A methodology for portfolio-level analysis of system commonality

Citation: Hofstetter, Wilfried K., and Edward F. Crawley. "A Methodology for Portfolio-Level Analysis of System Commonality." Research in Engineering Design 24, no. 4 (February 13, 2013): 349–373.

As Published: http://dx.doi.org/10.1007/s00163-012-0151-z

Publisher: Springer London

Persistent URL: http://hdl.handle.net/1721.1/106991

Version: Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

A Methodology for Portfolio-Level Analysis of System Commonality

Dr. Wilfried K. Hofstetter (corresponding author)
The Boston Consulting Group GmbH
Görresstrasse 11, 80798 München, Germany
wk_hof@alum.mit.edu, Phone: +49-(170)-334-4411

Dr. Edward F. Crawley
Ford Professor of Engineering
Massachusetts Institute of Technology
Massachusetts Avenue 77, Cambridge, MA 02139, USA
crawley@mit.edu, Phone: +1-(617)-253-7510

Abstract

Complex systems are increasingly being developed as part of portfolios or sets of related complex systems. This enables synergies such as commonality between portfolio systems that can significantly reduce portfolio life-cycle cost and risk. While offering these benefits, commonality usually also incurs up-front as well as life-cycle penalties in cost and risk due to increased design complexity. The resulting trade-off needs to be carried out during the architecting stage of the portfolio life-cycle when there is maximum leverage to improve life-cycle properties due to degrees of freedom available in architectural and design decisions. This paper outlines a 4-step methodology for the identification and assessment of commonality opportunities in complex systems portfolios during the architecting stage of the portfolio lifecycle. The methodology transforms a solution-neutral description of a portfolio of aerospace systems based on system functionality, requirements, and metrics into a set of preferred portfolio design solutions with commonality. The methodology is based on a 2-stage approach which identifies preferred architectures for each system in the portfolio individually prior to heuristic commonality analysis between systems based on a pair wise assessment of system overlap in functionality, technologies, operational environments, and scale. Application of the methodology is demonstrated with a retrospective analysis of NASA's Saturn launch vehicle portfolio.

Keywords: commonality, product platforming, system architecture

1. Introduction and Problem Statement

The last decades have seen an increasing trend towards the development of portfolios of complex systems rather than the development of individual complex systems. This trend can be observed across many industries such as the aerospace, automotive, and consumer goods industries (Cortright 1975) (Egbert, McCain 1994) (Kalligeros et al. 2006) (NASA 2005) (Suh. De Weck, Chang 2007). Developing complex systems as part of a portfolio enables synergies such as commonality which can significantly improve portfolio lifecycle properties such as cost, developmental risk, and operational risk. Commonality is usually defined as the concept of reusing elements of legacy systems for future systems or reusing elements between future systems (Bell 1967) (Coan, Bell 2006) (Siddiqui, de Weck 2007) (Waiss 1987).

As a general rule, the earlier commonality synergies can be identified during the portfolio life-cycle, the more pronounced the benefits of such synergies will be. Earlier identification means that less design decisions have been made and the disadvantages of commonality can be better mitigated (Hoffman 1995). Commonality should therefore already be investigated during the portfolio architecting stage (Maier, Rechtin 2000).

For commonality in a portfolio of complex systems to be implementable, three general conditions must hold (Boas, Crawley 2007):

- (1) Design reuse must be technically and operationally feasible, i.e. there must not be violations of physical laws or technical and operational practicality
- (2) Design reuse must be economically attractive, i.e. commonality must offer a reduction of cost compared to custom implementation
- (3) Design reuse must be managerially and organizationally feasible.

This paper is concerned with methodologies that provide portfolio design solutions which meet the first two conditions: technical and operational feasibility, as well as economic attractiveness. The challenge for the portfolio architect can be formally described with the following general problem statement: given a set $R = \{R_1, R_2, \ldots, R_K\}$ of K solution-neutral requirements for an aerospace systems portfolio $P = \{System_1, System_2, \ldots, System_N\}$ including N systems (also called use cases) related by a set of M constraints $C = \{C_1, C_2, \ldots, C_M\}$, find a set of L portfolio design solutions $PDS = \{PDS_1, PDS_2, \ldots, PDS_L\}$ for the portfolio. Each portfolio design solution in PDS must contain a set of N systems design solutions $SDS = \{SDS_1, SDS_2, \ldots, SDS_N\}$ for each of the systems in the portfolio and a description of the extent of commonality within and between systems design solutions. In addition to fulfilling conditions 1 and 2 above, each portfolio design solution in PDS must have the following attributes:

- It serves as input to more detailed design and development activities, providing a concept, selection of technologies for internal functionality, an operational description, as well as quantitative design information related to system scale for each of its system design solutions.

- It should be located close to or on the overall cost, risk, and performance Pareto front (or Pareto fronts) for the portfolio. For the purposes of this paper, the Pareto front is defined as the set of portfolio design solutions with commonality that are not dominated by any other solution, i.e. that are equal to or better than all other portfolio design solutions with regard to at least one portfolio metric. This conforms to the definition provided by (Smaling, de Weck 2005).

- It provides an explicit description of how commonality is being utilized for the portfolio design solution in terms of what functions, technologies, operations and elements of form are affected by commonality.

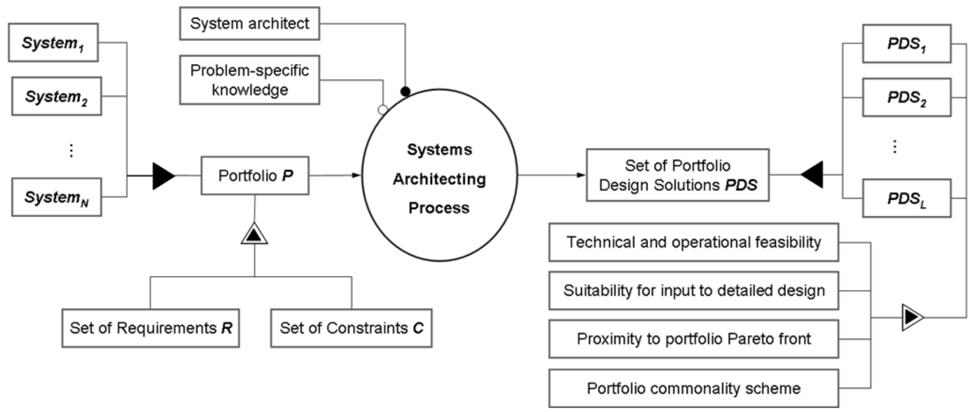


Figure 1: Black box representation of the general problem statement

Figure 1 provides a black box visualization of this general problem statement in the form of an Object-Process-Diagram (Dori 2002). The general research objective is to develop processes that conform to Figure 1 and to demonstrate their applicability by carrying out case studies with regard to commonality opportunities in specific aerospace systems portfolios.

In Section 2 a review of the state of the art with regard to existing systems architecting processes for portfolios with commonality is carried out as a basis for defining the specific gap in the state of the art addressed in this paper. Prior to discussing individual references in this review of the art, it is useful to provide a more concrete definition of what the "systems architecting" process or phase represents in the context of this paper. A general technical system (Figure 2) provides externally delivered functionality to the system stakeholders (and thereby delivers value) by using the system operating process. The system operating process in turn requires the system internal functionality (what the system does specifically), the system form (hardware and / or software), and the system operator as instrument objects. System internal functionality and system form are related through technology choices for the individual internal functions and operating processes associated with these technology choices. The operating processes in turn use elements of form as instrument objects. The elements of form are described by design parameters which capture the scale and characteristics of the elements of form and the form elements are related through the system structure. The description of the system as shown in Figure 2 represents the system architecture.

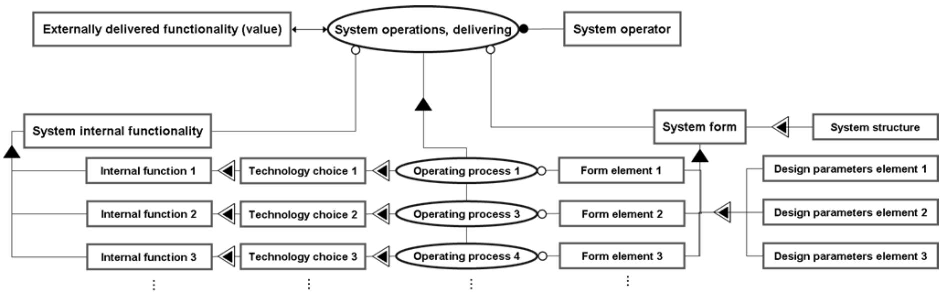


Figure 2: Overview of the system architecture of a technical system

For the context of this paper, the system architect is assumed to define the internal functions, associated technology choices and operating processes as well as the elements of form of the system with associated design parameters (i.e. the "architecture" of the system) based on a solution-neutral description of the externally delivered functionality. By extension, the portfolio architect defines the architecture of each of the systems in the portfolio. By contrast, during the design phases following the architecting phase, the design effort is typically concentrated on refining design parameters for the elements of system form. When analyzing the methodologies for commonality analysis proposed in the literature, it is important to assess whether they are applicable during the architecting phase (i.e. proceed from a solution-neutral description of the systems in the portfolio), or whether they require the system architecture to be known and are mainly concerned with commonality as expressed by similarity in design parameter values.

Section 3 outlines the Portfolio-Level Analysis of System Commonality Opportunities (PLASCO) methodology and also provides an illustration of PLASCO's workings by way of retrospective analysis of an example aerospace systems portfolio. The Saturn launch vehicle family developed by the National Aeronautics and Space Administration (NASA) was chosen for this example. In Section 4, the broader applicability as well as the limitations of the PLASCO methodology are discussed. Section 5 provides suggestions for future work, and Section 6 summarizes major insights and conclusions.

2. Review of the State of the Art

This section provides a review of the state of the art that is relevant to systems architecting of portfolios of complex systems with commonality. Specifically, four bodies of literature are covered in this review:

- (1) Function-based engineering design and modularization
- (2) Platforming based on multi-disciplinary optimization (MDO)
- (3) Commonality and standardization in the technology management literature
- (4) General commonality and platforming literature which covers remaining heuristic and other methodologies not captured in the three above categories, as well as architecture-level case studies with regard to aerospace systems portfolio commonality

The focus of the literature review is on publications describing practically applicable methodologies or strategies relevant to the architecting of aerospace systems portfolios with commonality.

2.1 Function-Based Design and Modularization

Pahl and Beitz (1996) provide two approaches to commonality and standardization in technical systems: one based on size ranges, and one based on functional modularization of products made possible by a classification of internal functionality into basic, special, auxiliary, and adaptive functions. The size range approach assumes system architecture to be fixed in order to apply similarity laws for scaling this approach is not helpful for commonality analysis during the architecting stage. The second method of enabling the use of modules which are common between different technical systems by clustering identical functions together (in particular special, auxiliary, or adaptive functions) is suitable for use during the architecting phase; however, the proposed method does not directly take into account the impact of function clustering on system / portfolio metrics such as mass, cost, developmental or operational risk which are essential for evaluation of aerospace systems commonality. Otto and Wood provide a more formalized and generalized version of the Pahl & Beitz modularization approach for general system function structures (Otto, Wood 2001). Dahmus, Gonzalez-Zugasti, and Otto (2001) describe a systematic modular architecture generating process using function structures and a matrix containing function-to-product mappings; in this matrix, shared and custom functionality can be indicated and thereby different platforming concepts generated. The functional modularization approach is also the basis for platforming approaches discussed in the book on product platform design by Meyer and Lenherd (1997). Thomas provides a partially automated version of the functional clustering algorithm (Thomas 1989) which is applied to investigating commonality opportunities between space station berthing mechanisms with different interfacing capabilities. The "penalty" of commonality is analyzed by counting the number of functions allocated to each berthing mechanism for custom and for common implementations; the overhead in functions for the common case represents the commonality penalty. Quantitative attributes of the mechanisms such as mass, cost, or developmental risk are not taken into account in the analysis. Note that Thomas provides another methodology in his work which is based on clustering of functions and investigating the quantitative attributes of the common and custom design solutions which is described below. Zhang, Tor, and Britton (2006) use a two-layered approach with a function- and a behavior-layer and a behavioral modularity matrix for another partially automated approach to commonality within a product portfolio. The approach is implemented in a software tool and allows for the comprehensive analysis of functional and behavioral modularization options. Reinhart, Schaefer, and Fricke (2001) provide an approach to the modularization of commercial airship functionality (specifically for the "Cargolifter" airship): their approach is focused on finding an assignment of functionality to a series of increasingly capable airships that provides for incremental build-up of technological capabilities while providing revenue and staying robust to changes in the market environment.

Summarizing, the strength of the approaches in this field of function-based modularization is that they do not require knowledge about the architectures of the systems in the portfolio and are therefore generally applicable during the systems architecting phase. The limitation of these approaches with regard to aerospace systems portfolios is that they tend to be limited to low- to mid-complexity systems (such as power tools, docking mechanisms, mechanical assemblies, etc.), and that function-based methodologies typically do not include explicit consideration of quantitative benefits or penalties of commonality opportunities. Given that the trade-off between the benefits and penalties of specific commonality opportunities for aerospace systems portfolios must be based on a quantitative assessment of the overall impact on life-cycle cost and risk, function-based methodologies in their present form do not provide all the attributes necessary for aerospace systems portfolio commonality analysis.

2.2 Platforming Based on Multi-Disciplinary Optimization (MDO)

Approaches in this field make use of the standard formulation of MDO problems for solving the portfolio design problem: the design is governed by a system of objective functions that relate design variables to figures of merits (such as weight, cost, etc.). These objective functions are then minimized given a system of equality and inequality constraints; commonality between systems in the portfolio is generally defined as similarity or identity in design variable values. Khajavirad, Michalak, and Simpson (2007) developed a decomposed multi-objective genetic algorithm to jointly optimize platform and variant design. Fujita (2002) provides a methodology for design optimization for product variety given a fixed product architecture. Messac, Martinez, and Simpson (2000) suggest the use of a Product Family Penalty Function (PFPF) in conjunction with physical programming that penalizes design parameters that are not common throughout the product family while optimizing the desired objectives. Gonzalez-Zugasti, Otto, and Baker (2000) outline an approach that includes a negotiation model in addition to the optimization approach with the goal to make the approach more applicable to conceptual design of planetary spacecraft; in Gonzalez-Zugasti, Otto, and Baker (2001) this was expanded upon through introduction of a real options approach to evaluating technically and economically feasible commonality opportunities. Fellini, Kokkolaras, and Papalambros (2005) introduced a Sharing Penalty Function (SPF) for products which have mild variation across design parameters to aid in the selection of design variables for the platform. Willcox and Wakayama (2002) provide an optimization approach for the simultaneous optimization of an aircraft family. Simpson, Maier, and Mistree (2001) introduce the Product Platform Concept Exploration Method (PPCEM) which starts with a market segmentation grid for the products in the portfolio and then uses design principles and meta-modeling to set up the MDO problem. Nayak, Chen, and Simpson (2002) developed the Variation-Based Platform Design Method (VBPDM), which uses aims to minimize variation between product designs in the family given certain platform requirements. Dobrescu and Reich (2003) provide a methodology based on a simulated annealing optimization algorithm to develop a common platform and variants from a set of pre-existing components using shape grammar rules for geometric layout. Sered and Reich (2006) introduce the SMDP (standardization and modularization driven by process effort) methodology which aims to minimize overall design process effort as objective function; this leads to the identification of commonality opportunities which reduce overall design effort.

