

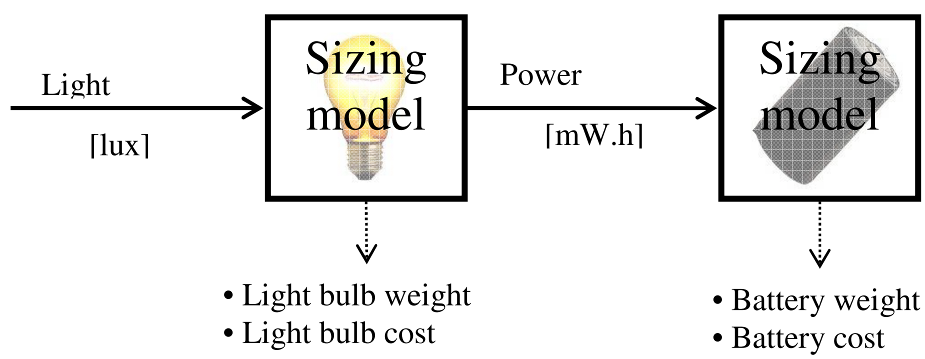


Figure 37: Power and Attribute Relationships

In Figure 37 we see two system elements: a flashlight and a battery. The relationship of each element to the design environment space is captured in terms of system attributes (e.g. weight and cost). These are needed in order to size the integrated system and ascertain overall effectiveness. However, the relationships between the two elements within the system are formed in an interactive energy/power/work related interaction.

There are multiple ways in which energy is transferred into a system or control volume to perform work: thermal, electrical, mechanical (rotational, translational), mass transfer [69], [70]. Thus, the transfer of any of these energy types across the boundary of a system can be handled by means of functional induction. Power variables can define the management of energy in any of these energy forms and must be characterized by the type of power and the attributes of that power as needed for the sizing models. For example, a specific element in the system may require a source of 110V AC electricity at some current rating. This device may, however, operate at different currents or voltages with some lesser degree of efficiency. Thus, the power quality begins to affect the total amount of electrical power that is required and the performance of the specific element within the architecture. The element requiring the power and the element providing the power must interact in their relationship, both providing information critical to sizing both components. These channels of information (power requirements and characteristics) must be defined for each power coupling.

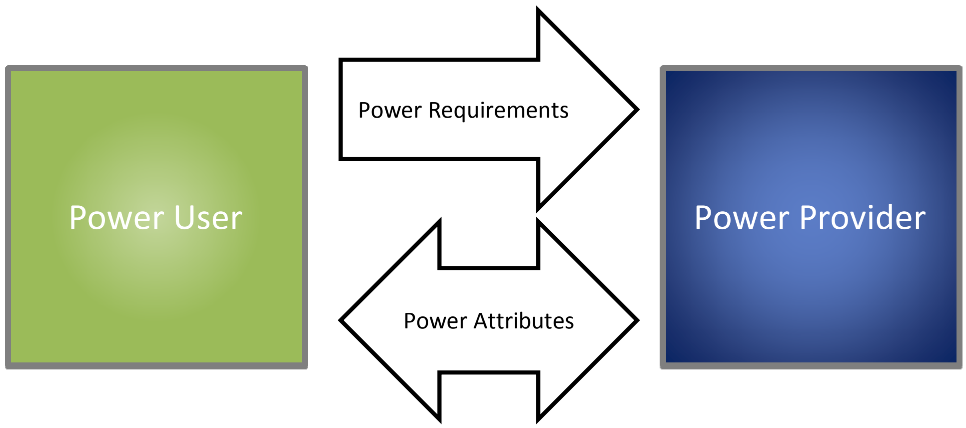


Figure 38: Power Relationship between User and Provider

The functions defining the use of power variables can be generalized into a few separate categories. Devices in the product can be used to transform, store, or distribute energy. All however are defined by the same function/attribute structure and are described in Figure 36. The categorization distinguishes these power elements by the relationships between power input and output and their sensitivity to system level attributes. This will be described in the following 3 sections.

<u>Transformation Functions</u>

These power management devices fulfill the function to change the power from one type to another or to change the power characteristics of a given power type. Each transformation device fulfills one power function (provide one type of power) and induces another (provide another type of power). A very straight forward example of a transformation device is a generator. The generator provides electrical power by utilizing a torque producing energy source.

All power devices can be considered to be some sort of transformation devices. However, the different power functions are defined in separate categories to manage the flow of energy from source to ultimate user.

Figure 39: Transformation Element

Storage Devices

Storage devices are elements which change their nature depending on operating scenario. During some portions of the aircraft operations the batteries are charged through flow of energy from the electrical distribution system. At other points in operation, these elements become the source of electrical power to the distribution system. Elements which induce the function to provide some power type during some portion of the mission and then fulfill the same function at other points can be categorized as power storage devices.

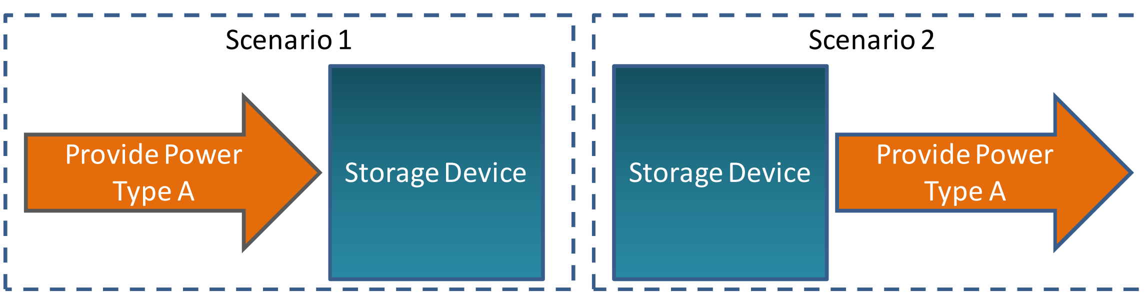


Figure 40: Storage Element

Distribution Devices

A distribution device fulfills the function to distribute energy from one location in space to another. All energy boundary, transformation, storage, and generation devices must receive their energy by means of a distribution element. Functionally, the distribution device induces the same function that it fulfills. However, it provides the connection between a power user and a power supplier.

One unique attribute of the distribution element is its sensitivity to higher level architecture attributes. Determining the attributes and performance of the distribution device requires knowledge regarding the physical location of the elements that this element connects and regarding the routing between those elements (distances, volumes, number of turns, etc.).

Figure 41: Distribution Element

Distinguishing the differences between common power-related functions allows for standardization of the means by which information is relayed between device models. Energy relationships are critical to the conceptual design of a product. The effectiveness of fulfilling functions for a product depends on the use and flow of energy throughout the system.

Secondary Functions

Not all actions required by a specific element within an architecture can be directly related to the power chain. New requirements can be attributed the selection of a specific element and defined similarly to a boundary function. These secondary induced functions are active only when this parent element is present in the system.

An example of a scenario in which physical means induces different functions is a comparison between a manned and unmanned air vehicle. All life support functions can be considered to be induced by the choice of a physical system (a pilot) fulfilling control related requirements. Having a pilot induces the functions to handle food or body waste, to provide external view (windows), provide lighting, and to support life (provide oxygen, pressurization, temperature management, cockpit area) [71]. If providing control was fulfilled by remote control or advanced autopilot system instead of a pilot a completely different set of functions are induced (e.g. real time data and video streaming to ground).

The differences between architectures of manned and unmanned aircraft are due to changes to the physical fulfillment of aircraft boundary functions, and the induction of different functions based on these decisions. Defining relationships in terms of induced functions allows designers to explore areas of the design space which were previously limited by assumptions regarding the physical nature of the product.

Functions are often induced by the combination of elements within an architecture and not directly induced by a single element. These higher-order induced functions become active in the design space when specific combinations of elements are selected. Logic characterizing the activation of the functions is so stated, “If all of these physical elements are present and none of these physical elements are present, then these new functions are induced on the system.”

Installation Induced Functions

Some induced functions cannot be inferred by the appearance of specific elements in the architecture. These functions can only be discovered by taking a system level view of the architecture. The spatial orientation of the aircraft and the placement of each element in a given location can impact the existence of new functions in the architecture design space. These functions are called installation induced functions.

Examples of potential functions which are affected by installation decisions are heat management, corrosion resistance, noise management, vibration control, and hazard protection. These functions are not always needed unless there is an adverse physical relationship between elements within the architecture.

Thermal management is often necessary only when heat sensitive equipment is located near equipment which emits heat. In such a situation the functions to insulate or to remove heat is required. The fact that both of these elements are present in the architecture does not induce the function; the fact that these elements are configured in a particular manner induces the function.

In general, installation induced functions can be categorized by the effect of an output of one element on the input of another. A power variable can be defined to represent these relationships within an installation scheme. For example, if one element emits radiation, and another element is sensitive to radiation, a variable must be defined which quantifies the amount of radiation produced. When radiation emitting elements are located within the aircraft the amount of radiation emitted becomes linked to the general area in which the element is located. A given zone's attributes are dependant on the outputs of the elements within the zone and relationships between neighboring zones. The sensitivity of other elements to these attributes causes new function to become activated. As elements are located in the architecture, induced functions (e.g. protect from radiation) can be initiated within the area in which an adverse relationships exists. These installation induced functions can be instantiated by multiple zones concurrently within the architecture.

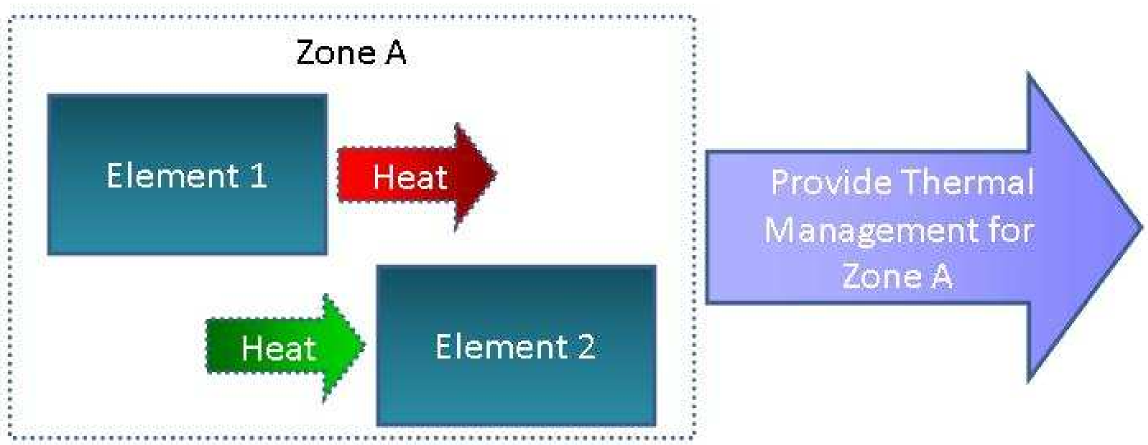


Figure 42: Installation Induced Function for thermal Management

Other induced functions can be induced by tagging zones with specific constraints (disk burst zones, fire suppression zones, etc.). Elements with given outputs can be prohibited or exclusively required in a given architecture location. Thus, as individual elements are situated in 3-dimensional space, side effects emerge and must be dealt with within the common context for functions.

Framework Definition

Power Variables

The threads which knit these function-based elements together are power and attribute variables. Each of these variables must be defined in order to characterize each element and each relationship appropriately. Power variables become the framework upon which all power functions are managed. The definition of power types and induced functions must be initially decided and detailed to provide appropriate flexibly within the design space. Designers must decide if some relationships are going to be handled through secondary functions or through power functions. For example, will thermal power be handled in terms of heat distribution and transformation or will it be handled as a secondary function with thermal attributes?

Power couplings are defined by both power requirements and attributes as shown in Figure 38. The power requirement is a simple statement of how that power is to be transferred. The characteristics are the attributes and qualities that may affect the attributes and performance of the either the power provider or the final power user. The conceptualization of this relationship is similar to relationships as defined by system dynamics. System dynamics defined relationships in terms of both an across and a through variable which are then used to define differential equations for the system. In the case for fluid flow, system dynamics uses pressure drop as the across variable and volume flow rate as the through variable.

The power variable definition for this architecting process considers the through variable (e.g. fluid flow rate) as the power requirement, while the across variable and all other attributes (pressure, temperature, Reynolds number, purity/contamination measurement, etc.) are defined as attribute variables. These characteristic can be constrained or specified by the power user but are defined by the distribution system itself based on upstream power conditions.

Attribute Variables

Some variables are not directly related to the power chain but are necessary for the overall sizing of the system. Attributes like cost, weight, size, reliability, and others are not only necessary to perform sizing and synthesis, but also act as metrics for comparison between architectural concepts. These variables must be generated by the models representing the individual systems and carried to the system level to size the system. These and other attribute variables are also necessary to perform analysis and zone attribute calculations.

Element Installation Definition

As these system elements are defined the next natural question is, “How will all of these pieces fit in the airplane?” or in other words, “What will this grouping of systems look like?” Will the fuel tanks be in the wing and belly faring, or will they hang from the wing in external tanks? Will the avionics bay be situated under the first class cabin? Are the engines on the wings or on the tail? Where will the landing gear be placed? Every element fulfilling the functions of the system must be situated within the system.

Volumetric considerations become paramount in order for sizing to occur. It is not enough to designate which technologies will be used, but these technologies must fit within the mold lines of the aircraft in an efficient way. Many existing conceptual design techniques consider the aircraft as a point mass with assumed aerodynamic and structural performance. In other approaches the layout of the architecture can be set to follow historical manufacturer conventions. These assumptions can be applied in this process of function based architecture design. However, true architectural trade-offs require physical information regarding the interactions between the systems and these interactions depend on the placement of these systems within 3-dimensional space.

Following the definition and integration of systems, elements must be situated relative to each other. The overall aircraft characteristics and new induced functions must be defined by where the elements comprising the architecture are located. The approach adopted for this design process was to discretize the conceptual layout of the aircraft into zones. These zones can represent different sections of the aircraft (fore fuselage, aft fuselage, belly faring, tail, wing, leading edge, nacelle, etc.) to the level of abstraction desired. Each zone is characterized by its relationships with the elements dedicated to it and by its relationships with the other zones in the architecture.

Not only does this allow for installation induced functions to be formulated as discussed previously, but it provides a means to capture the system level interrelationships between the zones that is necessary to size the distribution networks and perform various analyses (stability analysis, aerodynamics). These zones and their combined attributes are defined and interrelated in order to systematically arrange the hodgepodge of interconnected systems and prepare the system for sizing and synthesis. The inter-zone attributes are calculated based on geometry generated by the sizing process (wing area, span, sweep, etc).

Figure 43 is a graphical representation of a notional more electric aircraft architecture, including the control functions, air conditioning, avionics, galley functions, ice protection, and in-flight entertainment. The boundary functions are colored in orange, the induced functions are colored green, and the power sources are marked in blue. The connections between the elements are also color coded: blue as mechanical, dark pink as high voltage AC, pink as low voltage AC, orange as high voltage DC, and tan as low voltage power connection. Figure 44 shows the grouping of the architecture from Figure 43 into zones.

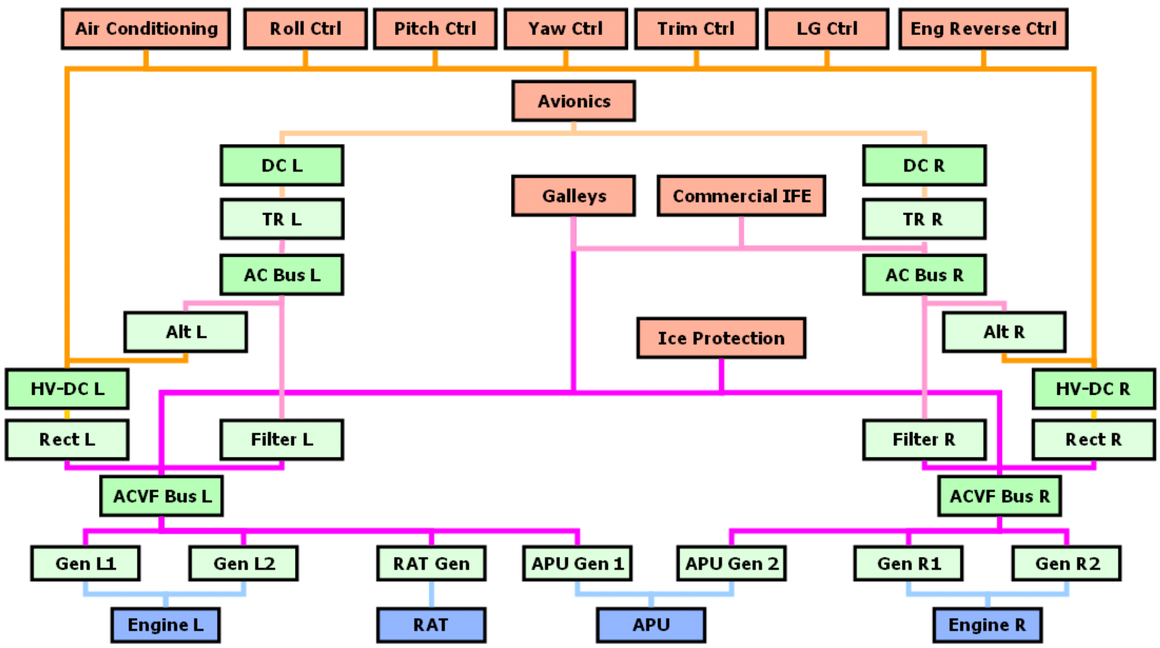


Figure 43: Notional More Electric Aircraft Architecture

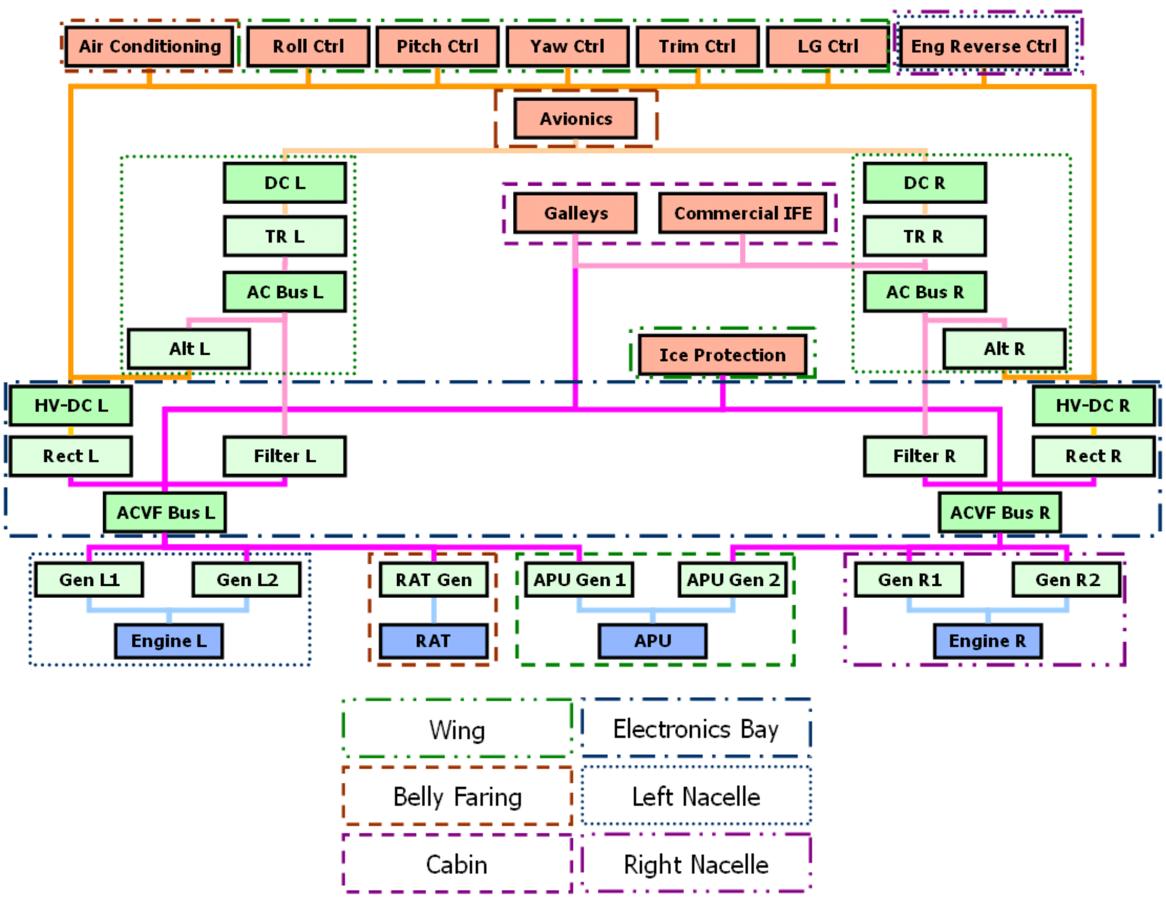


Figure 44: Elements Grouped into Zones

The spatial organization not only determines the overall "look" and performance of the aircraft, but also the immediate environment of each element within the system. Zone placement defines groupings of elements that share the same immediate environment. This allows for appropriate sizing of architectural elements. Element performance and attributes depend on the power requirements it is fulfilling and environment in which it is operating. For example, that attributes and performance of the variable frequency AC bus in Figure 44 are determined by its relationships with the rectifier, filer, and generator, and all physical relationships with the other elements in the electronics bay. Also, the size of the wing determines the power requirements and attributes of the ice protection system. The system level attributes and element zone location must be known to provide definition of zone environment conditions. Thereby, the zone attributes can be calculated and referenced by all the elements within the zone.

The attributes of each zone contributes, in return, contributed to the overall attributes of the architecture. By dividing the architecture into zones, the elements within the architecture not only receive information about the environment in which they operate, but the attributes of the system can be combined to allow the entire architecture to be sized as a whole. By placing the air conditioning and RAT in the belly faring, the system level attributes of the belly faring (volume, weight, impact on aircraft drag) are described. Changing the location of the elements can change the zone attributes, activate or eliminate installation induced functions, and change the performance of the overall architecture.

Summary

The architecture design space includes a complete listing of all possible elements which perform the actions defined by both boundary and induced functions. Every element must be defined with the intention of plugging this element into a grouping of other elements, exploring its effect on this system as a whole, and then replacing it with another element in order to determine which combinations of elements are the most appropriate for the architecture. The design space must alter and change with the introduction of new elements and induced functions.

To facilitate this process, the boundaries of each element must be characterized and sizing models for each element must be defined which are able to calculate all boundary relationships. This includes an understanding of which power types are required and provided and which attributes will be passed to the system (weight, geometry, etc.). Understanding each element also includes a study about which induced functions may be required by this element or by this element in combination with other elements.

Methods and guidelines have been presented in this section which allow the designer to define the architecture design space. Requirements map to functions, sizing scenarios, constraints, and mission definition. This mapping then leads to the exploration of induced functions and the brainstorming of alternatives which fulfill those functions. All of this must be integrated into a complete product requiring a consideration of installation considerations. This is achieved through architecture zone definition.

Once this design space has been defined, it must be explored. Tools are identified in the next section which allow for the exploration of architectures in terms of element selection, integration, and structure.

CHAPTER 7
CONCEPT DEFINITION

Physical descriptions must be made on multiple levels. The definition of the physical architecture can be reflected in three key activities; alternative selection, power relationships definition, and spatial layout configuration. Means must also be provided to address how the architecture is to be sized based on its mission and use-case scenarios.

Once the product concept space has been defined and has been decomposed into its basic elements (i.e. alternatives and functional relationships) these elements must be organized in a manner which allows candidate architectures to be easily defined. This means selecting the technologies dedicated to fulfill the boundary functions, and imposing the appropriate induced functions associated with the selection and networking of these technologies.

Deciding the manner by which the Function/Solution Chains will be fulfilled is a primary task in architecture design. The work represented by this thesis focuses primarily on the first two portions of this concept definition process: alternative selection, and alternative relationship definition. The tools developed to implement this process include interfaces in which installation and mission definition tools can be applied and the required variables can interact with the functional definition. However, tools performing sizing and analysis within this framework are areas for future research and development.

Alternative Selection

Alternative selection is the process of identifying which technologies will be involved in fulfilling the functions of the product. This exploration of the solution space was addressed through means of morphological analysis. Morphological analysis is a methodology by which the parameter space of a problem can be examined and the fulfillment of functions can be investigated. In the late 1940s, astronomer Fritz Zwicky developed a tool which was intended as a tool which organized and explored problems which are multi-element and unstructured [72]. This tool is called the morphological matrix. The morphological matrix is a mapping of a requirement to the means by which it can be fulfilled.

This morphological matrix has a very simple form. The functions that must be performed are listed on the vertical axis of a table and next to each function are all of the possible means of fulfilling these functions listed within a row. This matrix displays and organizes the library of concepts or alternatives that can be combined to fulfill the system functions. Thus, the morphological matrix is a tool and guide which is used to investigate potential combinations of technologies used within a system. A notional morphological matrix for a conventional commercial aircraft is displayed in Figure 45.

Vehicle Layout	Conventional	controll canard	Bi plane	Lifting Canard	Tandem wing	Three Surface	Aft Strake	Tailless	Flying Wing	Droop Wing outer Panels	Winglets	assymetric
Range (nm)	4000	5000	6000	7000	8000	9000	10000	11000				
Number of Engines	2	3	4	5								
Wing Config	C-Wing	Rear Sweep	Forward Sweep									
Dihedral	negative	zero	positive									
Tail configs	Conventional	T tail	Crucifor m	H tail	Triple Tail	V tail	Inverted Tail	Y tail	Twin Tail	Boom Mounted Tail	Boom Mounted Inveted V	Ring Tail
Fuel Type	Biodesiel	J-88	JP4	AvGas								
Number of Aisles	one	two	three									
Materials	aluminum	composites	titantium	steel	mixed							
Controls System	fly-by wire	hyrdraulics	fly-by light									
Trailing Edge	single-slotted	double-slotted	tripple-slotted	blown								
Leading Edge	Simple Krueger	Folding Bull-nose Krueger	Multi-pos. slat									
Auxilary Power	APU turbine	Fuel Cell	Batteries									
Engine Type	Turbofan	Turbojet	Ram Jet	JATO								
Wing tips	rounded	sharp	cutoff	Hoerner	Drooped	upswept	Aft Swept	Cutoff forward swept	Endplate	Winglet		
Fuselage Cross Section	Elliptical	Bell-shaped	Round	stacked	side-by-side							
# of crew	2	3	4	5	6							
# of pax	250-290	290-310	310-330	330-350								
Loc. of Crew Rest Area	behind cockpit	end of pax compartment	cargo hold	Overhead								
Wing Position	Hi	low	mid									

Figure 45: Morphological Matrix

Each element can be selected in combination with the other elements in the matrix. In this example the column represents both functions and configurations that are required from the product. A decision must be made regarding how each of these column elements is to be embodied. For example, there are 12 alternatives for the definition of tail configuration. As indicated by the yellow highlighting, the conventional tail has been selected.

The total number of possible combinations of alternative elements which can be sued to fulfill all of the functions and define all of the systems defined on the column can be calculated using the following equation.

Equation 1

$$N_A = \prod_{i=1}^{n} N_i$$

N_A = number of architecture alternatives

N_i = number of alternatives for function i

n = number of functions

For the example matrix in Figure 45 a total of 1.61×10^9 alternative configurations can be used in this solution space.

Interactive Reconfigurable Matrix of Alternatives

Tools have been developed which utilize the matrix of alternatives to capture the interdependent design options for very large systems. The Interactive Reconfigurable Matrix of Alternatives (IRMA) was developed as a means to "integrate objective and tacit information into the concept selection process" [73]. With the IRMA, engineers can use tacit knowledge to identify situations in which the selections of alternatives are interrelated and can filter the alternative selections available. The IRMA allows the designer to qualitatively explore design options, limit the alternative design space, and understand the dimensionality of the design decisions that must be made.

Consider the function to provide actuation for control on an aircraft. This function can be fulfilled by multiple different physical alternatives. Traditional large commercial aircraft actuators are typically hydraulically driven. Smaller aircraft and redundant systems often use mechanically driven actuation which rely on cables and pulleys to provide the force necessary for actuation. With the push towards implementing electrical technologies in the aircraft architecture, electrical actuation alternatives have become feasible alternatives to fulfill the control functions. These electrical systems include electro-mechanical actuation systems and electro-hydraulic systems, both utilizing electricity to move the control surfaces [74]. Entries in the morphological matrix regarding flight control functionalities and physical alternatives may look like Figure 46.

Functions	Alternatives				
Actuate Flight Control Surfaces	Hydraulic Actuators	Electro-Mechanical Actuators	Electro-Hydrostatic Actuators	Mechanical Actuators	
Provide Power for Flight Control	3-4k psi hydraulic distribution	5k psi hydraulic distribution	Push-Pull Control Rods	Cable and Pulley	DC Electrical Distribution
	AC Electrical Distribution				

Figure 46: Morph Matrix Entry for Actuate Flight Control Surfaces

Following IRMA methodology, a decision regarding the actuators used to fulfill this function narrows the design space regarding which physical element can be used to provide the power for flight control. For example, hydraulic actuators can be considered compatible with hydraulic power distribution systems only. Therefore, with the selection of a hydraulic actuator in the morphological matrix, the design space is limited. This is displayed in Figure 47 where green cells indicate selected alternatives and pink cells represent alternatives which are incompatible with the selected alternative. The functions to provide power for flight control must be fulfilled by a hydraulic distribution system. In this matrix, for simplicity's sake, it is assumed that only one alternative can be selected for each function.

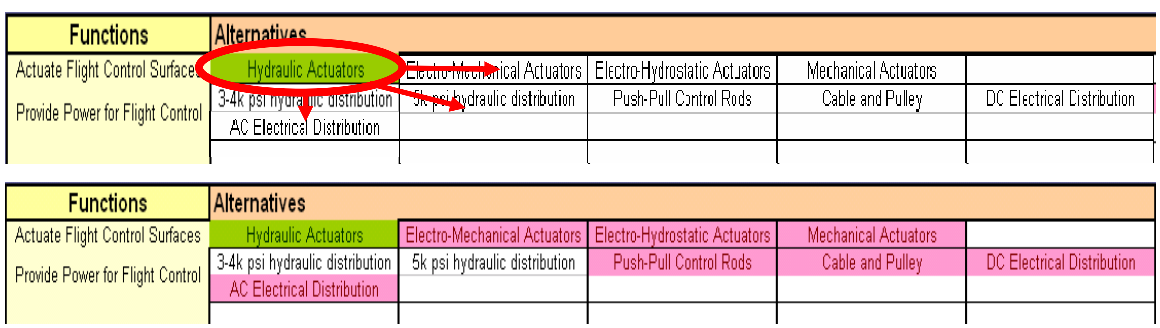


Figure 47: Design Space Limited by Compatibilities (IRMA)

Relationships between elements in the morphological matrix are determined by compatibility scenarios as defined by the design engineer and stored in the form of compatibility matrices. As specific elements are selected, other elements can be either eliminated based on incompatible relationships or automatically selected based on required architecture relationships. Figure 48 is a notional compatibility matrix given by

Engler, Biltgen, and Mavris [73]. The red cells indicate where an incompatible relationship exists. During the selection process, if an element is selected in the IRMA, all selections with incompatible relationships as indicated by this matrix are marked as invalid. For example, in this example matrix, the selection of a nose inlet position, the engine type can no longer take the form of a ramjet or rocket. This is indicated by the red 1 relationship in the matrix.

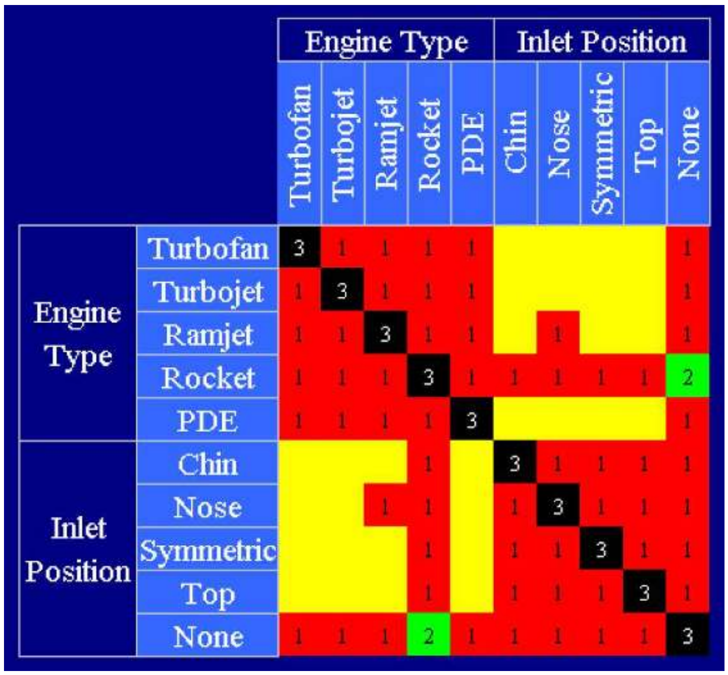


Figure 48: Compatibility Matrix for the IRMA [73]

Using the IRMA reduces the number of alternatives that are available to the user. The alternatives with incompatible combinations are removed from the design space. This is performed mathematically by augmenting Equation 1 as seen in the Equation 2.

Equation 2

$$N_c = \prod_{i=1}^{n_f} A_i - \sum_{j=1}^{n_{inc}} \prod_{i=1}^{n_f} C_{ij} + \sum_{j=1}^{n_{inc}-1} \sum_{k=j+1}^{n_{inc}} \left\{ \begin{array}{ll} \prod_{i=1}^{n_f} I_{ijk} & : F_{1j} = F_{1k}, F_{2j} \neq F_{2k}, A_{1j} = A_{1k} \\ \prod_{i=1}^{n_f} I_{ijk} & : F_{1j} \neq F_{1k}, F_{2j} = F_{2k}, A_{2j} = A_{2k} \\ 0 & : \text{otherwise} \end{array} \right\}$$

N_c = Total number of concepts available

n_f = Number of functions

n_{inc} = Number of incompatibilities

A_i = Number of alternatives for function i

$A = A_1, A_2, A_3, \cdots, A_n$

C_{ij} = Number of alternatives for function i during incompatible scenario j

$\mathrm{C_j} = C_{1j}, C_{2j}, C_{3j}, \cdots, C_{nj}$

I_{jk} = Intersection between incompatibility scenarios j and k

I_{ijk} = Number of alternatives for function i during incompatibility intersection $\mathrm{I_{jk}}$

F_{1j}, F_{2k} = Functions related by intersection $\mathrm{I_{jk}}$

A_{1j}, A_{2k} = Alternatives related by intersection $\mathrm{I_{jk}}$

Without taking compatibilities into account for the compatibility matrix in Figure 48, the number of possible combinations is 25. There are five options for engine type and five options for inlet position. With the compatibility conditions from Figure 48 the number of possible alternatives is reduced to 16 due to adverse relationships between rocket type and inlet position. A tool developed to investigate the effects of compatibility scenarios on the design is detailed in Appendix E.