In general, using MDO for finding commonality within a systems portfolio is a powerful approach due to the ability to investigate a large space of design alternatives. However, the MDO approaches investigated are focused on varying design parameter values and commonality opportunities are identified based on identity or similarity in design parameter values. This means that the architecture of the systems in the portfolio has to be known before the existing MDO methodologies can be applied, i.e. they are not directly suitable for use during the architecting phase.

2.3 Commonality in Technology Management

Standard texts on technology management (Khalil 2000) (Burgelman et al. 2003) mention commonality, platforming and standardization as important tools for improving life-cycle properties of product and technology portfolios, but provide no specific methodologies for or approaches to identifying commonality opportunities. Cooper, Edgett, and Kleinschmidt (2001) specifically distinguish technology platforms / commonality and marketing platforms from product design platforms, and suggest a strategic bucket funding approach for a project portfolio including platform projects in order to protect funding longer-term platform projects. Dickinson, Thornton, and Graves (2001) provide a quantitative optimization-based methodology for the management of a portfolio of interdependent projects; this interdependence could be interpreted to represent commonality. However, the approach requires quantitative information on the kind of interdependence between the systems in the portfolio (i.e. impact of commonality opportunities) as input. While not directly applicable during the systems architecting stage, the above works in the technology management literature do underscore the importance of the portfolio model for aerospace systems as a useful way of framing the commonality analysis problem.

2.4 General Methods for Commonality and Platforming

Thomas (Thomas 1989) provides a quantitative clustering-based commonality approach which was developed for the initial US space station designs; the method requires quantitative descriptions of architectures as input (i.e. it is intended for the preliminary and detailed design phases), and includes explicit consideration for the benefits and penalties of commonality. The approach can also be used for qualitative function-based clustering during conceptual design (see above section 2.1). Martin and Ishii (2002) propose the QFD-based Design For Variety (DFV) method which includes two indices: the generational variety index (GVI), a measure for the amount of redesign effort required for future designs of the product, and the coupling index (CI), a measure of the coupling among the product components; see Sered and Reich (2006) for a related approach. Both indices are used for designing a decoupled basic product architecture from which common variants can be easily generated; the approach is applied to the design of a family of water-coolers. Kalligeros (2006) developed a method based on sensitivity DSMs for identifying system components that can be standardized based on their robustness to changes in functional requirements and changes in other design variables. The limitation of this approach is that it requires a description of the system concept as input, and identifies commonality opportunities purely based on design parameter sensitivity to changes in requirements (the insensitive design parameters are candidates for commonality / standardization). The strengths of this approach are that it is amenable to mathematical treatment and can be coupled with real options analysis. Otto and Hölttä-Otto (2007) developed a 19-criteria platform assessment tool for use prior to proof-of-concept prototyping. Hodson (2007) performed spacecraft avionics systems commonality analysis and recommends commonality identification based on common internal functional and operational requirements, as well as a modular stack-based approach for hardware commonality.

Given the diversity of methodologies discussed in this subsection it is difficult to provide a single assessment that captures the advantages and disadvantages of each methodology. However, it is possible to identify the reliance on information about the system architecture of the systems in the portfolio as a characteristic feature of the general platforming and commonality analysis methodologies investigated here. Requiring partial or complete knowledge about the architecture of each system in the portfolio makes application of these methodologies during the architecting phase challenging, and also results in a focus on commonality as evidenced by identity or similarity in design parameter values (much as for the MDO methodologies discussed above).

2.5 Gap in the State of the Art

Based on the preceding review of the state of the art in the different areas the following gap in the state of the art was identified: existing methodologies for commonality analysis in portfolios of complex systems are mostly limited to application during the system design phases (i.e. require the system architecture as input), or are limited to portfolios of low- to mid-complexity systems if applicable during the systems architecting phase (i.e. starting with solution-neutral functional descriptions and requirements). The methodology presented in this paper aims to close this gap in the state of the art. The following is the subset of the above publications and approaches that are most relevant to and are extended by the work presented in this paper: (Reinhart, Schaefer, and Fricke 2001), (Thomas 1989), (Simpson, Maier, and Mistree 2001), (Kalligeros 2006), and (Hodson 2007).

3. PLASCO Methodology Description and Application

The PLASCO methodology consists of four major steps: portfolio definition (Step 1), architecture analysis without consideration for commonality (Step 2), commonality screening (Step 3), and sensitivity analysis and preferred portfolio selection (Step 4); see Figure 3 for a visual description of the methodology showing the inputs and outputs of each step. In Step 1, the portfolio scope is determined in terms of system use cases (and associated functionality and requirements) included in the portfolio and solution-neutral metrics to be used for relative ranking of portfolio design variants are defined. Use cases may either be future use cases or legacy use cases (i.e. existing systems). In Step 2, a comprehensive analysis of architecture alternatives is conducted for each of the future use cases in the portfolio individually (for legacy use cases the architecture is already known), leading to the selection of a set of preferred architectures for each use case in the portfolio. Commonality is not considered during this architecture analysis step, i.e. the architecture analyses for the individual systems in the portfolio are uncoupled and can be carried out in parallel. Based on the preferred architecture alternatives for each use case, a set of preferred portfolio design solutions without commonality can be enumerated combinatorially, to serve as input to the commonality analysis in Step 3.

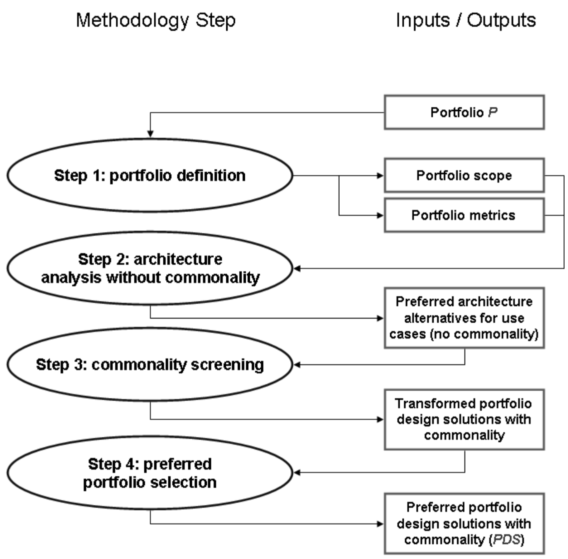


Figure 3: Overview of the PLASCO methodology for the architecting of aerospace systems portfolios with commonality

The commonality screening process in Step 3 comprehensively investigates the potential for commonality for all pairs of systems for each function for each of the portfolio design solutions. The specific definition of criteria for commonality screening may differ from application to application, but the following four heuristic criteria should be considered for every type of portfolio of complex systems:

- Functional overlap criterion: the two systems or subsystems in question must provide the same function, for example both subsystems must provide the function of "CO_2 removal from cabin atmosphere" in order to have a common implementation.

- Technology overlap criterion: the two systems or subsystems in question must utilize the same technology choice associated with the function, for example both subsystems with the function of "CO_2 removal from cabin atmosphere" must utilize 4-bed molecular sieve technology in order to have a common implementation.

- Operational overlap criterion: in addition to the above criteria, the two subsystem implementations also need to be operated in similar environments in order to ensure that requirements originating from these environments overlap. The similarity in operational environments is measured by calculating the number of common operational environments between two systems and then dividing this number by the number of total operational environments for each system. If both of these fractions are larger than or equal to a threshold δ, then the requirement for operational similarity is fulfilled. The value of δ is obviously an arbitrary choice, and therefore needs to be subject to sensitivity analysis.

- Scale overlap criterion: an additional criterion for the feasibility of specific commonality opportunities is similarity in the parameters describing implementation scale (i.e. similarity in physical size or scale). If the values of scale parameters (e.g. volumes) for a system pairing are within a factor *k* (the so-called overlap parameter) of each other, then commonality is assumed to be feasible for this system pairing (see Formula 1). The value for the overlap parameter *k* is obviously also an arbitrary choice and therefore also needs to be subject to sensitivity analysis.

$$Volume_{System\_1} \cdot \frac{1}{k} < Volume_{System\_2} < Volume_{System\_1} \cdot k \qquad \textbf{Formula 1}$$

These four general heuristic criteria need to be customized for a specific portfolio in order to carry out a commonality screening. The criteria are assessed using the so-called System Overlap Matrix (see Figure 4 for a generalized representation) which arranges the union of all functions and their associated technology choices in the portfolio on the left-hand vertical side, and the union of all operational environments in the portfolio on the top horizontal side. Thus, any system architecture in the portfolio can be represented in this matrix, and by overlaying matrices for two different systems we can assess the fulfillment of the functional, technical, and operational overlap by simple inspection whether the two matrices have the same entries in each cell. Assessment of similarity in scale can be carried out separately by assessing whether the values of the two systems for the same scale parameter are within a factor specified by the overlap parameter k.

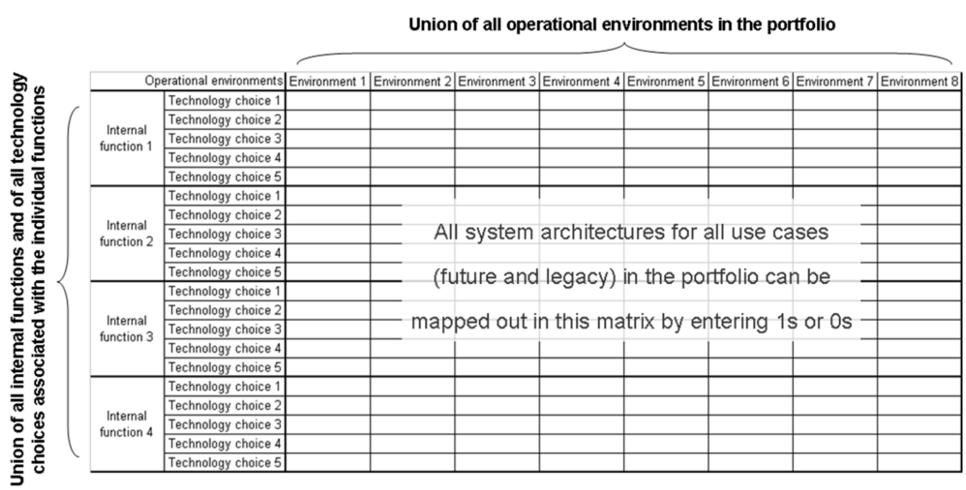


Figure 4: System Overlap Matrix (SOM) template featuring the union of all system functions, technology choices, and operational environments in the portfolio

The SOM is an extended morphological matrix to which operational environments have been added for each technology choice. By making the sets of functionality, technology choices, and operational building blocks the unions of all corresponding sets for the preferred architecture alternatives across the portfolio design solutions, a standardized matrix can be created and used to capture any architecture alternative in the portfolio (when only capturing one alternative, the matrix is called Concept Description Matrix - CDM) (Hofstetter et al. 2007). CDMs for two different use cases in the same portfolio design solution are then analyzed for overlap by determining which fields of the matrices have identical entries; this can be visualized as a process of overlapping the two CDMs, leading to a System Overlap Matrix (SOM). For each function, the number of overlapping fields is then normalized with the total number of entries for that function, resulting in two normalized overlap fractions which capture operational overlap for each function. The portfolio architect then specifies a minimum operational overlap fraction $\boldsymbol{\delta}$ for both systems: if the overlap fraction for each system is larger than this value then commonality is possible. This allows for an assessment of the above functional, technical, and operational overlap criteria for the feasibility of commonality opportunities.

Figure 5 shows a graphical representation of this overlap process: the individual CDMs for system 1 and system 2 are shown above the SOM. In the CDMs, fields with a "1" mark the entries indicating implementation of a particular function with a specific technology choice in a specific operating environment. The SOM entries are calculated by adding up the entries from the two CDMs; an entry of “2” (shaded) in a field in the SOM therefore means that both CDMs had entries for this field. This means that fields in the SOM with entries of “2” indicate overlap between the CDMs and therefore between the systems. By counting all the fields with entries of “2” for a specific function we can determine the degree of overlap for that function and therefore assess the fulfillment of the above criteria for that specific function.

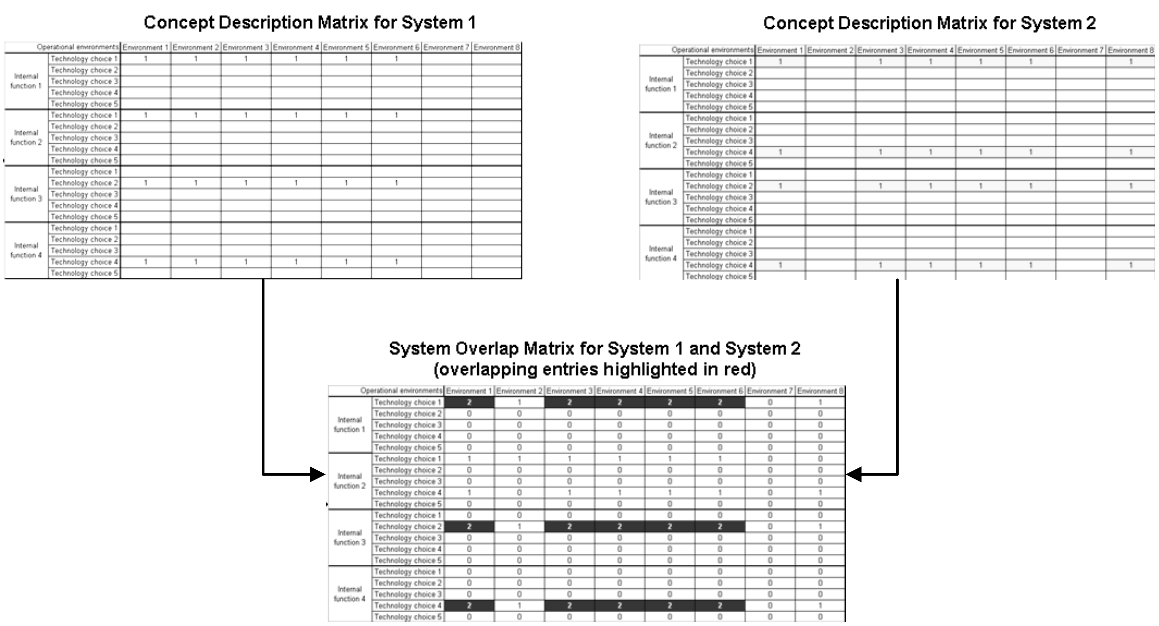


Figure 5: Assessment of overlap between two CDMs using the SOM. Fields with a value of“2” (marked in black) in the SOM indicate overlap in internal functionality, technology choice, and operating environment; all other entries indicate that there is no overlap.

The scale overlap criterion is assessed separately based on quantitative design parameters of the subsystem implementation such as mass, thrust, volume etc. Applying this commonality screening based on pair-wise comparison of functions for all use case pairs in the portfolio (future and legacy use cases) results in the comprehensive investigation of all possible commonality opportunities for all portfolio design solutions without commonality.

If all four criteria for commonality are satisfied, then the commonality opportunity is technically and operationally feasible and the more complex (and individually more costly) design of the two commonality candidates is chosen as the common design. This is repeated for each pairing of system design solutions in each portfolio design solution without commonality. Due to the symmetry of the heuristic commonality criteria defined above (see Formula 1) the results of overlap analysis are not dependent on the order in which these pair-wise commonality assessments are carried out. The output of Step 3 is a transformed set of portfolio design solutions where custom design solutions have been exchanged for common solutions where feasible according to the above criteria.

In Step 4, a selection of portfolio design variants is carried out based on the life-cycle properties of the original (no commonality or only accidental commonality without utilization of commonality benefits) as well as the transformed set of portfolio design solutions with commonality. Step 4 may also include an analysis of the sensitivity of the transformed portfolio to changes in key parameters and assumptions made for the heuristic commonality criteria (Criterion 3 and 4) in Step 3.

PLASCO is a representative of a 2-stage commonality analysis methodology: preferred architectures are identified individually for each system in the portfolio (stage 1) prior to commonality analysis based on these preferred system architectures (stage 2); the methodology described by (Gonzalez-Zugasti, Otto, Baker 2000) would be an example for a single stage methodology which combines the architecture analysis for each individual system in the portfolio with the commonality analysis between systems. For PLASCO, the 2-stage approach was deliberately chosen to allow the system architect to investigate the architecture spaces for the individual systems in the portfolio prior to commonality analysis and be actively involved in the selection of preferred architectures. Involvement of a human being in the architecture selection process is particularly desirable during the architecting stage of a portfolio of complex systems when additional non-quantifiable metrics such as maintainability need to be taken into consideration by the system or portfolio architect (Maier, Rechtin 2000).

The 4 steps of the framework will now be discussed in detail using the retrospective example of NASA’s Saturn launch vehicle family, a portfolio of launch vehicles used for Earth orbital and lunar missions of the United States Apollo human spaceflight program in the 1960s and 1970s (Cortright 1975) (Bilstein 1996).

3.1 Saturn Launch Vehicle Portfolio Definition

The historical Saturn launch vehicle portfolio was a set of launch vehicles developed initially for United States military applications but later adapted towards exclusively civilian use for human spaceflight to Earth orbit and beyond (Cortright 1975) (Belew 1977) (Ezell, Ezell 1978) (Bilstein 1996). The Saturn rockets were the first United States launch vehicle designs with clustered engines, enabling high lift-off mass and thrust using a number of smaller engines. The Saturn launch vehicle portfolio as implemented included three vehicle designs which serve as portfolio use cases:

The Saturn I use case: this vehicle was used in two forms (Block I and Block II), with and without an active upper stage. Ten units were built for the 1st stage and 6 units for the upper stage, all used for unmanned test flights. The payload to Low Earth Orbit was 9000 kg for a required velocity change along the vehicle trajectory (also called "delta-v", a measure for the change in momentum affected by the rocket) of approximately 9500 m/s.

The Saturn IB use case: this vehicle was used for Apollo CSM and lunar module unmanned test flights, as well as for 5 manned CSM flights: the Apollo 7, Skylab 2, 3, 4, and Apollo Soyuz Test Program missions. 12 units of the entire vehicle were produced. The payload to Low Earth Orbit was 17000 kg for a delta-v of approximately 9500 m / s.

The Saturn V use case: this vehicle was used to launch the Apollo lunar missions. 15 units of the entire vehicle were produced. The payload to trans-lunar injection was 47790 kg for a delta-v of approximately 12259 m / s.

The functionality of each of the Saturn vehicle propulsion stages can be captured in two main functions: provision of thrust to accelerate the vehicle and payload using rocket propulsion, and provision of propellant storage and load transmission from the engines to the payload of the stage, as well as provision of structural integrity. This 2-function breakdown of propulsion stage functionality is commonly used in the literature for architecture-level analysis of propulsion systems (Larson, Pranke 2000).

The following metrics were used to asses the relative cost of vehicle architecture alternatives: DDT&E (design, development, test, and evaluation) and unit production cost for engines and fuselages to assess the life-cycle cost of each use case, and vehicle height and vehicle wet mass at launch as proximate metrics for ground processing and operations cost. DDT&E and 1st unit production cost for individual engines and fuselage elements were calculated using dry-mass-based cost estimating relationships derived from NASA Air Force Cost Model (NAFCOM) (Formula 2,3,4, and 5) which are publicly accessible (NASA JSC 2007):

$$C_{Engine\_DDT\&E} = 32.264 \cdot m_{Engine}^{0.55}$$ **Formula 2**

$$C_{Fuselage\_DDT\&E} = 7.9875 \cdot m_{Fuselage}^{0.55}$$ **Formula 3**

$$C_{Engine\_1st\_unit} = 0.1776 \cdot m_{Engine}^{0.662}$$ **Formula 4**

$$C_{Fuselage\_1st\_unit} = 0.1898 \cdot m_{Fuselage}^{0.662}$$ **Formula 5**

The total unit production cost for a rocket engine or fuselage was calculated taking into account learning curve effects with a learning rate *LR* of 0.85 (Hoffman 1995) (NASA JSC 2007) (Formula 6,7, and 8):

$$C_{n-th\_Unit} = C_{1st\_Unit} \cdot n^{b} \qquad \textbf{Formula 6}$$

$$b = \frac{\ln(LR)}{\ln(2)} \qquad \textbf{Formula 7}$$

$$C_{Units} = C_{1st\_Unit} \cdot \sum_{n=1}^{k} n^{b} \qquad \textbf{Formula 8}$$

Calculation of vehicle mass and height is based on the mass and volume characteristics of the individual propulsion stages in the vehicle architecture and on payload mass and height, assuming a uniform diameter of 10 m for each propulsion stage in the portfolio. A 10 meter diameter is the maximum diameter that could be supported by the Saturn manufacturing infrastructure. While the actual Saturn launch vehicles had varying diameters for the individual stage designs, the assumption of a common diameter for all stages is appropriate for the relative ranking that the vehicle height metric is going to be used for. In addition, the use of the largest possible diameter for each stage design results in the most optimistic vehicle height achievable. It should be noted that the usage of lifecycle cost (consisting of DDT&E and production cost), as well as vehicle height and mass as metrics is consistent with evaluation criteria commonly used in launch vehicle design the 1960s (Bilstein 1996).

For the relative ranking of portfolio design solutions, portfolio life-cycle cost and the number of custom development projects required for implementation of a specific portfolio design solution were used as metrics for relative ranking of alternatives. Life-cycle cost for a portfolio design solution is defined as the sum of the life-cycle costs for the individual use case architecture alternatives in the portfolio design solution, either with or without commonality. The number of custom development projects is evaluated by summing up all the engine and fuselage development project for the three use cases in the portfolio: e.g. if the Saturn V use case requires three separate stages which each have different fuselages and engine development projects, the Saturn IB use case 2 stages which each have different engine and fuselage development projects, and the Saturn I use case 2 stages which each have different engine and fuselage development projects, then the total number of custom development projects is equal to (3+3)+(2+2)+(2+2) = 14.

3.2 Saturn Launch Vehicle Point Design Analyses

Step 2 of the commonality analysis methodology is devoted to the analysis of architectural alternatives for each of the use cases in the Saturn portfolio individually, i.e. without consideration for commonality opportunities. The analysis of architectural alternatives involves a comprehensive enumeration of architecture alternatives for each use case based on a set of architecture-level design factors and the subsequent evaluation of these alternatives with regard to the metrics outlined in Section 3.1. This evaluation is the basis for the down-selection to a preferred set of architecture alternatives for each use case as input to the commonality screening in Step 3 of the methodology. Each system architecture analysis in Step 2 represents an architectural decision problem as described by Simmons (2008).

Table 1: Morphological matrix for the Saturn V launch vehicle architecture analysis

Architectural decision	Technology choice 1	Technology choice 2	Technology choice 3	Technology choice 4
# of propulsion stages	3	2		
Thrust generation stage 3 – propellant type	LOX/LH2 (Isp = 421 s)	LOX/Kerosene (Isp = 310)	N2O4/UDMH (Isp = 308)	N/A
Thrust generation stage 3 – # of engines	1	2	5	N/A
Propellant storage stage 3	Common bulkhead tanks	2-tank structure	N/A	
Thrust generation stage 2 – propellant type	LOX/LH2 (Isp = 421 s)	LOX/Kerosene (Isp = 310)	N2O4/UDMH (Isp = 308)	
Thrust generation stage 2 – # of engines	1	2	5	
Propellant storage stage 2	Common bulkhead tanks	2-tank structure		
Thrust generation stage 1 – propellant type	LOX/Kerosene (Isp = 265)	N2O4/UDMH (Isp = 259)		
Thrust generation stage 1 – # of engines	2	5	8	
Propellant storage stage 1	2-tank structure	Multi-tank structure		

The enumeration of architecture alternatives for each of the use cases in the portfolio utilizes a morphological matrix (Pahl, Beitz 1996) which includes all feasible technology choices for the design factors of relevance for the use case. Table 1 shows the morphological matrix used for the Saturn V use case. The design factors included are: the number of propulsion stages on the vehicle (either 2 or 3; 1 is not practically feasible), and the technology choices for thrust generation, propellant storage, and propellant type for each of the stages on the vehicle. It should be noted that for 1^{st} stages, which would be used at ground-level, liquid hydrogen / liquid oxygen propulsion was not considered because this technology was immature when the Saturn launch vehicle family was being developed. For the same reason, solid propellant options were also not included in the analysis. For the 2-stage architectures 10 different values for the first stage delta-v are considered to vary the relative sizing of the propulsion stages (4000 – 6000 m/s first stage delta-v); this leads to different relative sizes of the 1^{st} and 2^{nd} stages in this case.

By selecting one technology choice from each row we can systematically enumerate a total of 3997 feasible architecture alternatives for the Saturn V use case, taking into account the constraint that if the number of propulsion stages equals 2, all choices for propulsion stage number 3 must equal "N/A", and that if the number of propulsion stages equals 3, the choices of "N/A" are not selectable. These constraints eliminate certain paths through the morphological matrix, thereby reducing the number of architecture alternatives from 20736 combinatorial ones in the matrix to the 3997 feasible ones.

The propulsion stages are sized in reverse order of usage using a mixture of physics-based models (such as the rocket equation, Formula 9) and scaling empirical models (e.g. Formula 11): the stage carrying the actual vehicle payload is sized first, then the next-lower propulsion stage using both the actual vehicle payload and the higher stage as payload, and so on. For each of these architecture alternatives, the propellant masses of the individual stages were determined using the rocket equation for each stage:

$$m_{\mathrm{Propellant}} = \left(m_{Payload} + m_{Fuselage} + n_{Engine} \cdot m_{Engine}\right) \left(\exp\left(\frac{\Delta v}{g_0 \cdot I_{sp}} \right) - 1 \right) \quad \textbf{Formula 9}$$

The engine dry mass can be estimated using the empirical relationship in Formula 10 adapted from (Larson, Pranke 2000); Formula 10 shows how to calculate the thrust required per engine:

$$m_{Engine} = \frac{\alpha \cdot Thrust_{Engine}}{\left(25.2 \cdot \ln\left(Thrust\right) - 80.7\right) 9.81} \quad \textbf{Formula 10}$$

$$Thrust_{Engine} = \left(m_{Payload} + m_{Fuselage} + n_{Engine} \cdot m_{Engine} + m_{\mathrm{Propellant}}\right) g_0 \cdot T/W \quad \textbf{Formula 11}$$

The mass of the stage fuselage can be estimated using the empirical relationship in Formula 12 which is based on interpolation of data provided by (Orloff 2001):

$$m_{Fuselage} = \beta \cdot Volume_{\mathrm{Propellants}} \cdot \left(\frac{V_{\mathrm{Re}\,ference}}{Volume_{\mathrm{Propellants}}} \right)^{0.1623} \quad \textbf{Formula 12}$$

The propellant volume is calculated using Formula 13:

$$Volume_{\mathrm{Propellants}} = \frac{m_{\mathrm{Propellant}}}{OTF + 1} \cdot \left(\frac{OTF}{\rho_{\mathrm{Oxidizer}}} + \frac{1}{\rho_{Fuel}} \right) \quad \textbf{Formula 13}$$

The constants α, β, and $V_{Reference}$ as well as the values for *OTF* (the ratio of oxidizer mass to fuel mass required by the engine), I_{sp}. *T/W* (the ratio of the stage thrust force to the weight force of the vehicle at the time of stage ignition), and n_{Engine} in Formulae 9-13 are determined by the choices in the Morphological Matrix, as is the number of engines. Table 6 in the Appendix provides values for these constants as a function of technology choice. Formulae 9-13 cannot be solved analytically; an iteration scheme was therefore implemented which initially sets the engine and fuselage masses to zero.