Adaptive Reconfigurable Matrix of Alternatives (ARM)

The basis behind the IRMA is a constricting design space. Incompatible relationships between physical elements limit the number of combinations that can be selected. However, it is typically used with a well-defined set of functions having assumed static relationships between functional requirements. This tool is subject to the same limitations as the functional analysis and allocation portions of the design process, where assumptions as to product structure can limit the design flexibility. Compatibility constraints between physical elements depend on assumed physical relationships required between these elements. In order to avoid these assumptions, a new method of alternative selection is proposed with tools based on this adapting functional breakdown. Relationships between alternatives and functions in this new tool are subject to the inductions of new functions based on physical requirements, not just elimination of elements do to assumed incompatibilities.

Configuring the means of providing actuation can be addressed using induced functions instead of incompatibilities. With this approach the design space grows with selection of physical elements based on induced functional requirements. Considering the same example as in Figure 47, the process begins with the same boundary function ("Actuate Flight Control Surface"). However, in contrast to the method of incompatibility elimination, the supplementary functional requirements depend on the selection of flight control device as displayed in Figure 49.

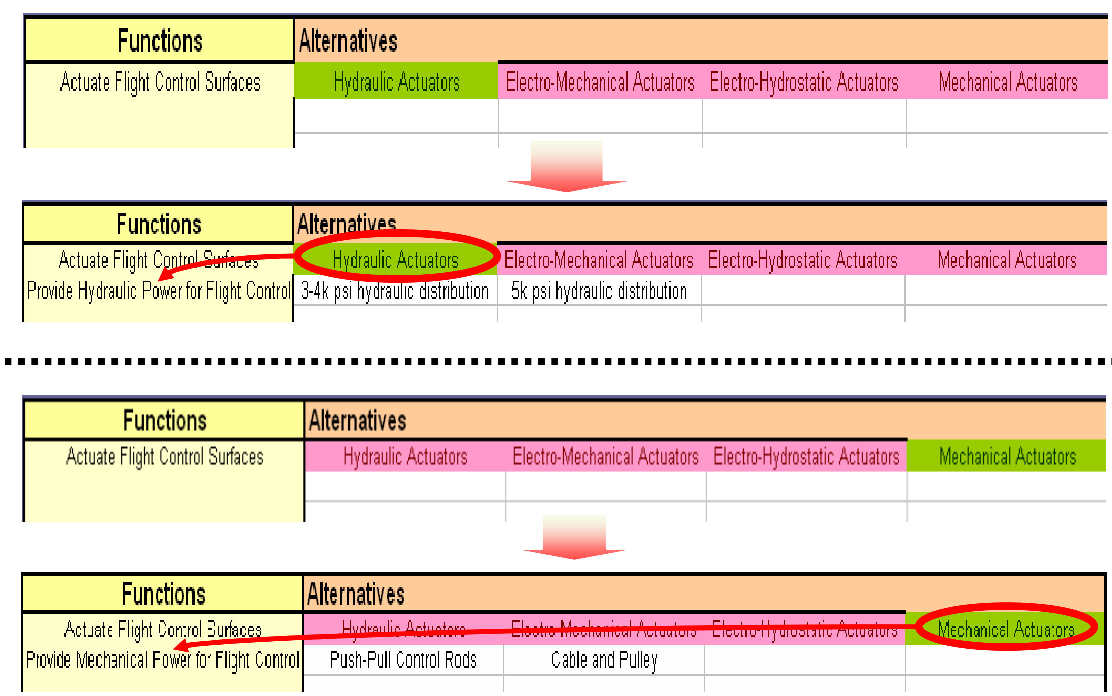


Figure 49: Design Space Expanded by Induced Function (ARM)

Another benefit of basing the morphological matrix on induced functions is an increased ability to allow for redundancies and multiple instantiations of elements within the morphological matrix. Incompatibilities cannot be imposed when each function can be fulfilled by multiple means. The hydraulic actuators are not incompatible with cables and pulleys, but cables and pulleys simply cannot fulfill the functional requirements of hydraulic actuators. A possible redundant mechanical or electrical actuation system would negate the incompatibility typically used in the IRMA.

The Adaptive Reconfigurable Matrix of alternatives (ARM) was designed to manage the functional and physical breakdown of the architecture through acting as a hybrid of the IRMA and the function/means tree. This tool relates both alternative interactions in the form of compatibilities and utilizes the induction of functions in a hierarchical manner [54]. The ARM provides an adaptive framework upon which the architecture can be defined. All boundary functions are listed in as the initial design space. As alternatives are selected to fulfill these boundary functions, induced functions appear in the matrix. After this manner, functional structure of the architecture adapts following the logic developed during design space definition. The ARM tool can also be configured to respond to compatibility constraints which can limit the design space. Using the ARM in concept definition provides a flexibility to explore options that are difficult to represent with the static functional breakdown used in the IRMA.

Power Relationship Definition

Once the alternatives have been selected the relationships between them must be established. Some induced functions have the capability to fulfill functional requirements for multiple boundary functions. This is often the case when considering the structure of the transformation and distribution of power. Many elements within the architecture can place requirements on different power distribution elements. These relationships dramatically change the structure of the functional chains and the requirements of the enabling technologies. Many of these relationships can exist and must be specified for a given architecture.

Design Structure Matrix

Defining and redefining relationships of elements within a system can be accomplished using the design structure matrix (DSM). The DSM (dependency structure matrix [75]) is a square matrix with identically named columns and rows. Each cell within the matrix represents a potential relationship between system elements. If a relationship exists between the elements this is indicated in the corresponding matrix cell.

The DSM can be applied to many forms of problems. It can be used for managerial purposes by representing relationships between people or teams. Analysis routines can also utilize the DSM by representing the data flow between modules in a simulations code [76].

The relationships can be directional when the matrix represents a system of sequential tasks, or it can be bi-direction when the matrix represents information regarding data, materials, and/or energy transfer relationships [75]. Depending on the direction of data flow in the matrix, one half of the matrix along the diagonal is feed forward and the other half is feed back. The DSM in Figure 50 is defined by seven elements labeled A through G. The Xs represent locations where interactions occur between these elements. In this matrix the horizontal elements require some input or relationship with the vertical elements (as designated by the Xs). Therefore, all relationships below the diagonal are feed forward relationship, while the elements above the diagonal are feed back relationships.

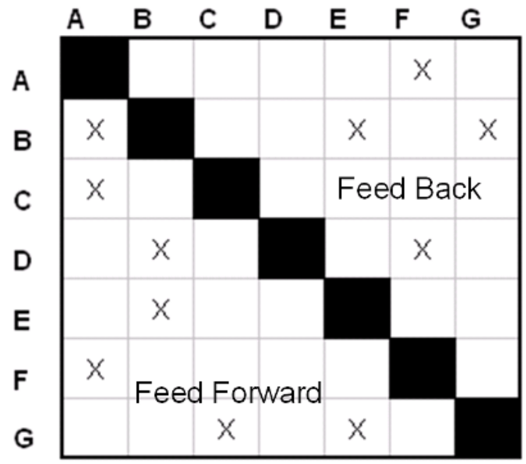


Figure 50: Design Structure Matrix

In traditional design, the DSM provides means to define systems' relationships and to either organize the elements or actions within the system to reduce the number of feed-backs (sequencing) or "cluster" all of the relationships close to the diagonal in order to characterize elements that should be grouped in tightly related systems [49]. A DSM application most relevant to this project is component based, or architecture DSM [49]. This method uses the matrix to explore various architectures configurations by interrelating physical elements within a system. With this tool each element represents a physical component and the relationships within the matrix designate how these components relate to each other.

Functional Mapping Matrix

Using an architecture based on a functional structure that changes with physical decisions, there must be a tight relationship between the means of alternative selection (ARM) and the tool defining the interrelationships. For a large system which includes multiple physical elements, the resulting matrix can be very large. However, not all elements within the system can have direct, logical relationships. In order to guide the architect in defining the relationships between elements, the DSM was adapted to have columns and rows which change with the selection of different physical elements in the design space. With knowledge regarding the boundaries of these elements this adapting DSM also limits the user to define connections where logical relationships can occur. This tool is called the functional mapping matrix (FMM).

When an element is selected in the ARM, information regarding its power inputs and outputs is delivered to the FMM. The FMM is generated by listing all of the power inputs for the technologies currently selected along the rows of a matrix, and all of the power outputs along the columns. This matrix changes every time the ARM's technology suite is redefined.

Consider the notional matrix in Figure 51. In this matrix the functions "condition air", "protect from ice", "provide lift", "interface with pilot", and "provide control" are either boundary functions or secondary/tertiary induced functions.

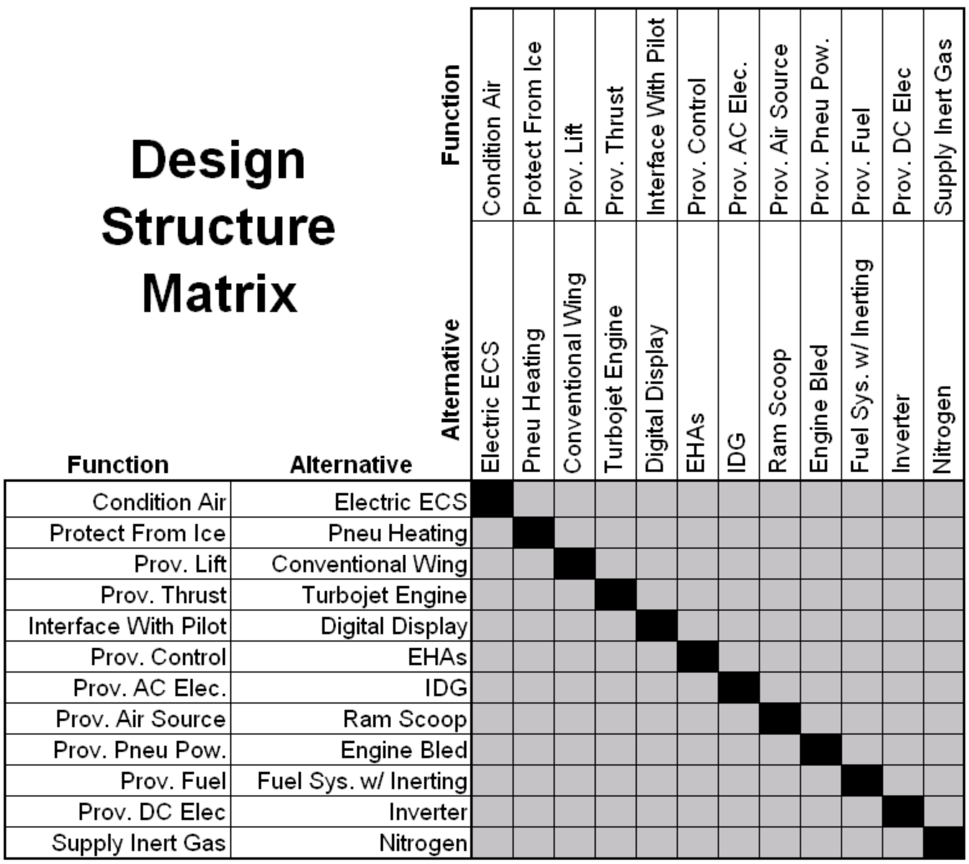


Figure 51: Notional Design Structure Matrix

These functions are all being fulfilled by a specific element. The diagonal of this matrix is marked in black, indicating that a physical element can not be configured to receive its power inputs from itself.

The power structure of each attribute limits the number of relationships that it can have with other elements. By replacing the function/alternative mapping with a function/alternative/power variable mapping the dimensions of the matrix changes. This is the form of the Functional Mapping Matrix. This matrix is displayed in Figure 52.

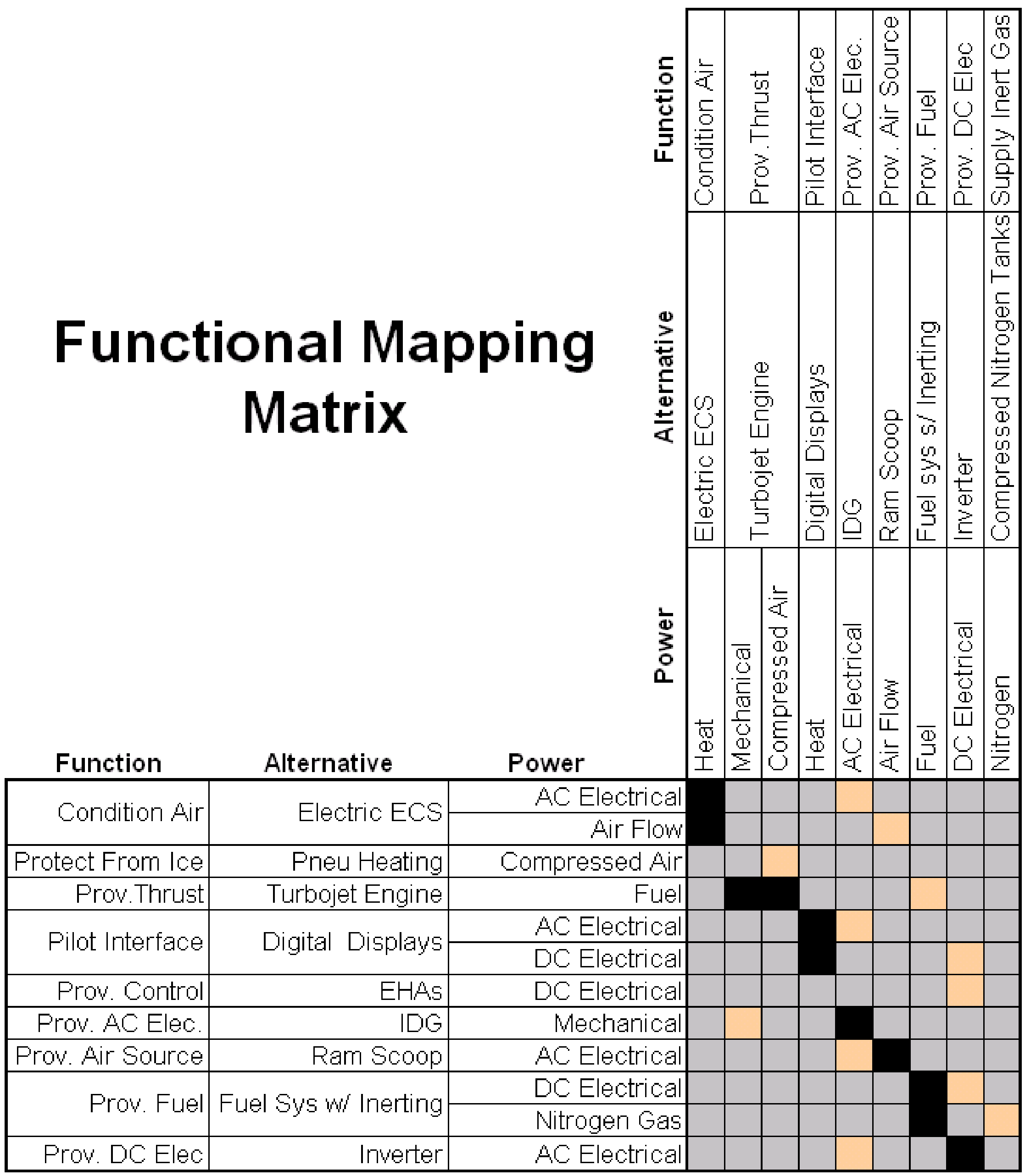


Figure 52: Notional Functional Mapping Matrix

Note that this matrix is not square. Some of the elements within the architecture are also not present. The conventional wing does not appear in this matrix because it has no power relationships with other elements in the current architecture conceptualization. There can also be more than one input or output from a given element. The electrical ECS system requires AC electricity and air flow. Similarly, the turbojet engine provides both mechanical power and compressed air. The FMM also indicates where connections can occur based on power type; the element on the left receiving power from the element on the top. These are indicated by the tan grid color. Interactions can be indicated by placing a token in one of the tan grid locations.

The matrix in Figure 52 does not show multiple sources available for each power type. However, as the number elements increases and redundant functions and systems are defined, alternative configurations are available. This not only allows for new architectures to be defined, but it also allows for the explorations of the failure scenarios. If certain functions cannot receive their power requirements when elements are deactivated, then a redundancy must be defined. If this potential power provider is no longer available during a failure scenario, this token can be moved to another location which provides this for this power requirement. Thereby, the designer indicates where power must be rerouted during different failure scenarios.

Some of these relationships cannot change in different flight scenarios. An integrated drive generator has a mechanical power connection with an engine. In the event that the engine is disabled, the IDG, by its nature, cannot be reconfigured to receive power from another engine. In contrast, an electrical distribution system can be configured to receive power from any number of generators. If one generator fails, this distribution network can receive power from another generator. The electrical network will have different performance attributes to be taken into account in the sizing, but it has the capability to receive energy from multiple sources at different times. Therefore, when a connection is defined it is characterized as a fixed or flexible relationship.

Spatial Orientation/Installation

The third task in configuring candidate architectures is defining of spatial relationships between the alternative elements. As discussed in the previous chapter, induced functions like cooling requirements, hazard zones, and fire protection zones are all related to the installation considerations of the aircraft. Other systems attributes like efficiency, length, and size of distribution systems depend directly on where the elements are situated in relationship to the others. Thus, a means must be provided whereby these considerations can be taken into account by the system architect.

An interface is required in which the designer designates where each elements exists in the system. A notional zone breakdown is displayed in Figure 53. Elements selected to fulfill the architecture functions can be given a numerical value representing where they are physically placed. This image represents the concept for the aircraft and the framework within which it will be sized. Each zone must be individually sized based on the mission requirements and the geometry of the components within them.

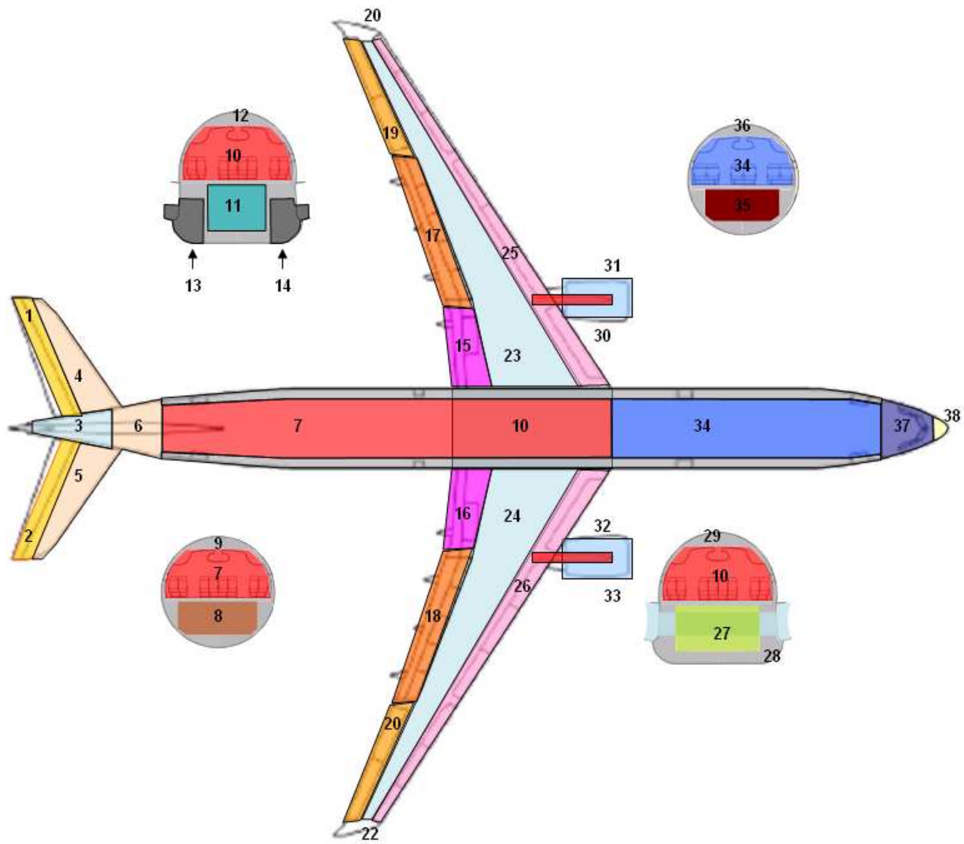


Figure 53: Aircraft Installation Zone Definition

The details of the calculations governing all inter and intra-zone relationships for an aircraft depends on what attributes and requirements are necessary to perform the analysis and size the architecture. The spatial arrangement of the architecture must translate the overall aircraft geometries and attributes into variables used in the modeling of each physical element. The zone attributes, in turn, determine the structural, aerodynamic, and stability attributes of the aircraft.

Mission Definition

One fundamental purpose of this process of architecture definition is to infuse more detail and information into conceptual design by linking the concept (requirements, functions) directly to the physical attributes (elements, relationships). This would allow the conceptual designer to concurrently define the conceptual architecture and the corresponding modeling and simulation environment. The modeling is accomplished by mathematically defining the elements and their relationship, while simulation inherently requires the replication of states in which this model operates and how the requirements change for all operational states. The ADEN environment provides means by which alternative use-case scenarios can be identified and linked to their corresponding requirements mapping. This allows for more detail to be introduced in the mission analysis of the architecture and a configuration of redundancy scenarios.

Tools were developed which allow the definition of the states in which architecture operates, or use-cases. This was designed in an object oriented manner to allow the user to alter the mission conditions or requirements in order to explore improvement to the design. An object oriented mission definition would also allow for flexible reconfiguration of the order of mission tasks or segments.

Three things are necessary for the sizing situations to be defined: the configuration of the architecture, the external conditions, and the attributes of this sizing case relative to the other sizing scenarios.

Mission Segment Architecture Configuration

It can be assumed that the technology set indicated for an architecture will not change mid mission. However, the way in which power is transferred between elements can change during operation. Functions can be fulfilled by receiving power from different sources during different portions of its mission. Therefore, for a given technology set defined in the ARM, multiple FMMs can and should be generated for different sizing, use-case, and failure scenarios. Each sizing scenario must be linked with a specific FMM, thereby defining the configuration of the architecture during that sizing case. This changes the way in which functions are fulfilled during different mission scenarios. In turn, this allows for deeper levels of architectural trade-offs and exploration into product performance.

The overall physical parameters of the product will always be derived from the physical parameters of the elements composing the architecture. The weight will be the sum of the weights; the volume will depend on the volume of the components, and so on. However, depending on the boundary of the system, power variables may be transferred from the environment to the components. If the boundary of architecture concept lets the environment (or assumed architecture configuration external to the design space) handle the functions to provide fuel, for example, the fuel tanks and distribution system will not be modeled as system elements. In this case fuel systems modeling must be done in combination with the sizing and synthesis calculations. Conceptualization of the architecture with power relationships crossing the boundary must occur at the system level.

The boundaries of the system must be defined to provide adequate flexibility in the configuration of the architecture. If power relationships cross the boundaries of the system it may represent a physical assumption which limits the flexibility of the design space.

Mission Segment External Conditions

The means in which an aircraft or any product is used greatly influences its overall design and performance in fulfilling its functions. For example, simply changing the operating altitude and cruise speed required significantly changes the optimal propulsion cycles, sweep angle, and wing size required. Fulfilling the function to carry passengers from one location to another can be accomplished more effectively at one specific operating condition, depending on the technologies applied in the architecture. A turboprop aircraft will have a much different optimal operating point than a turbofan driven aircraft. At the very least, altitude, range, and Mach number for each mission segment must be defined in order to allow the designer to investigate changes to the design depending on variations in external operating conditions and stimuli.

Mission Segment Sequential Calculation

The sequence in which these sizing scenarios are calculated must take be taken into account in order to capture the change in overall aircraft attributes during the mission. Each mission segment requires information from previous mission segment to perform accurate analyses.

An example of these changing attributes throughout a mission is weight. Weight is one of the driving factors in aircraft design because of its impact on the requirements on the thrust and lift requirements of the aircraft. Weight is so important that it is often used as a surrogate measure of effectiveness for aircraft performance. As the aircraft goes through its mission it loses weight due to fuel burn. Weight lapse (β) is defined as the ratio of the aircraft weight after a mission segment to the aircraft take off gross weight as shown in Equation 3.

$$\beta \equiv \frac{W_2}{W_0}$$

Equation 3

As the aircraft proceeds through the mission the weight fraction is continuously changing. The total fuel burn for each β_n is calculated as the product of all the previous weight fractions for each mission segment (Equation 4).

$$\beta_n = \frac{W_n}{W_0} = \frac{W_{n-1}}{W_{n-2}} \frac{W_{n-2}}{W_{n-3}} \cdots \frac{W_1}{W_0}$$

Equation 4

To find each weight fraction one must capture the effects of fulfilling the functions required in this mission segment. Energy based relationships can be developed which relates weight loss, range, performance, aerodynamics, and propulsion. One of the simplest of these calculations is the Breguet Range Equation (Equation 5).

$$Range = \frac{V(L/D)}{gTSFC} \ln\left(\frac{Winitial}{Wfinal}\right)$$

$$\beta = e^{-\left(\frac{Range(g)(TSFC)}{V(L/D)}\right)}$$

Equation 5

In order to allow for the reconfiguration of the mission definition the analysis of each segment can be handled discreetly. Each is defined as a follow on to a previous segment, thereby referencing the appropriate β and other factors. The order in which mission segments are performed change the initial weight fraction for each mission segment. Using an object oriented process for sizing, each segment can be adapted and new segments can be included while still preserving the ability to calculate the total fuel burn, weights, and other attributes which require sequential sizing. This is displayed in Figure 54.

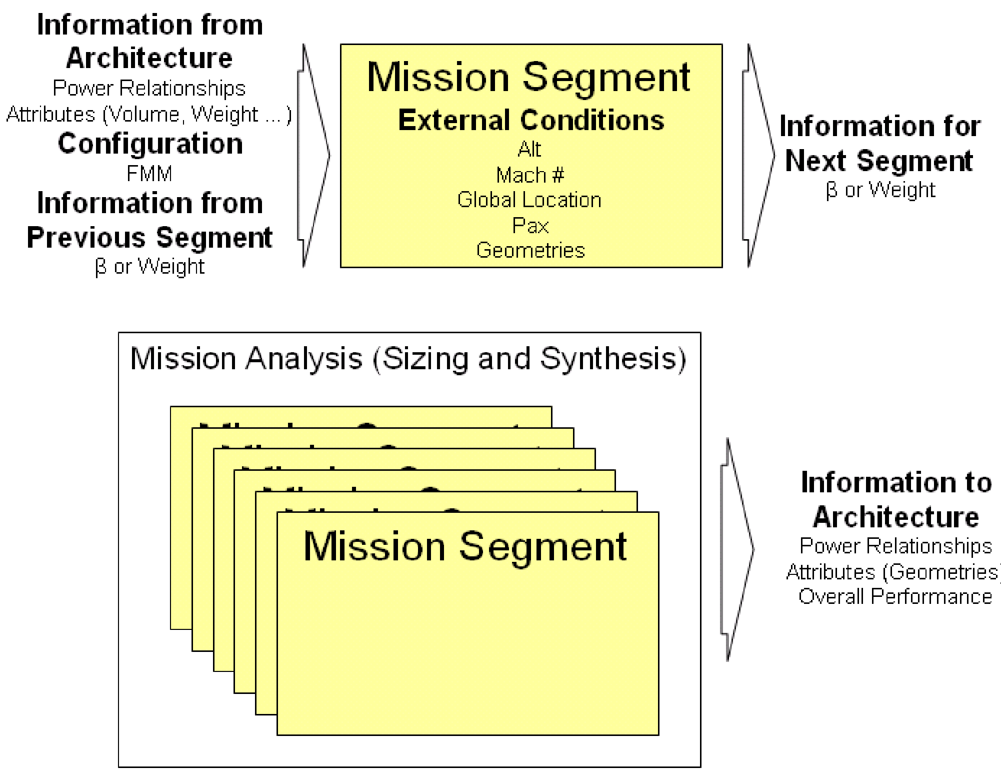


Figure 54: Mission Segment Integration

The sizing of the aircraft includes high levels of interaction between the fulfillment of specific functions, explicit conditions imposed on the architecture, and the physical relationships defined for the system. This depth can allow for intricate interrelationships between component attributes and the entire mission of the aircraft. By integrating the use-case scenario definition with functional mapping, the conceptual designer can truly assess the impact of architecting decisions on the performance of the aircraft.

Summary

This process of architecture design manages the assumptions made in assigning groups of physical elements to a boundary function or a functional chain. The defined physical elements must be limited to the task of individual function specified in the breakdown. The architecture design space is only subject to assumptions in the matching of specific physical technology to a function, in the modeling of architecture elements, and the definition of induced functions.

The purpose of the ADEN Concept Definition tools (ARM, FMM, Installation, Mission definition) is to provide the means of investigating possible combinations and relationships that most effectively utilize the potential technologies to be applied to the architecture. Each tool provides a means of making specific decisions which define the architectural concept and configuring an architecture model. The adapting functional breakdown ensures that side effects are not overlooked which may lessen the benefit of a technology on the product as a whole. The FMM provides a means to configure architecture relationships and investigate alternative relationships during all use-case scenarios. As the physical fulfillments of the architecture functions are chosen, relationships can be configured and changed for each operating condition. The installation definition interface provides means by which each architecture element can be housed and contained somewhere in the architecture. Finally, all interactions between the use-case environment conditions and the architecture can be defined with the ADEN toolset. In this way all three dimensions of architecture definition (elements, relationships, structure) can be configured and changed through interactions with simple and flexible tools without requiring drastic revision and reconfiguration of the models used in sizing the element of the architecture.

CHAPTER 8
AIRCRAFT ARCHITECTURE DESIGN SPACE DEFINITION

The architecture design space is defined by the functions of the aircraft, the alternatives included in the architecture, and the potential relationships involved in with these alternatives. This chapter reviews the functions and alternatives involved in a proof of concept design space for commercial aircraft.

Functions

In order to develop a flexible design space, the definition of a product's functions must occur without biases and constraints developed in previous design projects. Lester Faleiro, project manager for the European Union Technology Project "Power Optimised Aircraft" and assistant chair of the EOAsys Technical Committee in the AIAA, stated that in order to address the development of the "more electric aircraft" a new vision is necessary. He asserts that this revisiting of the integration of the architecture requires approaching the design "through the eyes of a child [77]." In keeping this statement, the functions of the aircraft were designated in a way which did not over-define the actions described by the function, and allows for general fulfillment of these function.

Boundary Functions

In simplifying the statements of what an aircraft must do, the functional breakdown shown in Figure 55 was developed. This tree structure is very helpful in the definition of functions. It allows the designer to be as generic as possible, and refine the functions to the level of detail desired. The difficulty and beauty of the functional breakdown lies in the ability to succinctly and accurately embody the purposes and uses of a product in the most straightforward manner possible.

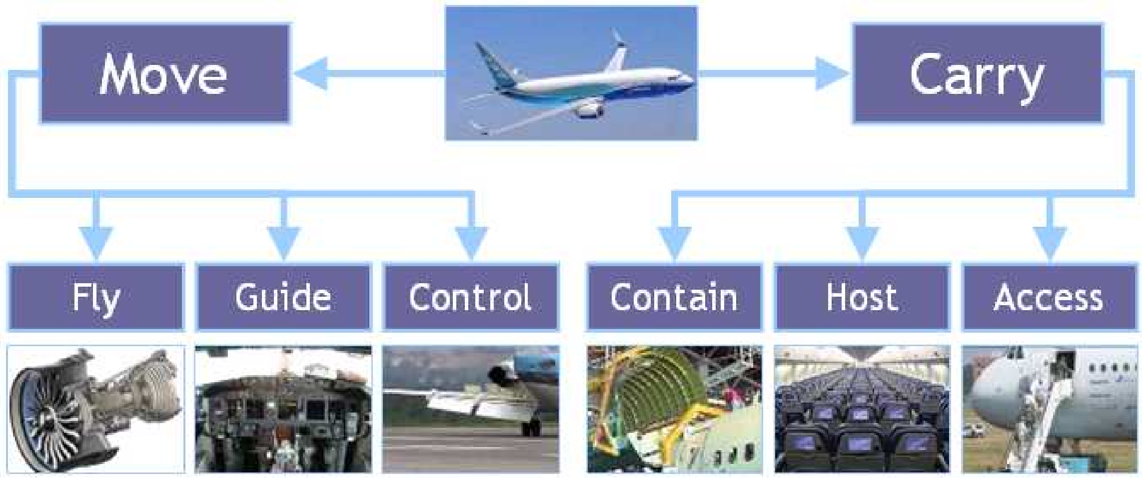


Figure 55: Functional Breakdown of a Commercial Transport Class Aircraft.

The underlying requirement of this type of aircraft is to “transport.” The dictionary definition of the term transport is “to carry, move, or convey from one place to another” [78]. Fundamentally, the functions of an aircraft are to move from one location to another and to carry objects (passengers and cargo) on this route.

The two upper level functions defined for a commercial aircraft belong to any transport vehicle. Distinctions are made between an aircraft and other transport vehicle in exploring the subsequent hierarchy of functions. The function “to move” is broken into three lower level functions: to fly, to guide, and to control. First, operation within the air transportation system requires that the aircraft must fly. Second, in order to travel from one place to another the vehicle must have a means to ascertain its position relative to its source, destination, and potential obstacles en route (Guide). Finally, the aircraft must also allow response to guidance information to control its movement (Control). Exploring the function to “carry” yields three sub-functions: to contain, to host or maintain, and to access. All elements must be kept in the vehicle as it moves (contain); they must be kept in a desired state during transport (host); and they must be deposited and retrieved in the transport vehicle (access). Hence, the functional breakdown in Figure 55

All induced functions will be discussed later in this chapter as alternatives and power types are defined. The function “to contain” is partially managed by the installation tools described earlier in which these elements are provided a place in the architecture. The structural requirements to contain the architecture elements may also be managed through zone definition.

By exploring each of these functions and the underlying actions embodied by these functions, an extensive hierarchy of functions can be defined. The lowest levels of this tree are the boundary functions for a transport category aircraft. This tree is displayed in Figure 56.

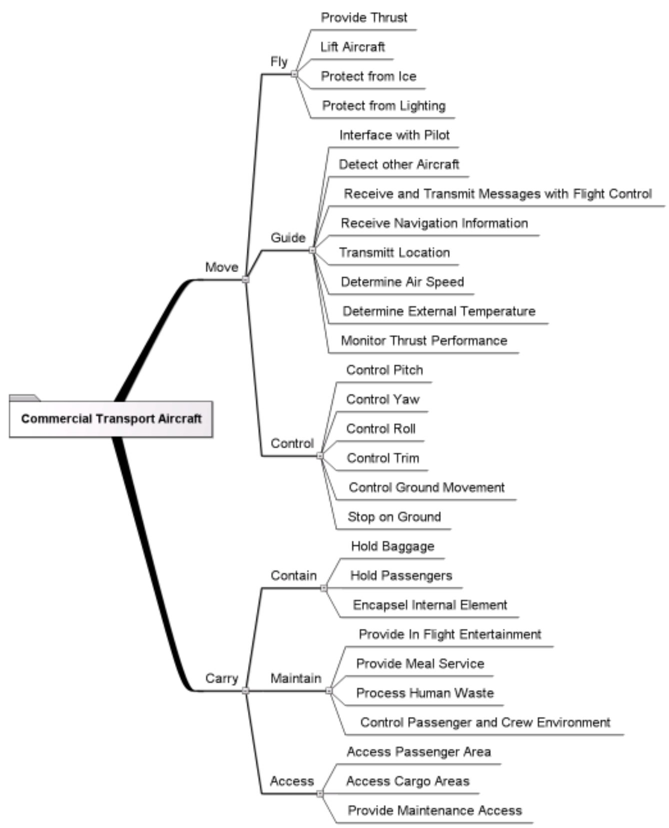


Figure 56: Boundary Functions for a Transport Category Aircraft

For the purposes of proving this process and generating candidate architecture the number of functions has been limited to those typically associated in trades between more electric and conventional aircraft. As seen in Figure 57, the number of functions comprising the function to guide has been reduced from eight to one ("Provide Navigation"), and the airframe and wing related functions are removed from the fly, access, and contain categories.