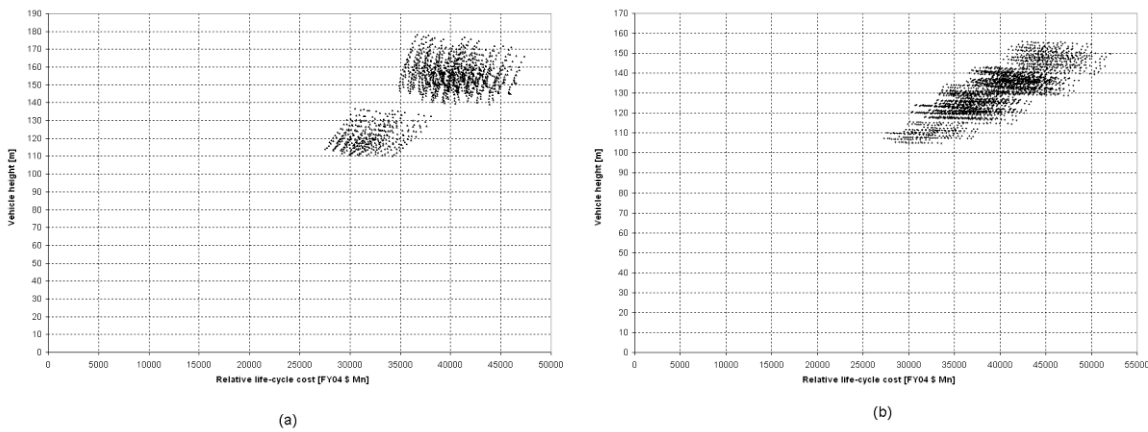


Figure 6: Point design architecture analysis results for the Saturn V use case: vehicle height vs. relative life-cycle cost ("Mn" stands for "million"); part (a) shows results for the 2-stage architectures, part (b) for the 3-stage architectures.

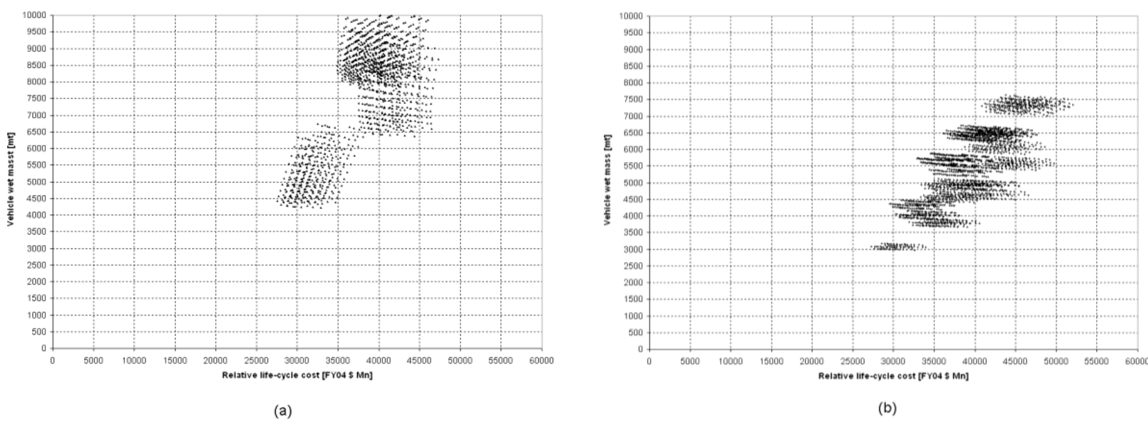


Figure 7: Point design architecture analysis results for the Saturn V use case: vehicle wet mass vs. relative life-cycle cost ("Mn" stands for "million"); part (a) shows results for the 2-stage architectures, part (b) for the 3-stage architectures

Figure 6 and Figure 7 show results from this enumeration and evaluation of architecture alternatives for the Saturn V use case. 2-stage alternatives generally results in increased vehicle height for similar life-cycle cost; this is understandable given that the significantly higher delta-v per stage well above the value of the exhaust velocity of the engine leads to much larger stage size. For vehicle launch mass, the increase due to choosing a 2-stage design is more pronounced than for vehicle height: the lowest-mass 2-stage alternatives require nearly 50% more launch mass than the lowest-mass 3-stage alternatives for similar life-cycle cost. The increased height and mass of two-stage alternatives would result in increased ground processing expenditures due to more demanding infrastructure requirements (building height, launch pad foundations, etc.) while not offering any life-cycle cost benefit (i.e. benefit in engine and fuselage development and production cost). In addition, 2-stage designs leave less performance margin for this high-delta-v use case (this is an attribute of the rocket equation). This makes 2-stage design solutions unattractive for the Saturn V use case; the further down-selection towards preferred architectures therefore focuses on 3-stage architectures only.

Table 2: Preferred point design architectures for the Saturn V use case; the historical preferred architecture is number 5.

Preferred architecture	LCC [FY04 $ Mn]	Stage 3			Stage 2			Stage 1		
		Structure	# engines	Propellant	Structure	# engines	Propellant	Structure	# engines	Propellant
1	27647	Common BH	2	LOX/LH2	Common BH	5	LOX/LH2	2-tank	8	LOX/RP1
2	28069	Common BH	1	LOX/LH2	Common BH	5	LOX/LH2	2-tank	8	LOX/RP1
3	28299	Common BH	2	LOX/LH2	Common BH	5	LOX/LH2	2-tank	5	LOX/RP1
4	28475	Common BH	2	LOX/LH2	Common BH	2	LOX/LH2	2-tank	8	LOX/RP1
5	28720	Common BH	1	LOX/LH2	Common BH	5	LOX/LH2	2-tank	5	LOX/RP1
6	28896	Common BH	1	LOX/LH2	Common BH	2	LOX/LH2	2-tank	8	LOX/RP1
7	28961	Common BH	2	LOX/LH2	Common BH	5	LOX/LH2	Multi-tank	8	LOX/RP1
8	29126	Common BH	2	LOX/LH2	Common BH	2	LOX/LH2	2-tank	5	LOX/RP1
9	29381	Common BH	1	LOX/LH2	Common BH	5	LOX/LH2	Multi-tank	8	LOX/RP1
10	29500	Common BH	2	LOX/LH2	Common BH	1	LOX/LH2	2-tank	8	LOX/RP1
11	29546	Common BH	1	LOX/LH2	Common BH	2	LOX/LH2	2-tank	5	LOX/RP1
12	29621	Common BH	2	LOX/LH2	Common BH	5	LOX/LH2	Multi-tank	5	LOX/RP1
13	29786	Common BH	2	LOX/LH2	Common BH	2	LOX/LH2	Multi-tank	8	LOX/RP1
14	29913	Common BH	2	LOX/LH2	Common BH	5	N2O4/UDMH	2-tank	8	LOX/RP1
15	29919	Common BH	1	LOX/LH2	Common BH	1	LOX/LH2	2-tank	8	LOX/RP1
16	30040	Common BH	1	LOX/LH2	Common BH	5	LOX/LH2	Multi-tank	5	LOX/RP1
17	30110	Common BH	2	LOX/LH2	Common BH	5	LOX/RP1	2-tank	8	LOX/RP1
18	30150	Common BH	2	LOX/LH2	Common BH	1	LOX/LH2	2-tank	5	LOX/RP1
19	30205	Common BH	1	LOX/LH2	Common BH	2	LOX/LH2	Multi-tank	8	LOX/RP1
20	30331	Common BH	1	LOX/LH2	Common BH	5	N2O4/UDMH	2-tank	8	LOX/RP1
21	30366	Common BH	2	LOX/LH2	Common BH	5	LOX/LH2	2-tank	2	LOX/RP1
22	30445	Common BH	2	LOX/LH2	Common BH	2	LOX/LH2	Multi-tank	5	LOX/RP1
23	30529	Common BH	1	LOX/LH2	Common BH	5	LOX/RP1	2-tank	8	LOX/RP1
24	30569	Common BH	1	LOX/LH2	Common BH	1	LOX/LH2	2-tank	5	LOX/RP1
25	30687	Common BH	2	LOX/LH2	Common BH	5	N2O4/UDMH	2-tank	5	LOX/RP1
26	30717	Common BH	2	N2O4/UDMH	Common BH	5	LOX/LH2	2-tank	8	LOX/RP1
27	30784	Common BH	1	LOX/LH2	Common BH	5	LOX/LH2	2-tank	2	LOX/RP1
28	30810	Common BH	2	LOX/LH2	Common BH	1	LOX/LH2	Multi-tank	8	LOX/RP1
29	30829	Common BH	2	LOX/RP1	Common BH	5	LOX/LH2	2-tank	8	LOX/RP1
30	30863	Common BH	1	LOX/LH2	Common BH	2	LOX/LH2	Multi-tank	5	LOX/RP1

The preferred point design solutions for the Saturn V are selected from the 3-stage architecture alternatives based on life-cycle cost ranking; life-cycle cost is used as the driving metric since the lowest-lifecycle cost architectures also are among architectures with low vehicle height and mass (see Figure 6 and Figure 7). Table 2 provides an overview of the 30 preferred point design architecture alternatives selected for the Saturn V use case ranked by life-cycle cost. Preferred architecture number 5 corresponds to the historical Saturn V design as implemented. From the preferred architectures it is apparent that the choice of propellant type is quite robust: for the third and second stages LOX / LH_2 propellants are preferred, and for the first stage LOX / RP1 propellants. In addition, common bulkhead fuselage designs are preferred for the third stage and the second stage, and 2-tank designs for the first stage. The preferred number of engines is significantly more variable in the set of preferred architectures; this is beneficial for enabling commonality opportunities with regard to engines in the Saturn portfolio. The use of hypergolic propellants does not reduce life-cycle cost; given their toxicity and the associated special ground processing requirements at the launch pad, hypergolic propellants do not seem to offer an advantage for the Saturn V use case and are not considered as preferred architectures.

An architecture enumeration, evaluation, and selection process identical to that for the Saturn V use case was carried out for the Saturn IB and Saturn I use cases; Figure 15 and Figure 16 and Table 7 and Table 8 in the Appendix show the associated results. For both the Saturn IB and Saturn I use cases, 2-stage vehicle architectures are preferred because 3-stage architectures exhibit somewhat higher life-cycle cost. The 30 preferred point design solutions based on life-cycle cost for these use cases are shown in Table 3 and Table 4 in order of life-cycle cost ranking. Preferred architectures 23, 26, and 28 in Table 3 are comparable to the historical Saturn IB vehicle architecture, albeit with varying delta-v allocations to the propulsion stages. Preferred architectures 12, 15, 19, 23 and 27 in Table 4 are comparable to the historical Saturn I vehicle architecture, also with varying delta-v allocations to the propulsion stages.

Table 3: Preferred point design architectures for the Saturn IB use case; different colors indicate different technology choices (delta-v values for stages not shown). Historical preferred architectures are marked with red boxes.

Preferred architecture	LCC [FY04 $ Mn]	Stage 2			Stage 1		
		Structure	# engines	Propellant	Structure	# engines	Propellant
1	10637	Common BH	5	LOX/LH2	2-tank	8	LOX/RP-1
2	11145	Common BH	5	LOX/LH2	2-tank	8	LOX/RP-1
3	10932	Common BH	5	LOX/LH2	2-tank	5	LOX/RP-1
4	11475	Common BH	5	LOX/LH2	2-tank	5	LOX/RP-1
5	11095	Common BH	2	LOX/LH2	2-tank	8	LOX/RP-1
6	11479	Common BH	2	LOX/LH2	2-tank	8	LOX/RP-1
7	11102	Common BH	5	LOX/LH2	Multi-tank	8	LOX/RP-1
8	11859	Common BH	5	LOX/LH2	Multi-tank	8	LOX/RP-1
9	11388	Common BH	2	LOX/LH2	2-tank	5	LOX/RP-1
10	11807	Common BH	2	LOX/LH2	2-tank	5	LOX/RP-1
11	11399	Common BH	5	LOX/LH2	Multi-tank	5	LOX/RP-1
12	11764	Common BH	5	LOX/LH2	Multi-tank	5	LOX/RP-1
13	11558	Common BH	2	LOX/LH2	Multi-tank	8	LOX/RP-1
14	12084	Common BH	2	LOX/LH2	Multi-tank	8	LOX/RP-1
15	11853	Common BH	2	LOX/LH2	Multi-tank	5	LOX/RP-1
16	12073	Common BH	2	LOX/LH2	Multi-tank	5	LOX/RP-1
17	11667	Common BH	1	LOX/LH2	2-tank	8	LOX/RP-1
18	11902	Common BH	1	LOX/LH2	2-tank	8	LOX/RP-1
19	11853	Common BH	5	LOX/LH2	2-tank	2	LOX/RP-1
20	12073	Common BH	5	LOX/LH2	2-tank	2	LOX/RP-1
21	11959	Common BH	1	LOX/LH2	2-tank	5	LOX/RP-1
22	12109	Common BH	1	LOX/LH2	2-tank	5	LOX/RP-1
23	12128	Common BH	1	LOX/LH2	Multi-tank	8	LOX/RP-1
24	12144	Common BH	5	LOX/LH2	2-tank	2	LOX/RP-1
25	12144	Common BH	2	LOX/LH2	Multi-tank	5	LOX/RP-1
26	12153	Common BH	1	LOX/LH2	Multi-tank	8	LOX/RP-1
27	12164	Common BH	1	LOX/LH2	2-tank	5	LOX/RP-1
28	12183	Common BH	1	LOX/LH2	Multi-tank	8	LOX/RP-1
29	12189	Common BH	2	LOX/LH2	Multi-tank	8	LOX/RP-1
30	12194	Common BH	5	LOX/LH2	Multi-tank	5	LOX/RP-1

Table 4: Preferred point design architectures for the Saturn I use case; different colors indicate different technology choices (delta-v values for stages not shown). Historical preferred architectures are marked with red boxes.

Preferred architecture	LCC [FY04 $ Mn]	Stage 2			Stage 1		
		Structure	# engines	Propellant	Structure	# engines	Propellant
1	7595	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
2	7629	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
3	7664	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
4	7702	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
5	7743	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
6	7789	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
7	7839	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
8	7850	Common BH	6	LOX/LH2	2-tank	5	LOX/RP-1
9	7885	Common BH	6	LOX/LH2	2-tank	5	LOX/RP-1
10	7895	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
11	7923	Common BH	6	LOX/LH2	2-tank	5	LOX/RP-1
12	7933	Common BH	6	LOX/LH2	Multi-tank	8	LOX/RP-1
13	7957	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
14	7964	Common BH	6	LOX/LH2	2-tank	5	LOX/RP-1
15	7985	Common BH	6	LOX/LH2	Multi-tank	8	LOX/RP-1
16	8009	Common BH	6	LOX/LH2	2-tank	5	LOX/RP-1
17	8026	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
18	8037	Common BH	2	LOX/LH2	2-tank	8	LOX/RP-1
19	8038	Common BH	6	LOX/LH2	Multi-tank	8	LOX/RP-1
20	8054	Common BH	2	LOX/LH2	2-tank	8	LOX/RP-1
21	8058	Common BH	6	LOX/LH2	2-tank	5	LOX/RP-1
22	8074	Common BH	2	LOX/LH2	2-tank	8	LOX/RP-1
23	8095	Common BH	6	LOX/LH2	Multi-tank	8	LOX/RP-1
24	8097	Common BH	2	LOX/LH2	2-tank	8	LOX/RP-1
25	8112	Common BH	6	LOX/LH2	2-tank	5	LOX/RP-1
26	8124	Common BH	2	LOX/LH2	2-tank	8	LOX/RP-1
27	8156	Common BH	6	LOX/LH2	Multi-tank	8	LOX/RP-1
28	8156	Common BH	2	LOX/LH2	2-tank	8	LOX/RP-1
29	8172	Common BH	6	LOX/LH2	2-tank	5	LOX/RP-1
30	8189	Common BH	6	LOX/LH2	Multi-tank	5	LOX/RP-1

Based on the 30 preferred point design architectures for each of the three use cases, we can enumerate a total of 30 x 30 x 30 = 27000 point design portfolio architectures without commonality; the choice of 30 preferred solutions each was based on processing limits of the Java compiler used for the implementation of the case study. It should be noted that this limit is not due to the calculation time imposed by the commonality screening in Step 3: the commonality screening took on the order of seconds of computing time, using a standard off-the-shelf laptop personal computer. This indicates that significantly more preferred architectures can be subjected to commonality screening in acceptable timeframes when using optimized computer architectures. Nevertheless, Tables 2-4 suggest that even with a limit of 30 preferred architectures per use case, considerable diversity in paths through the associated morphological matrices (as evidenced by different propellant combinations, numbers of engines, and fuselage types) can be included in the analysis.

Figure 9a shows the number of custom development projects of these portfolios plotted over the relative life-cycle cost for each of the custom portfolio design solutions. The number of development projects is constant at 14 for all portfolio design solutions: 7 engine developments and 7 fuselage developments, one for each of the three stages in the Saturn V use case, for each of the 2 stages in the Saturn IB use case, and for each of the 2 stages in the Saturn I use case; the result is that all portfolio design solutions without commonality lie on the same line with y-axis value of 14 in the diagram, while their lifecycle cost values vary considerably (see Figure 9a). The 27000 portfolio design solutions without commonality serve as input to the commonality screening process which transforms them into 27000 portfolio design solutions with commonality (see following Section 3.3).

3.3 Saturn Launch Vehicle Family Commonality Screening

		Operational environments	
		Sea-level	Altitude
Provide thrust generation	Scale: thrust [N]		
	N2O4/UDMH propulsion		
	LOX/Kerosene propulsion		
	LOX/LH2 propulsion		
Provide propellant storage and load transmission	Scale: fuel volume [m3]		
	Scale: oxidizer volume [m3]		
	N2O4/UDMH - Multi-tank structure - 8 engines		
	N2O4/UDMH - Multi-tank structure - 6 engines		
	N2O4/UDMH - Multi-tank structure - 5 engines		
	N2O4/UDMH - Multi-tank structure - 2 engines		
	N2O4/UDMH - Multi-tank structure - 1 engine		
	N2O4/UDMH - 2-tank structure - 8 engines		
	N2O4/UDMH - 2-tank structure - 6 engines		
	N2O4/UDMH - 2-tank structure - 5 engines		
	N2O4/UDMH - 2-tank structure - 2 engines		
	N2O4/UDMH - 2-tank structure - 1 engine		
	N2O4/UDMH - Common bulkhead tanks - 8 engines		
	N2O4/UDMH - Common bulkhead tanks - 6 engines		
	N2O4/UDMH - Common bulkhead tanks - 5 engines		
	N2O4/UDMH - Common bulkhead tanks - 2 engines		
	N2O4/UDMH - Common bulkhead tanks - 1 engine		
	LOX/Kerosene - Multi-tank structure - 8 engines		
	LOX/Kerosene - Multi-tank structure - 6 engines		
	LOX/Kerosene - Multi-tank structure - 5 engines		
	LOX/Kerosene - Multi-tank structure - 2 engines		
	LOX/Kerosene - Multi-tank structure - 1 engine		
	LOX/Kerosene - 2-tank structure - 8 engines		
	LOX/Kerosene - 2-tank structure - 6 engines		
	LOX/Kerosene - 2-tank structure - 5 engines		
	LOX/Kerosene - 2-tank structure - 2 engines		
	LOX/Kerosene - 2-tank structure - 1 engine		
	LOX/Kerosene - Common bulkhead tanks - 8 engines		
	LOX/Kerosene - Common bulkhead tanks - 6 engines		
	LOX/Kerosene - Common bulkhead tanks - 5 engines		
	LOX/Kerosene - Common bulkhead tanks - 2 engines		
	LOX/Kerosene - Common bulkhead tanks - 1 engine		
	LOX/LH2 - 2-tank structure - 8 engines		
	LOX/LH2 - 2-tank structure - 6 engines		
	LOX/LH2 - 2-tank structure - 5 engines		
	LOX/LH2 - 2-tank structure - 2 engines		
	LOX/LH2 - 2-tank structure - 1 engine		
	LOX/LH2 - Common bulkhead tanks - 8 engines		
	LOX/LH2 - Common bulkhead tanks - 6 engines		
	LOX/LH2 - Common bulkhead tanks - 5 engines		
	LOX/LH2 - Common bulkhead tanks - 2 engines		
	LOX/LH2 - Common bulkhead tanks - 1 engine		

Figure 8: system overlap matrix for commonality analysis of propulsion stages in the Saturn launch vehicle portfolio

Step 3 of the framework is the systematic screening of preferred architecture pairs for commonality opportunities for each of the 27000 portfolio design solutions without commonality. In the case of the Saturn portfolio, all pairs of propulsion stage designs were subjected to the commonality screening, resulting in 21 pairs for the 7 stages in the portfolio for each of the 27000 portfolio design solutions.

For the identification of opportunities for commonality as part of the screening process the four heuristic commonality criteria described above were used (see system overlap matrix in Figure 8):

Functional overlap criterion: commonality can only occur between pairs of engines and pairs of fuselage elements, but not between an engine and fuselage element.

Technology overlap criterion: for engine elements this means that the same propellant choice is required. For fuselage elements this means that the same propellant choice is required and in addition the same number of engines per stage and the same tank arrangement.

Operational overlap criterion: rocket engines are either designed for sea-level operations or for operations at altitude in near vacuum, resulting in radically different expansion ratios and nozzle designs; since the performance penalty of using an altitude engine at seas level or vice versa would be prohibitive, the overlap in operational environments for engines needs to be 100%. Stage fuselages typically also carry specialized requirements for ground interfacing, structural loading, and thermal control due to their particular active operating environments; this means that a seal-level fuselage would also not be used for altitude operations and vice versa, requiring 100% overlap in operational environments also for fuselages. This corresponds to a selection of a value of 100 % for the operational overlap parameter δ (complete operational overlap required).

Scale overlap criterion: the values of quantitative design parameters must be within a factor k (overlap parameter) of each other (see Formula 14 and 15) in order for two engines or two fuselage element designs to be common; this criterion applies to propellant volume for fuselage elements and to thrust for engine elements which are continuous variables and can therefore be subjected to numerical overlap analysis using Formulae 14 and 15.

$$Volume_{System\_1} \cdot \frac{1}{k} < Volume_{System\_2} < Volume_{System\_1} \cdot k \qquad \textbf{Formula 14}$$

$$Thrust_{System\_1} \cdot \frac{1}{k} < Thrust_{System\_2} < Thrust_{System\_1} \cdot k \qquad \textbf{Formula 15}$$

Figure 9b shows the transformed portfolio design solutions with commonality based on the application of the 4 above commonality criteria to each of the portfolio design solutions without commonality; the overlap parameter k was set to k = 2.0 for this analysis.

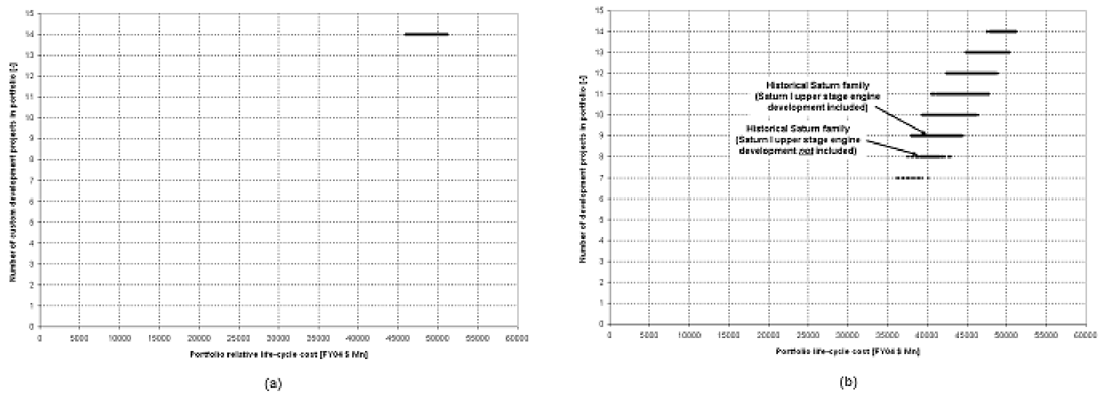


Figure 9: Evaluation of portfolio design solutions for the Saturn launch vehicle family for a scale overlap parameter setting of k = 2.0: # of development projects vs. portfolio life-cycle cost; 27000 portfolio design solutions without commonality are shown on the left in part (a), the same 27000 portfolio design solutions with commonality after screening in Step 3 shown on the right in part (b).

The introduction of commonality enables significant reductions both in life-cycle cost and in the number of custom propulsion stage fuselage and engine development projects. The minimum number of development projects is down to 7 from 14, and the minimum life-cycle cost moves from just over FY04 $ 45000 million ("Mn") to just over $ 35000 Mn. Figure 9b also shows the location of the historical common Saturn launch vehicle family as modeled in this case study. Figure 10 shows the life-cycle cost breakdown of the 27000 portfolio design solutions with and without commonality, ranked by life-cycle cost in the common case. In both cases, life-cycle cost consists of approximately equal parts of DDT&E and unit production costs for the propulsion stages. The introduction of commonality results in a reduction in both of these cost components: DDT&E cost is reduced due to the elimination of custom designs for fuselage and engine elements, and unit production cost is reduced due to the increased number of units produced for the common elements and the associated learning curve benefits.

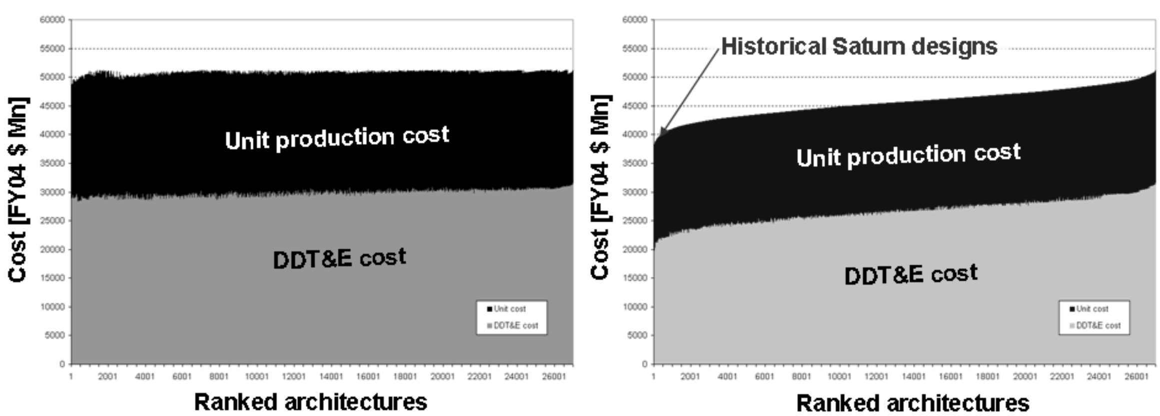


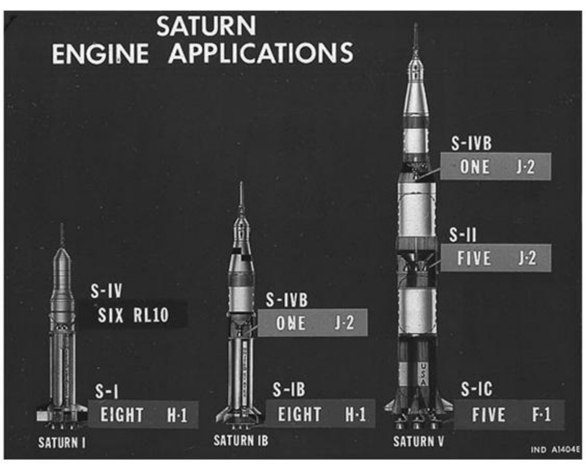
Figure 10: Life-cycle cost breakdown into DDT&E and unit cost for portfolio design solutions without commonality (left-hand side) and with commonality (right-hand side); k = 2.0

Figure 11 shows the commonality scheme implemented in the best-ranked portfolio design solution with commonality (see also Figure 12 for the Saturn family as implemented). A common engine design is utilized for the Saturn V third stage and for the Saturn IB and Saturn I second stages, as well as for the Saturn V second stage. In addition, a common fuselage design is employed for the Saturn V third stage and Saturn IB and Saturn I second stages.

Element	Common engine 1	Common engine 2	Common fuselage 1	Common fuselage 2
Saturn V: 3rd stage fuselage			X	
Saturn V: 3rd stage engine	X			
Saturn V: 2nd stage fuselage				
Saturn V: 2nd stage engine	X			
Saturn V: 1st stage fuselage				
Saturn V: 1st stage engine				
Saturn IB: 2nd stage fuselage			X	
Saturn IB: 2nd stage engine	X			
Saturn IB: 1st stage fuselage				X
Saturn IB: 1st stage engine		X		
Saturn I: 2nd stage fuselage				
Saturn I: 2nd stage engine				
Saturn I: 1st stage fuselage				X
Saturn I: 1st stage engine		X		

	Life-cycle cost [FY04 $ Mn]	# of development projects [-]
Custom	46780	14
Common	36124	7

Figure 11: Commonality opportunities for the portfolio design solution with commonality with the lowest life-cycle cost; k = 2.0



- **Propulsion stages:**
  - **S-IV**
  - **S-IV B**
  - **S-I**
  - **S-IC**
  - **S-II**
- **Rocket engines:**
  - **J-2**
  - **F-1**
  - **H-1**
  - **RL-10**

Figure 12: Saturn launch vehicle family design solution as implemented; image credit NASA

The historical Saturn portfolio design solution and commonality scheme was identified by the commonality screening process: it is ranked number 351, requires 9 development projects instead of 7, and has a higher life-cycle cost estimate ($39794 million as opposed to $36000 million, see also Figure 9b). The major difference between the historical Saturn portfolio and the best-ranked portfolio is the custom Saturn I upper stage. The use of a custom Saturn I upper stage in the historical portfolio can be understood when taking into account the development of the RL-10 engine for the Centaur upper stage preceded the Saturn I development (it was a de facto legacy element), thereby reducing the development cost associated with it. Given that discounting Saturn I upper stage engine development does not significantly reduce cost (see Figure 9b), the motivation for building a custom fuselage for the Saturn I upper stage must have been rooted in the desire to gain design and production experience for high-performance common bulkhead and multi-engine upper stages before the development of the S-IVB and S-II which needed to provide unprecedented structural performance for the low-margin lunar use case.