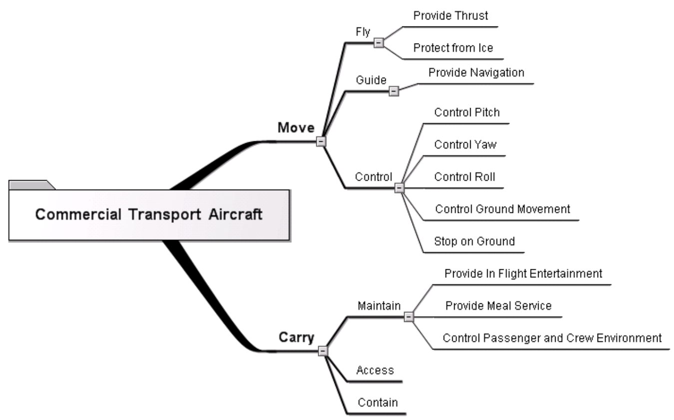


Figure 57: Condensed Boundary Functions for a Transport Category Aircraft

This set of boundary functions will be used to drive the design of the candidate architectures. This reduced set of boundary functions is appropriate for this exercise of comparing conventional and more electric aircraft architectures and will suffice for a proof of concept. However, in consequence to this reduction of functions considered in the architecture, the degrees of freedom associated with this design space and the ability to innovate on revolutionary concepts is reduced.

Induced Functions

As discussed in the previous chapters, induced functions appear in the architecture as a result of power relationships within the system, installation conditions, and directly induced functions. All induced functions stem from the requirements induced by the physical elements which will be used to fulfill the functions described above.

<u>Power Induced Functions</u>

Power Variables

The number of induced power functions depends on the number of power types that will be used to define this system. For every power type, three induced functions are automatically created; to transform, to distribute, and to store this type of power. If these power types are to be used in the architecture, distribution elements, and transformation or generation elements must be defined for this power type. Power generation functions are not automatically defined. These must be generated by the user, depending on which power types are needed.

The power types defined for this architecture are shown in Table 1. This table lists the power variables defined for this architecture and the power characteristic variables that are included in the relationships. For this test case 7 power types were defined.

Table 1: Power Types

Power Type		Power Characteristics	
AC Electric	kVa	Voltage	V
		Current	amps
		Frequency	Hertz
DC Electric	kVa	Voltage	V
		Current	amps
Pneumatic	lbm/s	Temperature	psi
		Pressure	°F
Hydraulic	lbm/s	Pressure	psi
		Temperature	°F
Mechanical	hp	Torque	lb-ft
		RPM	rpm
HVAC Electric	kVa	Voltage	V
		Current	amps
		Frequency	Hertz
HVDC Electric	kVa	Voltage	V
		Current	amps
Thermal	Btu	Source Temp	°F
		Sink Temp	°F
		Overall Heat Transfer Coefficient	Btu/h·ft^2

With these power variables defined, 25 power functions appear in the functional breakdown. These are displayed in Figure 58. This list of power variables could be expanded to include many more in order to either allow more technologies to be considered or facilitate the modeling and simulation of this system. Not all power induced functions must be fulfilled by physical alternatives unless required by other architecture elements. Power variables may also be primarily identified as channels for the definition of installation induced functions and do not require specific alternative fulfillment. For example, thermal relationships do not necessarily mean that there must be elements which distribute, store, or transform thermal energy. This power type may be used to indicate where induced thermal protection functions must occur on a zone-by-zone basis as discussed in chapter 7.

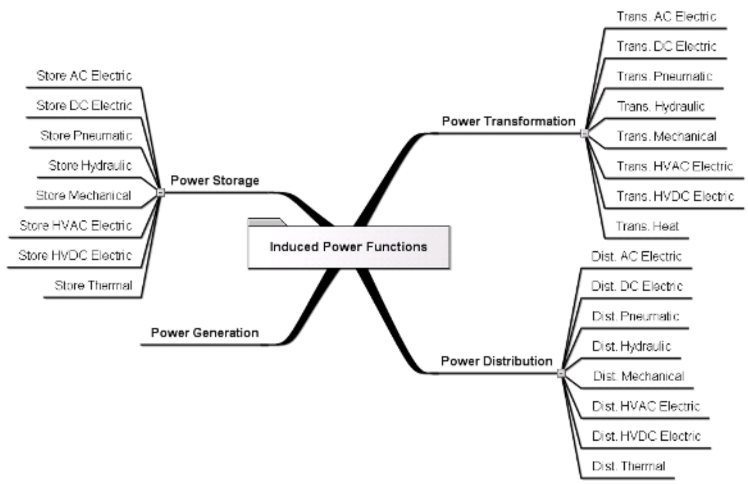


Figure 58: Power Induced Functions

Installation Induced Functions

For this conventional layout, the aircraft was divided into 38 zones to house components. This zoning is displayed in Figure 53. Dividing the aircraft into zones allows functions to be induced if elements with certain inputs and outputs are installed in close proximity to each other. This zone breakdown also provides the management of attributes for aircraft and element sizing.

For this test scenario, only thermal management issues are taken into account as installation induced functions. When an element having a thermal output is placed in a zone a new function appears in the functional breakdown to provide thermal protection for the elements in the zone. These functions must then be fulfilled by physical elements which manages the heat produced by this system. The physical attributes of this thermal management device, or even its presence in the system, is based on the amount of heat produced and the sensitivity of elements within the zone. The physical model for the element intended to insulate or remove the heat from the system may indicate that it has no weight or volume, and requires no power, if at the amount of heat provided by a heat producing element is enough to warrant the presence of the system.

Secondary/Tertiary Induced Functions

Secondary/Tertiary induced functions are directly induced by specific elements when selected in the design space. They allow for the elimination of assumptions regarding how the architecture will be configured. For this test case, it was assumed that the aircraft is manned and carries passengers. It has a conventional layout and is fueled by Jet A. Many of the structural and installation designations have been defaulted and excluded from this specific design study. These assumptions remove much of the necessity of including many secondary/tertiary functions. However, thermal protection induced functions were included for the electrical power distribution elements and the electrical actuation elements. Regardless of proximity to other devices, these elements must be cooled to prevent damage to themselves.

Including secondary/tertiary induced functions is necessary to create a robust design space. However, for brevity sake the number of functions considered was reduced. This, in turn, reduces number of physical elements and technologies that must be placed in the architecture.

Technologies

After all of the functions of this aircraft have been designated, alternative technologies must be brainstormed for the fulfillment of each of the functions. These elements must be characterized by their power relationships, attributes, and all other inputs and outputs. In order to develop architectures that are “more electric” and conventional, a variety of conceptual alternatives must be defined. The conventional aircraft can be classified as a hybrid mix of power usage. Devices on the aircraft utilize electric, hydraulic, mechanical, and pneumatic power to fulfill the functions. The motivation behind the "more electric aircraft" is the utilization of a single power type. Electricity is more assessable and controllable than other power types, and reducing the number of transformations between power types removes unneeded losses in the system. The theory is that by requiring only electricity and distributing this power efficiently (high voltage) the aircraft becomes lighter (less components), more efficient (less transformation and more controllable), and also more dependable (reduce maintenance time). In this section, technologies associated with conventional and more electric technologies will be discussed and included in the architecture design space.

Not all possible technologies have been reviewed in this work. This discussion of alternative technologies represents some of the concepts which are generally associated with the conventional vs. more electric aircraft trades. In full application, this catalogue of technologies could represent a survey of all individual part suppliers' inventory of potential technology solutions. This would require representative characteristic and power models for each of the available technologies.

Actuation Technology

The function to provide roll, pitch, yaw, and ground movement control are fulfilled by some type of actuation device. This device provides a translational force to deflect the flight control surfaces or provide steering. Four types of technologies were included in the design space to provide the translational energy required to actuate the control surfaces. Each of these devices requires a different type of power transmission to be provided. These technologies are the mechanical actuator, the hydraulic actuator, the electro-hydrostatic actuator, and the electro-mechanical actuator. Other actuator concepts exist which utilize one or combinations of these power sources, but were not included in this design space for the sake of brevity.

<u>Mechanical Actuator</u>

Mechanical actuators were commonly used early in aircraft design. Originally these actuation systems consisted of elements which directly translated the movement and force of the cockpit controls to movement of the control surfaces. This was done by means of pulleys, rods, and cables. In larger aircraft where the forces on the control surfaces are large, this simple actuation system is not as feasible. Thus, servo tabs and other devices were developed to reduce the amount of force necessary to move the control surface [79]. The mechanical actuator can be characterized as a physical element with a weight and volume which requires the function to distribute mechanical power.

Hydraulic Actuators

The amount of force required to deflect the control surfaces increases with larger aircraft. Hydraulic power provides the capability of delivering a large load. The hydraulic system is powered by pumps which generate pressure in the hydraulic distribution system. This pressure is translated to a translational force by means of a servo-valve which allows pressure to be applied to a hydraulic piston. This hydraulic piston applies force to the control surfaces. The hydraulic actuator is characterized by a volume and weight, and requires hydraulic power distribution. Its attributes depend on the mission and size of the aircraft, as well as the pressure of the hydraulic fluid provided by the hydraulic distribution system.

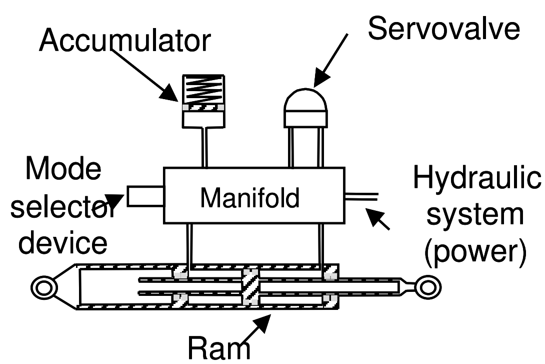


Figure 59: Hydraulic Actuator [80]

Electro-Mechanical Actuators

As seen in Figure 60, the electro-mechanical actuator (EMA) is made up of a mechanical gear and screw system which converts rotational torque from an electric motor into a translational force to actuate the control surface. These actuators can receive either AC or DC electrical power to drive their motors. These types of actuators have traditionally only been used for functions of the aircraft like trim of door actuation because of problems with response time and translational force. Jamming is an issue that is also involved with mechanical actuators due to many interfaces in which failures can occur. Advances in rare earth material for high power DC motors, solid state switching devices, and lightweight controls have increased the viability of using EMA for flight control applications [74]. With application of these technologies, failures can be reduced with penalties in complexity, cost, and weight. For these reasons the EMA is typically not used for primary actuation, however, with technology advances, performance may increase to the point of increased feasibility [81].

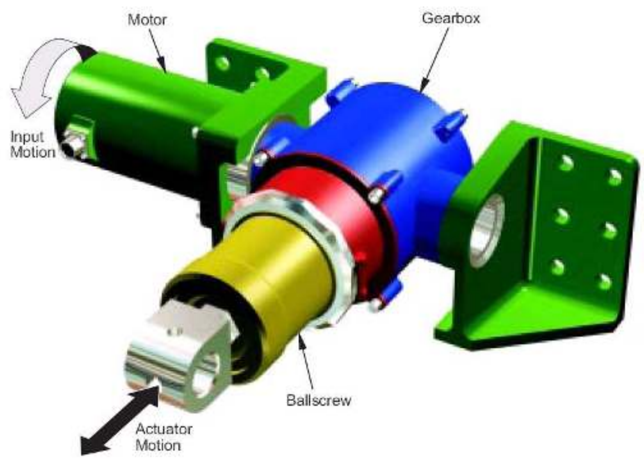


Figure 60: Electro-Mechanical Actuator [81]

Electro-Hydrostatic Actuator

In order to mitigate jamming and response time issues problems which are typically associated with the EMA, electro-hydrostatic actuators (EHA) utilize a localized hydraulic force. These actuators use a hydraulic piston to provide translational force; however, this piston is not connected to a central hydraulic system. This device has its own supply of hydraulic fluid and its own pressure/flow generating device, often in the form of an electrically driven fixed displacement hydraulic pump.

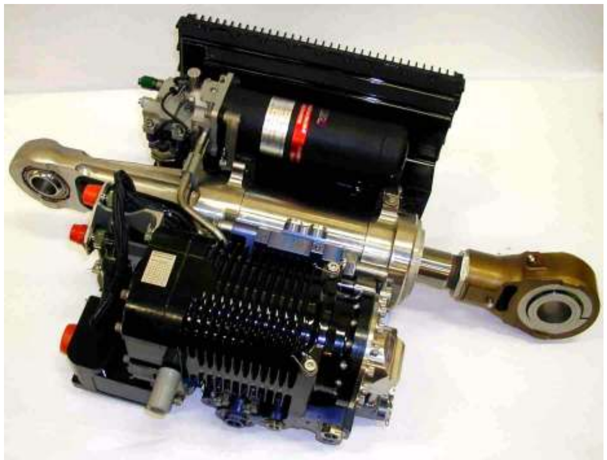


Figure 61: Electro-Hydrostatic Actuator [80]

The EHA eliminates the requirement to distribute hydraulic power, thus eliminating the weight and volume from the aircraft as well as other maintenance issues associated with the central hydraulic system. This device still must be connected to a power distribution system to drive a hydraulic pump. Each EHA is also far more complex than a simple hydraulic actuator. It not only requires its own built in power transformation device, but it also becomes a source of thermal energy which must be managed. The attributes and characteristic generated by a model representing this device have similar relationship to the EMA. However, this element does exhibit a thermal output that was not present with the other actuation devices.

ECS Technology

The environment control system is tasked with providing a comfortable and safe passenger environment under potentially dangerous conditions. This environment manipulation includes elevating temperatures and pressures higher than those present and cruise altitudes, removing ozone and other particulates from the incoming airflow, and providing oxygen. Cabin pressure is maintained at a minimum of 8000 ft pressure-altitude with temperatures ranging between 65 and 73 °F. Humidity is also managed in order to prevent ice from forming in the pneumatic tubing and ducting, condensation from occurring inside the cabin, and fungus and bacteria from growing. FAR regulations also stipulate that 0.55 lbs of outside air must be provided per passenger per minute. These requirements are met through providing external air and filtering and recirculating spent air back into the ECS system. As a result, the aircraft has a completely new cabin full of air ever 2 to 3 minutes [82].

The ECS system is very complex. It is comprised of multiple physical elements, each fulfilling individual functions of the system (mix, filter, cool, remove O_3, etc). All of these functions are generalized by the single function to condition air. Although this function could be decomposed further, applying a higher level of abstraction allows the architect to simply compare different types of environment control system concepts; bleed, and bleedless. If the functional breakdown was more detailed, trade-offs could also be made on the components used within these systems. However, for this proof of concept a higher level view is taken regarding the ECS definition.

Conventional Environment Control System

The major differences between the conventional and electric ECS are the source of the air and power driving the system. The conventional ECS, or "bleed" system, delivers air to the cabin which bled from a high or low pressure port in the engine's compressor. The conditions of the air vary, depending on the flight phase in which the airplane is operating and which bleed port is active. Pressure and temperature of the bleed air delivered by the engine vary greatly as displayed in Table 2.

Table 2: Bleed Temperatures and Pressures [82]

Mode of Operation	Temperature (℉)	Absolute Pressure (psi)	Extraction Stage
Takeoff - Maximal Power	660	170	Low Pressure
Top of Climb	590	100	Low Pressure
Cruise	480	50	Low Pressure
Initial Descent	365	29	High Pressure
End of Descent (Ground Level)	445	67	High Pressure
Switchover from High to Low Pressure	535	70	High Pressure
Ground Operations	340	--	Auxiliary Power Unit

In order to distribute and condition the air, temperatures in the pneumatic system are typically maintained to at approximately 350°F, not to exceed 470°F [82]. This is done using a heat exchanger positioned within the engine nacelle, extracting pre-compressor air from the fan, and dumping excess heat in this cooler air. This air is then discharged from the engine. It has been estimated that at certain points of the mission up to about 30% of the energy delivered by the engine to the pneumatic system is lost during this precooling operation [9]. Once this air has been cooled, it is delivered to the pneumatic distribution system. This system delivers the air to devices which perform all of the ECS functions. The cooling of the air is traditionally performed in a bootstrap cycle. This cycle cools the air through compression, expansion, and heat exchange with ram air.

Electric Environment Control System

In contrast to the conventional ECS, the "more electric" ECS system does not rely on pneumatic power from the engine to drive the system. With this concept, air is received by ram ports generally located in the belly faring of the aircraft and is compressed to much lower temperature and pressure than that provided by the engine compressor. In so doing, the need for a precooler is eliminated. The pressure and temperature is directly determined by the electrically driven compressor. However, with lower operating temperatures, the design of the elements tasked with ozone removal and heat exchange, must adapt and generally grow in size.

Ice Protection Technology

Ice buildup can have detrimental and hazardous effects on the performance of an aircraft. Icing occurs at altitudes and conditions where there is enough moisture in the air and cold enough temperatures such that ice begins to freeze on the skin of the aircraft. This generally only occurs during low altitude maneuvers, when the amount of moisture in the air can be problematic (e.g. takeoff, climb, descent, approach, holding, and landing). Ice buildup on the lifting and control surfaces can lead to flight instability, lack of control, and an inefficient production of lift. Systems must be used in the aircraft, which either prevent ice formation from occurring (anti-icing), or remove ice when present on the aircraft (deicing).

Pneumatic Heating Ice Protection

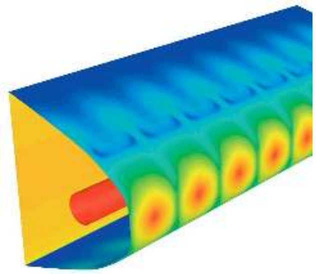


Figure 62: Pneumatic Ice Protection [83]

Conventional ice protection devices on large scale commercial aircraft utilize the high temperature air available in the pneumatic system to heat the leading edges of the wings in order to protect the wing from icing. This heated air is directed along the surface of the leading edge of the wing, providing thermal energy to melt the ice or increase the temperature of the surface of the wing to prevent ice from forming. After the thermal energy has been used, this air is discarded overboard, representing a loss in thermal energy. This alternative is characterized by a weight and volume attributes, as well as a use of pneumatic energy.

Electrical Ice Protection

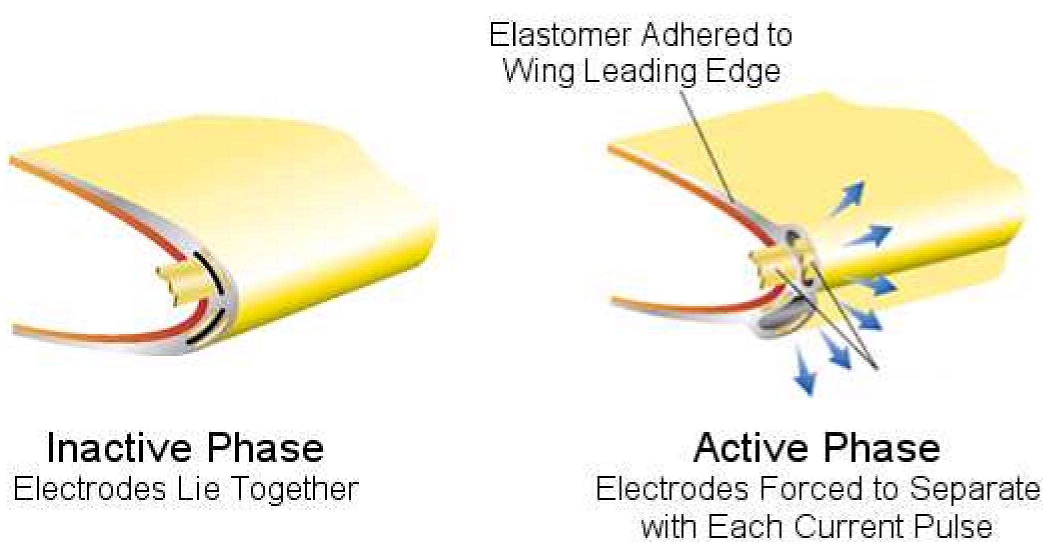


Figure 63: Electro Expulsive Deicing [84]

In order to remove the bleed system all elements that require pneumatic elements must be eliminated from the architecture. The “more electric” aircraft utilizes electrical energy to provide the functionality to protect from ice. Many ice protection devices utilize electricity. Deicing elements like electro-impulsive, and electro-expulsive, use electrical energy to provide a mechanical force and motion to the surface of the wing, breaking up accumulated ice [85]. Electrical anti-icing devices also exist which prevent ice from forming on the wing. Electro-thermal device convert electrical energy into heat, which protects the critical lifting and control surfaces. Heaters do not require a stable electrical signal. Therefore, variable frequency signals, which are more efficient to generate, are often used to power electrical heating [74]. Eddy current devices have also been proven to be able to provide ice protection [85]. In defining their place and interaction with the aircraft, these devices are characterized by their use of low voltage electricity, and their physical attributes in the aircraft.

Thermal Management Technology

For the design space defined in this test case, the function to provide thermal management can be accomplished by one of two means. Heat can be removed from the element by means of a cooling system which removes heat by means of convection to some delivered fluid, or the heat sources and heat sensitive elements can be protect by means of low conductance materials or insulation. Many types of cooling devices and insulations exist which can be used in this architecture. However, it is assumed here that they have a typical I/O structure, and therefore, in this proof of concept study not all thermal management tools devices are included in the design space. In an actual trade-off study more detailed characterizations and physics based models would be required. For this study, three alternatives are listed: vapor cooling, air cooling and insulation.

Power Distribution Technology

<u>Mechanical Distribution</u>

Mechanical distribution systems are elements which require mechanical power to operate and which deliver mechanical power to other power users. Several types of mechanical distribution systems have been defined in this trade-off environment. The first is typically used with flight control. Mechanical distribution systems are often used for flight control on smaller aircraft and backup systems for large commercial transports. These distribution elements are made up of pulleys, cables, levers, and rods. The system must receive its mechanical power source either from the pilot himself (unlikely in such a large aircraft) or some type of transformation device that receives some other source of power and provides sufficient force. This cable/pulley system is one type of mechanical distribution.

Flight control is not the only discipline that utilizes mechanical distribution. Due to the nature of the power relationships, other mechanical elements within the architecture must be defined as distribution devices. Power elements must always be connected by a distribution device. Therefore, elements like the generator must receive mechanical power from the engine by means of a distribution element. This device is defined as either the shaft itself, or an accessory gear box. Both devices receive mechanical energy from an engine, RAT, or APU and provide mechanical energy to other elements.

Performance and attributes of mechanical distribution devices depend highly on the type of material properties these elements exhibit. Multiple elements composed of different materials or of varying designs can be used in the architecture. However, for this conceptual case the only listed alternatives for mechanical distribution elements are cables/pulleys, shaft, and gear box.

Electrical Distribution

For this trade study there are four types of electrical power being used by elements within the system. These are high and low voltage AC and DC power. The trade off between high or low voltage electricity has to do with the efficiency of electrical distribution. High voltage DC electricity is distributed most efficiently. However, power usability and transformation efficiency must be taken into account when using higher voltages. In order to distribute HVDC power, the variable frequency signal must be rectified and if the end user requires AC power the DC signal must again be transformed to provide the correct power type. The hardware must also be in place to manage the high voltage signal and mitigate damage if failure should occur.

Four alternative distribution systems have been defined in this architecture; one for each power type. Each is represented by an electrical bus which distributes the electricity to all power users. A bus is a large conductor which provides an interface to which elements can be connected to receive the same electrical signal.

Sizing the bus system depends on the voltage and current of the signal being passed by the network, the power required by the power users, and the locations of both the power users and producers in the architecture. Electrical elements are generally not defined with static input relationships. Multiple sources for electrical power are available in the aircraft. Depending on failure and use-case scenarios the electrical network receives power from any of these sources. Having multiple sources of power simply increases the length of the network and increases the system reliability as a whole.

Pneumatic Distribution

The pneumatic distribution system is composed of ducting or tubing which connects the sources and end user of pneumatic energy. The tubing and valve sizes are governed by the pressure, mass flow rate, and temperature of the air distributed by the system. The lower pressures and temperatures seen by the pneumatic system are 50 psi and 180° C. However, during take-off takeoff the pressures and temperatures can typically raise to 410 psi and 540° C [74]. The operating conditions that are seen by the pneumatic system during all use-case scenarios determine the attributes of the pneumatic system.

Pneumatic distribution alternatives may differ in their material properties, and configuration. This, in turn, affects their weights and volumes. For high level trades, the pneumatic distribution system is handled as a whole in order to limit trade-off available for this proof of concept. In the design space created for this paper, only pneumatic tubing is available as a conceptual alternative in the design space. The bleed system is also handled as a distribution network.

Hydraulic Distribution

The sizing of hydraulic actuation system depends on the performance requirements of the aircraft, the pressure of the system, and the flow rate required. Many current aircraft have hydraulic systems with an operating pressure of 3,000 psi, however, pressures upwards of 8,000 psi have purported benefits in overall system mass [86]. Two options are provided in the design space: a high and a low pressure distribution system.

Power Transformation Technology

Depending on which power types are required by elements in the aircraft architecture and which power types are being provided, transformation elements must be used to utilize the power available to fulfill the functions. Many different types of transformation devices are used in an aircraft. As discussed in chapter 6, a transformation device is one which fulfills the requirement to provide some type of power and in turn requires a different type of power from another distribution network. All of these transformation devices are categorized by the type of power that they provide to the power users.

Table 3 shows a list of all of the transformation devices being used for this architectural trade off. These devices are characterized by their input and output.

Table 3: Transformation Devices

Power Output	Transformation Device	Power Input
AC Electric	Alternator	DC Electric
	Integrated Drive Generator	Mechanical
	AC Generator	Mechanical
	AC Filter	HVAC
DC Electric	Transformer Rectifier Unit	HVAC
	Transformer	HVDC
Pneumatic	Compressor	Electrical
	Heat Exchanger	Pneumatic (different characteristics)
Hydraulic	Electro-Motor Pump	HVAC
	Mechanically Driven Pump	Mechanical
Mechanical	Motor	JetA
	Electrical Motor	AC Electric
Hi Voltage AC	Imbedded Starter Generator	Mechanical
	HVAC Generator	Mechanical
Hi Voltage DC	HV Rectifier	HVAC

Power Generation Technology

In order to provide adequate reliability and performance, multiple power sources are made available in the aircraft architecture. For this trade off exercise only conventional power generation devices are considered. Four main elements are present in the design space: a turbofan engine, a more electric engine, a conventional auxiliary power unit (APU), and a ram air turbine (RAT). Other more advanced or unconventional technologies, like fuel cell APU, that can fit in this design space are not included in this exercise.

Engine and APU

Both the engine and the APU are turbo machinery which receive jet fuel and provide mechanical and pneumatic outputs. The efficiency of this power production is based on the cycle of the engine, the atmospheric conditions, and the component efficiencies. Energy is extracted from engines shaft by means of a gear box or an embedded generator. The main difference between the engine and APU is the generation of thrust. The primary function of the APU is to generate usable power for aircraft systems while the task to provide propulsion for the aircraft is the largest load required from the engine.

Turbofan engines have very different cycles, each engine defined by particular performance maps and engine decks. The functions associated with the engine cycle and other attributes (compress, combust, provide mechanical energy, house engine, lubricate, etc) are not addressed in this design space. Various implementations of these functions provide vastly different engine concepts.

One engine concept that is represented in this design space is the “more electric engine” (MEE). The more electric engine attempts to use electrical power in the fulfillment of all engine accessory functions. Magnetic bearings are used to provide lubrication, negating the requirement to provide oil and pressurize the oil. In order to remove accessory gear box (creating a symmetric nacelle) and pneumatic off take, the MEE only allows the extraction of mechanical energy from the shaft by means of an embedded starter generator. This device utilizes electrical energy provided by the architecture or external sources to drive the embedded generator as a motor. This action ramps up the spool for engine starting. Traditionally pneumatically driven starting system performs the same function. Other engine auxiliaries are altered with the increase of electrical energy generation. Thermal energy management becomes a more significant issue with the more electric engine.

RAT

The final technology discussed in this section is a backup energy generation device; the ram air turbine (RAT). The RAT is a device that provides mechanical power to be used when all other power sources have failed or are no longer available in the aircraft. This small turbine drops from a hatch into the free stream air around the aircraft body. This turbine generates shaft power which can be used to pressurize hydraulic lines, generate electricity, or power other transformation elements.

Figure 64: Ram Air Turbine [87]

The power generated by the RAT is used to fulfill the critical aircraft functions that are needed to get the passengers to the ground safely (flight control functions, pilot interface, avionics, ECS, etc). The size of the turbine depends on the speed and flight conditions of aircraft and the power required by the ultimate power users.

Summary

Boundary functions, induced functions, alternative technologies, and all possible relationships characterize the design space of the architecture. The functions and alternatives in this trade-off space are organized in a tree structure as shown in Figure 65. This tree is categorizes these by type of function: boundary, power, secondary induced, and installation induced.

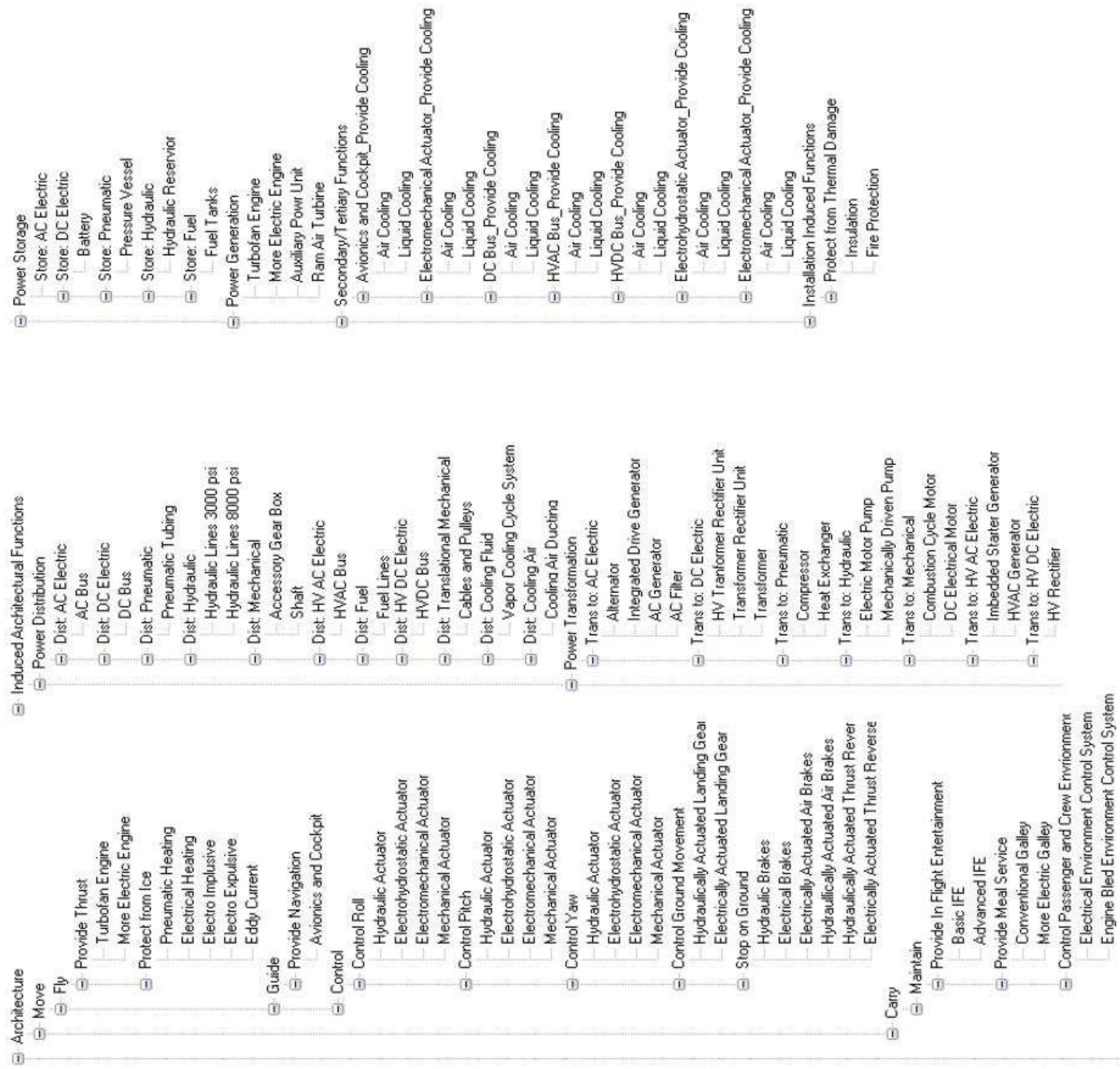


Figure 65: Architecture Design Space

With the design space based on functions the physical nature of the architecture depends solely on the boundaries of each alternative defined. Any combination of physical elements represents a feasible architecture as long as the boundary functions of the aircraft are fulfilled. The next chapter will discuss the exploration of these functions and alternatives to create a conventional, a more electric, and an all electric architecture.

CHAPTER 9

ALTERNATIVE CONCEPT DEFINITION

With the design space defined on the basis of functions, the architect can now define and configure the elements of the architecture. Using object oriented programming allows for easy definition and redefinition of the architecture. The rules and attributes of the design space described in chapter 8 become the conditions and conventions used in the definition of the architecture. The tools described in chapter 6 enable the definition of multiple architectures. The application of these tools to this design space is discussed as well as three architecture alternatives generated by this process. All alternatives generated and compared here are subject to the limitation of the design space defined for this architecture. More accurate representations of architecture may be defined with a more detailed design space.

ARM

The adaptive reconfigurable matrix of alternatives is defined directly from the architecture's functional breakdown and alternative technology lists. If each function in the design space is to be defined by a single alternative, this space would represent, in total, 48.3 billion distinct technology sets. However, using this object oriented process and allowing the designer as much freedom as possible, the number of solutions increases drastically. The ADEN tools not only provide the mapping of function to alternative, but also provide means to configure redundancy and failure case scenarios. The number of alternative configurations increases when redundancies and alternative operational configurations are introduced to the design space. Each function can be fulfilled by multiple means at the same time. An example can be seen in flight control. Multiple means of providing basic flight control actuation are always used in current commercial aircraft. If one system fails, another system is available to fulfill the functions in its place. Redundancy and segregation ensure that the system exhibits appropriate reliability in performing the critical boundary functions [88]. Federal regulations require that the reliability of a commercial transport aircraft be extremely high, requiring a maximum of $1x10^{-9}$ probability of catastrophic failure [89]. Requiring redundancies on the propulsion, control, transformation, and generation functions increases the number of alternative combinations to 4.75 x 10^{15}. To provide appropriate reliability, some functions may require multiple instantiations of the same element to fulfill the defined function. An example of this is the instantiation of multiple engines which act to provide thrust and non-propulsive power.