3.4 Sensitivity Analysis and Selection of Preferred Portfolio Design Solutions

The results from commonality screening presented in Section 3.3 are based on a value of k = 2.0 for the overlap parameter associated with Criterion 4. In order to assess the robustness of the results, a sensitivity analysis with regard to changes in the value of the overlap parameter k is carried out; k is varied between 1.0 (identical propellant volume and thrust required for commonality) and 3.0. Figure 13 provides an overview of the variation in the average number of custom development projects required across the 27000 portfolio design solutions as a function of the value of the overlap parameter k. For a parameter value of k = 1.0, no commonality opportunities are identified and the average number of developments remains 14 as in the custom case.

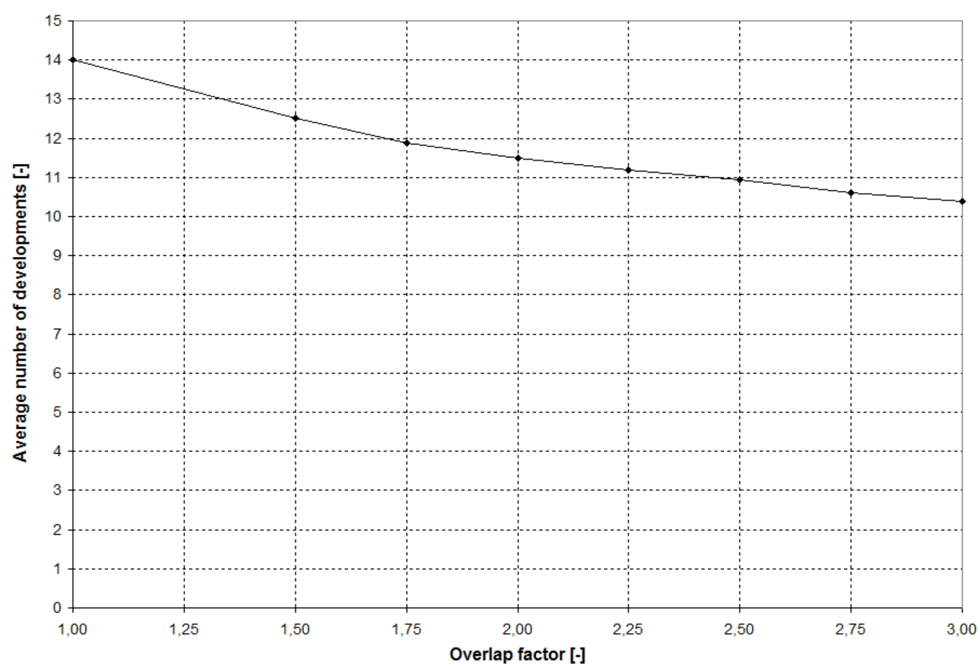


Figure 13: Sensitivity of the average number of development projects for the portfolio design solutions considered in the commonality analysis as a function of the overlap parameter k

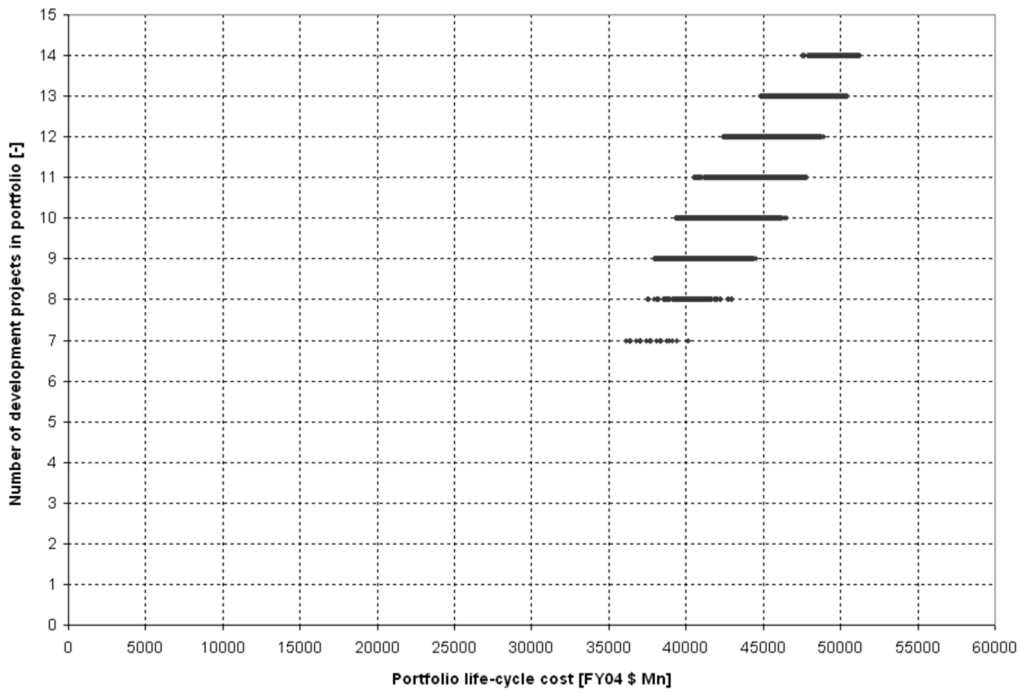


Figure 14: Evaluation of the 27000 portfolio design solutions with commonality for the Saturn launch vehicle family for a scale overlap parameter value of k = 1.5: # of development projects vs. portfolio life-cycle cost for each of the 27000 portfolio design solutions with commonality

In order to assess the impact of changing k on specific commonality opportunities and on life-cycle cost, a variant commonality analysis for a value of k = 1.5 was carried out. Results from this analysis are shown in Figure 14 with the number of development projects in the portfolio design solutions plotted over relative life-cycle cost. It is apparent that the implementation still results in significant reductions in the number of development projects and life-cycle cost, albeit slightly less pronounced than for the k = 2.0 case. It is interesting note that the commonality screening still identified portfolio design solutions similar to the historical Saturn portfolio design solution (identical technology choices and number of engines, somewhat different stage delta-v allocations); the first of these portfolio design solutions is now number 50 (instead of 351 for k = 2.0).

We can now also investigate the similarity in technology choices for the 50 lowest-ranked portfolio design solutions with regard to life-cycle cost for k = 1.5 and k = 2.0: see Table 5 for a line-by-line comparison of the 50 lowest-ranked portfolio design solutions in both cases. Shaded cells mark differences in technology choices between the portfolio design solutions with the same lifecycle cost rank (i.e. the same line), white cells indicate identity in technology choices between the portfolio design solutions with the same lifecycle cost rank). It is apparent that the preferred propellant choices, as well as the preferred fuselage structure choices are very robust to changes in the overlap parameter k. Pronounced variations occur for the number of engines for all three use cases; this observation corresponds well with the results from the architecture analysis in Section 3.2 with regard to variations in the preferred number of engines per stage for the individual use cases themselves.

Table 5: Overlap of technology choices for the 50 best-ranked portfolio design solutions with commonality for overlap parameter values of k = 2.0 and k = 1.5

Saturn V propellants			Saturn V # of engines			Saturn V stage structures			Saturn IB propellants		Saturn IB # of engines		Saturn IB stage structures		Saturn I propellants		Saturn I # of engines		Saturn I stage structures	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0	0	1	0	0	1	0	0	1	0	0	1
0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	1	0	1	0	0	0	0	0	1	0	0	1	0	0	1	0	0	1
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	1	1	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	1	0	0
0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	1	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	1	0	0
0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	0
0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1

4. Discussion

The following observations can be made based on the above application case study:

Observation 1: in Step 2, it is essential to not only consider system architectures which are non-dominated but to apply the concept of the fuzzy Pareto front (Smaling, de Weck 2007) because dominated solutions may yield superior portfolio design solutions with commonality in Steps 3 and 4 (this is an artifact of the choice of a 2-stage methodology). Ultimately it would be desirable to consider all system architectures for each use case; practically we are limited in the number of preferred system architectures per use case by the upper limit on portfolio design solutions which can be subjected to Steps 3 and 4 in acceptable time.

Observation 2: in Step 1, the number of dedicated subsystem developments in the portfolio was introduced as a proximate metric for developmental risk. A reduction in the number of dedicated development projects can generally be regarded as desirable because developmental risk scales with the number of projects. The Saturn case indicates that while a low number of development projects is good, lowest may not always be best. This is illustrated by the fact that the architects of the Saturn launch vehicle family deliberately gave up the moderate cost advantage of merging the Saturn I and Saturn IB use cases in order to gain developmental experience with the unprecedented development of clustered-engine upper stages. This was presumably in order to reduce developmental risk for the low-mass-margin Saturn IVB upper stage. This indicates that once a low number of dedicated developments has been reached, there is an additional trade-off between additional up-front developmental risk and total portfolio developmental risk.

Observation 3: in the Saturn case, sensitivity analysis with the regard to the overlap criteria in Step 4 indicates that the average number of dedicated developments varies only modestly with the value of the overlap parameter k and that there do not appear to be any significant step changes. Furthermore, the best-ranked portfolio design solutions appear to be robust to changes in the overlap factor from 2.0 to 1.5. This kind of sensitivity analysis should always be performed to check for and analyze step changes. This holds for both the operational and scale overlap criteria: while in the Saturn case the threshold for the overlap fraction δ was set to 1 (100%: only commonality between altitude stages and engines as well as between ground-level engines and stages considered), in a general case both k and δ may be different from 1 and therefore require sensitivity analysis. Please refer to (Hofstetter 2009) for a life support system case study which features a sensitivity analysis for parameter δ.

Observation 4: while the PLASCO methodology itself is intended for use during the portfolio architecting stage, elements of the methodology may be useful during later stages of the design process: a generalized SOM concept could be useful for identifying suitable candidates for commonality based on detailed design attributes on almost any level of design detail. It is also conceivable that the SOM could even serve as an automated tool to systematically search part databases for suitable commonality matches during the regular design process in order to encourage opportunistic reuse based on legacy elements (the type of commonality strategy which entails no up-front disadvantages).

There are three main limitations of the PLASCO methodology: the first is the limited number of preferred system architectures for each use case that can be considered in the commonality screening process due to the combinatorially increasing computational complexity as well as the need for some amount of manual evaluation of alternatives. The second limitation is that the output of the methodology is constrained by the quality of the system models used for calculating the metrics used in Step 2, Step 3, and Step 4. In order to ensure adequate quality of metric modeling, it is necessary to either use physics-based models or empirical engineering models grounded in past design experience. For the Saturn case study, a combination of physics-based models, such as the rocket equation, and empirical models, such as the sizing functions for engines and fuselages, was used. A further limitation by design is that PLASCO does not include an assessment of managerial and organizational feasibility of commonality, which is the third condition for the implementation of commonality (Boas, Crawley 2007).

5. Suggestions for Future Work

A number of opportunities for follow-on work to further develop the PLASCO methodology have been identified:

- The further application of the methodology to architecture and commonality studies for portfolios of complex systems other than aerospace systems, thereby demonstrating its general applicability.

- The development of higher-fidelity and generalized cost and risk models that capture aspects of portfolio lifecycle cost and risk that traditional analogy-based models cannot capture.
- Integration of an assessment of managerial and organizational feasibility into the commonality assessment and evaluation in Steps 3 and 4 of the methodology.
- Integration of MOD approaches into the methodology in order to increase the number of portfolio design solutions that can be included in the commonality screening.
- Integration of the custom architecture and commonality analysis codes into a single software application. The current implementation of the methodology features individual Java codes for the architecture analysis for each system in the portfolio as well as for the commonality analysis for each function; this leads to a significant need for manual data management which proved to be among the most time-consuming tasks when applying the methodology to a case study.

6. Conclusion

The development of portfolios of complex systems offers significant opportunities for using commonality between the constituent systems as a means to improve life-cycle properties such as cost and risk. The gap in the current state of the art is a lack of methodologies that allow for the identification of commonality opportunities in portfolios of aerospace systems during the architecting stage, when disadvantages of commonality can be mitigated most effectively.

This paper presents a novel 2-stage methodology called PLASCO for the architecting of aerospace systems portfolios with commonality (see Figure 1). The first stage, corresponding to Steps 1 and 2 of the methodology, is concerned with identifying preferred architectures for each of the systems in the portfolio individually (no consideration for commonality, point design solutions for each system, and thereby for the portfolio). This is achieved through a definition of portfolio scope (use cases and functionality) and metrics in Step 1, and through the enumeration and evaluation of architecture alternatives for each use case individually in Step 2, leading to the selection of preferred architecture alternatives for each use case. In the second stage, corresponding to Steps 3 and 4 of the methodology, a comprehensive screening of commonality opportunities among the preferred point design solutions in the portfolio is carried out based on four heuristic commonality criteria (Step 3). Step 4 involves a sensitivity analysis of portfolio design solutions with regard to changes in the heuristics as well as the subsequent selection of preferred portfolio design solutions with commonality. The methodology was applied to the retrospective analysis of NASA’s Saturn launch vehicle family which was used for the missions of the Apollo program in the 1960s. The Saturn case study demonstrates the practical usability of the methodology for the task of architecting a real aerospace systems portfolio.

Acknowledgements

The authors would like to thank Prof. Dr. Olivier L. de Weck (Massachusetts Institute of Technology) and Prof. Dr. rer. nat. Ulrich Walter (Technische Universität München, Germany) for their support and advice.

References

Belew LF (editor) (1977) Skylab - Our First Space Station, NASA-SP-400, National Aeronautics and Space Administration, Washington, D.C.

Bell J (1967) An Evolutionary Program for Manned Interplanetary Exploration, AIAA Journal of Spacecraft, VOL. 4, NO. 5, MAY 1967.