Figure 66 displays the list of boundary functions for a conventional aircraft. Redundant boundary functions are required for the control and stopping functions. This allows for multiple means to be used for the fulfillment of a single function. The ADEN tools also allow multiple instances to be identified for the fulfillment of the one architecture function.

Functions	Alternatives	Instances
Provide Thrust	Turbofan Engine	2
Protect from Ice	Pneumatic Heating	2
Provide Navigation	Avionics and Cockpit	
Control Roll	Hydraulic Actuator\Control Roll	2
Control Roll_1	Hydraulic Actuator\Control Roll_1	2
Control Pitch	Hydraulic Actuator\Control Pitch	2
Control Pitch_1	Hydraulic Actuator\Control Pitch_1	2
Control Yaw	Hydraulic Actuator\Control Yaw	
Control Yaw_1	Hydraulic Actuator\Control Yaw_1	
Control Ground Movement	Hydraulically Actuated Landing Gear	
Stop on Ground	Hydraulic Brakes	
Stop on Ground_2	Hydraullically Actuated Air Brakes	2
Stop on Ground_1	Hydraulically Actuated Thrust Reversers	2
Provide In Flight Entertainment	Basic IFE	
Provide Meal Service	Conventional Galley	
Control Passenger and Crew Envrionment	Engine Bled Environment Control System	2

Figure 66: ADEN Boundary Functions (including redundancies) and List with Conventional Alternatives with number of instances

For the conventional case, the alternatives selected to fulfill the boundary functions of the aircraft varied in their input. The ECS requires pneumatic energy, the control systems required hydraulic energy, the engines require fuel, and the avionics, IFE, and galley require electricity. These induced requirements determine the power types to be provided by the power systems of the aircraft. The induced functions, their redundancies, the fulfilling alternatives, and the number of instances of these alternatives are shown in Figure 67.

Dist: AC Electric	AC Bus	2
Dist: DC Electric	DC Bus	2
Dist: Pneumatic	Pneumatic Tubing	4
Dist: Hydraulic	Hydraulic Lines 3000 psi	3
Dist: Mechanical	Accessory Gear Box	4
Dist: Mechanical_1	Shaft	4
Dist: HV AC Electric		
Dist: Fuel	Fuel Lines	2
Dist: HV DC Electric		
Dist: Translational Mechanical		
Dist: Cooling Fluid		
Dist: Cooling Air	Cooling Air Ducting	
Trans to: AC Electric	Alternator	2
Trans to: AC Electric_1	AC Generator	4
Trans to: DC Electric	Transformer Rectifier Unit	2
Trans to: Pneumatic	Heat Exchanger	2
Trans to: Hydraulic	Mechanically Driven Pump	4
Trans to: Mechanical		
Trans to: HV AC Electric		
Trans to: HV DC Electric		
Power Generation	Turbofan Engine\inst	2 inst's
Power Generation_2	Ram Air Turbine	
Power Generation_1	Auxiliary Powr Unit	
Store: DC Electric	Battery	2
Store: Pneumatic	Pressure Vessel	2
Store: Hydraulic	Hydraulic Reservior	3
Store: Fuel	Fuel Tank	2
DC Bus_Provide Cooling	Air Cooling	2
AC Bus_Provide Cooling	Air Cooling	2

Figure 67: Induced Functions and Conventional Fulfillment

Redundancies are very important in the fulfillment of the induced functions of the architecture. The generation of power in an aircraft is done in very different ways depending on the flight and failure conditions. While sitting at the gate and during starting and diagnostics, the APU generally provides the power necessary to run essential functions while the engines are off. The RAT is also only used during specific scenarios. It is not used in general operation of the aircraft, but must be available to provide power with the failure of both engines [74]. Multiple power generation functions were defined so that multiple elements can be designated to fulfill this function.

Another interesting situation occurs when the turbofan engines are defined as power generation devices when they have been previously been selected to fulfill the “Provide Propulsion” function. This element must be allowed to fulfill multiple functions at the same time. The ARM tool provides the option of referencing previously defined elements to fulfill these new functional requirements. In Figure 67 the third column of the row entitled “Power Generation” reads, “2 inst’s.” This label means that the 2 instances of the turbofan engines defined as the means to provide thrust to the aircraft are also intended to be used as non-propulsive power sources.

Exploration of this design space defined by 11 boundary functions yielded a conventional aircraft architecture with over 60 physical elements. This number of elements can also be greatly increased, depending on the reliability of the elements chosen to fulfill the architecture functions and potential failure situations that must be defined.

An all electric architecture was also generated using this process of architecture concept definition. The same redundancy requirements were applied to the boundary functions of this design space as those applied to the conventional architecture. However, the structure of the induced functions of the architectures is extremely different. This is driven by the reliability required for the fulfillment of the critical boundary functions while utilizing different induced technologies. It should be noted that the idea of an “all electric” power source is somewhat of a misnomer. Thrust is generated through some mechanical means of thrust production. The more electric engine utilizes jet fuel and generates mechanical energy. For these reasons it may not be classified as “all electric” because of its use of power types other than electricity. However, for the purpose of this thesis the “all electric” concept utilizes the fuel driven, mechanical energy producing, more electric engine. A list of the functions, alternatives, and instances is displayed in Figure 68.

Functions	Alternatives	Instances
Provide Thrust	More Electric Engine	2
Protect from Ice	Electrical Heating	2
Provide Navigation	Avionics and Cockpit	
Control Roll	Electrohydrostatic Actuator	2
Control Roll_1	Electromechanical Actuator	2
Control Pitch	Electrohydrostatic Actuator\Control Pitch	2
Control Pitch_1	Electromechanical Actuator\Control Pitch_1	2
Control Yaw	Electrohydrostatic Actuator\Control Yaw	
Control Yaw_1	Electromechanical Actuator\Control Yaw_1	
Control Ground Movement	Electrically Actuated Landing Gear	
Stop on Ground	Electrical Brakes	
Stop on Ground_2	Electrically Actuated Air Brakes	2
Stop on Ground_1	Electrically Actuated Thrust Reversers	2
Provide In Flight Entertainment	Advanced IFE	
Provide Meal Service	More Electric Galley	
Control Passenger and Crew Envrionment	Electrical Environment Control System	2
Dist: AC Electric	AC Bus	2
Dist: DC Electric	DC Bus	2
Dist: Pneumatic		
Dist: Hydraulic		
Dist: Mechanical	Shaft	4
Dist: HV AC Electric	HVAC Bus	2
Dist: Fuel	Fuel Lines	2
Dist: HV DC Electric	HVDC Bus	2
Dist: Translational Mechanical		
Dist: Cooling Fluid	Vapor Cooling Cycle System	
Dist: Cooling Air		
Trans to: AC Electric	AC Filter	2
Trans to: AC Electric_2	AC Generator	2
Trans to: AC Electric_1	Alternator	2
Trans to: DC Electric	HV Tranformer Rectifier Unit	2
Trans to: DC Electric _1	Transformer	2
Trans to: Pneumatic		
Trans to: Hydraulic		
Trans to: Mechanical		
Trans to: HV AC Electric	Imbedded Starter Generator	2
Trans to: HV DC Electric	HV Rectifier	2
Power Generation	More Electric Engine\inst	2 inst's
Power Generation_2	Auxiliary Powr Unit	1
Power Generation_1	Ram Air Turbine	1
Store: DC Electric	Battery	2
Store: Pneumatic		
Store: Hydraulic		
Store: Fuel	Fuel Tank	2
Avionics and Cockpit_Provide Cooling	Air Cooling	1
DC Bus_Provide Cooling	Air Cooling	2
HVAC Bus_Provide Cooling	Air Cooling	2
HVDC Bus_Provide Cooling	Air Cooling	2
Electrohydrostatic Actuator_Provide Cooling	Air Cooling	5
Electromechanical Actuator_Provide Cooling	Air Cooling	5
AC Bus_Provide Cooling	Air Cooling	2

Figure 68: Functions and Alternatives for “Electric Architecture”

FMM

The FMM allows the architect to direct the flow of energy between elements within the system. This is necessary to provide the interplay of requirements between all fundamental elements within the architecture. These relationships allow each of the elements to be sized with all information necessary. This matrix also provides an interface in which trades can be made between interrelationship mappings. The FMM is used in correlation with the ARM tool to identify the relationships between selected technologies.

These mapping matrices increase the number of decisions that must be made to define the architecture. The FMMs for the conventional and electric architectures are displayed in Figure 69 and Figure 70.

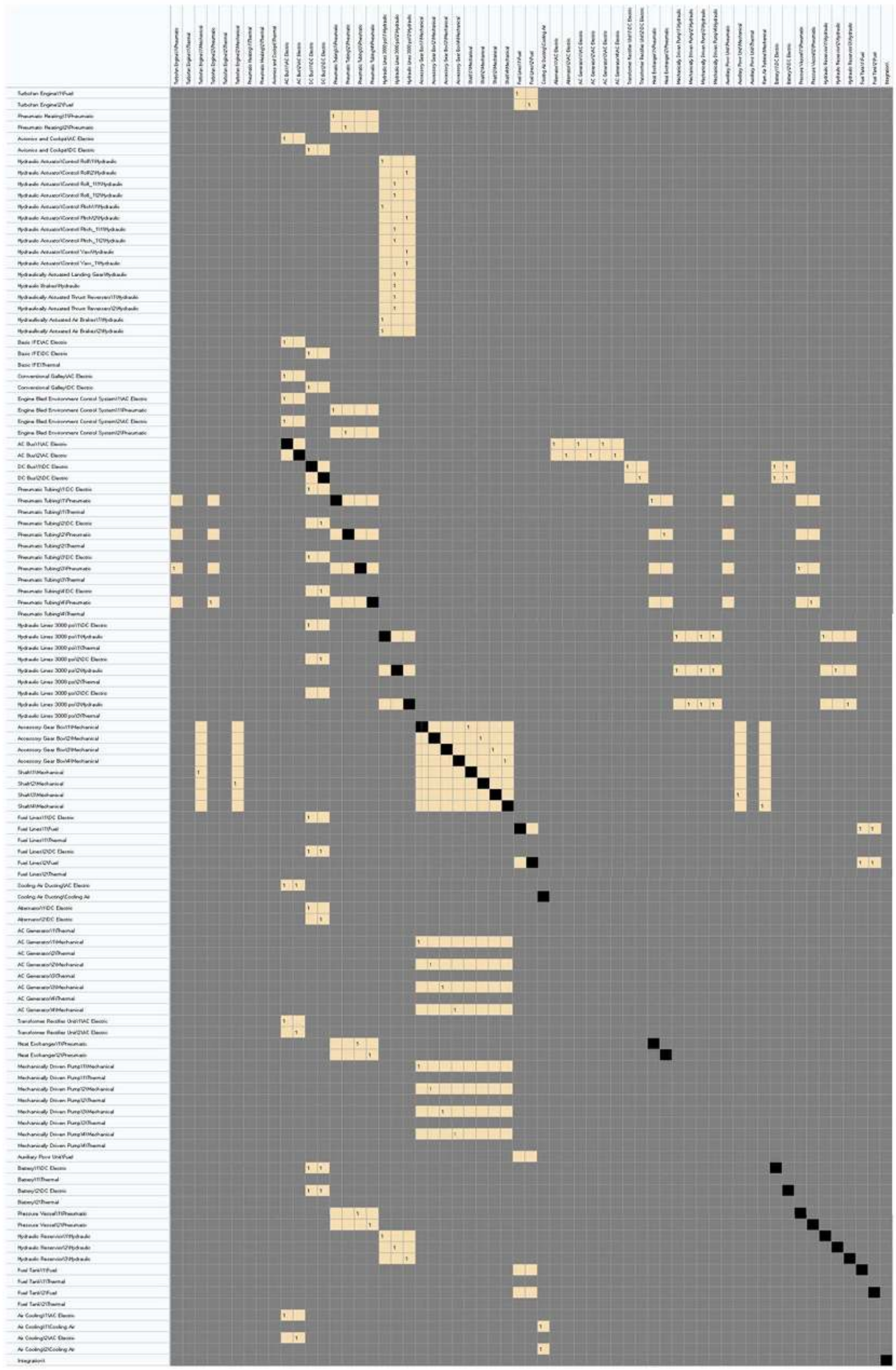


Figure 69: Conventional Aircraft Architecture FMM

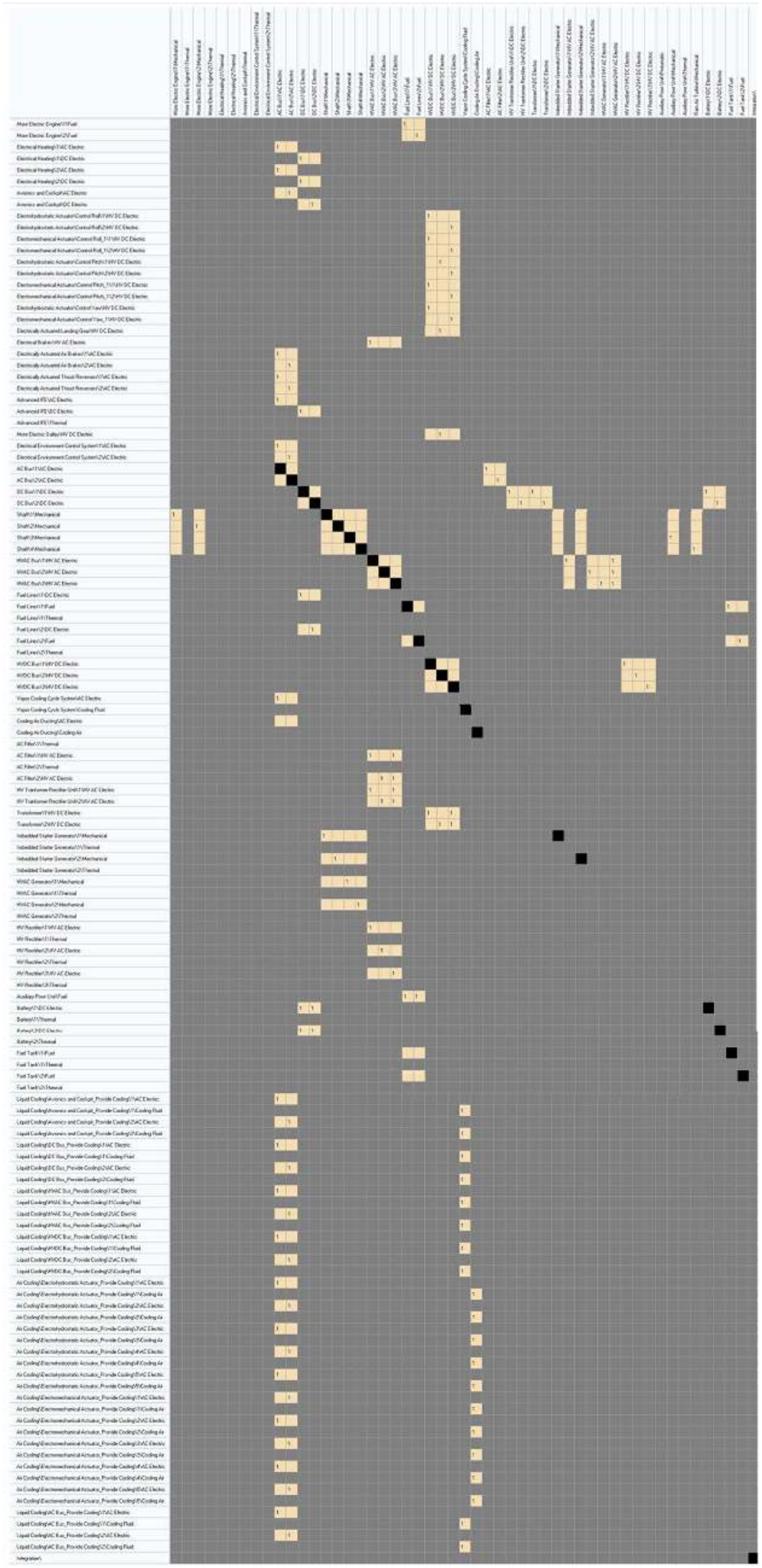


Figure 70: More Electric Aircraft Architecture FMM

Even after the technologies have been selected and introduced into the system, a multitude of decisions have to be made as to their relationships. Each technology power input listed along the left hand side of these figures must be attached to an appropriate power provided marked in the tan cells of the matrix. Assuming that only one link is necessary for each element, there are 5.21×10^{39} possible network configurations for the conventional architecture, and 8.57×10^{29} possible for the more electric architecture configuration. This number grows if the architect must decide how many redundant connections must exist.

These large matrices, indicating vast numbers of possible interconnection scenarios, require the input of expert aircraft architects who understand the implications of redundancies and how these are to be configured in the architecture. For example, the redundant control roll element would most likely not receive its power from the same distribution network as the primary control roll element. In case of the failure of one of the distribution networks, the other would be able to provide the power necessary to fulfill this critical function. Each architecture configuration should be logically defined to provide desired functionality and reliability. The myriad of configuration possibilities must be addressed and reduced by technical reasoning. This tool allows for the investigation of the critical scenarios governing these decisions.

Each redefinition of the relationships between elements within the system represents a completely different architecture with different attributes and performance. The amount of power that must be distributed or provided by a specific induced function changes with the number, type, and functions of the elements attached to it downstream. For example, if all the power is provided to actuation and control by the same electrical distribution network, the attributes of this system will be different than a system which carries half the load. The passing of functionality through all elements must be designated for each operating scenario. All of these scenarios supply different sets of requirements to the boundary and induced functions. In turn, multiple networking configurations must be defined for each mission segment and failure scenario. Another tool is necessary to define the re-networking of the system for each flight scenario. This will be discussed later in the "Mission Definition" section of this chapter.

The FMM tool provides the opportunity to investigate failure cases by designating if the relationship between components as fixed or flexible, as discussed in chapter 7. This tool also allows the designer to designate which of the elements has failed during the scenario. In the case that the left engine fails in the conventional architecture, the relationships must be re-designated to provide functionality to critical functions. This condition is displayed in Figure 71.

Figure 71: Failed Engine FMM for Conventional Architecture

If this engine fails, elements which have fixed relationships with this engine in turn cannot provide their functionality. Other elements which interact with the engine and have flexible relationships can be configured to receive fulfillment of their induced requirement by another power element. As seen in Figure 71, the FMM indicates that the engine has been failed by the designer by coloring the matrix cells associated with the engine in dark red. A fixed relationship between the mechanical power source provided by the turbofan engine and the accessory gear box is indicated by the red circle. This causes the accessory gear box to fail, in turn causing the generator attached to it to lose its power source. These relationships are indicated by the blue and black circles. All elements that lose their functionality due to the failure of the upstream power providers are indicated by light red matrix cells. In order to fulfill the critical functions, selected technology sets must allow for power to be provided to all of the elements fulfilling critical functions.

With this interface in place and the modeling tools appropriately defined, drastic changes in the structure and interconnectedness of the architecture do not compromise the structure of the modeling and simulation environment. Although the matrices generated for these examples may look very large and add to the complexity to the architecting task, they provide a flexible interface which does not prematurely infer the structure of the system. They also allow for the investigation of intricacies that are not generally captured until the later phases of design.

Installation

For the architectures generated is this test case, a conventional layout was adopted and the elements were configured in a typical fashion. The engines are placed under the wing, the ECS system and electronics are primarily located in the belly faring, the cockpit is fore of the passenger cabin at the front of the fuselage, the APU is placed in the tail section, the pneumatic storage elements are placed in the pylons, and the avionics and flight control are placed below the cockpit. Trade-offs can be made between the locations of elements within the architecture. With this installation tool in the architecting framework, standard trade-offs can be developed between element placement, and generalizations can be developed regarding the architecture level impact of technology placement within the architecture. A listing of all the locations of the elements is displayed in Appendix F.

Based on this installation configurations three thermal protection zones were identified. These zones are the ones which contain the engines and APU. The thermal outputs from the engine and the thermal sensitive elements within the same zone require some type of insulation or fire suppression technology to be applied. For this example the avionics bay cooling is handled on an element per element basis. This, however, could very easily be configured on a zonal basis. For the all electric candidate architecture, the high power electronics were placed in the belly faring with the electrical buses. In the actual design these elements may be distributed throughout the aircraft or be grouped elsewhere. This is one example of an aspect of the aircraft which could be handled in conceptual architecture design and is enabled by this process and toolset.

Mission Definition

The last dimension in which the architect can impact the performance of the design is through the manipulation of how this architecture will be used. Each "use-case" scenario must be defined by the requirements imposed on the architecture via the boundary functions. All architecture elements can be sized based on the requirements imposed by one or many operating scenarios. This sizing can be based on the peak, minimum, or nominal power requirements for all scenarios. Use-cases are derived from the mission profile and from failure scenarios.

In the ADEN toolset, 22 use-case scenarios were defined as potential sizing cases for the architecture. These use cases include 11 mission segments (at gate, taxi, take off, climb, cruise, descent, loiter, approach, land, proceed to gate, and reserve) and 11 failure scenarios, including engine failures, electrical power distribution system failures (AC/DC, hydraulic network), and ECS failures. These 22 scenarios must be defined by the external conditions placed on the system and the internal configuration of the power relationships. 16 different architecture configurations were defined for these 22 mission segments.

Output File

Once the 3-dimensions of architectural concept definition have been configured and the use-cases of the architecture have been identified, this information must be packetted in a manner which allows this architectural concept to be synthesized in a modeling and simulation environment. The technologies used to define the architecture, the relationships of these models with each other, and the interfaces in which these elements can be embodied to achieve the overall product requirements. Some of these relationships remain the same and others vary for every mission scenario.

Three types of variables can be traded between models synthesizing the architecture; conditions, attributes, and power requirements. These classes of variables have different natures during the modeling and simulations process and must be passed between the 3 different types of modeling elements; technology models, zone groupings, and architectural sizing and analysis.

The first type of variable is defined directly by the mission definition and the use-case scenarios. Variables like cruise altitude and Pax stay constant regardless of how the attributes of the architecture change and are needed to find the power requirements and attributes of some of the technologies interacting with the environment. Attributes like the weight and power requirements of the galley and IFE system depend on the number of passengers. The performance and attributes of an electrical ECS system and the engine depend on the temperature and pressure of the air they receives. These variables are user defined values defined in the mission definition.

The second class of variables changes with each iteration of architecture sizing and have a fixed I/O during the modeling and simulation process. These variables include attributes of the elements, zones, and overall aircraft. This list of attributes includes all measures of effectiveness and physical descriptions of the elements and architecture which are not passed as power variables. Variables of this type can change with each iteration of architecture sizing and are handled by the zone definition of the architectures.

The third type of variables changes during modeling iterations and for each use case sizing operation. These are the power variable relationships defined in the FMMs. On the lowest level of the architecture definition, power relationships interrelate the performance and attributes of the individual elements used in the architecture. The I/O of these power variables depends on the FMM defined for each use-case scenario. Arrays of power relationships for every sizing scenario can be used to interrelate these elements and allow them to be sized based on the requirements from all operating conditions. Some of these relationships can also be configured to place requirements on the external environment. If that is the case, product level integration routines will have to handle these variables and use these requirements in aircraft level sizing.

Relationships between technologies, zones, and analysis and sizing routines are displayed in Figure 72.

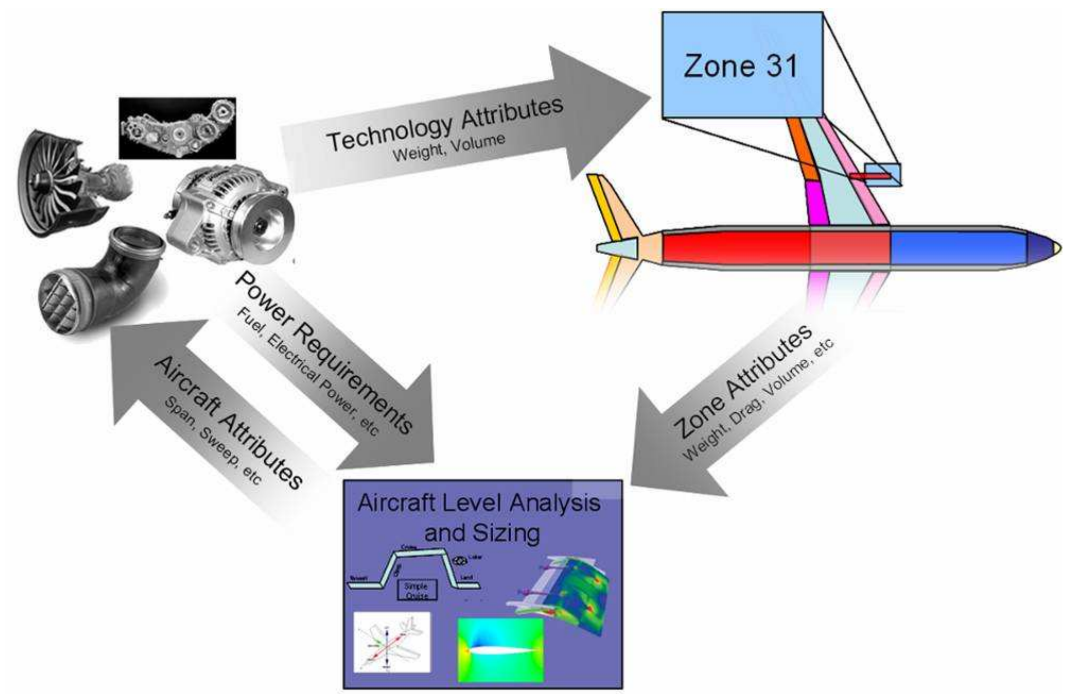


Figure 72: Relationships between Technologies, Zones, and Analysis

In order to size some of the systems, overall aircraft attributes are needed. The size and power requirements of the ice protection system, for example, depend on the size and shape of the wing. Wing span, sweep, area, and other attributes are calculated by aircraft sizing and analysis. These values will always be managed at the overall architecture level. Aircraft level attributes may change after it has been sized, however, with a defined set of technologies, these relationships between the outputs of sizing and the inputs of the architectural elements follow the same path.

Once the technologies and architecture elements have been sized, their attributes are grouped into their zones. This packaging allows the attributes of elements which are tightly grouped to be combined. Each zone is defined by the attributes of the systems contained it. Handling the elements in zones allows the attributes of the individual technologies to be rolled up to product level.

The attributes of the zones combine to define the overall attributes of the aircraft. Depending on the attributes of the individual zones, aircraft attributes change. For example, the weights of the zones composing the fuselage act in determining the location of the center of gravity. The requirements on technologies within fuselage zones could increase their size and weight, thereby changing the center of gravity, effecting stability, and in turn change the location wing. Other attributes can also be affected by the placement and sizing of individual technologies. Wing thickness may increase with the volume of technologies located in the wing, and leading edge zones. As a result, the drag polar is modified; changing the amount of fuel required to fulfill the mission. Zone attributes must be integrated in sizing the overall aircraft.

All of these relationships are managed by a variable interaction between models. The ADEN tool writes an output file describing all of the relationships between elements, zones, and integration routines for sizing and synthesis. An example output file can be seen in Appendix G. This process of modeling this architecture was also previously discussed in Chapter 6.

Conventional vs. Electric Technologies

The ADEN toolset makes it possible to change the architecture very simply. This has been shown in the introduction of the tools and examples given. Current comparisons between the electrical and conventional architectures must be done in a purely descriptive manner. With the definition and integration of technology and integration models actual physical comparisons can be made between architecture concepts. Work is being done at the ASDL in exploring integration environments suitable to storing and integrating models which have been described by the ADEN toolset outputs. A qualitative comparison of the number and type of elements is provided. This short commentary highlights the possible implications of implementing electric technologies.

All architectures developed were characterized by the same architecture design space, and have the same number of elements fulfilling the same boundary functions. Two of these concepts are the conventional and electrical architectures introduced earlier in this chapter and the third is a “more electric” architecture. This more electric concept uses hydraulic and electrohydrostaic actuation for flight control functions, in turn, requiring the generation of high voltage DC electricity. The AC generators are replaced with high voltage AC generators. Electrical heating was also used for the function to protect from ice. This function utilizes also requires the generation high voltage AC power in the architecture. Filters are then required to provide the low voltage AC electricity and rectifiers are included to produce a high voltage DC signal from the high voltage AC signal. This provides the electricity for actuation. Finally, transformers are implemented to reduce the voltage of the DC power for elements which require low voltage DC electricity.

The boundary functions, the number of boundary elements, and the number of power sources are kept the same. Therefore, comparisons between these architectures were made in terms of number and type of distribution, transformation, storage, and general induced elements. For each one of these alternative sets, many different configurations and layouts are possible. These three architectures were defined by the elements required to provide adequate functionality and redundancy. A breakdown of the functional elements of the three concept architectures can be seen in Figure 73.

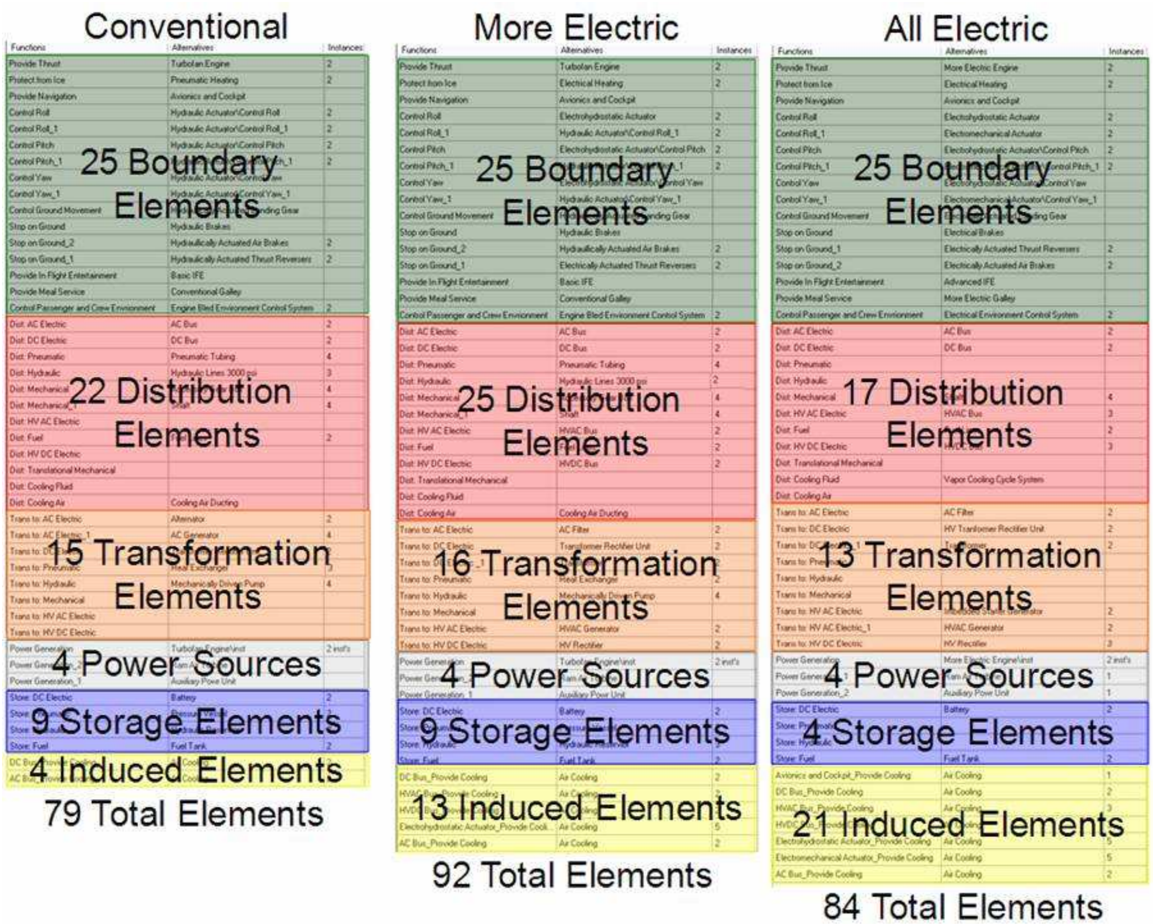


Figure 73: Breakdown of the Functional Elements of Three Aircraft Architecture Concepts

Main differences between these three architectures occur in the type of power utilized in the induced functions. The conventional architecture utilized low voltage AC and DC power, as well as engine bled pneumatic, and hydraulic power. The more electric architecture adds high voltage AC and DC to its power types in order to facilitate electrical actuation and pneumatic heating. Finally, the all electric architecture eliminates hydraulic and pneumatic power usage by replacing the elements requiring these power types with electrical elements. The all electric aircraft utilizes high and low voltage AC and DC power in fulfillment of its boundary functions.

The changes in boundary technologies naturally impact the number of distribution and transformation devices necessary in the architecture. The conventional aircraft uses redundant systems delivering electrical, hydraulic, and pneumatic power to deliver energy to its devices. In total the conventional architecture uses 22 distribution networks, 15 transformations, and 9 storage elements to deliver this energy. Due to the addition of high power devices, the "more electric" architecture adds high voltage distribution systems, giving it 25 distribution devices in total. The size of the low voltage distribution system will be smaller in the more electric system because high power electricity is being used for the control and generation functions. One hydraulic network is removed because hydraulic primary flight control is replaced by electric actuation. In the more electric example the number of transformation devices also increases. These transformation devices, however, are quite different when it comes to electrical power. The conventional architecture relies on low voltage generators and TRUs. The more electric concept has high voltage generators, filters and rectifiers to provide high voltage DC power. 9 new induced functions also appear in the architecture design space. These include potential cooling of the high voltage actuators and high voltage distribution systems.

The results of the architecture selection process indicate that the number of elements in the all electric architecture is larger than that of the conventional. Even with a reduction of all types of power elements, this increase of total elements is due to cooling functions that are required with the introduction of high powered devises. These and other side effects are key considerations in determining the effectiveness of architecture configurations. If the heat removal functions for the actuation elements are not required or the heat generated by these actuators is not sufficient to warrant the inclusion of a whole new system, the total number of elements in the all electric architecture would be 74 and the number of elements in the more electric architecture would be 87.

All benefits to all electric technologies must lie in the efficiency in which they perform their functions, or an improvement of individual technology attributes. Potential benefits may emerge in the "all electric" aircraft concept in terms of reducing the number of power induced functional elements in the architecture. However, the motivation of employing more electric technologies should not lie in the reduction of the number of components in the architectures. The number of distribution, transformation, and storage elements for the "all electric" concept is less than those required by the conventional or more electric alternatives. This does not necessarily mean that this architecture is better.

The effectiveness of an architecture depends on the communitive performance of the architecture elements in terms of reliability, segregation, redundancy, and the occurrences of induced functions instigated by these new technologies. With high voltage electrical elements, many new thermal management induced functions are introduced into the system.

Conventional vs. Electric Structures

All of these aircraft architectures have been laid out conceptually after a similar fashion. In the future and with appropriate models in place, layout definition has the potential to greatly impact the usability, manufacturability, and performance of an aircraft architecture. The FMM was used to ensure that appropriate redundancy occurred in providing power to all of the elements of the architecture. This tool is essential to designate relationships between models, however, because this test case does not continue all the way through modeling and simulation they are not discussed further in this section.

A graphical representation of these functional/physical chains of these three architectures can be seen in Figure 74. These charts are primarily illustrative in nature. Each displays the relationships created in the FMM and how the architectures are characterized by different power relationships. The points organized in the circle represent component or technologies selected using the ARM for all of the alternatives and the lines represent power relationships configured in the FMM.