Bilstein RE (editor) (1996) Stages to Saturn - A Technological History of the Apollo/Saturn Launch Vehicles, NASA-SP-4206, The NASA History Series, National Aeronautics and Space Administration, NASA History Office, Washington, D.C.

Boas R, Crawley EF (2007) Divergence: The Impact of Lifecycle Changes on Commonality, Seventeenth Annual International Symposium of the International Council On Systems Engineering, 24 - 28 June 2007.

Burgelman RA, Christensen CM, Wheelwright SC, Maidique MA (2003) Strategic Management of Technology and Innovation, McGraw Hill Higher Education; 4th Revised edition.

Caffrey RT, Simpson TW, Henderson R, Crawley EF (2002) The Technical Issues with Implementing Spacecraft Open Avionics Platforms, AIAA 2002-0319, 40th Aerospace Sciences Meeting & Exhibit, 14-17 January 2002, Reno, Nevada.

Coan DA, Bell ER (2006) Essential Commonality for Effective Future Extravehicular Activity Operations, AIAA 2006-5953, Space Operations 2006 Conference.

Cooper RG, Edgett SJ, Kleinschmidt EJ (2001) Portfolio Management for New Products, 2nd Edition, Perseus Books, Cambridge, MA.

Cortright EM (editor) (1975) Apollo - Expeditions to the Moon, NASA-SP-350, Scientific and Technical Information Office, National Aeronautics and Space Administration Washington, DC.

Dahmus JB, Gonzalez-Zugasti JP, Otto KN (2001) "Modular Product Architecture," Design Studies, 22(5), 409-424.

Dickinson MW, Thornton AC, Graves S (2001) Technology Portfolio Management: Optimizing Interdependent Projects Over Multiple Time Periods, IEEE Transactions on Engineering Management, Vol. 48, No. 4.

Dobrescu G, Reich Y (2003) Progressive sharing of modules among product variants, Computer-Aided Design 35, 791–806.

Dori D (2002) Object-Process Methodology – A Holistic Systems Paradigm, Springer, Berlin/New York.

Egbert NF, McKain TF (1994) Allison Engine Company, Common Core Development Approach for Allison T406/AE Family of Turboshaft, Turboprop, and Turbofan Engines, AIAA 94-2829, 30th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, June 27-29, 1994, Indianapolis, Indiana.

Ezell EC, Ezell LN (editors) (1978) The Partnership: A History of the Apollo-Soyuz Test Project, NASA-SP-4209, The NASA History Series, National Aeronautics and Space Administration Scientific and Technical Information Office, Washington, DC.

Fellini, R., Kokkolaras, M., Papalambros, P. and Perez-Duarte, A. (2005) "Platform Selection Under Performance Loss Constraints in Optimal Design of Product Families," ASME Journal of Mechanical Design, 127(4), 524-535.

Fujita, K. (2002) "Product Variety Optimization Under Modular Architecture," Computer-Aided Design, 34(12), 953-965.

Gonzalez-Zugasti, J. P., Otto, K., Baker, J. D., A Method for Architecting Product Platforms, Journal of Research in Engineering Design (2000)12:61-72.

Gonzalez-Zugasti, J. P., Otto, K. N. and Baker, J. D. (2001) "Assessing Value for Platformed Product Family Design," Research in Engineering Design, 13(1), 30-41.

Hodson, R. F. (2007) Reuse and Interoperability of Avionics for Space Systems, AIAA 207-2918, 2007.

Hoffman EJ (editor) (1995) NASA Systems Engineering Handbook, NASA-SP-610S, Washington DC.

Hofstetter WK, Wooster PD, de Weck OL, Crawley EF (2007) The System Overlap Matrix – A Method and Tool for the Systematic Identification of Commonality Opportunities in Complex Technical Systems, 9th International Design Structure Matrix Conference, DSM’07 16–18 October 2007, Munich, Germany.

Hofstetter WK (2009) A Framework for the Architecting of Aerospace Systems Portfolios with Commonality, doctoral thesis according to the requirements of the Department of Aeronautics and Astronautics, Massachusetts Institute of Technology.

Kalligeros K, de Weck O, de Neufville R, Luckins A (2006) Platform Identification using Design Structure Matrices, Sixteenth Annual International Symposium of the

International Council On Systems Engineering (INCOSE), Orlando, Florida, 8 - 14 July 2006.

Kalligeros K (2006) Platforms and Real Options in Large-Scale Engineering Systems, Doctoral Thesis, Engineering Systems Division, Massachusetts Institute of Technology, Cambridge.

Khajavirad, A., Michalak, J. J. and Simpson, T. W. (2009) "An Efficient Decomposed Multiobjective Genetic Algorithm for Solving the Joint Product Platform Selection and Product Family Design Problem with Generalized Commonality," Structural and Multidisciplinary Optimization, 39(2), 187-201.

Khalil TM (2000) Management of Technology: The Key to Competitiveness and Wealth Creation, McGraw-Hill, Boston.

Larson W J, Pranke LK (editors) (2000), Human Spaceflight – Mission Analysis and Design, McGraw-Hill, New York.

Maier MW, Rechtin E (2000) The Art of Systems Architecting, CRC Press Inc.

Martin, M. V., Ishii, K. (2002) Design for Variety: Developing Standardized and Modularized Product Platform Architectures, Journal of Research in Engineering Design 13 (2002) 213–235.

Meyer MH, Lenherd AP (1997) The Power of Product Platforms – Building Value and Cost Leadership, The Free Press, New York.

Messac, A., Martinez, M. P. and Simpson, T. W. (2002) "Effective Product Family Design Using Physical Programming," Engineering Optimization, 34(3), 245-261.

Nayak RU, Chen W, Simpson TW (2002) A Variation-Based Method for Product Family Design, Journal of Engineering Optimization, 2002, Vol. 34(1), pp. 65–81.

National Aeronautics and Space Administration (2005), Exploration Systems Architecture Study –Final Report, NASA-TM-2005-214062.

NASA JSC (2007) Spacecraft / Vehicle Level Cost Model http://cost.jsc.nasa.gov/SVLCM.html

Orloff RW (editor) (2001) Apollo By The Numbers: A Statistical Reference, NASA-SP-4029, National Aeronautics and Space Administration, NASA History Division.

Otto K, Wood K, Product Design – Techniques in Reverse Engineering and New Product Development, Prentice Hall, NJ, 2001.

Otto K, Hölttä-Otto K (2007) "A Multi-Criteria Assessment Tool for Screening

Preliminary Product Platform Concepts," Journal of Intelligent Manufacturing, 18(1), 59-75.

Pahl G, Beitz W (1996) Engineering Design – A Systematic Approach, 2nd edition, Springer, London.

Reinhart O, Schaefer I, Fricke E (2001) Platform Strategy in Airship Development, AIAA Lighter Than Air Convention.

Sered Y, Reich Y (2006) Standardization and modularization driven by minimizing overall process effort, Computer-Aided Design 38, 405–416

Siddiqi A, de Weck O (2007) "Spare Parts Requirements for Space Missions with Reconfigurability and Commonality", Journal of Spacecraft and Rockets, 44 (1), 147-155, January-February 2007.

Simmons WL (2008) A Framework for Decision Support in Systems Architecting. PhD Thesis, Dept. of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA.

Simpson, T. W., Maier, J. R. A., Mistree, F. (2001) Product Platform Design: Method and Application, Journal of Research in Engineering Design 13 2-22.

Smaling R, de Weck O (2007) Assessing Risks and Opportunities of Technology Infusion in System Design, Systems Engineering, 10 (1), 1-25.

Suh ES, de Weck OL, Chang D (2007) "Flexible product platforms: framework and case study", Journal of Research in Engineering Design, 18 (2), 67-89.

Thomas LD (1989) A Methodology for Commonality Analysis, With Application to Selected Space Station Systems, NASA-TM-100364, Science and Engineering Directorate, Space Station Project Engineering Office, NASA.

Waiss RD (1987) Cost Reduction on Large Space Systems through Commonality, AAA - 1987-0585, AIAA 25th Aerospace Sciences Meeting,.

Willcox, K., Wakayama, S. (2002) Simultaneous Optimization of a Multiple-Aircraft Family, AIAA 2002-1423, 43rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, 22-25 April 2002, Denver, Colorado.

Zhang WY, Tor SY, Britton GA (2006) Managing Modularity in Product Family Design with Functional Modeling, International Journal of Advanced Manufacturing Technology 30: 579–588.

Appendix

Table 6: Design parameters for the modeling of engines and propulsion stage fuselages

Function: thrust generation					
Propellant combination	Isp altitude [s]	Isp sea level [s]	OTF [-]	Density oxidizer [kg/m3]	Density fuel [kg/m3]
LOX / LH2	421	N/A	5.5	1141	70.8
LOX / RP1	310	265	2.27	1141	817
N2O4 / UDMH	308	259	1.6	1434	870
Function: propellant storage					
Fuselge design	Constant beta [kg/m3]		Reference volume [m3]		
Common bulkhead design	39.409		310		
2-tank design	65.522		2110		
Multi-tank design	103.234		359		
Thrust parameters					
Parameter	Stage 1	Stage 2		Stage 3	
T/W [-]	1.17	1.17		1.17	
Constant alpha [-]	1	0.8		0.6	

Table 7: Morphological Matrix of technology choices for the Saturn IB use case

Function	Technology choice 1	Technology choice 2	Technology choice 3	Technology choice 4
# of propulsion stages	3	2		
Thrust generation stage 3 – propellant type	LOX/LH2 (Isp = 421 s)	LOX/Kerosene (Isp = 310)	N2O4/UDMH (Isp = 308)	N/A
Thrust generation stage 3 – # of engines	1	2	5	N/A
Propellant storage stage 3	Common bulkhead tanks	2-tank structure	N/A	
Thrust generation stage 2 – propellant type	LOX/LH2 (Isp = 421 s)	LOX/Kerosene (Isp = 310)	N2O4/UDMH (Isp = 308)	
Thrust generation stage 2 – # of engines	1	2	5	
Propellant storage stage 2	Common bulkhead tanks	2-tank structure		
Thrust generation stage 1 – propellant type	LOX/Kerosene (Isp = 265)	N2O4/UDMH (Isp = 259)		
Thrust generation stage 1 – # of engines	2	5	8	
Propellant storage stage 1	2-tank structure	Multi-tank structure		

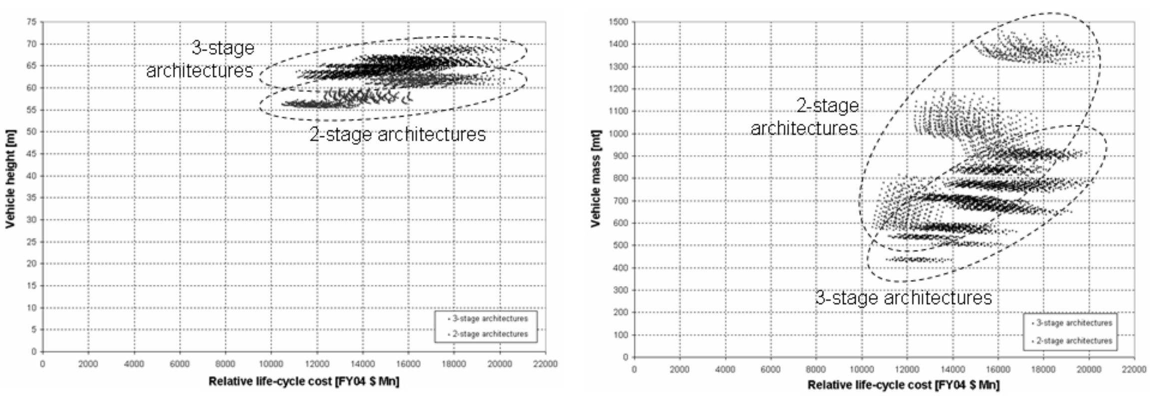


Figure 15: Point design architecture analysis results for the Saturn IB use case. Left diagram: vehicle height [m] vs. lifecycle cost; right diagram: vehicle wet mass vs. relative life-cycle cost

Table 8: Morphological Matrix of technology choices for the Saturn I use case

Function	Technology choice 1	Technology choice 2	Technology choice 3	Technology choice 4
# of propulsion stages	3	2		
Thrust generation stage 3 – propellant type	LOX/LH2 (Isp = 421 s)	LOX/Kerosene (Isp = 310)	N2O4/UDMH (Isp = 308)	N/A
Thrust generation stage 3 – # of engines	1	2	6	N/A
Propellant storage stage 3	Common bulkhead tanks	2-tank structure	N/A	
Thrust generation stage 2 – propellant type	LOX/LH2 (Isp = 421 s)	LOX/Kerosene (Isp = 310)	N2O4/UDMH (Isp = 308)	
Thrust generation stage 2 – # of engines	1	2	5 / 6	
Propellant storage stage 2	Common bulkhead tanks	2-tank structure		
Thrust generation stage 1 – propellant type	LOX/Kerosene (Isp = 265)	N2O4/UDMH (Isp = 259)		
Thrust generation stage 1 – # of engines	2	5	8	
Propellant storage stage 1	2-tank structure	Multi-tank structure		

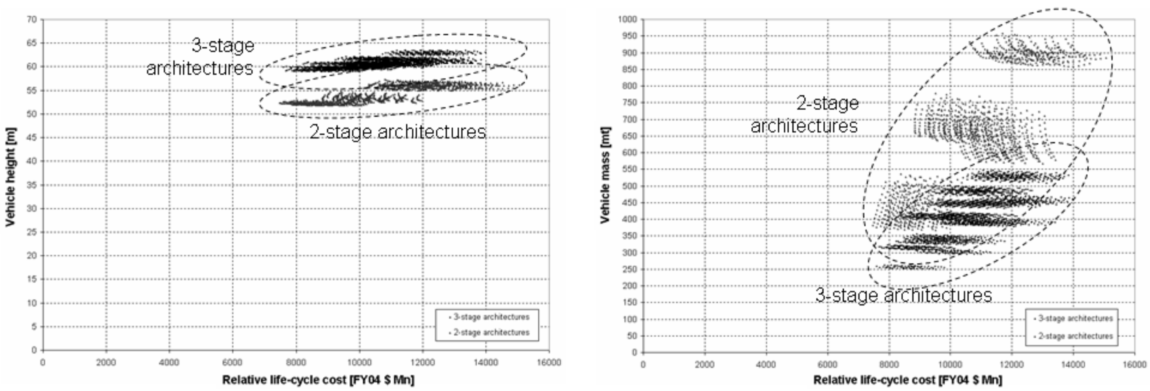


Figure 16: Point design architecture analysis results for the Saturn I use case. Left diagram: vehicle height [m] vs. lifecycle cost; right diagram: vehicle wet mass vs. relative life-cycle cost

Table 1: Morphological matrix for the Saturn V launch vehicle architecture analysis

Architectural decision	**Technology choice 1**	**Technology choice 2**	**Technology choice 3**	**Technology choice 4**
# of propulsion stages	3	2		
Thrust generation stage 3 – propellant type	LOX/LH2 (Isp = 421 s)	LOX/Kerosene (Isp = 310)	N2O4/UDMH (Isp = 308)	N/A
Thrust generation stage 3 – # of engines	1	2	5	N/A
Propellant storage stage 3	Common bulkhead tanks	2-tank structure	N/A	
Design delta-v stage 3 [m/s]	4159 for 3-stage case	N/A in 2-stage case		

Thrust generation stage 2 – propellant type	LOX/LH2 (Isp = 421 s)	LOX/Kerosene (Isp = 310)	N2O4/UDMH (Isp = 308)	
Thrust generation stage 2 – # of engines	1	2	5	
Propellant storage stage 2	Common bulkhead tanks	2-tank structure		
Design delta-v stage 2 [m/s]	4700 for 3-stage case	6259 – 8259 in 2-stage case		
Thrust generation stage 1 – propellant type	LOX/Kerosene (Isp = 265)	N2O4/UDMH (Isp = 259)		
Thrust generation stage 1 – # of engines	2	5	8	
Propellant storage stage 1	2-tank structure	Multi-tank structure		
Design delta-v stage 1 [m/s]	3400 for 3-stage case	4000 – 6000 for 2-stage case		

Table 2: Preferred point design architectures for the Saturn V use case; the historical preferred architecture is number 5.