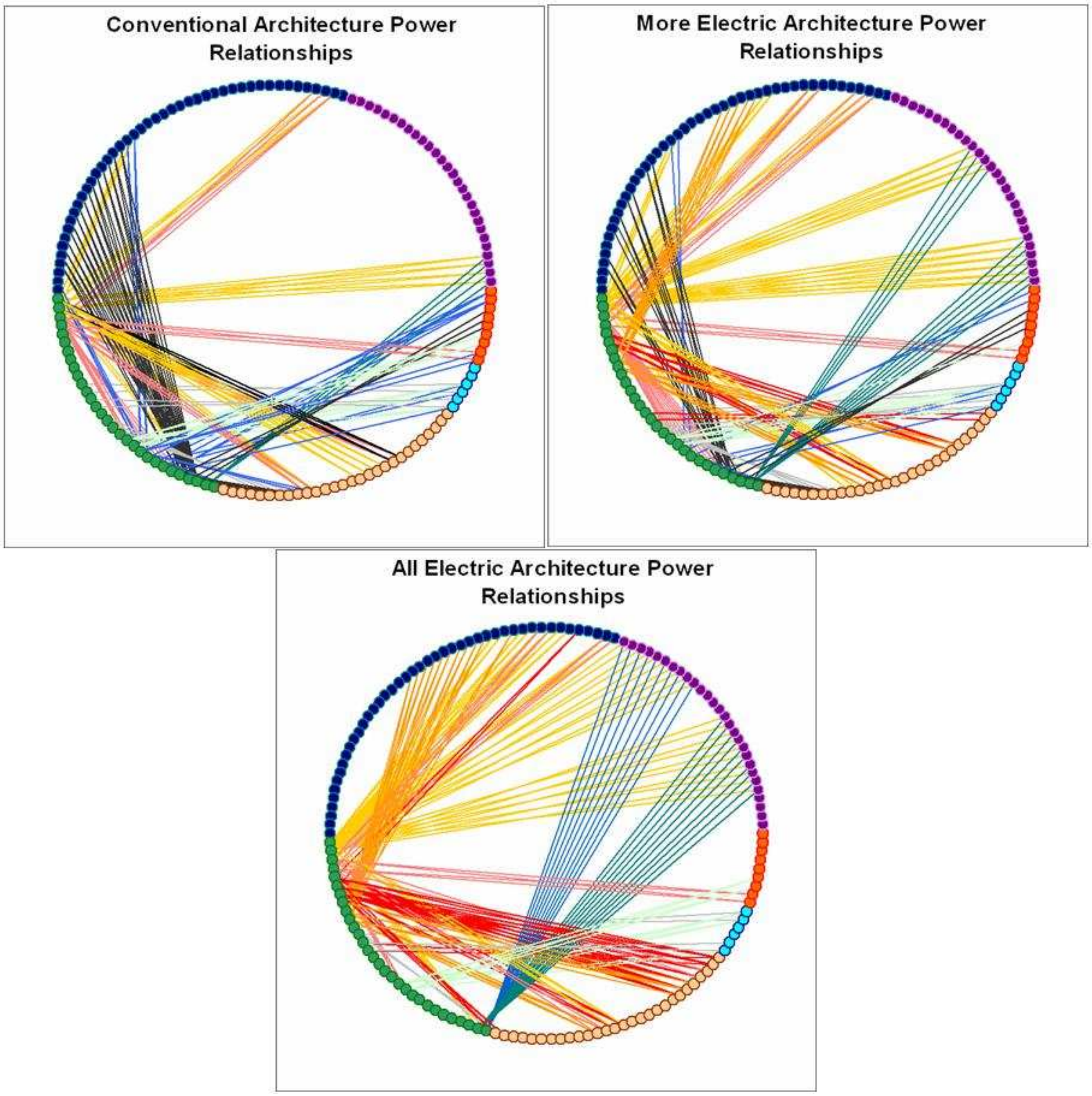


Figure 74: Power Relationship Mapping for Three Architectural Concepts

The different colored points on the circle represent different types of architecture elements. The dark blue points represent boundary elements, the green represents distribution elements, the tan represent transformation elements, the light blue represent power source elements, the orange represent power storage elements, and the purple represent general induced elements. The relationships between the elements are also color-coded by power type. This coloring is detailed in Table 4.

Table 4: Relationship Graph Line Color Coding

Line Color	Power Type
Red	HVAC
Orange	HVDC
Yellow	AC Bus
Blue	Pneumatic Air
Green	Cooling Air
Dark Blue	Cooling Liquid
Pink	DC Bus
Light Green	Fuel Lines
Black	Hydraulic
Grey	Mechanical

The distribution of relationships within the circle and the colors of the connections show differences in the technologies applied and the power types required in the architecture. As displayed by these charts, the alternatives generated with the ADEN utilize different technologies and transfer energy between these technologies through different means. The relationships visualized with these circle charts in Figure 74 are not the only configurations possible for each technology set. A large number of possible relationships can be configured for each set of technologies. These configurations are singular examples of the many relationships possible for each technology set.

Summary

The tools outlined in the previous chapters allow for the definition of vastly different architectures. The three concepts outlined in this chapter provide illustrative examples of the freedom which this toolset provides the architect in the definition of an architectural concept and the configuring of modeling and simulation. Each architecture utilizes different technologies and different redundancy and segregation methods. With this toolset, trades can be made regarding which technologies are implemented, how these architectures elements are interrelated, and where they are spatial located.

Although, no detailed modeling and simulation was performed for these candidate architectures, the three architectures were qualitatively compared by considering the number of elements and the complexity of their interrelationships.

Qualitative comparisons can provide limited information regarding the effectiveness of the architecture. The conventional architecture has the least number of elements, but the all electric architecture has a reduction in the number of power induced functions representing distribution and transformation. The introduction of high power elements without completely unifying the power type may not yield all potential benefit due to the remaining requirements and redundancies of other power networks.

The benefits of this function-based architecture definition process and toolset lies in its application and interface with modeling and simulation. The ADEN toolset provides the I/O for architecture modeling and simulation. Accuracy in estimating the performance of a given architecture is still subject to proper definition of the induced functions associated with each technology. This chapter highlights the need of a design space which allows any desired architectural change. Rigorous exploration of a complex architecture requires the identification and classification of all induced functions that may be associated with the technologies used in architecture concepts. The simplified examples of aircraft architectures which were generated in this proof of concept scenario illustrate the capability of functions and induced functions to provide flexibility to configure any architecture within the design space.

CHAPTER 10
CONCLUSIONS AND FUTURE WORK

The demands placed on aircraft performance require much innovation on the part of technology developers. With emerging functional requirements, performance constraints, and increasingly stringent operating conditions, the success of an airframer depends on the full utilization of new technologies within the aircraft architecture. Integrated or monolithic architectures are descriptions of the physical elements, their relationships and structure, whose combination is intended to fulfill some function or set of functions. The elements in an integrated architecture are expressly designed and sized for interoperation within the context of the architecture. Traditional complex systems architecture utilizes assumptions which govern the structure and relationships within the architecture.

Revolutionary technologies introduce new relationships that are not provided during evolutionary design. Technologies can only be easily integrated within a previously defined architecture if the boundaries of these elements correspond to the boundary relationships of the technology which it replaces in this architecture. At higher conceptual levels, assumed relationships between technologies and systems do not always allow for easy integration of revolutionary technologies. The architecture of the system as a whole must adapt during technology infusion. In order to provide maximum benefit of a technology it must be sized and implemented within a compatible architecture. Optimal performance at the scope of an individual technology may not provide the best global system performance.

Functions were identified as a guiding element by which new architecture design can emerge. Boundary functions are the only truly generic attribute of the architecture design process. The ultimate purposes and functions of the product are entirely independent of the physical structure used to define the architecture. In order to manage the relationship between function and physical form, functions were classified into two categories: boundary and induced functions. The boundary functions are the ultimate goals of the product or the boundary relationships of this product and the environment. As physical elements are defined to fulfill the boundary functions, new functionality must be provided because of the application of a specific technology. These induced functions enable the application of a specific technology to other architecture functions. By classifying functions in these terms, flexibility was introduced in the relationship between function and physical form. Functions require physical fulfillment, and physical elements in turn induce new functions. The adaptive reconfigurable matrix of alternatives (ARM) was developed to serve as a means of managing this relationship between function and physical definition.

Architectures not only involve elements, but also relationships and structure. Functions provide information regarding the physical elements required in the architecture and some guidance concerning structure. However, complex relationship between elements can also have large impact on the performance and definition of the architecture. These complex relationships can occur when single architecture elements can fulfill multiple functions and when spatial organization impacts the relationships between elements. The architect must not only define physical elements, but also configure their relationships. Standard classes of induced functions were developed in order to enable this redefinition of relationships. More architecture detail was also included in this architecture definition process by allowing the definition and placement of elements in 3-dimensional space.

The Functional Mapping Matrix (FMM) allows for the developing alternative relationships between system elements in terms of power requirements. Standard classes of induced functions were termed power functions. Power functions (power distribution, transformation, and generation) can be induced often within an architecture by multiple architecture elements simultaneously. The interrelation of power users and providers can be configured to accommodate all use-case scenarios defined for the architecture. These relationships govern the power structure of the architecture.

The physical structure of the architecture depends on the attributes of each element based on the requirements which it is fulfilling and the placement of this element in relationships with the others in 3-dimensional space. By allowing the architect to define the spatial integration of these elements, new functional relationships can be addressed and the overall attributes of the architecture can be determined. For an aircraft, spatial relationships of elements have critical impact on the drag attributes of the aircraft as well as on the center of gravity, wing placement, and other contributors to key performance. The integration tool developed allows these decision be made which potentially impact the performance of the product.

These three dimensions of concept definition (function-driven physical element selection, configuration of functional relationships between physical elements, and the spatial integration of the elements) completely define the physical attributes of the architecture. The Architecture Design Environment (ADEN) is the combination of the tools required to both design the functional/physical design space of the architecture and then define architecture alternatives in terms of the these three architecture dimensions. This object oriented toolset provides the flexibility of relationships between functional and physical elements and allows the architecture to define any architecture that is defined within the design space provided by the designers. This flexibility was displayed by the definition of three aircraft architecture concepts: conventional, more electric, and all electric.

Not only does the ADEN toolset allow for the definition of vastly different architectures, it also provides a management of modeling information. Each technology element must be defined by its power relationships, both the power it requires and the power it provides. It can also be characterized by the other variables necessary to size these elements and determine its attributes. With all of this information defined and intended to be provided by an architecture model, output files are generated which provide the I/O for modeling and simulation models. This provides a potential bridge over the gap between conceptual architecture definition and the analysis, by allowing architectures to be easily defined and compared based on physics based models.

Hypothesis Review

In addressing the thesis scope, it has been shown that configuring existing tools within an integrated interface has allowed functions to become the defining element of architecture definition and has provided flexibility in the definition of architectures. This process and toolset, which utilizes functional flexibility, allows for the definition of vastly different architecture concepts. The hypothesis was tested in the generation of widely variant architecture concepts using the developed tools and methods. These concepts exhibited various underlying structures and technology implementations. The connection to modeling and simulation was illustrated through the definition of output files. However, much work still remains in linking this architecture design process to its analytical conclusion and developing processes of architecture trade-offs

The traditional limitations of assumed relationships between predefined systems modules are avoided in the process of building functional chains. Not all assumptions, however, are eliminated in this process. Postulations may be present in the formulation of the control volume for each of the physical elements and the sizing or analysis models which are used to assess the architecture and its constituents. Tacit knowledge will also be involved in the definition of the boundary functions, induced functions, environment relationships, and architecture zone interactions. Nevertheless, this process aids the architect by forcing the explicit definition of the assumptions which govern the interactions between functions and physical embodiments and the tracking of all model interactions. It also allows the relationships between physical elements to be easily changed without breaching modeling constructs.

The grouping of architecture definition tool in the Architecture Design Environment provides means of defining critical levels of detail during architecture definition. The ADEN tools allow for the appropriate interplay between the selection of elements and the definition of structure. These tools also manage the induction of functions based on groupings of architecture decisions. Combining definition tools allows architecture configuration to be adapted during different mission scenarios. Safety and reliability can also be explored through allowing the rerouting of power during specific use-case scenarios.

In order to guide the selection of an architecture scheme, more knowledge must be integrated early in the design process. Modeling and simulation must be performed to capture the physics of the architecture and justify the choice of a specific architecture concept. Each element in the design space must be represented and catalogued in terms function, sizing model, and I/O. To allow these models to interact in the same design space, they must follow standard variable conventions. All of this requires much more up front development of the design space.

The design space becomes much broader with the introduction of new boundary functions, induced functions, and potential physical elements. With technologies, networks, redundancy schemes, mission segments, and spatial installation being managed in a flexible architecture, many decisions associated with detailed design must be made up front. This flexible concept definition and model definition process requires that robust alternative sizing, zone relationship, and architecture analysis models be created for concept exploration. This represents a significant effort and commitment on the part of the conceptual designers in the initial definition of the architecture design space.

The benefit of these tools and processes lies in the flexibility of the design space and the generic nature of its constructs. By modularizing the design space on the basis of functions, architectures of any level of abstraction can be defined and changed. Once the design space has been defined, the functions and physical alternatives are the same for any future architecture. The design space is not predicated on the specific physical relationships of one architecture concept. For redesign or the development of next generation architectures, new functionalities or technologies can be easily added to the design space without breaching the fundamental assumptions of the architecture. Reformulation of the architecture model can also be done with little difficulty using these tools. If knowledge becomes availed during later phases of the design process, it can be easily integrated into the concept space where trades can be made. Models can be added, new induced functions can be defined, new power types can be introduced, and this new information can be used to size more detailed architecture relationships.

In the face of increased performance requirements, regulations, emissions constraints, and fuel prices, aircraft architectures must adapt in order to fully utilize promising technologies. A process which allows for logical and directed definition of candidate architectures is necessary to explore possible ways of providing the correct combination of function, physical element, and structure in the employment of these technologies. This function based process offers a promising approach to architecture design and analysis by providing adaptability in the conceptualization and integration of architectures. With further development in the flexibility of this process and the definition of tools which manage these complex functional and physical relationships in a modeling environment, this architecture framework can further link the decisions made during conceptual, preliminary, and detailed design and facilitate the definition of revolutionary architectures.

Future Research

The work done for this project can be categorized as primarily preparatory. The real benefit of the developmental work outlined in this thesis lies in its implementation in a large scale architectural trade-off exercise. This framework provides the interface in which the architect can have the flexibility to make different levels of decisions and see the direct impact of those decisions through modeling and simulation. With this framework in place the future work enabled by this process includes, first, adding the tools necessary to finish this process and enhance user interaction with the design space, and second, using this interface to make trades in the architecture design space.

Process Development

In order to improve the process and develop the modeling pieces to allow for this guided generation of architectures based on function the following future research should be performed.

-Sizing tools must be defined for each element within the architecture. Tools which provide the attributes and performance requirements of the elements must be created for each architectural element, which can provide attributes (size, weight, operating temperature, etc) and performance requirements based sizing inputs. These models can be represented by response surface equations which can increase the speed of calculation while still providing accurate sizing responses.

-As it stands now, the definition of a concept is guided solely through compatibilities and induced functions. However, once these functions and alternatives have been identified, the definition of multiple instances and redundancies is based on segregation and reliability. Currently, these conversations are being handled by means of indicating failure scenarios and instances. Somehow, this information must be integrated in the concept definition decision making process. Research must be done to leverage information regarding reliability to the configuration of redundancy and segregation scenarios. Similar information could be used in the linking of architecture elements to others in the FMM.

-In order to roll up the effect of individual elements on the architecture as a whole, they must be placed in 3D space and combined to provide the overall geometry of the aircraft. The product must be broken down into zones with attributes that are made to interact, sharing information and attributes in order to provide values required in the sizing of the aircraft (drag polar, overall weight, etc). Models representing these architectural zones and their relationships must be defined and developed for the architecture to facilitate the sizing of the architecture as a whole.

-All zone models must be sized at all mission scenarios, and the aircraft attributes must be determined based on the mission defined in concept definition. A mission analysis routine must be developed which can integrate discrete mission conditions and segments to calculate overall fuel burn. The requirements from each mission section must then be packaged in a way which provides the correct requirements and attributes to the right FMM element map.

-The process and toolset begins with the formulation of boundary functions. Additional tools and processes should be identified which facilitate the development of functions based on a flexible requirements document. The importance of these requirements could be linked to the definition of critical functions and guide the selection specific technologies for these critical functions and requirements.

-The sizing of a dynamic system depends highly on transient behaviors which emerge when time plays a role in the performance of the system. This sizing process was developed with the intent of being used with steady-state sizing models of the technologies. Further research work must be done to include time dependant modeling in the conceptual sizing of aircraft components within this framework and process.

Process Implementation

With the sizing models, zone sizing tools, analysis tools, and integration routines in place this toolset enables architects to make trades with technologies, redundancy scenarios, interrelationships, technology placement, and mission conditions. Once these modeling pieces are in place and trades can be made, research can be done into the process in which effective trade scenarios can be configured. This involves how this process can be most successfully utilized to find and develop the best performing architecture. Depending on the level of flexibility that is to be used, vehicle trade-offs can be made on any number of levels.

-Architectural trades can be explored for a given design space on one of many levels within two general categories: fixed technology trades and overall architecture trades. Fixed technology trades would include situations which include the same set of technologies in an architecture and changes made to the relationships, locations, mission conditions, or combinations of these elements. Flexible technology trades would mean a redefinition of the elements used to fulfill the functions of the aircraft. This can be done while holding locations and mission static, but with flexible FMM. The FMM and functional design space will change with changes in technology selection.

-In the purposes of connecting the efforts of conceptual, preliminary, and detail design, this tool can act as a communication medium and a common interface in which models are integrated with increasingly accurate levels of detail. Along this vein, tools should be included in this interface which can provide detailed schematics that are understandable in multiple levels of conceptualization.

-Once architectures have been developed, methods and tools must also be developed for this toolset which manage the performance results of the models defined by the architecture interface. These outputs must link back to the requirements driving the functional breakdown and provide metrics which can be compared and justify architecture selection. Tools must be developed to interface with this process which recognize performance outputs and provide guidance as to how the architecture needs to be altered to provide for better performance.

-With guidance from the metrics for success and the rules set up in the design space, an automated process for architecture redefinition could be explored. This would allow the computer to explore the design space, eliminating infeasible architectures and exploring positive architectures, and consider and evaluate a larger number of architectural concepts. Logic and guidance would need to be worked into the optimization/search routines and provide a rational approach to this multidimensional architecture design space exploration exercise.

APPENDIX A: REFERENCES

[1] U.S. Department of Transportation, Research and Innovative Technology Administration, Bureau of Transportation Statistics, Office of Airline Information, August, 2007; available at: http://www.bts.gov/oai.

[2] “Environmental Protection: Volume I, Aircraft Noise”, International Standards and Recommended Practices, Annex 16 to the Convention on International Civil Aviation, ICAO, July 2005.

[3] Conner, L. N. M., Boeing Commercial Airplanes, Airport Noise Regulations, Copyright 1995 – 2007; available at: www.boeing.com/commercial/index.html.

[4] Scott Campbell, “Air pollution on high: computer model assesses impact of aviation-related emissions”, MIT Center for Aviation Studies, July 31, 2001.

[5] “European Aeronautics: A Vision for 2020, Meeting Society’s Needs and Winning Global Leadership”, Advisory Council for Aeronautics Research in Europe, Luxembourg: Office for Official Publications of the European Communities, 2001.

[6] David Quick & James Ordo, “More-Electric Engine (MEE) Impact on the Environment”, Presentation at the TEOS Forum, Sept 2007.

[7] “Commercial Jet Fuel Supply: Impact on U.S. Airlines,” Statement of John P. Heimlich, Vice President and Chief Economist of the Air Transport Association of America, Inc., Before the Aviation Subcommittee of the Committee on Transportation and Infrastructure of the U.S. House of Representatives (Feb. 15, 2006).

[8] Lester Failero, Power Optimised Aircraft, TEOS 2006.

[9] Michal A. Dornheim, “Boeing 787 Technology: Massive 787 Electrical System Pressurizes Cabin,” Aviation Week and Space Technology, March 2005.

[10] Utterback, J. M. 1994. Mastering the Dynamics of Innovation: How Companies Can Seize Opportunities in the Face of Technological Change. Boston: Harvard Business School Press.

[11] “TEOS Forum Overview,” Technologies for Energy Optimized Equipment Systems: available at; http://www.teos-forum.com (September 25, 2007).

[12] CY Baldwin, KB Clark, "Design Rules." MIT Press, Cambridge, Mass, 2000.

[13] Sage, A., Lynch, C. "Systems Integration, and Architecting: An Overview of Principles, Practices, and Perspectives," Systems Engineering, 1998.

[14] "decision." Merriam-Webster Online Dictionary. 2004. http://mw1.merriam-webster.com (October 1, 2007).

[15] INCOSE, SE Handbook, Version 2a, 2004.

[16] Maier, M.W., "Architecting principles for system of systems," Systems Engineering, 1998.

[17] Shenhar, A., A New Systems Engineering Taxonomy, Proc. 4th Int. Symp. Natl. Council Syst. Eng., National Council on System Engineering, Vol. 2, 1994, pp. 261.276.

[18] Eisner, H., "RCASSE: Rapid Computer-Aided Systems of Systems (S2) Engineering," Proc. 3rd Int. Symp. Natl. Council Syst. Eng., NCOSE, Vol. 1, 1993, pp. 267.273.

[19] Committee on Advanced Engineering Environments, Aeronautics and Space Engineering Board, Commission on Engineering and Technical System, National Research Council, National Academy of Engineering, "Design in the New Millennium: Advanced Engineering Environments, Phase 2," National Academy Press, Washington, D.C., 2000.

[20] Xu, Q. L., Ong, S. K., Nee, A. Y. C., "Function Based Design Synthesis," Research in Engineering Design, 2006

[21] Dieter, George E., "Engineering Design," 3rd Edition, McGraw Hill, 2000.

[22] Moir I, Seabridge A., "Design and development of aircraft systems." 1st edition, AIAA Education Series, 2004.

[23] Ulrich, K.T., Eppinger, S.D., "Product Design and Development," 3rd edition, New York, NY: McGraw-Hill Inc, 2004.

[24] Pahl, G., Beitz, W., "Engineering Design: A Systematic Approach," 2d English edition, Springer-Verlag, Berlin, 1996.

[25] Crawley, E., de Weck, O., Eppinger, S., Magee, C., Moses, J., Seering, W., Schindall, J., Wallace, D., & Whitney, D., "The Influence of Architecture in Engineering Systems." the MIT Engineering Systems Symposium. Cambridge, MA. March Paper presented at 29-31, 2004. esd.mit.edu/symposium/monograph.

[26] Sage, A.P., Lynch, C.L., "Systems integration and architecting: An overview of principles, practices, and perspectives ," Systems Engineering, 1998.

[27] Rechtin, E., "Art of systems architecting," Spectrum, IEEE, 1992.

[28] Sharman,D.M., Yassine, A.A., "Characterizing Complex Product Architectures," Systems Engineering, 2004.

[29] J. Hartigan, "Clustering Algorithms," Wiley, New York, 1975.

[30] Allen, K. R. and Carlson-Skalak, S., 1998, "Defining Product Architecture During Conceptual Design," ASME Design Engineering Technical Conferences: Design Theory & Methodology Conference, Atlanta, GA, ASME, DETC98/DTM-5650.

[31] Ullman, D.G., "The Mechanical Design Process," 2nd Edition, McGraw-Hill, 1997.

[32] Mavris, Dimitri N., & Daniel DeLaurentis, "Methodology for Examining the Simultaneous Impact of Requirements, Vehicle Characteristics, and Technologies on Military Aircraft Design," 22ndCongress of the International Council on the Aeronautical Sciences (ICAS), Harrogate, England, Aug. 27-31. ICAS-200-1.4.4.

[33] Wagenhals, L., Shin, I., Kim, D., and Levis, A., "C4ISRarchitectures: I. A Structured Analysis Approach for Architecture Design." Systems Engineering, 2000.

[34] Sharman,D.M., Yassine, A.A., "Characterizing Complex Product Architectures," Systems Engineering, 2004.

[35] Kirby, M. R., "A Method for Technology Identification, Evaluation, and Selection in Conceptual and Preliminary Aircraft Design," Ph. D. Thesis, Georgia Institute of Technology, March, 2001.

[36] Gerwin, D., Barrowman, N.J., "An Evaluation of Research on Integrated Product Development," Management Science, 2002.

[37] Sterman, J.D., "All Models Are Wrong, Some Are Useful," System Dynamics Review, 2002.

[38] MCDP 6. Command and Control. Department of the Navy, Headquarters United States Marine Corps, Washington, DC 20380-1775, 1996., Jeffrey R. Cooper, "The Coherent Battlefield—Removing the 'Fog of War.' " Unpublished paper, SRS Technologies, June 1993.

[39] "understanding." Merriam-Webster Online Dictionary. 2004. mw1.merriam-webster.com (October 1, 2007).

[40] Roskam, J., "Airplane Design: Part V Component Weight Estimation," DARcorporation, 2003.

[41] Raymer, Daniel P., "Aircraft Design: A Conceptual Approach," AIAA Education Series, 2006.

[42] Meadows, D. H. Robinson, J. M., "The Electronic Oracle: Computer Models and Social Decisions," System Dynamics Review, 2002.

[43] Tversky, A., Kahneman, D., "Judgment Under Uncertainty: Heuristics and Biases," Science, 1974.

[44] Clark, K.B., Fujimoto, T., "Product development performance: strategy, organization, and management in the world auto industry," Harvard Business School Press Boston, Mass, 1991.

[45] Thomke, S., T. Fujimoto. 2000. The effect of front-loading problem solving on product development performance. Journal of Product Innovation Management 17(2) 128–142.

[46] NASA, "NASA Systems Engineering Handbook," NASA SP-6105, Washington D.C., June1995.

[47] Box, G. E. P., "Science and Statistics," Journal of the American Statistical Association, Vol. 71, No. 356 (Dec., 1976), pp. 791-799.

[48] Burnham, K. P., Anderson, D. R., "Model Selection and Inference: A Practical Information-Theoretic Approach," Springer, New York, NY, 1998

[49] Browning, T. R., "Applying the Design Structure Matrix to System Decomposition and Integration Problems," Engineering Management, 2001.

[50] Forsberg, Kevin, Mooz, Hal, Cotterman, Howard, "Visualizing Project Management: Models and Frameworks for Mastering Complex Systems," John Wiley & Sons, Inc, New York, 1996.

[51] Mukkamala, R., Ramesh, R., “A Process Model for Engineering of Complex Systems Architectures with Complex Subsystem Architecture Reuse in Mind,” The 24th Digital Avionics Systems Conference, 2005. DASC 2005.

[52] McClelland, J .L., Rumelhart, D. E., “Parellel Distributed Processing,” Cambridge, Mass., MIT Press., 1995.

[53] Pahl, G., Beitz, W., "Engineering design : a systematic approach", translated by Arnold Pomerans and Ken Wallace , edited by Ken Wallace, London, Springer-Verlag, 1988.

[54] Suh, N. P., "The principles of design", New York, Oxford University Press, 1990.

[55] Department of Defense Systems Management College, “Systems Engineering Fundamentals,” Defense Acquisition University Press, Fort Belvior, Virginia, January 2001.

[56] “Power Optimised Aircraft,” Liebherr Aerospace, September 30, 2007, http://www.poa-project.com.

[57] INCOSE, SE Handbook, Version 2a, 2004.

[58] Department of Defense Systems Management College, “Systems Engineering Fundamentals,” Defense Acquisition University Press, Fort Belvior, Virginia, January 2001.

[59] Mavris, D. N., Phan, L. L., Garcia, E., ”Formulation and Implementation of an Aircraft – System – Subsystem Interrelationship Model for Technology Evaluation,” 25th International Congress of the Aeronautical Sciences, Hamburg, Germany, 2006.

[60] Air Transport Association, “ATA Specification 100: Manufacturers’ Technical Data.” Air Transport Association of America (www.airlines.org), 1999.

[61] Browning, T. R., .Integrative Mechanisms for Multiteam Integration: Findings from Five Case Studies,. Systems Engineering, Vol. 1, No. 2, 1998, pp. 95.112.

[62] Grady, J. O., Systems Integration, CRC Press, Boca Raton, FL, 1994.

[63] Clarkson, P. J., Simons C., Eckert C. (2001) Predicting change propagation in complex design. In: Proceedings of 13th international conference on design theory and methodology, 9–12 September, Pittsburgh, Pennsylvania.

[64] Hashemian, M, "Design for Adaptability," Ph. D. thesis, Department of Mechanical Engineering, University of Saskatchewan, 2005.

[65] Bryant, C., Stone, R., McAdams, D., Kurtoglu, T., Campbell, M., "Concept Generation from the Functional Basis of Design," International Conference of Engineering Design, Melbourne, Australia, August 15-18, 2005.

[66] Hays, D. C., "Requirements Analysis: From Business Views to Architecture," Prentice Hall, 2003.

[67] Mavris, D., De Tenorio, C., Armstrong, M., "Methodology for Aircraft System Architecture Definition," 46th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, 2008.

[68] Hanke, S., Pufe, S., Maré, J. C., "A Simulation Framework for Aircraft Power Management," 2nd European conference for Aerospace Sciences (EUCASS), 2007.

[69] Çengel, Y. A., Boles, M. A., "Thermodynamics: An Engineering Approach, 4th Edition," McGraw-Hill, Boston, 2002.

[70] Rowell, D., Wormley, E. N., "System Dynamics: An Introduction," Prentice Hall, NJ, 1997.

[71] Scholz, D., Seeckt, K., "Green Freighter Sysems," 46th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, 2008.

[72] Zwicky, F., "Morphological Analysis and Construction", Wiley Inter-science, New York, 1948.

[73] Engler, W.O, Biltgen, P. T., Mavris D. N., "Concept Selection Using an Interactive Reconfigurable Matrix of Alternatives (IRMA)," 45th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, 2007.

[74] Moir,I, Seabridge, A, "Aricraft Systems: Mechanical, Electrical, and Avionics Subsystems Integration," AIAA Education Series, Reston, VA, 2001.

[75] Sharman, D. and A. Yassine, Characterizing Complex Product Architectures, Systems Engineering Journal, Vol. 7, No. 1, 2004.

[76] J. Andersson, "On engineering systems design: A simulation and optimization approach," M. E. thesis, Linköpings Universitet, Linköping, Sweden, 1999.

[77] Faleiro, L., "'Power by Wire' Aricraft – Vision or Myth?" SAE Power Systems Conference, New Orleans, November 7, 2006.

[78] "transport." Random House Unabridged Dictionary, 2006. http://dictionary.reference.com (February 12, 2008).

[79] Lombardo, D. A., "Advanced Aircraft Systems," TAB books, McGraw-Hill Inc, Blue Ridge Summit, PA, 1993.

[80] Van Den Bossche, D., "The A380 Flight Control Electrohydrostatic Actuators: Achievements and Lessons Learnt," 25th International Congress of the Aeronautical Sciences, Hamburg Germany, Sept 5, 2006.

[81] Botten, S., Whitley, C., King, A., "Flight Control Actuation Technology for Next-Generation All-Electric Aircraft," Technology Review Journal — Millennium Issue, Fall/Winter, 2000.

[82] Committee on Air Quality in Passenger Cabins of Commercial Aircraft, "The Airliner Cabin Environment and the Health of Passengers and Crew," National Academy Press, Washington, D.C., 2002.

[83] "Controlling Ice Build Up on Aircraft Wings," Fluent, accessed on February, 15, 2008, http://www.fluent.com.

[84] Lawson, C. P., "Electrically Powered Ice Protection Systems for MALE UAVs – Requirements and Integration Challenges," ICAS, 2006.

[85] Federal Aviation Administration, "Advisory Circular: Aircraft Ice Protection," US Department of Transportation, August 16, 2006.

[86] Fielding, F. P., "Introduction to Aircraft Design," Cambridge University Press, Cambridge, UK, 1999.

[87] "AirBus 380 Boeing 787 Ram Air Turbine RAT," Tronair, Inc, http://www.tronair.com.

[88] Wise, J., Hopkin, V.D., Stager, P., "Verification and Validation of Complex Systems: Human Factors Issues," Springer, Verlag Berlin Heidelberg, 1993.

[89] ARP 4761, "Guidelines And Methods For Conducting The Safety Assessment Process On Civil Airbome Systems And Equipment", 1995 Dec.

[90] Hirtz, J., Stone, R., McAdams, D., Szykman, S., Wood, K., "A functional Basis for Engineering Design: Reconciling and Evolving Previous Efforts," National Institute of Standards and Technology, Technical Note 1447, US Government Printing Office, 2002.

[91] Pahl, G., Beitz, W., "Engineering Design: A Systematic Approach," Design Council, London, 1984.

[92] Hundal, M., "A Systematic Method for Developing Function Structures, Solutions, and Concept Variants," Mech. Mach. Theory, 25(3):243-256, 1990.

[93] Little, A., Wood, K., McAdams, D., "Functional Analysis: A Fundamental Empirical Study for Reverse engineering, Benchmarking and Redesign," Proceedings of DETC2000, DETC2000/DTM-14563, Baltimore MD, 2000.

[94] Altshuller, G. "Creativity as an Exact Science," Gordon and Branch Publishers, Luxembourg, 1984.

APPENDIX B: ATA CHAPTERS

Table 5: ATA Chapters

Chapter 21 21-00 AIR CONDITIONING 21-10 Compression 21-20 Distribution 21-30 Pressurization control 21-40 Heating 21-50 Cooling 21-60 Temperature control 21-70 Moisturization/air contamination	**Chapter 22** 22-00 AUTOFLIGHT 22-10 Autopilot 22-20 Speed-attitude correction 22-30 Autothrottle 22-40 System monitor 22-50 Aerodynamic load alleviating
Chapter 23 23-00 COMMUNICATIONS 23-10 Speech communication 23-20 Data transmission,auto. calling 23-30 Passenger address and ent. 23-40 Interphone 23-50 Audio integrating 23-60 Static discharging 23-70 Audio & video monitoring 23-80 Integrated automatic tuning	**Chapter 24** 24-00 ELECTRICAL POWER 24-10 Generator drive 24-20 AC generation 24-30 DC generation 24-40 External power 24-50 AC electrical load dist. 24-60 DC electrical load dist.
Chapter 25 25-00 EQUIPMENT & FURNISHINGS 25-10 Flight compartment 25-20 Passenger compartment 25-30 Buffet/galley 25-40 Lavatories 25-50 Cargo compartments 25-60 Emergency 25-70 Accessory compartments 25-80 Insulation	**Chapter 26** 26-00 FIRE PROTECTION 26-10 Detection 26-20 Extinguishing 26-30 Explosion suppression

Chapter 27 27-00 FLIGHT CONTROLS 27-10 Aileron & tab 27-20 Rudder & tab 27-30 Elevator & tab 27-40 Horizontal stabilizer 27-50 Flaps 27-60 Spoiler, drag devices, fairings 27-70 Gust lock & damper 27-80 Lift augmenting	**Chapter 28** 28-00 FUEL 28-10 Storage 28-20 Distribution 28-30 Dump 28-40 Indicating
Chapter 29 29-00 HYDRAULIC POWER 29-10 Main 29-20 Auxiliary 29-30 Indicating	**Chapter 30** 30-00 ICE & RAIN PROTECTION 30-10 Airfoil 30-20 Air intakes 30-30 Pitot and static 30-40 Windows, windshields & doors 30-50 Antennas & radomes 30-60 Propellers & rotors 30-70 Water lines 30-80 Detection
Chapter 31 31-00 INDICATING & RECORDING SYS. 31-10 Instrument & control panels 31-20 Independent instruments 31-30 Recorders 31-40 Central computers 31-50 Central warning systems 31-60 Central display systems 31-70 Automatic data reporting systems	**Chapter 32** 32-00 LANDING GEAR 32-10 Main gear & doors 32-20 Nose gear & doors 32-21-01 ACTUATOR, CENTERING 32-30 Extension & retraction 32-40 Wheels & brakes 32-50 Steering 32-60 Position and warning 32-70 Supplementary gear
Chapter 33 33-00 LIGHTS 33-10 General compartment 33-20 Passenger compartments 33-30 Cargo and service compartments 33-40 Exterior 33-50 Emergency lighting	**Chapter 34** 34-00 NAVAGATION 34-10 Flight environment data 34-20 Attitude & direction 34-30 Landing & taxiing aids 34-40 Independent position determining 34-50 Dependent position determining 34-60 Flight management computing 34-70 X-PONDER, MODE S

Chapter 35 35-00 OXYGEN 35-10 Crew 35-20 Passenger 35-30 Portable	**Chapter 38** 38-00 WATER/WASTE 38-10 Potable 38-20 Wash 38-30 Waste disposal 38-40 Air supply
Chapter 51 51-00 STRUCTURE, GENERAL	**Chapter 52** 52-00 DOORS 52-10 Passenger/crew 52-20 Emergency exit 52-30 Cargo 52-40 Service 52-50 Fixed interior 52-60 Entrance stairs 52-70 Door warning 52-80 Landing gear
Chapter 53 53-00 FUSELAGE	**Chapter 54** 54-00 NACELLES/PYLONS 54-10 Nacelle section 54-50 Pylon section
Chapter 55 55-00 HORIZ. & VERT. STABILIZERS 55-10 Horizontal stabilizer or canard 55-20 Elevator 55-30 Vertical stabilizer 55-40 Rudder	**Chapter 56** 56-00 WINDOWS 56-10 Flight compartment 56-20 Passenger compartment 56-30 Door 56-40 Inspection & observation
Chapter 57 57-00 WINGS 57-10 Center wing 57-20 Outer wing 57-30 Wing tip 57-40 Leading edge and leading edge 57-50 Trailing edge and trailing edge 57-60 Ailerons and elevons 57-70 Spoilers	**Chapter 61** 61-00 PROPELLERS 61-10 Propeller assembly 61-20 Controlling 61-25-01 GOVERNOR, PROPELLER 61-30 Braking 61-40 Indicating 61-50 Propulsor duct

Chapter 71 71-00 POWER PLANT 71-10 Cowling 71-20 Mounts 71-30 Fireseals 71-40 Attach fittings 71-50 Electrical harness 71-60 Air intakes 71-70 Engine drains	**Chapter 72** 72-00 ENGINE - TURBINE
Chapter 73 73-00 ENGINE FUEL AND CONTROL 73-10 Distribution 73-15 Divider Flow 73-20 Controlling 73-25 Unit Fuel Control 73-30 Indicating	**Chapter 74** 74-00 IGNITION 74-10 Electrical power supply 74-15-01 Box, Ignition exciter 74-20 Distribution
Chapter 75 75-00 AIR 75-10 Engine anti-icing 75-20 Cooling 75-30 Compressor control 75-35-01 Valve HP & LP Bleed 75-40 Indicating	**Chapter 76** 76-00 ENGINE CONTROLS 76-10 Power control 76-20 Emergency shutdown
Chapter 77 77-00 ENGINE INDICATING 77-10 Power 77-20 Temperature 77-30 Analyzers 77-40 Integrated engine instrument sys.	**Chapter 78** 78-00 EXHAUST 78-10 Collector/nozzle 78-20 Noise suppressor 78-30 Thrust reverser 78-40 Supplementary air
Chapter 79 79-00 OIL 79-10 Storage 79-20 Distribution 79-30 Indicating	**Chapter 80** 80-00 STARTING 80-10 Cranking

APPENDIX C: ARCHITECTURE DESIGN ENVIRONMENT (ADENII)

The Architecture Design Environment (ADEN) was initially based in Excel with a fixed functional breakdown and models defined and executed in spreadsheet operations. However, excel it did not allow for the induction of new functions into the design space, adapt well to changes to the functional breakdown, allow redundant functions, or allow easy redefinition of variables and relationships because of its rigid cell references. Therefore, an object-oriented environment was designed to allow for constant manipulation and alteration of the design space.

ADEN II was based in Visual Basic, utilizing the forms, treeviews, gridviews, and other built in tools to manage and organize information stored in memory. The information generated by the design space generation and concept definition interfaces can be saved as ADEN design space files (adends) and ADEN architecture concept files (adenarch). These are text files which list out all of the array data necessary to completely define the architecture. This allows multiple versions of the design space and architectures to be saved and recalled without changing the structure of the tool.

The final product of the tool is a description of the concept architecture model's I/O in the form a CSV file. This file is intended to act as an interface between the architecture design environment and the modeling and simulation tool which assesses the performance of the architecture generated. More information about the output file is available in Appendix F.

Architecture design is a process of decomposition followed by a process of definition. Many unique architectures can be defined within an architecture design space. Therefore, the interface and files associated with the architecture design space are segregated from the concept definition process and interface. The design space is the backbone upon which an architecture is defined. It includes the definition of the boundary and induced functions of the architecture, and the physical elements which are intended to fulfill these functions. All models representing architecture elements or integration would also be included in the design space. These models must be configured and defined before going into the process of architecting and represent portions of the design space. However, these tools are utilized till later, after architectural concepts have been defined and are to be analyzed. The concept definition portion of the architecture includes the selection of physical alternative, the configuration of the relationships between these alternatives, selection the location of these elements in relationship to each other in physical space, and adapting the architecture to take on different attributes during different mission segments. No architecture can be developed with the concept definition interface which does not lie in the design space defined by the user in the first portion of this process.

Design Space Definition

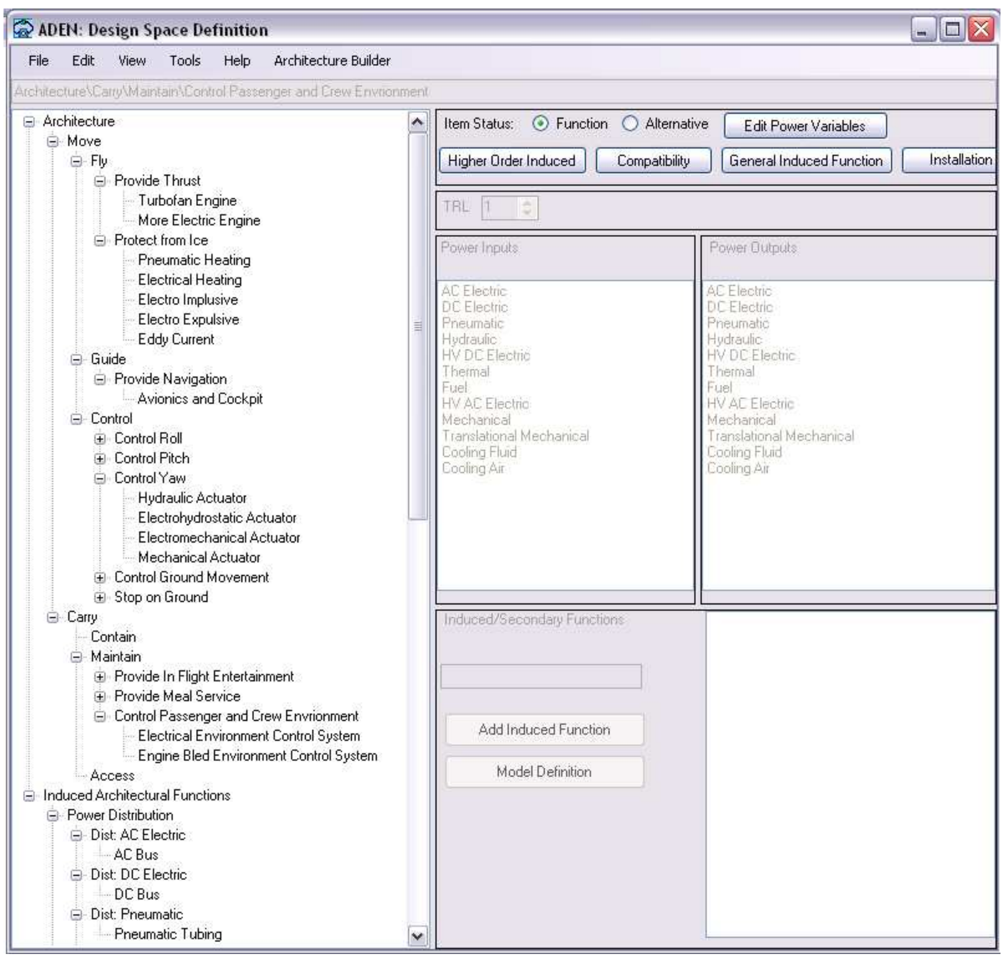


Figure 75: ADEN II Design Space Definition Interface

Function and Alternative Definition

The ADEN II Design Space Definition (ADENDSD) tool's main interface is the treeview seen on the left side of the interface as displayed in Figure 75. Using a treeview provides an intuitive interface within which the architect can configure the architecture. However, the logic behind defining, recalling, and changing this treeview requires recursive algorithms which search the tree and either access the appropriate inform concerning parent and child nodes, or add new nodes or attributes to the tree based on other function and alternative information stored previously.

The default design treeview includes three functions under the architecture heading in the design space. There are no secondary/tertiary functions or installation induced functions. four power variables are loaded default in the architecture. The default treeview is shown in Figure 76.

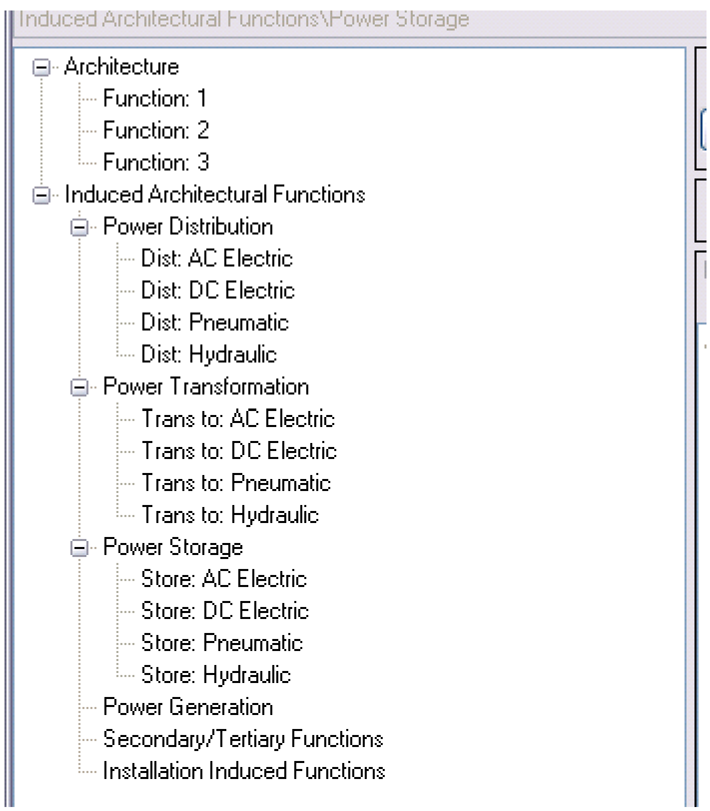


Figure 76: ADENDSD Default Functional Breakdown Treeview

The default power variable groupings are defined in a text file which must be located in the a folder with a program files path (C:\Program Files\System Architect\Tools\ default_pow_var.txt). This text file contains one entry per line in saving data. The first line contains the words `default_power_variables` and is directly followed by the title of the first power variable. The file then contains the variable name representing the type of power distributed by this power variable grouping and the units of this power variable each on an individual line. Power attributes are then listed with their variables. Each power variable grouping is separated by four asterisks. The end of the file is indicated by 8 asterisks. An example of this default power variable text file is seen in Figure 77.

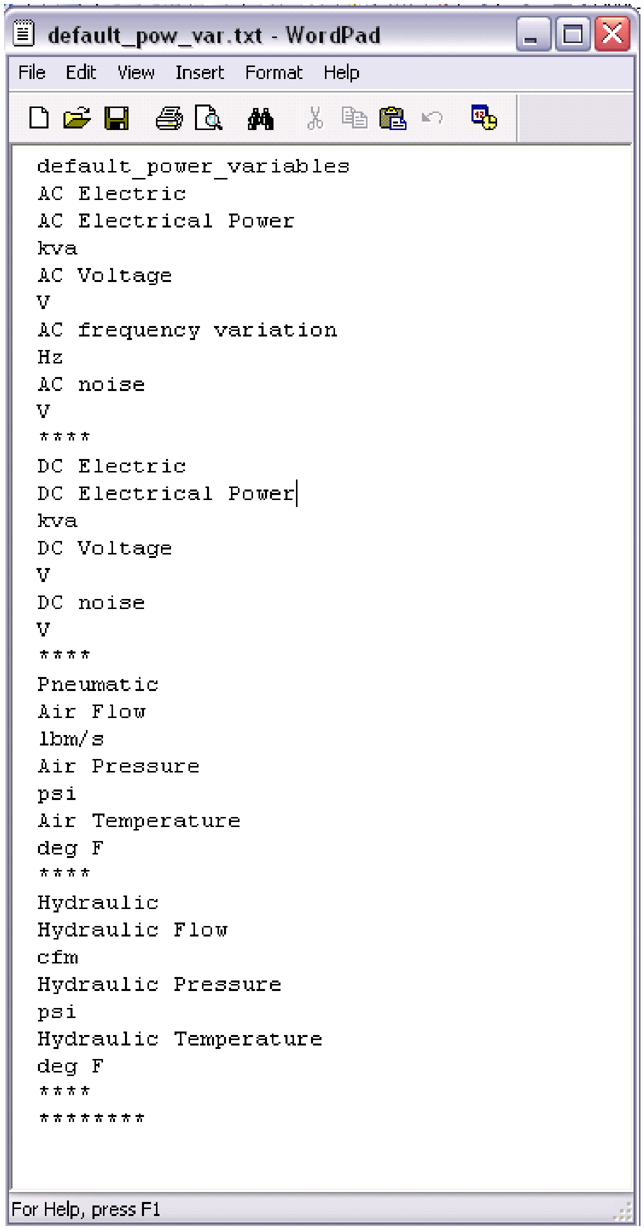


Figure 77: Default Power Variables Input Text File

The definition of each power variable results in the automatic definition of three induced functions within the treeview shown in Figure 76. These functions include a distribution, transformation, and storage power function. The addition and removal of other power variables will be discussed later in this section.

Adding and removing elements in the tree is accomplished by right clicking a given node in the treeview and selecting "Add Sub Branch" or "Remove Sub Branch" in the contextual menu which is displayed. Elements will always be defined as functions unless otherwise toggled. Added branches can be named by again selecting the node and entering the new text.

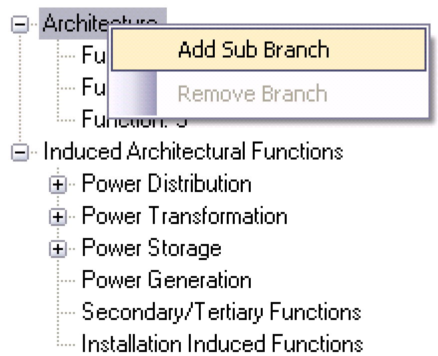


Figure 78: Adding or Removing Functions

This treeview structure allows for the hierarchical definition of the functional breakdown starting from a broad sweeping description of the function to a specific function which can be fulfilled by a single physical element. When a tree node is toggled the text box directly above the functional tree displays which element is selected as displayed in Figure 79.

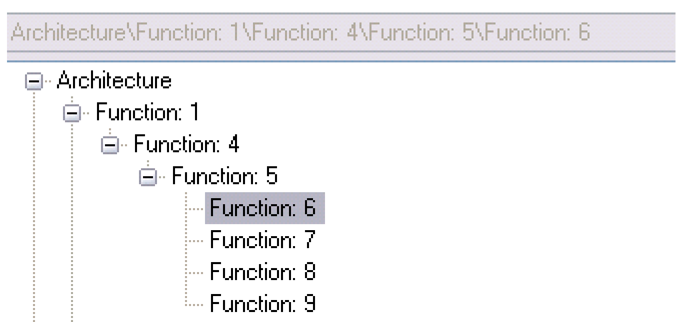


Figure 79: Functional Tree Selected Element

Once the lowest desired level function has been described (e.g. "Function: 5" in the Figure 79), all sub elements of this function must be toggled as alternatives. This is done by changing the item status from function to alternative to the upper right of the treeview.

Figure 80: Item Status Toggle

These items are then handled as “alternatives”, meaning that the architect must now configure the power inputs and outputs of the alternative, define the model by all inputs required for sizing of the model, indicate location of the model within the model repository file, and specify any compatibility or induction characteristics of this element or the combination of this and other elements. All of this is done by interfacing with the panels to the right of the treeview interface shown in Figure 81.

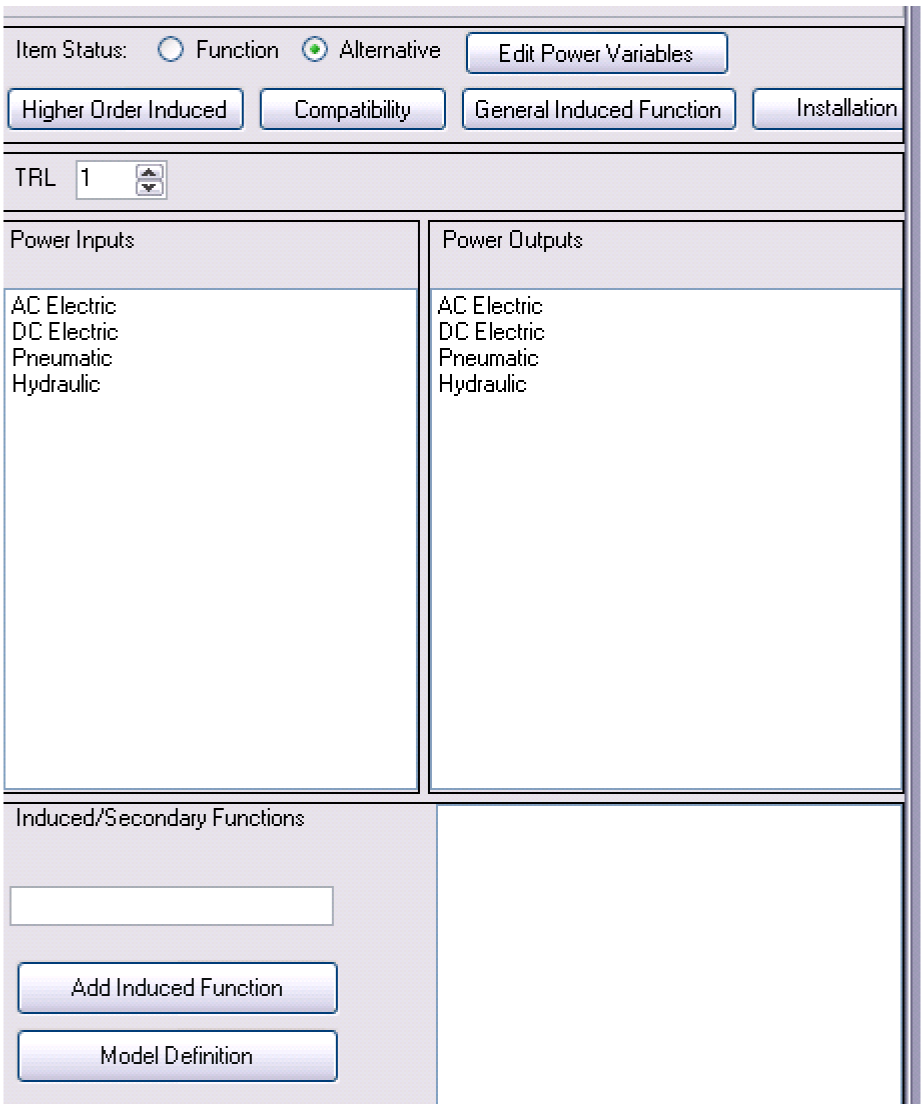


Figure 81: Alternative Characterization Interface

These alternatives must be described by their power inputs and all other information required during sizing. The power variable inputs and outputs can be indicated by simply selecting the appropriate power variables in the textboxes labeled power inputs and power outputs shown in Figure 81.

Each alternative must then be characterized by all other information required during sizing. This can be performed by selecting the "Model Definition" button and accessing the model definition interface. Here the architect configures the I/O for that specific physical element. An example of the model definition interface for a turbofan engine is shown in Figure 82.

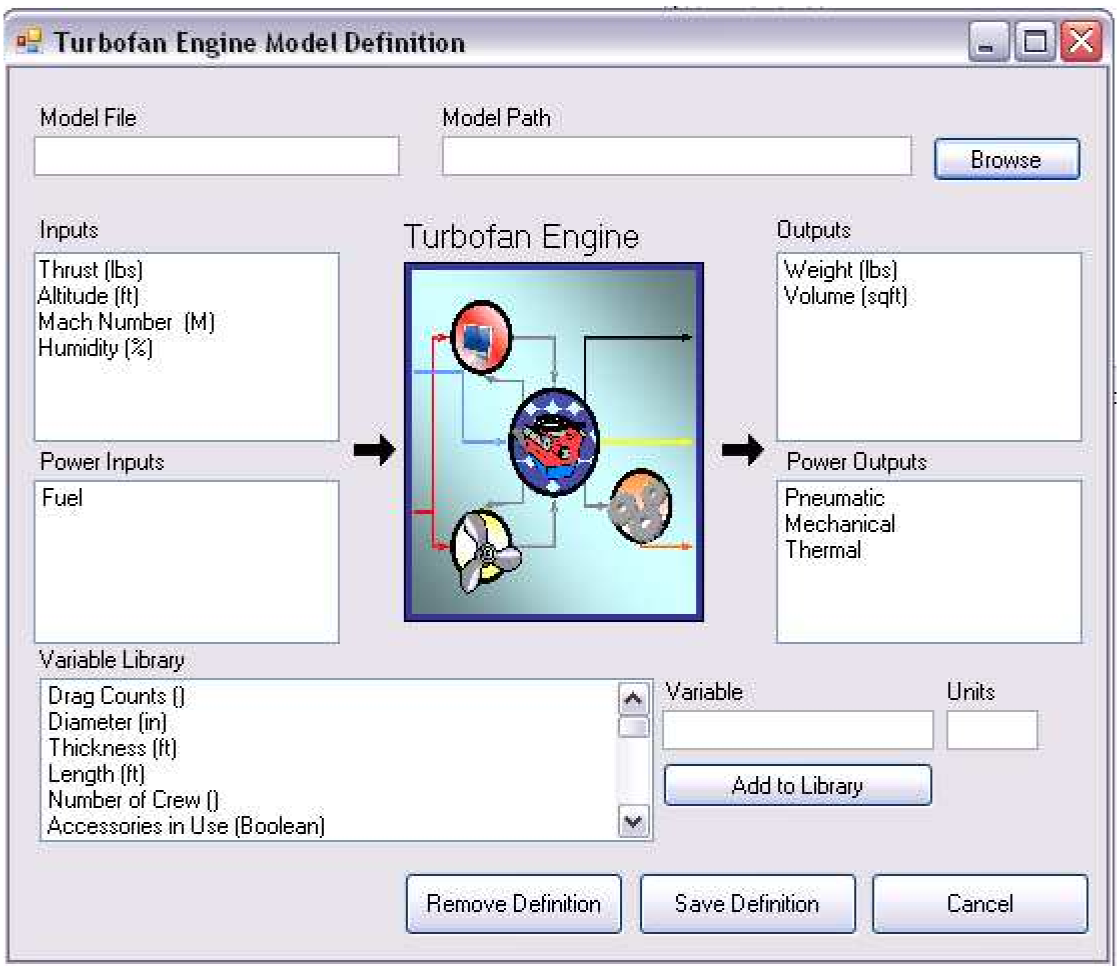


Figure 82: Model Definition Interface for a Turbofan Engine

Here the turbofan is characterized as receiving fuel as an input while potentially providing pneumatic mechanical and thermal power to other architectural elements. Operating conditions govern the attributes and fuel flow rate defined for this engine in providing the necessary thrust and power. These operating conditions are listed in the inputs section to the upper left of this figure and are available from a variable library that is generated by the architect. Variables are added to the library by adding its name and unit in the textboxes at the bottom right of this form and selecting "Add to Library." These actions update the overall variable library and allow this new variable to be used by any element model in the architecture. In order to connect this tool to a modeling and simulation environment, the model and its path must be indicated so the model can be accessed when it is active in the architecture concept.

Power Variables

In order to capture the exchange of information necessary between the architecture elements, new power variables must be defined. Power variables are defined by accessing the power variable definition form through the power variables button. This power variables definition form is shown in Figure 83.

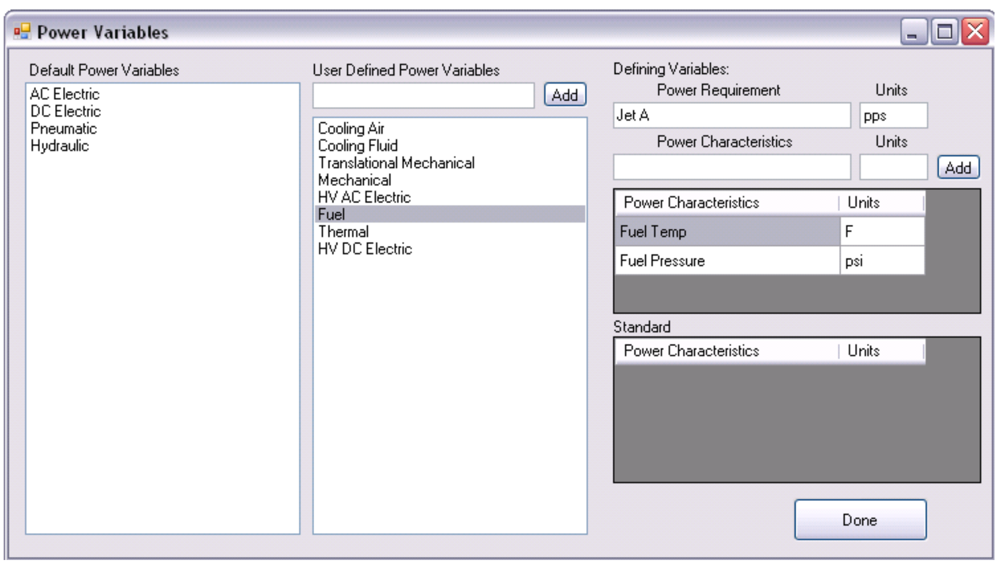


Figure 83: Power Variables Definition Form

Any relationship that manages the passing of energy, mass, information, etc. between elements within the architecture can be treated as a power variable. When a new power variable is to be defined, its title is added in the user defined power variables textbox and selecting add. When this is done, the power requirements and characteristics can be added which will be passed as the variable relationships between the technology models. As many variables can be defined as necessary in order to manage the relationships between variables. The characteristics and requirements of default power variables can be augmented for the design space by changing the default variable file described earlier. As power variables are defined induced functions are added to the functional tree in the distribution, transformation, and storage categories.

Induced Functions

All non-power related induced functions must be individually defined. This is done by assigning these induced functions to specific technologies, zones, or combinations of technologies. Assigning induced functions directly to a technology is performed by adding the name of the function to the textbox in the lower right hand corner of the initial interface and selecting "Add induced function." Induced functions created by multiple physical elements can be done by selecting the "Higher Order Induced", "General Induced", or "Installation Induced" buttons. When an induced function is assigned directly to a technology, this new induced function is listed in the box to the right in Figure 84 when the technology inducing it is selected. A function also appears in the tree under the "Secondary/Tertiary Function" node with the same name. Sub branches and physical alternatives can then be added and defined as described earlier. If an induced function is applied to a specific technology, a unique function will be induced for every element selected in the architecture.

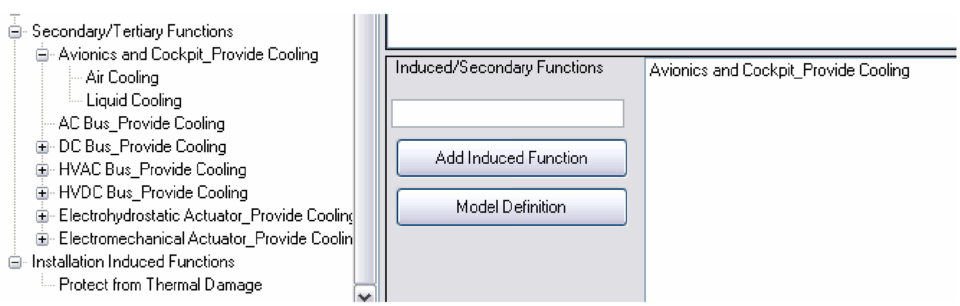


Figure 84: Adding Induced Function for a Technology

Some functions can be induced with the selection of any one of many alternatives. These functions can be defined by selecting the "General Induced Functions" button. With the interface toggled with this button, an induced function can be defined as a result of selecting one of many functions.

Higher order induced functions are functions whose instantiation in the architecture is defined by complex logical conditions. The interface indicates the occurrence of this function when specific elements have been selected and other elements have not been selected. This interface is shown in Figure 85.

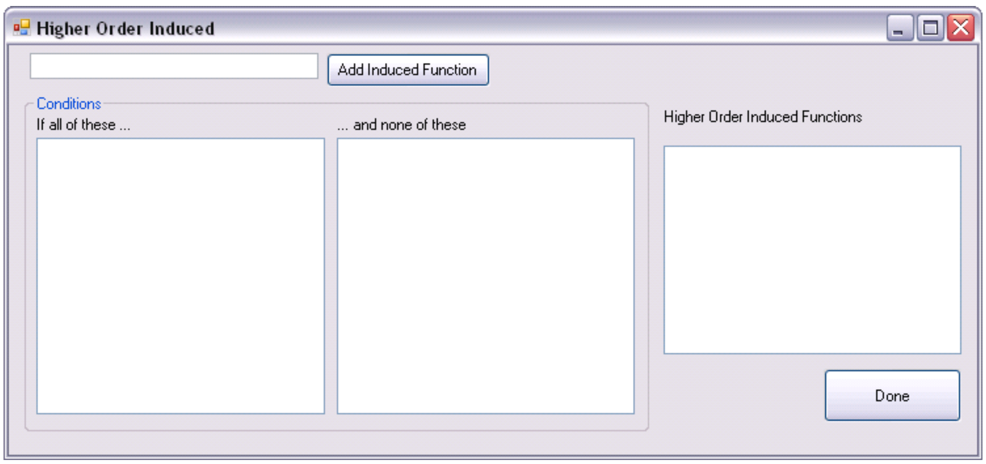


Figure 85: Higher Order Induced Functions Interface

Installation induced functions are the last type of induced functions that can be defined for the design space. Installation induced functions are not driven by elements within specific zones but by elements exhibiting specific input or output power types within the same zone. The example shown in Figure 86 displays the installation induced function to "Protect from Thermal Damage."

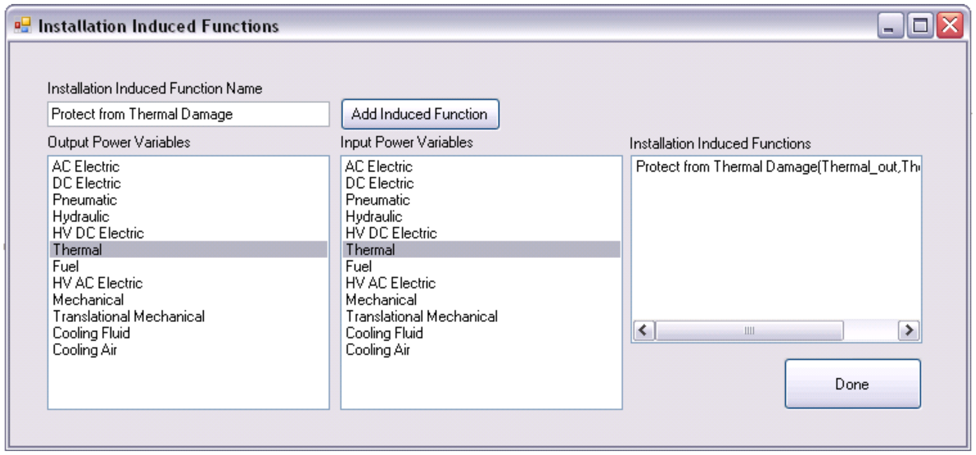


Figure 86: Installation Induced Functions Definition Interface

The "Protect from Thermal Damage" function is induced when a thermal input interacts with a thermal output in the same zone. The attributes of the element which fulfills this thermal management function depend on the amount of thermal energy exchanged in the interaction. Although the function appears in the design space and must be fulfilled by a specific element. This element may not need to exist in the architecture unless the thermal interaction requires it. This function may be induced but will remain dormant or inactive (the model returns weights and volumes of 0 lbs and sqft).

Compatibilities

Compatibilities can be defined between architectural elements. These compatibility conditions can be of two natures: "Incompatible" or "Gotta Have." These compatibility scenarios can also be designated to work bi-directionally or not. Toggling the "Bi-Directional" check box indicates the nature of this relationship. Indicating that one element requires another element to be selected does not require that the second element require the first. The compatibilities definition interface is shown in Figure 87.