Preferred architecture	LCC [FY04 $ Mn]	Stage 3			Stage 2			Stage 1		
		Structure	# engines	Propellant	Structure	# engines	Propellant	Structure	# engines	Propellant
1	27647	Common BH	2	LOX/LH2	Common BH	5	LOX/LH2	2-tank	8	LOX/RP1
2	28069	Common BH	1	LOX/LH2	Common BH	5	LOX/LH2	2-tank	8	LOX/RP1
3	28299	Common BH	2	LOX/LH2	Common BH	5	LOX/LH2	2-tank	5	LOX/RP1
4	28475	Common BH	2	LOX/LH2	Common BH	2	LOX/LH2	2-tank	8	LOX/RP1
5	28720	Common BH	1	LOX/LH2	Common BH	5	LOX/LH2	2-tank	5	LOX/RP1
6	28896	Common BH	1	LOX/LH2	Common BH	2	LOX/LH2	2-tank	8	LOX/RP1
7	28961	Common BH	2	LOX/LH2	Common BH	5	LOX/LH2	Multi-tank	8	LOX/RP1
8	29126	Common BH	2	LOX/LH2	Common BH	2	LOX/LH2	2-tank	5	LOX/RP1
9	29381	Common BH	1	LOX/LH2	Common BH	5	LOX/LH2	Multi-tank	8	LOX/RP1
10	29500	Common BH	2	LOX/LH2	Common BH	1	LOX/LH2	2-tank	8	LOX/RP1
11	29546	Common BH	1	LOX/LH2	Common BH	2	LOX/LH2	2-tank	5	LOX/RP1
12	29621	Common BH	2	LOX/LH2	Common BH	5	LOX/LH2	Multi-tank	5	LOX/RP1
13	29786	Common BH	2	LOX/LH2	Common BH	2	LOX/LH2	Multi-tank	8	LOX/RP1
14	29913	Common BH	2	LOX/LH2	Common BH	5	N2O4/UDMH	2-tank	8	LOX/RP1
15	29919	Common BH	1	LOX/LH2	Common BH	1	LOX/LH2	2-tank	8	LOX/RP1
16	30040	Common BH	1	LOX/LH2	Common BH	5	LOX/LH2	Multi-tank	5	LOX/RP1
17	30110	Common BH	2	LOX/LH2	Common BH	5	LOX/RP1	2-tank	8	LOX/RP1

18	30150	Common BH	2	LOX/LH2	Common BH	1	LOX/LH2	2-tank	5	LOX/RP1
19	30205	Common BH	1	LOX/LH2	Common BH	2	LOX/LH2	Multi-tank	8	LOX/RP1
20	30331	Common BH	1	LOX/LH2	Common BH	5	N2O4/UDMH	2-tank	8	LOX/RP1
21	30366	Common BH	2	LOX/LH2	Common BH	5	LOX/LH2	2-tank	2	LOX/RP1
22	30445	Common BH	2	LOX/LH2	Common BH	2	LOX/LH2	Multi-tank	5	LOX/RP1
23	30529	Common BH	1	LOX/LH2	Common BH	5	LOX/RP1	2-tank	8	LOX/RP1
24	30569	Common BH	1	LOX/LH2	Common BH	1	LOX/LH2	2-tank	5	LOX/RP1
25	30687	Common BH	2	LOX/LH2	Common BH	5	N2O4/UDMH	2-tank	5	LOX/RP1
26	30717	Common BH	2	N2O4/UDMH	Common BH	5	LOX/LH2	2-tank	8	LOX/RP1
27	30784	Common BH	1	LOX/LH2	Common BH	5	LOX/LH2	2-tank	2	LOX/RP1
28	30810	Common BH	2	LOX/LH2	Common BH	1	LOX/LH2	Multi-tank	8	LOX/RP1
29	30829	Common BH	2	LOX/RP1	Common BH	5	LOX/LH2	2-tank	8	LOX/RP1
30	30863	Common BH	1	LOX/LH2	Common BH	2	LOX/LH2	Multi-tank	5	LOX/RP1

Table 3: Preferred point design architectures for the Saturn IB use case; different colors indicate different technology choices (delta-v values for stages not shown). Historical preferred architectures are marked with red boxes.

Preferred architecture	LCC [FY04 $ Mn]	Stage 2			Stage 1		
		Structure	# engines	Propellant	Structure	# engines	Propellant
1	10637	Common BH	5	LOX/LH2	2-tank	8	LOX/RP-1
2	11145	Common BH	5	LOX/LH2	2-tank	8	LOX/RP-1
3	10932	Common BH	5	LOX/LH2	2-tank	5	LOX/RP-1
4	11475	Common BH	5	LOX/LH2	2-tank	5	LOX/RP-1
5	11095	Common BH	2	LOX/LH2	2-tank	8	LOX/RP-1
6	11479	Common BH	2	LOX/LH2	2-tank	8	LOX/RP-1
7	11102	Common BH	5	LOX/LH2	Multi-tank	8	LOX/RP-1
8	11859	Common BH	5	LOX/LH2	Multi-tank	8	LOX/RP-1
9	11388	Common BH	2	LOX/LH2	2-tank	5	LOX/RP-1
10	11807	Common BH	2	LOX/LH2	2-tank	5	LOX/RP-1
11	11399	Common BH	5	LOX/LH2	Multi-tank	5	LOX/RP-1
12	11764	Common BH	5	LOX/LH2	Multi-tank	5	LOX/RP-1
13	11558	Common BH	2	LOX/LH2	Multi-tank	8	LOX/RP-1
14	12084	Common BH	2	LOX/LH2	Multi-tank	8	LOX/RP-1
15	11853	Common BH	2	LOX/LH2	Multi-tank	5	LOX/RP-1
16	12073	Common BH	2	LOX/LH2	Multi-tank	5	LOX/RP-1
17	11667	Common BH	1	LOX/LH2	2-tank	8	LOX/RP-1
18	11902	Common BH	1	LOX/LH2	2-tank	8	LOX/RP-1
19	11853	Common BH	5	LOX/LH2	2-tank	2	LOX/RP-1
20	12073	Common BH	5	LOX/LH2	2-tank	2	LOX/RP-1
21	11959	Common BH	1	LOX/LH2	2-tank	5	LOX/RP-1
22	12109	Common BH	1	LOX/LH2	2-tank	5	LOX/RP-1
23	12128	Common BH	1	LOX/LH2	Multi-tank	8	LOX/RP-1
24	12144	Common BH	5	LOX/LH2	2-tank	2	LOX/RP-1
25	12144	Common BH	2	LOX/LH2	Multi-tank	5	LOX/RP-1
26	12153	Common BH	1	LOX/LH2	Multi-tank	8	LOX/RP-1
27	12164	Common BH	1	LOX/LH2	2-tank	5	LOX/RP-1
28	12183	Common BH	1	LOX/LH2	Multi-tank	8	LOX/RP-1
29	12189	Common BH	2	LOX/LH2	Multi-tank	8	LOX/RP-1
30	12194	Common BH	5	LOX/LH2	Multi-tank	5	LOX/RP-1

Table 4: Preferred point design architectures for the Saturn I use case; different colors indicate different technology choices (delta-v values for stages not shown). Historical preferred architectures are marked with red boxes.

Preferred architecture	LCC [FY04 $ Mn]	Stage 2			Stage 1		
		Structure	# engines	Propellant	Structure	# engines	Propellant
1	7595	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
2	7629	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
3	7664	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
4	7702	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
5	7743	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
6	7789	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
7	7839	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
8	7850	Common BH	6	LOX/LH2	2-tank	5	LOX/RP-1
9	7885	Common BH	6	LOX/LH2	2-tank	5	LOX/RP-1
10	7895	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
11	7923	Common BH	6	LOX/LH2	2-tank	5	LOX/RP-1
12	7933	Common BH	6	LOX/LH2	Multi-tank	8	LOX/RP-1
13	7957	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
14	7964	Common BH	6	LOX/LH2	2-tank	5	LOX/RP-1
15	7985	Common BH	6	LOX/LH2	Multi-tank	8	LOX/RP-1
16	8009	Common BH	6	LOX/LH2	2-tank	5	LOX/RP-1
17	8026	Common BH	6	LOX/LH2	2-tank	8	LOX/RP-1
18	8037	Common BH	2	LOX/LH2	2-tank	8	LOX/RP-1
19	8038	Common BH	6	LOX/LH2	Multi-tank	8	LOX/RP-1
20	8054	Common BH	2	LOX/LH2	2-tank	8	LOX/RP-1
21	8058	Common BH	6	LOX/LH2	2-tank	5	LOX/RP-1
22	8074	Common BH	2	LOX/LH2	2-tank	8	LOX/RP-1
23	8095	Common BH	6	LOX/LH2	Multi-tank	8	LOX/RP-1
24	8097	Common BH	2	LOX/LH2	2-tank	8	LOX/RP-1
25	8112	Common BH	6	LOX/LH2	2-tank	5	LOX/RP-1
26	8124	Common BH	2	LOX/LH2	2-tank	8	LOX/RP-1
27	8156	Common BH	6	LOX/LH2	Multi-tank	8	LOX/RP-1
28	8156	Common BH	2	LOX/LH2	2-tank	8	LOX/RP-1
29	8172	Common BH	6	LOX/LH2	2-tank	5	LOX/RP-1
30	8189	Common BH	6	LOX/LH2	Multi-tank	5	LOX/RP-1

Table 5: Overlap of technology choices for the 50 best-ranked portfolio design solutions with commonality for overlap parameter values of k = 2.0 and k = 1.5

Saturn V propellants			Saturn V # of engines			Saturn V stage structures			Saturn IB propellants		Saturn IB # of engines		Saturn IB stage structures		Saturn I propellants		Saturn I # of engines		Saturn I stage structures	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0	0	1	0	0	1	0	0	1	0	0	1
0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	1	0	1	0	0	0	0	0	1	0	0	1	0	0	1	0	0	1
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	1	1	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	1	0	0
0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	1	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	1	0	0

0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	0
0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1

Table 6: Design parameters for the modeling of engines and propulsion stage fuselages

Function: thrust generation					
Propellant combination	Isp altitude [s]	Isp sea level [s]	OTF [-]	Density oxidizer [kg/m3]	Density fuel [kg/m3]
LOX / LH2	421	N/A	5,5	1141	70,8
LOX / RP1	310	265	2,27	1141	817
N2O4 / UDMH	308	259	1,6	1434	870
Function: propellant storage					
Fuselge design	Constant beta [kg/m3]		Reference volume [m3]		
Common bulkhead design	39,409		310		
2-tank design	65,522		2110		
Multi-tank design	103,234		359		
Thrust parameters					
Parameter	Stage 1	Stage 2	Stage 3		
T/W [-]	1,17	1,17	1,17		
Constant alpha [-]	1	0,8	0,6		

Table 7: Morphological Matrix of technology choices for the Saturn IB use case

Function	**Technology choice 1**	**Technology choice 2**	**Technology choice 3**	**Technology choice 4**
# of propulsion stages	3	2		
Thrust generation stage 3 – propellant type	LOX/LH2 (Isp = 421 s)	LOX/Kerosene (Isp = 310)	N2O4/UDMH (Isp = 308)	N/A
Thrust generation stage 3 – # of engines	1	2	5	N/A
Propellant storage stage 3	Common bulkhead tanks	2-tank structure	N/A	
Thrust generation stage 2 – propellant type	LOX/LH2 (Isp = 421 s)	LOX/Kerosene (Isp = 310)	N2O4/UDMH (Isp = 308)	
Thrust generation stage 2 – # of engines	1	2	5	
Propellant storage stage 2	Common bulkhead tanks	2-tank structure		
Thrust generation stage 1 – propellant type	LOX/Kerosene (Isp = 265)	N2O4/UDMH (Isp = 259)		
Thrust generation stage 1 – # of engines	2	5	8	
Propellant storage stage 1	2-tank structure	Multi-tank structure		

Table 8: Morphological Matrix of technology choices for the Saturn I use case

Function	Technology choice 1	Technology choice 2	Technology choice 3	Technology choice 4
# of propulsion stages	3	2		
Thrust generation stage 3 – propellant type	LOX/LH2 (Isp = 421 s)	LOX/Kerosene (Isp = 310)	N2O4/UDMH (Isp = 308)	N/A
Thrust generation stage 3 – # of engines	1	2	6	N/A
Propellant storage stage 3	Common bulkhead tanks	2-tank structure	N/A	
Thrust generation stage 2 – propellant type	LOX/LH2 (Isp = 421 s)	LOX/Kerosene (Isp = 310)	N2O4/UDMH (Isp = 308)	
Thrust generation stage 2 – # of engines	1	2	5 / 6	
Propellant storage stage 2	Common bulkhead tanks	2-tank structure		
Thrust generation stage 1 – propellant type	LOX/Kerosene (Isp = 265)	N2O4/UDMH (Isp = 259)		
Thrust generation stage 1 – # of engines	2	5	8	
Propellant storage stage 1	2-tank structure	Multi-tank structure		