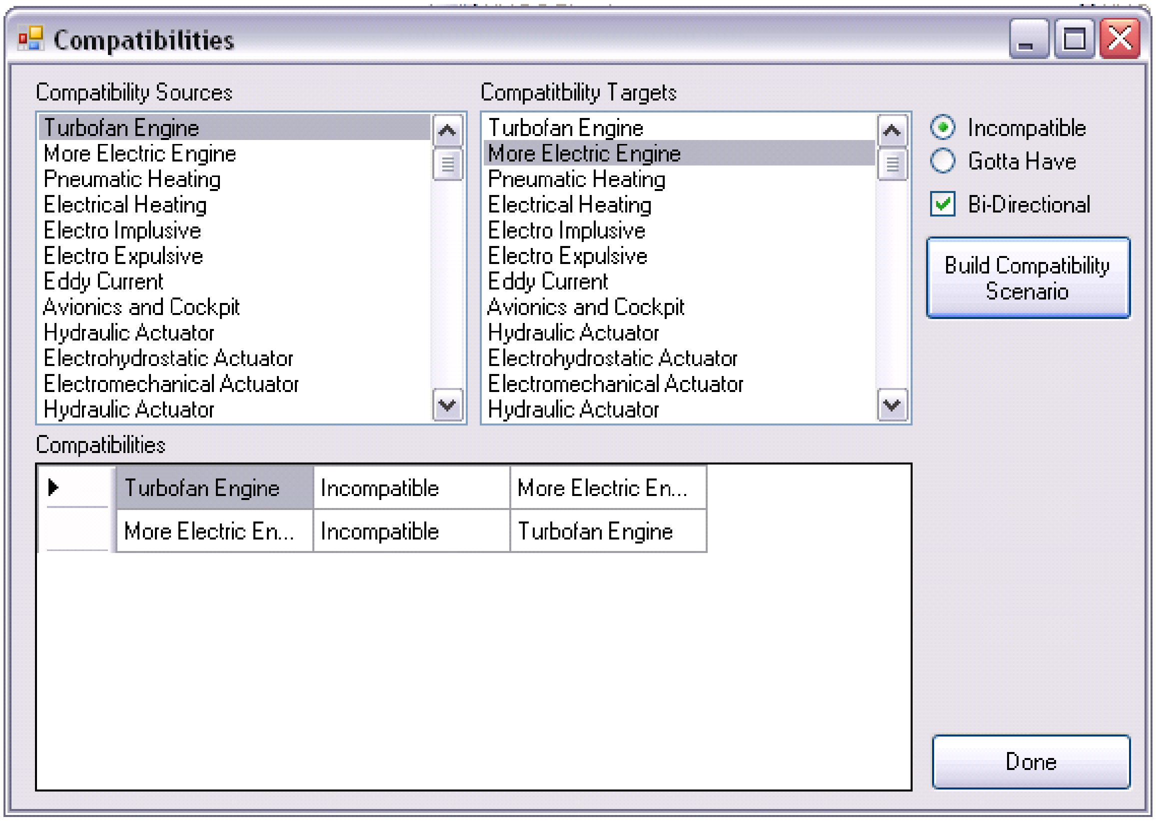


Figure 87: Compatibilities Interface

It should be noted that, these compatibility scenarios currently overrule any redundancy situations used in concept definition. Regardless of how many redundant functions able to be fulfilled by an incompatible element, this element is not selectable in the architecture if it is found incompatible with any other element.

Compatibility scenarios infer some physical relationship which does not allow a given set of elements to be used at the same time in the architecture. Formulating this design space through means of induced functions allows these incompatibility assumptions to be circumvented with the definition of induced functions. Therefore, it is not recommended incompatibility relationships be used in the design space unless the assumptions interrelating these element are maintained with any definition of the architecture. More work could be done to investigate more complex compatibility relationships. These may include the introduction of installation zone based incompatibility constraints or other means of managing the defined relationships that can or cannot occur within the architecture.

Tree Data

All of the information governing this architecture is visible by accessing the "View" menu and selecting "Tree Info" underscored by "Show." This opens a pane which displays gridviews containing data about the tree structure of the architecture, power inputs for each model, power outputs for each model, induced functions, default power variables, user defined power variables, the path of the technology models, compatibility scenarios, and installation induced functions. This panel appears in the lower section of the design space definition interface as seen in Figure 88. This an more information is used to characterize the design space of the architecture and

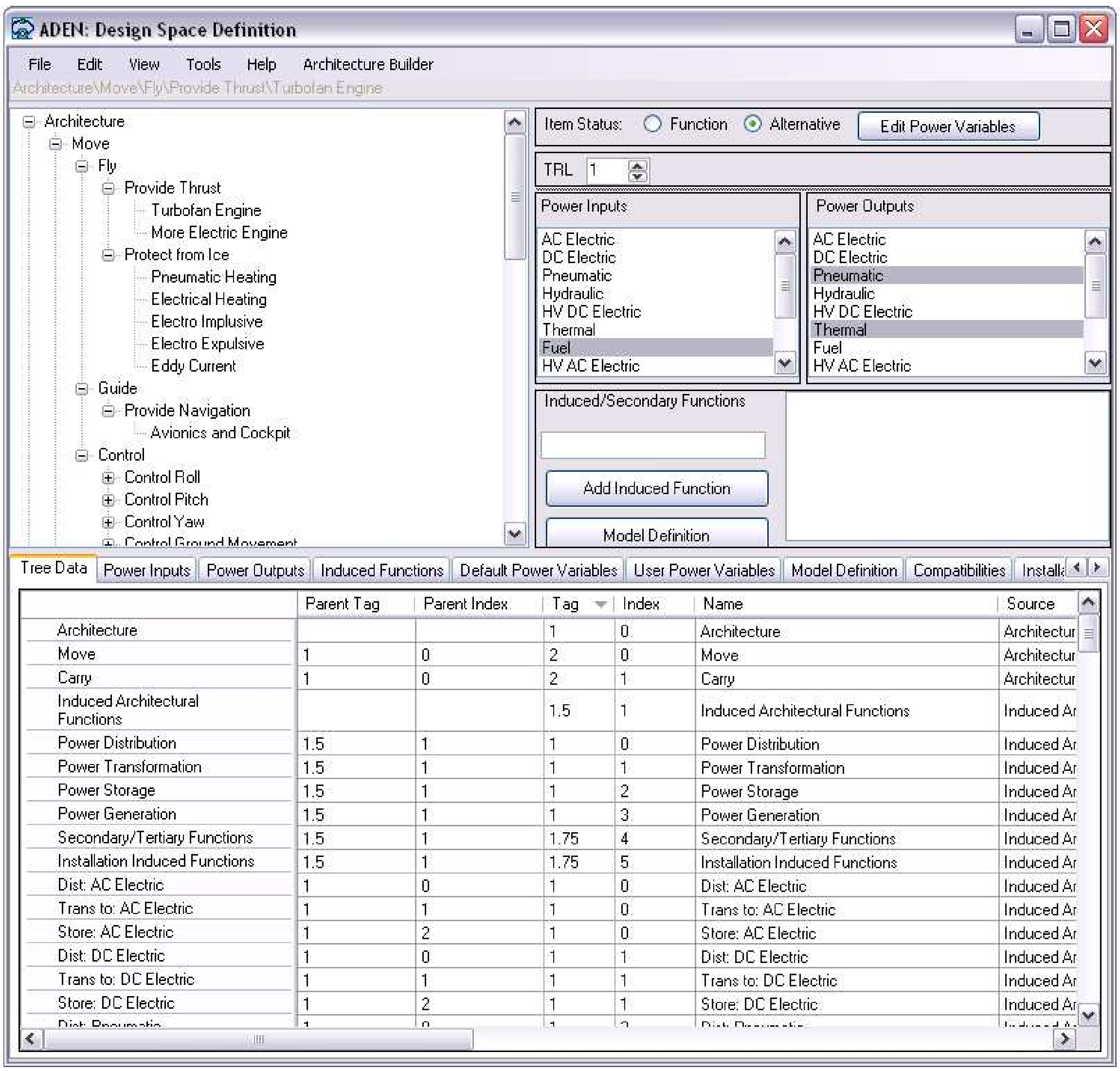


Figure 88: Design Space Definition Interface Displaying Definition Information

With functions, alternatives, power variable groupings, model I/O, induced functions, and compatibility constraints defined, the design space is complete. It can be saved, recalled, and changed and updated without jeopardizing information that has been stored previously. The information describing this design space can be used to begin selecting alternative architectures using the ADEN Concept Definition interface. This is accessed by selecting the "Architecture Builder" file menu item.

Concept Definition

The ADEN II concept definition (ADENCD) interface is composed of four main sections. Three of the sections define the physical attributes of the architecture, and the 4^{th} defines the use-case scenarios that will be used to size the architecture. The adaptive reconfigurable matrix of alternatives (ARM) is situated in the upper left panel of the ADENCD toolset. Here the alternative elements are selected and redundant functions are assigned. The functional mapping matrices (FMM) are located in the lower left panel. Here the relationships between elements are designated and failure scenarios are investigated. The installation definition tool is located in the lower right panel. With this interface each element assigned in the architecture is assigned a zone location based on a previously defined aircraft discretization. The mission definition tool is in the upper right panel. Here the use-case scenarios are defined by operating conditions and the applied FMM. All of these interfaces can be expanded to fill the whole screen by selecting the blue and white icon in their lower corner.

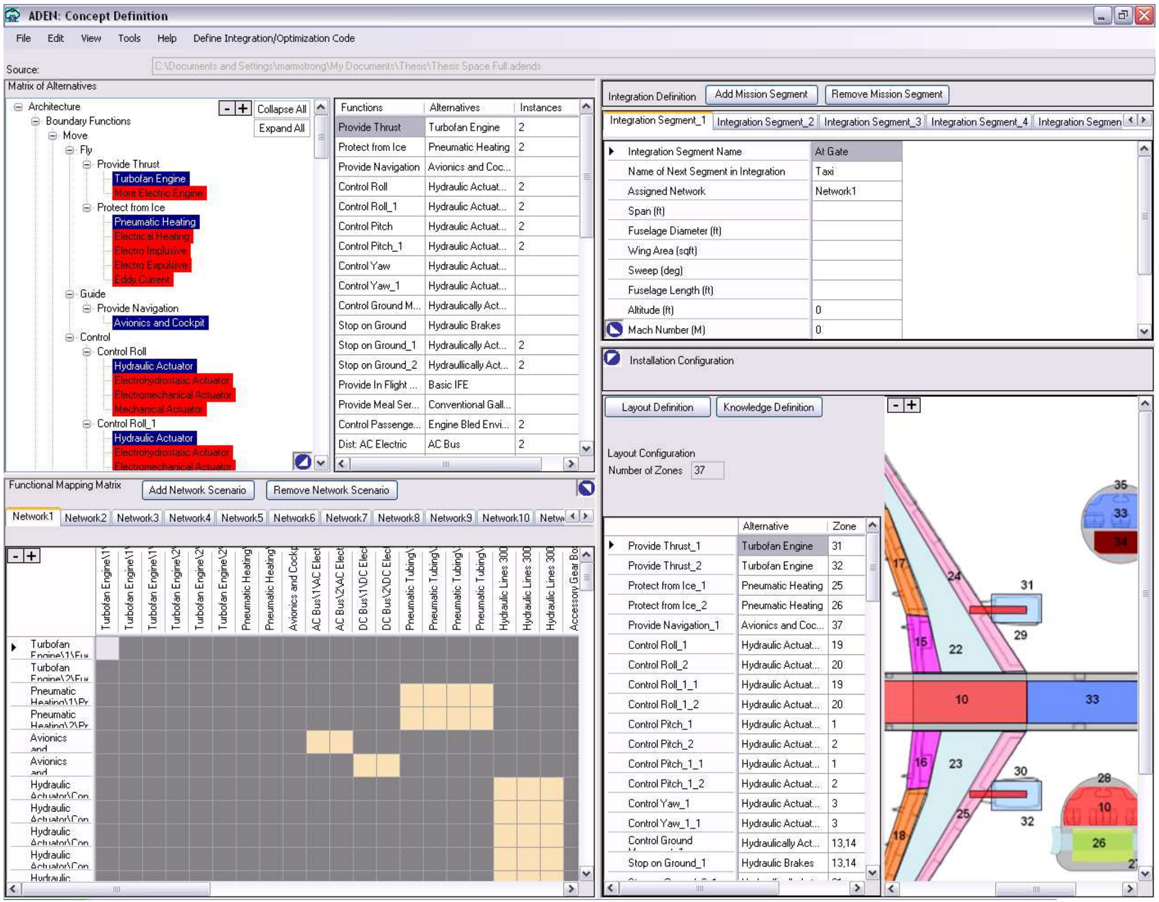


Figure 89: ADEN Concept Definition Interface

ARM Treeview and Table

The adaptive reconfigurable matrix of alternatives (ARM) is based in a treeview which has the same structure as the treeview created during design space definition. Alternatives are selected with this tool to fulfill each function by clicking on this node. When an alternative is toggled this node turns blue and all other elements which are rendered unselectable by this choice turn red. The selected elements appear in the table to the right of the tree. If multiple instances are to be used in the fulfillment of this function, this is can be indicated in the third column of the table.

The ADENCD interface also allows redundant functions to be defined. If the function to be duplicated is selected and right clicked, a context menu appears in which the architect can indicate if a redundancy is to be defined. Selecting "Add Redundancy" inserts another branch in the treeview following the initial function which has the same title as the duplicated function. This redundant function also appears in the table.

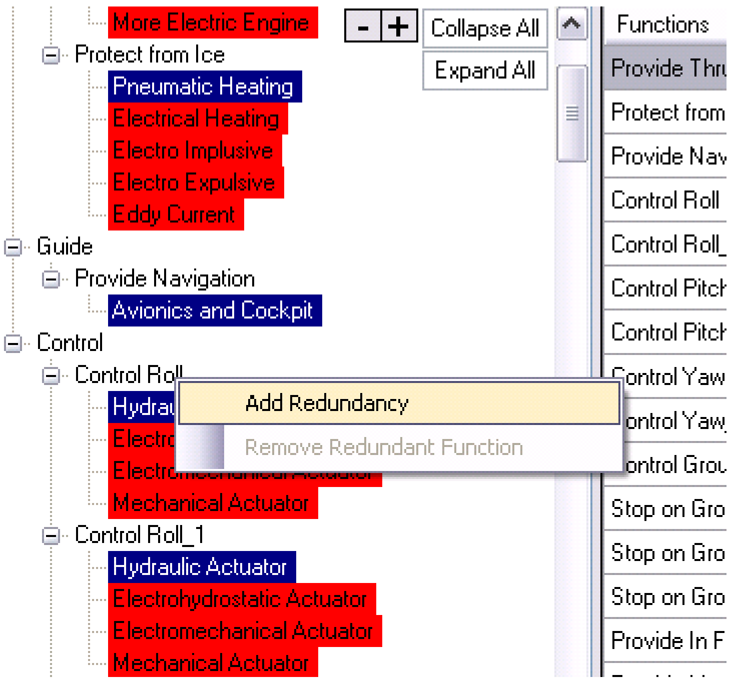


Figure 90: Adding Redundancies to the ADEN II

As elements which require different power types are selected functions for each power related induced function are highlighted. This indicates that some element must be selected to fulfill requirements upstream. When non-power related induced functions are introduced into the design space through other architecture choices, these functions appear at bottom of the tree under branch names "Secondary/Tertiary Functions" or "Installation Induce Functions". The physical elements selected to fulfill these functions can be interfaced with in the same manner as described earlier for boundary functions.

Functional Mapping Matrix (FMM)

As physical elements are selected to fulfill the various functions, all power variables associated with these elements are built into the FMM. The FMM lists all of the power inputs of this element on the left hand column and all of the power outputs in a row along the top of this matrix. The cells in the matrix indicate where elements can receive the power necessary to operate. Tan cells indicate where connections can occur, gray cells indicate where connections cannot occur, and black cells indicate where the source and user in the column and row are the same element and power type. Multiple FMMs can be defined in order redirect requirement mappings for each use-case scenarios. This is done by selecting "Add" or "Remove Network Scenario."

Relationships can be designated as fixed or flexible as explained in the body of this text. This is done by right clicking on an input or output element and indicating if this element has fixed or flexible relationships as shown in Figure 91.

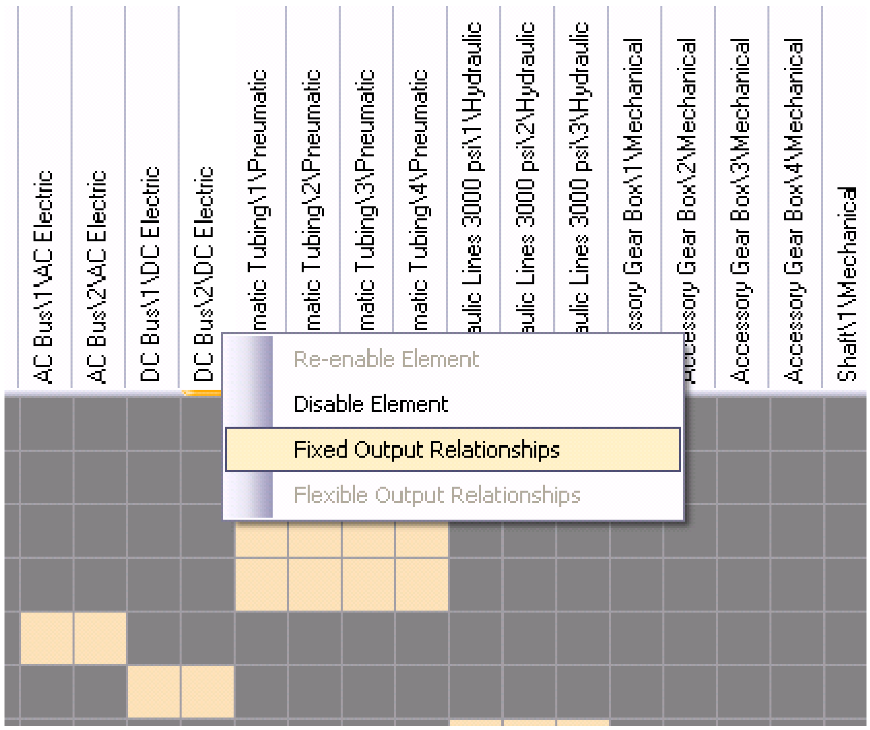


Figure 91: FMM Element Relationships Classification

Failure scenarios can be configured in the FMM. By disabling element through the same context menu seen in Figure 91, the architect can assess whether the proper elements have been included in the architecture to allow all critical functions to be performed. When a elements are disabled, all columns cells of these failed elements are labeled in red. This indicates that these relationships are not valid. Elements which have relationships with the failed elements which are fixed are in turn disabled.

Installation/Layout Configuration

The image seen at the right side of the interface as shown in Figure 92 was generated external to this toolset and acts as a reference for the architecture in placing the architectural elements.

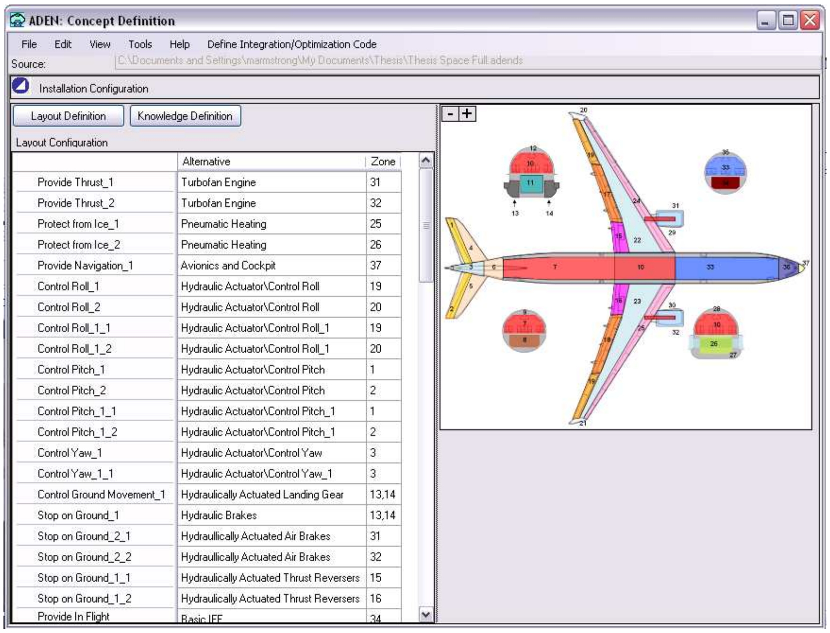


Figure 92: ADEN II Concept Definition Installation/Layout Definition Interface

The locations of the elements of the architecture effect how the architecture is to be sized. These effects must be quantified in a zone integrating tool which provides attributes required by the technologies themselves and by the overall sizing and analysis tools. The characteristics of the layout sizing tool can be managed by selecting “Layout Definition.” In this form the architect indicates the name of the layout, the path of the image used to guide the layout, and any distribution attribute variables which will be calculated by this interface. This layout definition form is shown in Figure 93.

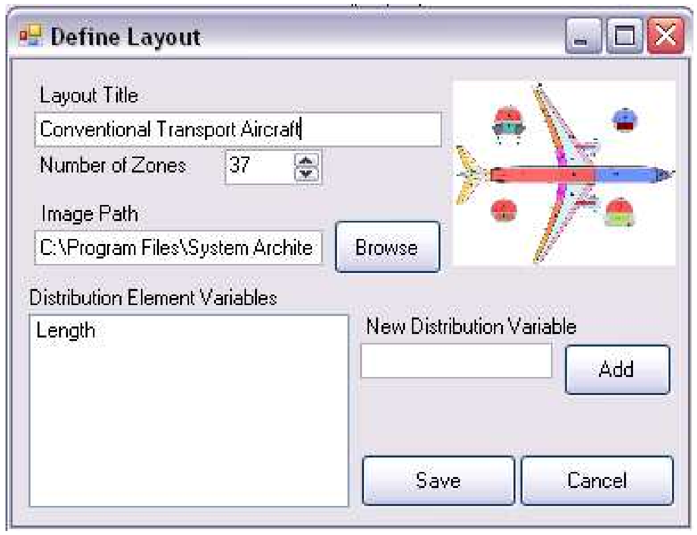


Figure 93: Defining the Installation Layout Reference Image, Variables, and Zone Count

Other layout model zone attributes and power variables must also be configured. These variables are necessary for the sizing of specific elements and the sizing and synthesis of the entire aircraft (zone geometry, weight, temperature, etc). Depending on the boundaries of the architecture design space, power variables may also interact between the zones and the elements assigned to it. The user can configure these relationships with the interface displayed in Figure 94.

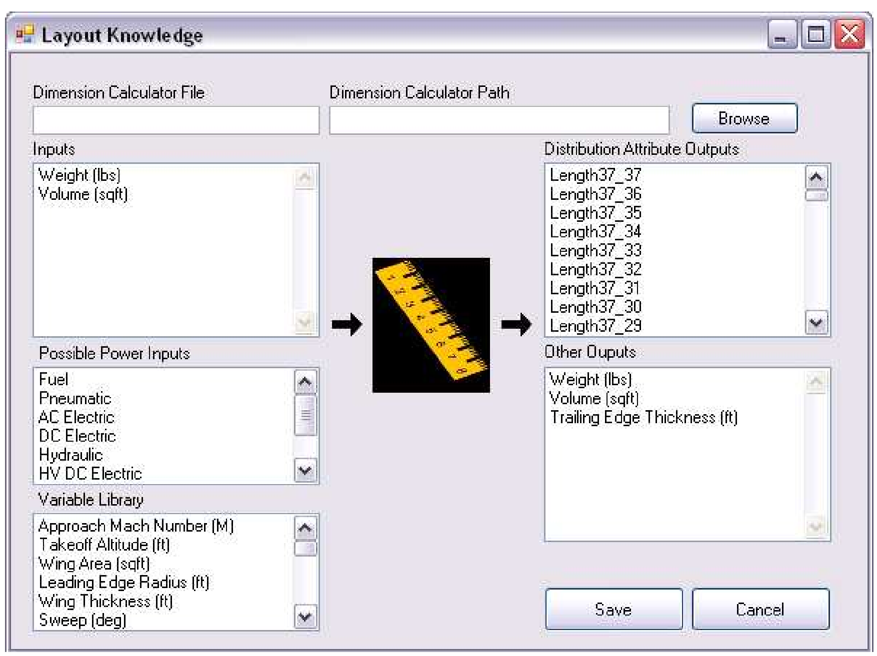


Figure 94: Zone Attribute Modeling I/O

This interface allows the user to define all variables which are to be provided by a zone attribute calculation tool and the inputs required to generate these variables. The actual tool generating this I/O must be defined externally and integrated following the information generated in this toolset.

Integration Interface, power variables, variable list

The last model which must be integrated with the ADEN toolset is one which provides the overall attributes of the aircraft and calculates total performance. This is called the integration element. Integration includes sizing and well as any analysis that is to be done on the architecture generated by the ADENCD. These tools resize the aircraft in terms of overall attributes (e.g. wing area, thrust required, and span). The architect must indicate which variables are being calculated by the model integration tool and which are user inputs which are assigned by the user. These user defined variables and constraints are entered for each use-case scenario during mission definition. The integration I/O manager interface can be seen in Figure 95.

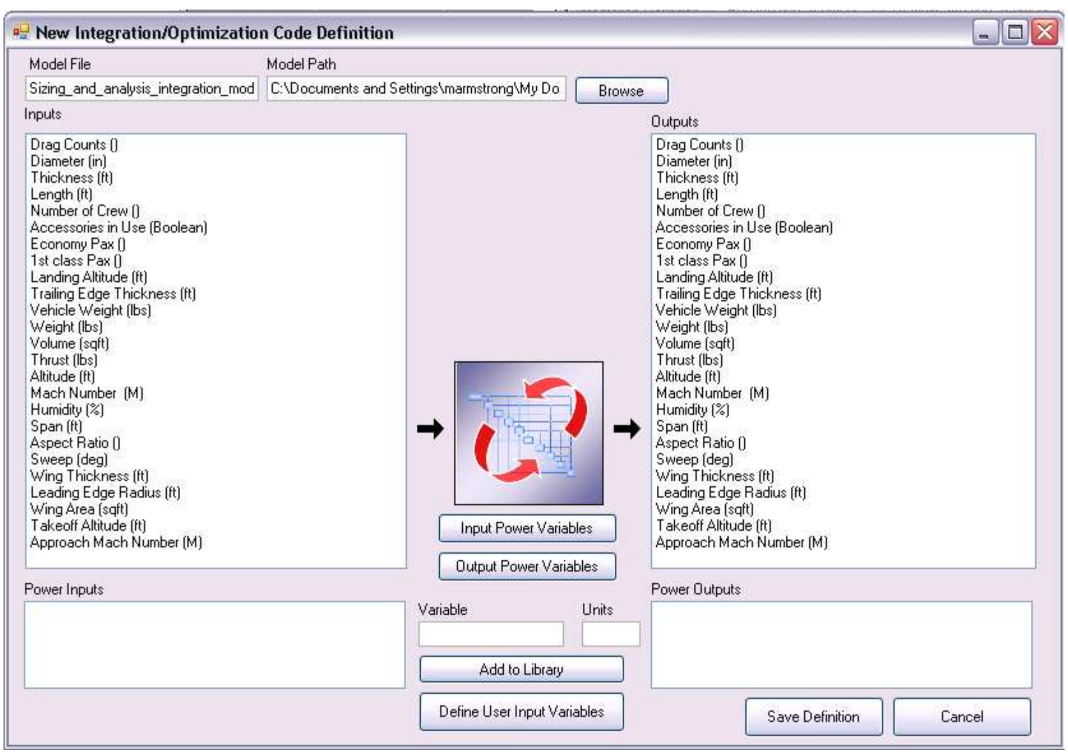


Figure 95: Architecture Sizing and Analysis Integration Interface

Each use-case scenario is described by the boundary conditions in which it operates and the network configuration it uses in its operation. This information is accessible in the upper right pane of ADENCD. Each tab of this interface represents a different use-case or mission segment. When the variables are defined the cells in which represent variables that must be defined before in order for the architecture to be sized. The gray cells indicate variables which are automatically calculated by the sizing and analysis routines.

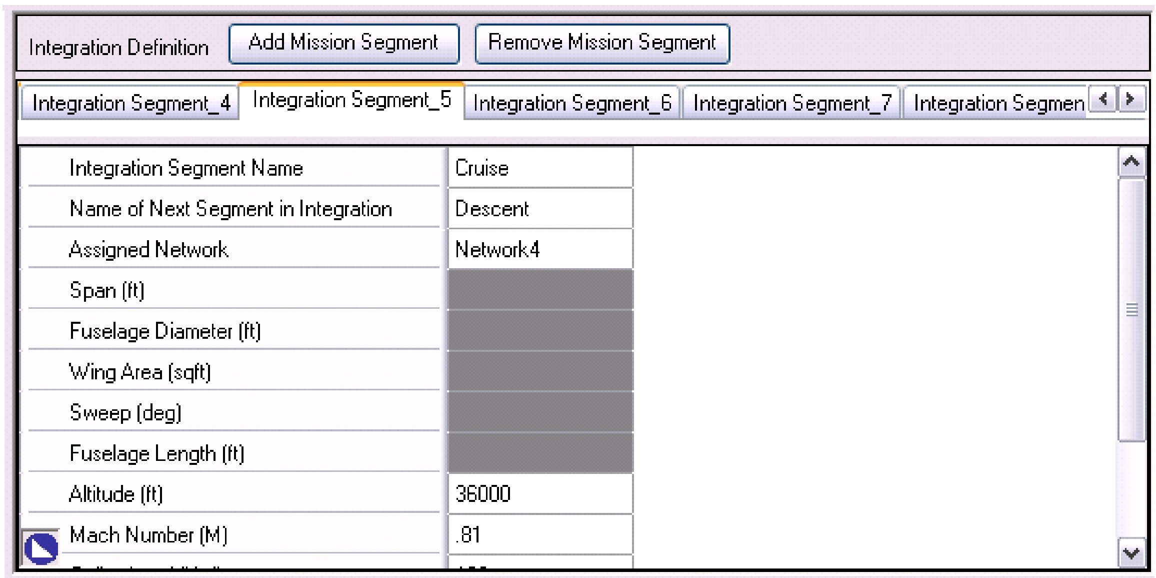


Figure 96: Mission Configuration Interface

The output of the ADEN toolset is a file detailing all of the models which are to be used in the sizing of this architecture, and a mapping of all variables interacting between the models for each use-case scenario and mission segment. This output file is discussed in Appendix F.

APPENDIX D: FUNCTIONAL CATEGORIES

Comparison of typical functions as defined by multiple authors in terms of the functional classifications introduced in this paper [90], [91], [92], [93], [94].

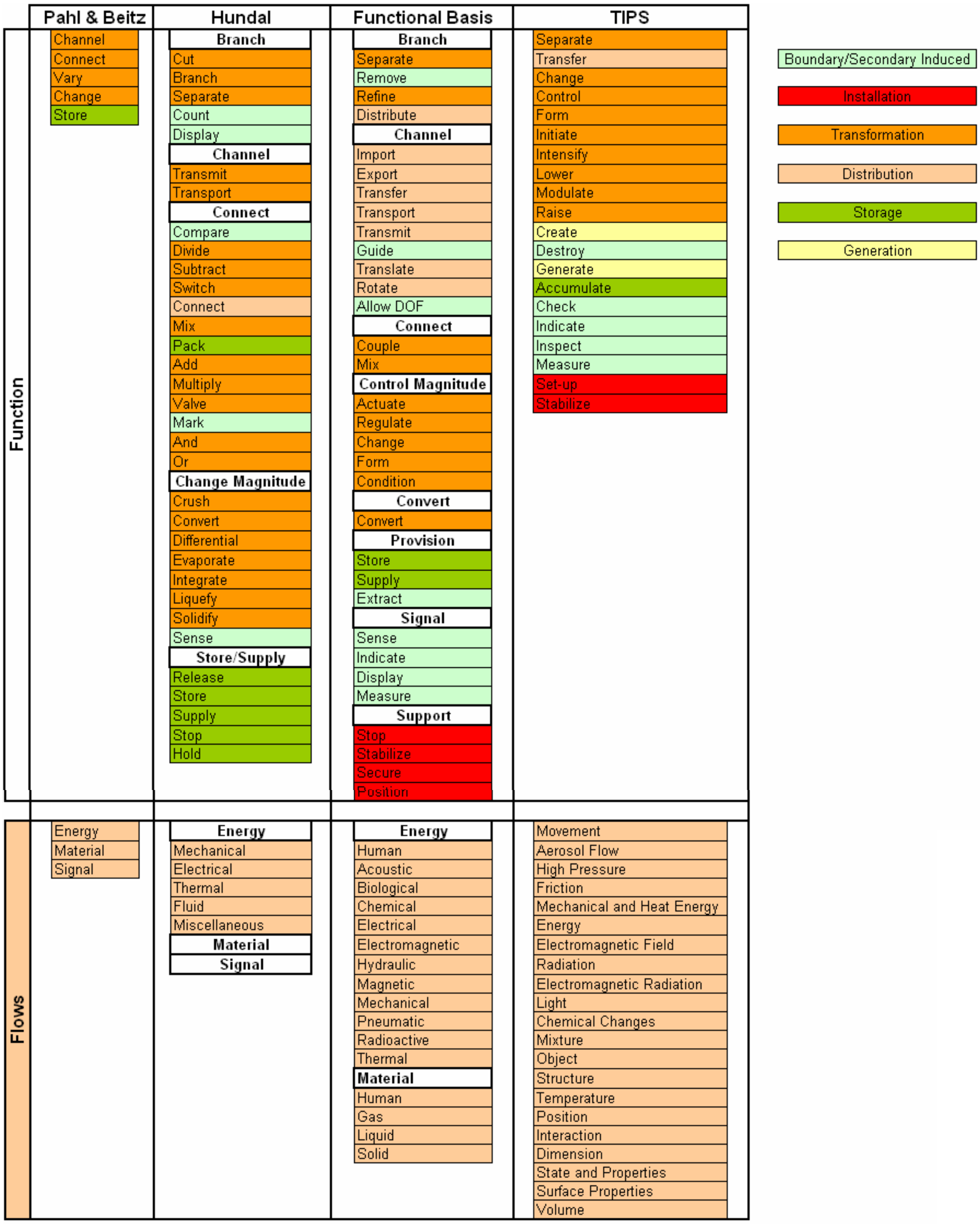


Figure 97: Comparison of Functional Syntax Definitions

Boundary/Secondary Induced	Installation	Transformation	Distribution	Storage	Generation
Count	Stop	Channel	Connect	Release	Create
Display	Stabilize	Connect	Distribute	Store	Generate
Compare	Secure	Vary	Import	Supply	
Sense	Position	Change	Export	Stop	
Remove	Set-up	Cut	Transfer	Hold	
Guide	Stabilize	Branch	Transport	Accumulate	
Allow DOF		Separate	Transmit		
Extract		Transmit	Translate		
Sense		Transport	Rotate		
Indicate		Divide	Transfer		
Display		Subtract	**Flows**		
Measure		Switch			
Destroy		Mix			
Check		Add			
Indicate		Multiply			
Inspect		Valve			
Measure		And			
		Or			
		Crush			
		Convert			
		Differential			
		Evaporate			
		Integrate			
		Liquefy			
		Solidify			

Figure 98: Boundary/Induced Function Categorization of Functions from Figure 97

APPENDIX E: INCOMPATIBILITIES TOOL

With every incompatibility scenario imposed on the morphologic matrix, the design space shrinks to exclude all incompatible scenarios. The typical tools used at the ASDL use incompatibilities to guide the selection of elements defining a system. The tool developed here uses Equation 2 to calculate the actual number of alternatives with every incompatibility scenario.

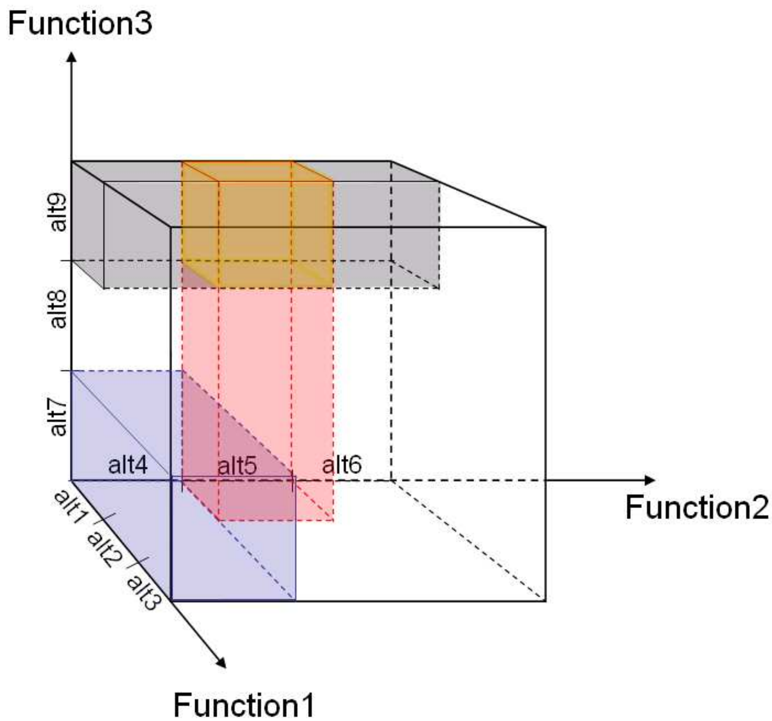


Figure 99: Reduction of Design Space Due to Incompatibilities

Consider the design space in Figure 99. This design space is composed of three functions labeled 1, 2, and 3, and three alternatives for each functions labeled 1 through 9. The total number of possible combination of elements without compatibilities is 27 ([alt1,alt4,alt7], [alt2,alt4,alt7], [alt3,alt4,alt7], [alt1,alt5,alt7], [alt2,alt5,alt7], [alt3,alt5,alt7], [alt1,alt6,alt7], [alt2,alt6,alt7], [alt3,alt6,alt7], [alt1,alt4,alt8], [alt2,alt4,alt8], [alt3,alt4,alt8], [alt1,alt5,alt8], [alt2,alt5,alt8], [alt3,alt5,alt8], [alt1,alt6,alt8], [alt2,alt6,alt8], [alt3,alt6,alt8], [alt1,alt4,alt9], [alt2,alt4,alt9], [alt3,alt4,alt9], [alt1,alt5,alt9], [alt2,alt5,alt9], [alt3,alt5,alt9], [alt1,alt6,alt9], [alt2,alt6,alt9], [alt3,alt6,alt9]).

Applying one incompatibility in the 3D design space removes a line of alternatives (3 total alternatives removed with 1 incompatibility). If this was a 4 dimensional design space a plane would be removed; if the space were a 5 dimensional a volume would be removed; and so on. In Figure 99 there are 3 incompatibilities. If each incompatibility removes 3 elements from the design space the space would be reduced from 27 to 18 total alternative combinations. However, because of intersections between the incompatibility scenarios, one element has been counted twice in the tally of removed elements. Intersections occur when two incompatibilities involve one common function alternative pair and another function alternative with a dissimilar function. The space occupied by both of these incompatibilities should only be tallied once to be removed from the total number of possible alternatives count. In the case of the design space in Figure 99, the total number of feasible alternatives is 19.

An Excel based tool was created to tally the overall number of feasible alternative combinations. This tool is displayed in Figure 100.

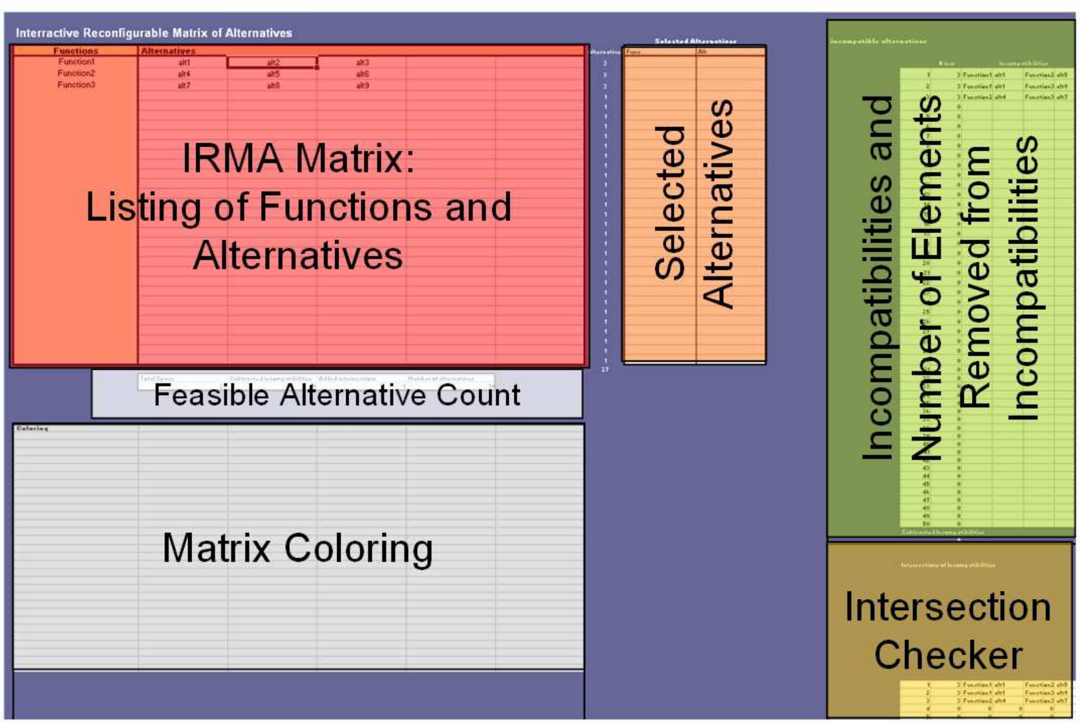


Figure 100: Incompatibilities Tool

This tool includes a listing of all of the functions and alternatives associated with the design space in the section at the upper-left. All incompatibility scenarios are also listed to the right of the IRMA matrix interface. When an alternative is selected, it is indicated in this column and, based on incompatibility relationships, the number of elements to be removed from the design space is calculated and subtracted from the feasible alternative count below the matrix. If there are intersections between the matrix incompatibilities, these are calculated and added back to the count. The coloring section below the matrix provides information which changes the color the matrix cells indicating whether these elements are unselected (white), selected (green), or not selectable (pink).

The design space from Figure 99 was input into the incompatibilities tool as shown in Figure 101. With the incompatibilities and intersections as shown in the figure the number of alternatives is 19.

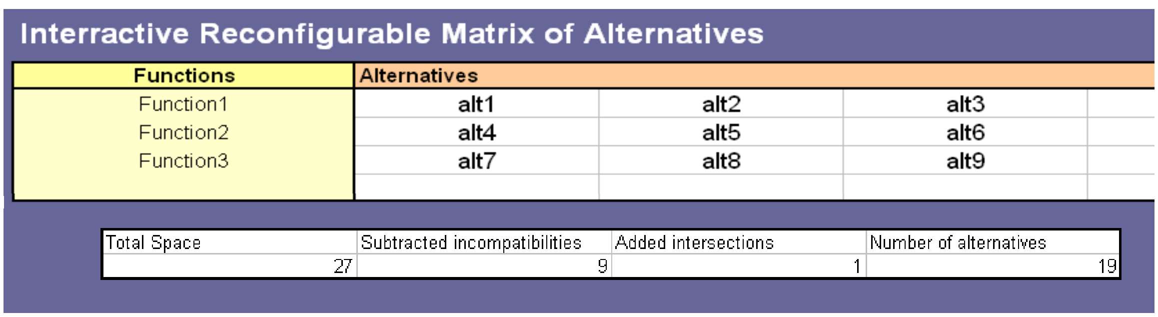


Figure 101: Three function/alternative design space

These numbers change as alternatives are selected. When alt1 is selected to fulfill function1, the incompatibilities show that alt5 and alt9 are now not selectable due to incompatibilities and due to the incompatibility between alt4 and alt7 there are now only 3 feasible alternatives with alt1 selected. This is displayed in Figure 102.

Interractive Reconfigurable Matrix of Alternatives

Functions	Alternatives		
Function1	alt1	alt2	alt3
Function2	alt4	alt5	alt6
Function3	alt7	alt8	alt9

Total Space	Subtracted incompatibilities	Added intersections	Number of alternatives
4	1	0	3

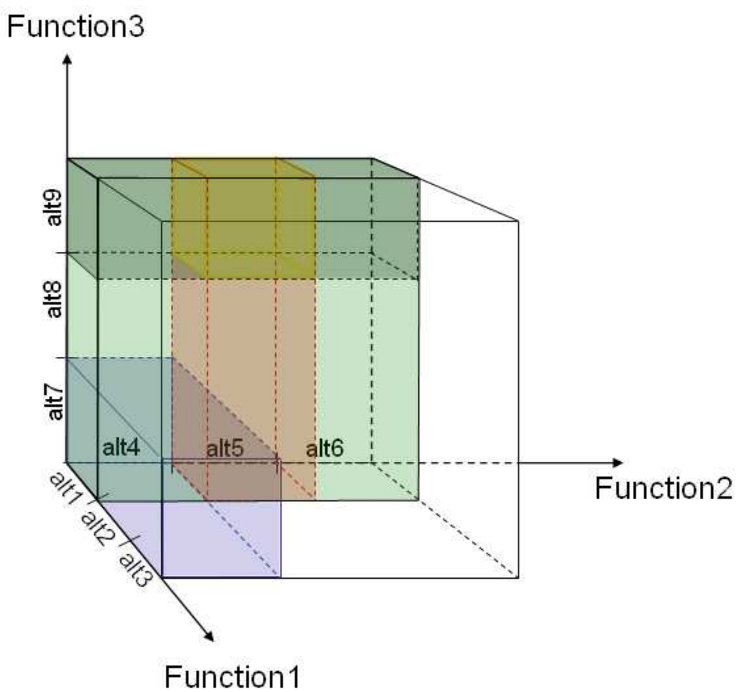


Figure 102: Three function/alternative design space with alt1 selected

With alt1 selected the combinations [alt1 ,alt6 ,alt7], [alt1 ,alt4 ,alt8], and [alt1 ,alt6 ,alt8] are feasible. These elements are green and compatible in Figure 102.

APPENDIX F: INSTALLATION LOCATIONS

This appendix contains the aircraft zone breakdown used for this sizing process and a listing of the zone assignments for each technology defined for the conventional and more electric aircraft examples. A conventional aircraft layout was defined with 38 zones, providing a framework wherein the technologies chosen in the ARM tool can be 3-dimensionally laid out in preparation for sizing.

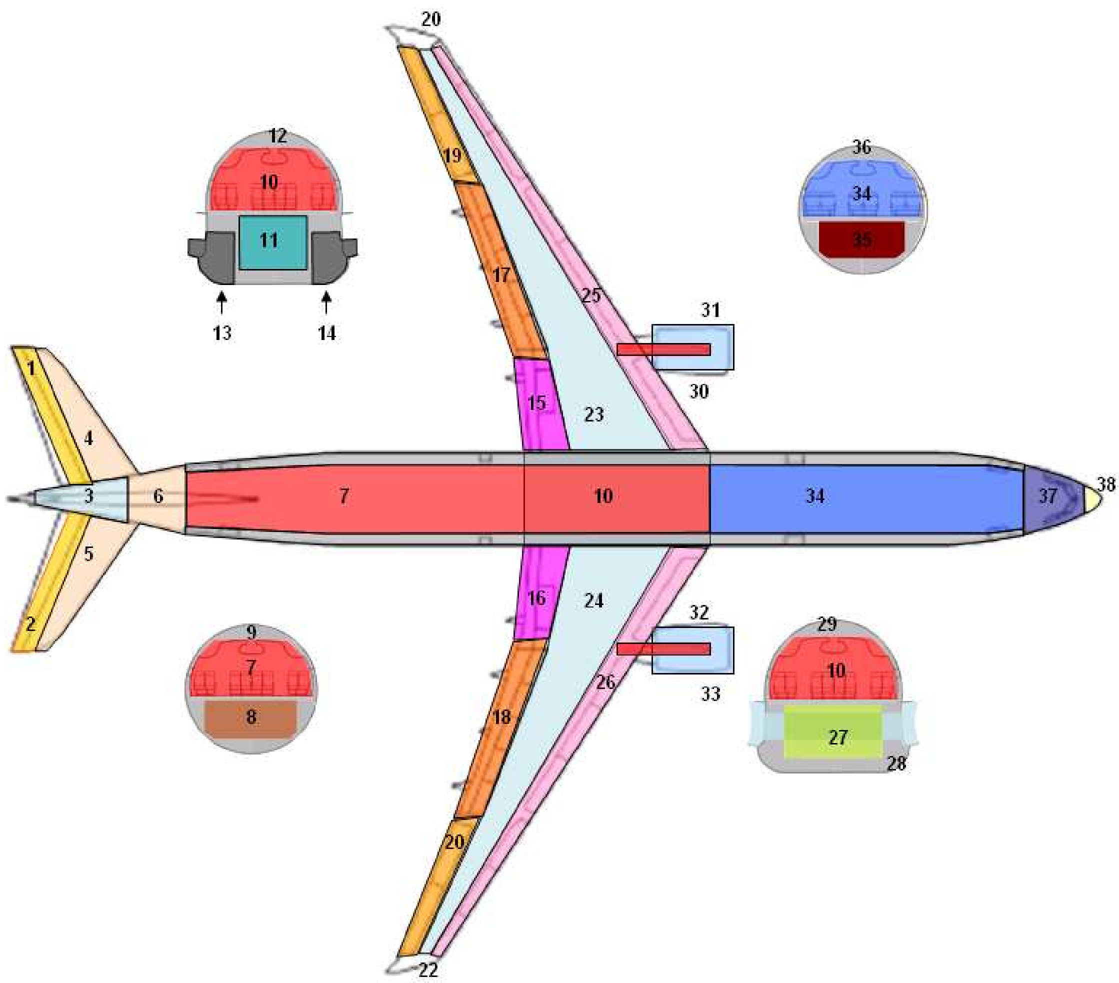


Figure 103: Aircraft Installation Zone Definition

	Alternative	Zone
Provide Thrust_1	Turbofan Engine	31
Provide Thrust_2	Turbofan Engine	32
Protect from Ice_1	Pneumatic Heating	25
Protect from Ice_2	Pneumatic Heating	26
Provide Navigation_1	Avionics and Cockpit	37
Control Roll_1	Hydraulic Actuator\Control Roll	19
Control Roll_2	Hydraulic Actuator\Control Roll	20
Control Roll_1_1	Hydraulic Actuator\Control Roll_1	19
Control Roll_1_2	Hydraulic Actuator\Control Roll_1	20
Control Pitch_1	Hydraulic Actuator\Control Pitch	1
Control Pitch_2	Hydraulic Actuator\Control Pitch	2
Control Pitch_1_1	Hydraulic Actuator\Control Pitch_1	1
Control Pitch_1_2	Hydraulic Actuator\Control Pitch_1	2
Control Yaw_1	Hydraulic Actuator\Control Yaw	3
Control Yaw_1_1	Hydraulic Actuator\Control Yaw_1	3
Control Ground Movement_1	Hydraulically Actuated Landing Gear	13,14
Stop on Ground_1	Hydraulic Brakes	13,14
Stop on Ground_2_1	Hydraullically Actuated Air Brakes	31
Stop on Ground_2_2	Hydraullically Actuated Air Brakes	32
Stop on Ground_1_1	Hydraulically Actuated Thrust Reversers	15
Stop on Ground_1_2	Hydraulically Actuated Thrust Reversers	16
Provide In Flight Entertainment_1	Basic IFE	34
Provide Meal Service_1	Conventional Galley	6,37
Control Passenger and Crew Envrionment_1	Engine Bled Environment Control System	28
Control Passenger and Crew Envrionment_2	Engine Bled Environment Control System	28
Dist: AC Electric_1	AC Bus	35
Dist: AC Electric_2	AC Bus	35
Dist: DC Electric _1	DC Bus	35
Dist: DC Electric _2	DC Bus	35
Dist: Pneumatic _1	Pneumatic Tubing	29
Dist: Pneumatic _2	Pneumatic Tubing	30
Dist: Pneumatic _3	Pneumatic Tubing	
Dist: Pneumatic _4	Pneumatic Tubing	
Dist: Hydraulic_1	Hydraulic Lines 3000 psi	
Dist: Hydraulic_2	Hydraulic Lines 3000 psi	
Dist: Hydraulic_3	Hydraulic Lines 3000 psi	
Dist: Mechanical_1	Accessory Gear Box	31
Dist: Mechanical_2	Accessory Gear Box	32
Dist: Mechanical_3	Accessory Gear Box	3
Dist: Mechanical_4	Accessory Gear Box	28
Dist: HV AC Electric_1		
Dist: Fuel_1	Fuel Lines	
Dist: Fuel_2	Fuel Lines	
Dist: HV DC Electric_1		
Dist: Translational Mechanical_1		
Dist: Cooling Fluid_1		
Dist: Cooling Air_1	Cooling Air Ducting	11
Trans to: AC Electric_1	Alternator	35
Trans to: AC Electric_2	Alternator	35
Trans to: AC Electric_1_1	AC Generator	31
Trans to: AC Electric_1_2	AC Generator	32
Trans to: AC Electric_1_3	AC Generator	3
Trans to: AC Electric_1_4	AC Generator	28
Trans to: DC Electric _1	Transformer Rectifier Unit	35
Trans to: DC Electric _2	Transformer Rectifier Unit	35
Trans to: Pneumatic _1	Heat Exchanger	31
Trans to: Pneumatic _2	Heat Exchanger	32
Trans to: Hydraulic_1	Mechanically Driven Pump	31
Trans to: Hydraulic_2	Mechanically Driven Pump	32
Trans to: Hydraulic_3	Mechanically Driven Pump	3
Trans to: Hydraulic_4	Mechanically Driven Pump	23
Trans to: Mechanical_1		
Trans to: HV AC Electric_1		
Trans to: HV DC Electric_1		
Power Generation_2_1	Ram Air Turbine	23
Power Generation_1_1	Auxiliary Powr Unit	3
Store: DC Electric _1	Battery	35
Store: DC Electric _2	Battery	35
Store: Pneumatic _1	Pressure Vessel	11
Store: Pneumatic _2	Pressure Vessel	11
Store: Hydraulic_1	Hydraulic Reservior	11
Store: Hydraulic_2	Hydraulic Reservior	11
Store: Hydraulic_3	Hydraulic Reservior	11
Store: Fuel_1	Fuel Tank	23
Store: Fuel_2	Fuel Tank	24
DC Bus_Provide Cooling_1	Air Cooling	35
DC Bus_Provide Cooling_2	Air Cooling	35
AC Bus_Provide Cooling_1	Air Cooling	35
AC Bus_Provide Cooling_2	Air Cooling	35

Figure 104: Conventional Architecture Technology Zone Locations

	Alternative	Zone
Provide Thrust_1	More Electric Engine	31
Provide Thrust_2	More Electric Engine	32
Protect from Ice_1	Electrical Heating	25
Protect from Ice_2	Electrical Heating	26
Provide Navigation_1	Avionics and Cockpit	37
Control Roll_1	Electrohydrostatic Actuator	19
Control Roll_2	Electrohydrostatic Actuator	20
Control Roll_1_1	Electromechanical Actuator	19
Control Roll_1_2	Electromechanical Actuator	20
Control Pitch_1	Electrohydrostatic Actuator\Control Pit...	1
Control Pitch_2	Electrohydrostatic Actuator\Control Pit...	2
Control Pitch_1_1	Electromechanical Actuator\Control Pi...	1
Control Pitch_1_2	Electromechanical Actuator\Control Pi...	2
Control Yaw_1	Electrohydrostatic Actuator\Control Yaw	3
Control Yaw_1_1	Electromechanical Actuator\Control Y...	3
Control Ground Movement_1	Electrically Actuated Landing Gear	13,14
Stop on Ground_1	Electrical Brakes	13,14
Stop on Ground_1_1	Electrically Actuated Thrust Reversers	31
Stop on Ground_1_2	Electrically Actuated Thrust Reversers	32
Stop on Ground_2_1	Electrically Actuated Air Brakes	15
Stop on Ground_2_2	Electrically Actuated Air Brakes	16
Provide In Flight Entertainment_1	Advanced IFE	34
Provide Meal Service_1	More Electric Galley	6,37
Control Passenger and Crew Envrionment_1	Electrical Environment Control System	28
Control Passenger and Crew Envrionment_2	Electrical Environment Control System	28
Dist: AC Electric_1	AC Bus	35
Dist: AC Electric_2	AC Bus	35
Dist: DC Electric _1	DC Bus	35
Dist: DC Electric _2	DC Bus	35
Dist: Pneumatic _1		
Dist: Hydraulic_1		
Dist: Mechanical_1	Shaft	31
Dist: Mechanical_2	Shaft	32
Dist: Mechanical_3	Shaft	3
Dist: Mechanical_4	Shaft	23
Dist: HV AC Electric_1	HVAC Bus	35
Dist: HV AC Electric_2	HVAC Bus	35
Dist: Fuel_1	Fuel Lines	
Dist: Fuel_2	Fuel Lines	
Dist: HV DC Electric_1	HVDC Bus	35
Dist: HV DC Electric_2	HVDC Bus	35
Dist: Translational Mechanical_1		
Dist: Cooling Fluid_1	Vapor Cooling Cycle System	35
Dist: Cooling Air_1		
Trans to: AC Electric_1	AC Filter	11
Trans to: AC Electric_2	AC Filter	11
Trans to: AC Electric_1_1	Alternator	11
Trans to: AC Electric_1_2	Alternator	11
Trans to: AC Electric_2_1	AC Generator	3
Trans to: AC Electric_2_2	AC Generator	28
Trans to: DC Electric _1	HV Tranformer Rectifier Unit	11
Trans to: DC Electric _2	HV Tranformer Rectifier Unit	11
Trans to: DC Electric _1_1	Transformer	11
Trans to: DC Electric _1_2	Transformer	11
Trans to: Pneumatic _1		
Trans to: Hydraulic_1		
Trans to: Mechanical_1		
Trans to: HV AC Electric_1	Imbedded Starter Generator	31
Trans to: HV AC Electric_2	Imbedded Starter Generator	32
Trans to: HV DC Electric_1	HV Rectifier	11
Trans to: HV DC Electric_2	HV Rectifier	11
Power Generation_1_1	Ram Air Turbine	28
Power Generation_2_1	Auxiliary Powr Unit	3
Store: DC Electric _1	Battery	35
Store: DC Electric _2	Battery	35
Store: Pneumatic _1		
Store: Hydraulic_1		
Store: Fuel_1	Fuel Tank	23
Store: Fuel_2	Fuel Tank	24
Avionics and Cockpit_Provide Cooling_1	Air Cooling	37
DC Bus_Provide Cooling_1	Air Cooling	35
DC Bus_Provide Cooling_2	Air Cooling	35
HVAC Bus_Provide Cooling_1	Air Cooling	35
HVAC Bus_Provide Cooling_2	Air Cooling	35
HVDC Bus_Provide Cooling_1	Air Cooling	35
HVDC Bus_Provide Cooling_2	Air Cooling	35
Electrohydrostatic Actuator_Provide Cooling_1	Air Cooling	19
Electrohydrostatic Actuator_Provide Cooling_2	Air Cooling	20
Electrohydrostatic Actuator_Provide Cooling_3	Air Cooling	1
Electrohydrostatic Actuator_Provide Cooling_4	Air Cooling	2
Electrohydrostatic Actuator_Provide Cooling_5	Air Cooling	3
Electromechanical Actuator_Provide Cooling_1	Air Cooling	19
Electromechanical Actuator_Provide Cooling_2	Air Cooling	20
Electromechanical Actuator_Provide Cooling_3	Air Cooling	1
Electromechanical Actuator_Provide Cooling_4	Air Cooling	2
Electromechanical Actuator_Provide Cooling_5	Air Cooling	3
AC Bus_Provide Cooling_1	Air Cooling	35
AC Bus_Provide Cooling_2	Air Cooling	35

Figure 105: More Electric Architecture Technology Zone Locations

APPENDIX G: ADEN OUTPUT FILE

The output file to the ADEN tool is intended to be queried by model and simulation integration tool which can instantiate the architecture defined in ADEN. It is provided in CSV formation to be recognizable in Excel as well as many other integration programs. This file contains a list all models necessary to size the architecture, as well as all of the variable relationships between these models. The variable mappings can be split into 2 portions: static or use-case independent variable relationships, and use-case dependant variable mappings.

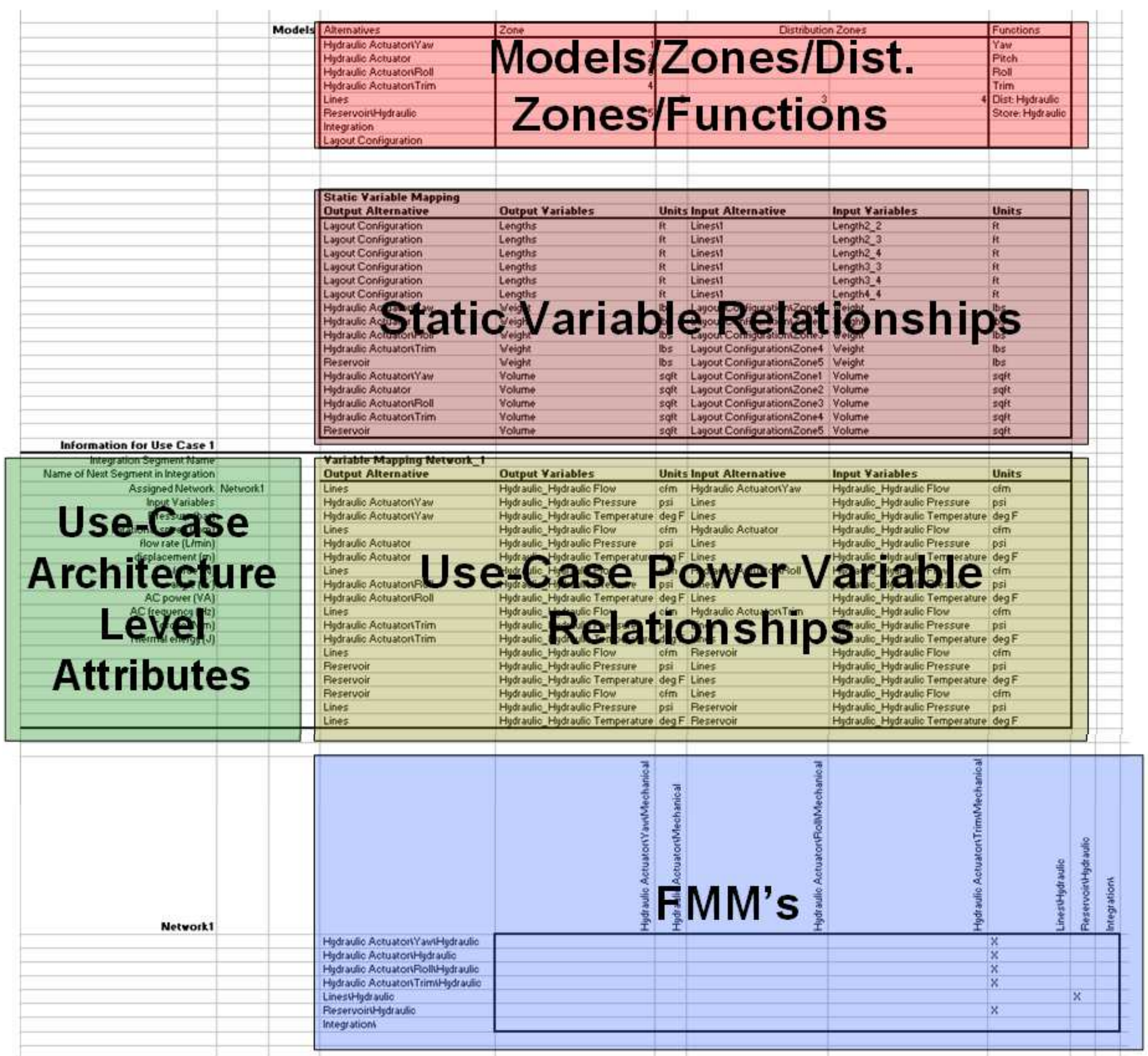
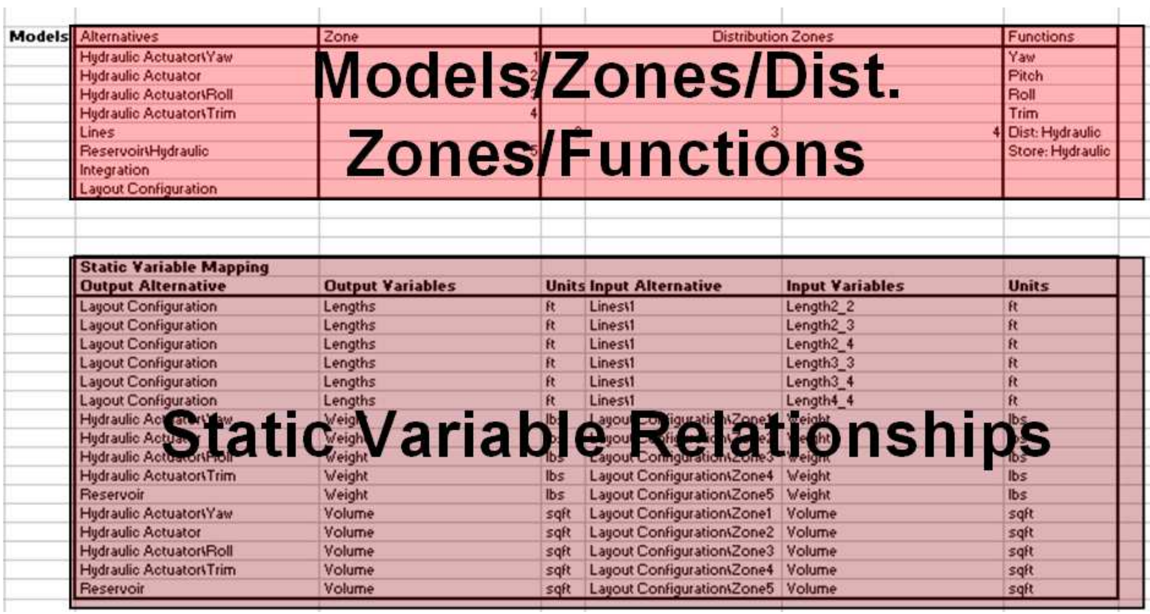


Figure 106: ADEN Output File

The first portion of the static variable mapping is a simple listing of all of the models that must be used for this architecture sizing. Elements, their locations and distribution zones, and the function they are fulfilling are listed in the model definition portion of the output file. The models listed here include every necessary technology models, the layout configuration attribute calculator, and the integration model including sizing and analysis. The second portion of static mappings contains the static variable relationship I/O mappings section. This section of the output file includes all attributes or length relationships between the layout configuration sizing model and the distribution elements and the variable mapping between technology attributes and the zone in which they are housed.



Models	Alternatives	Zone		Distribution Zones		Functions
	Hydraulic Actuator\Yaw	1				Yaw
	Hydraulic Actuator	2				Pitch
	Hydraulic Actuator\Roll	3				Roll
	Hydraulic Actuator\Trim	4				Trim
	Lines		2	3	4	Dist: Hydraulic
	Reservoir\Hydraulic	5				Store: Hydraulic
	Integration					
	Layout Configuration					

Static Variable Mapping					
Output Alternative	**Output Variables**	**Units**	**Input Alternative**	**Input Variables**	**Units**
Layout Configuration	Lengths	ft	Lines\1	Length2_2	ft
Layout Configuration	Lengths	ft	Lines\1	Length2_3	ft
Layout Configuration	Lengths	ft	Lines\1	Length2_4	ft
Layout Configuration	Lengths	ft	Lines\1	Length3_3	ft
Layout Configuration	Lengths	ft	Lines\1	Length3_4	ft
Layout Configuration	Lengths	ft	Lines\1	Length4_4	ft
Hydraulic Actuator\Yaw	Weight	lbs	Layout Configuration\Zone1	Weight	lbs
Hydraulic [illegible]	Weight	[illegible]	[illegible]	[illegible]	[illegible]
Hydraulic [illegible]	Weight	lbs	Layout Configuration\Zone3	Weight	lbs
Hydraulic Actuator\Trim	Weight	lbs	Layout Configuration\Zone4	Weight	lbs
Reservoir	Weight	lbs	Layout Configuration\Zone5	Weight	lbs
Hydraulic Actuator\Yaw	Volume	sqft	Layout Configuration\Zone1	Volume	sqft
Hydraulic Actuator	Volume	sqft	Layout Configuration\Zone2	Volume	sqft
Hydraulic Actuator\Roll	Volume	sqft	Layout Configuration\Zone3	Volume	sqft
Hydraulic Actuator\Trim	Volume	sqft	Layout Configuration\Zone4	Volume	sqft
Reservoir	Volume	sqft	Layout Configuration\Zone5	Volume	sqft

Figure 107: Use-Case Independent Selected Models and Relationships Output

The second set of variable mappings is listed below the static variable relationships. These use-case relationships include all power relationships involved in the FMM selected for that given use-case. It also includes a listing of all of the variables that are defined for that specific sizing scenario. These variables are necessary to size some of the physical alternatives. Each use-case will include its own mapping of variable relationships and architecture level attributes.

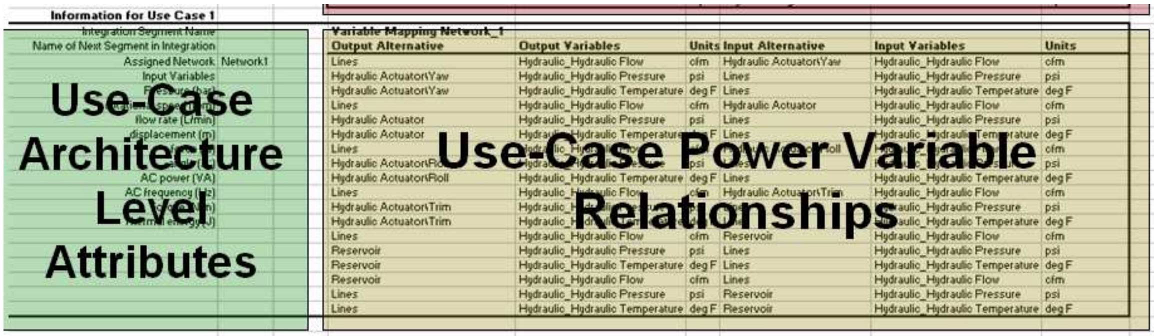
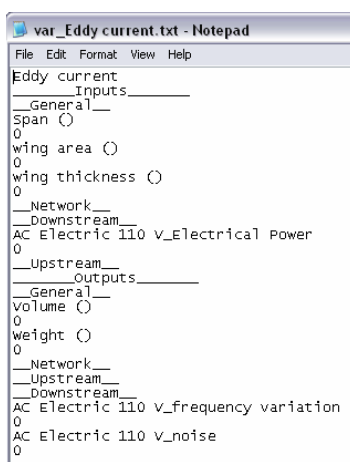


Figure 108: Use-Case Model Relationships Output

Functional Mapping Matrices are also output if they are desired in the integration of the tools.

Figure 109: Functional Mapping Matrix Output

In the case that scripts are used for the passing of variables between models. A text file wrapper can also be written when each technology model is identified. This wrapper includes upstream and downstream power relationships as well as all required input and output attributes required for sizing. Figure 110 is an empty notional wrapper file for an eddy current ice protection sizing model.




```
Eddy current
_______Inputs_______
__General__
Span ()
0
wing area ()
0
wing thickness ()
0
__Network__
__Downstream__
AC Electric 110 V_Electrical Power
0
__Upstream__
_______Outputs_______
__General__
Volume ()
0
weight ()
0
__Network__
__Upstream__
__Downstream__
AC Electric 110 V_frequency variation
0
AC Electric 110 V_noise
0
```

Figure 110: Technology Model Wrapper File