

Model-Based Systems Engineering in Concurrent Engineering Centers

Curtis Iwata¹

The Aerospace Corporation, 2310 E. El Segundo Blvd., El Segundo, CA 90245-4691

Samantha Infeld²

Analytical Mechanics Associates, Inc., 21 Enterprise Parkway Suite 300, Hampton, VA, 23666-6413

Jennifer Medlin Bracken³

NASA Goddard Space Flight Center, 8800 Greenbelt Rd., Greenbelt, MD, 20771

Melissa McGuire⁴

NASA Glenn Research Center, 21000 Brookpark Road, Cleveland, Ohio 44135

Christina McQuirk⁵ and Aron Kisdi⁶

RAL Space, Harwell Oxford, Didcot, Oxfordshire, OX11 0QX, UK

and

Jonathan Murphy⁷, Bjorn Cole⁸, and Pezhman Zarifian⁹

Jet Propulsion Laboratory, 4800 Oak Grove Ave, Pasadena, CA 91109

Concurrent Engineering Centers (CECs) are specialized facilities with a goal of generating and maturing engineering designs by enabling rapid design iterations. This is accomplished by co-locating a team of experts (either physically or virtually) in a room with a narrow design goal and a limited timeline of a week or less. The systems engineer uses a model of the system to capture the relevant interfaces and manage the overall architecture. A single model that integrates other design information and modeling allows the entire team to visualize the concurrent activity and identify conflicts more efficiently, potentially resulting in a systems model that will continue to be used throughout the project lifecycle. Performing systems engineering using such a system model is the definition of model-based systems engineering (MBSE); therefore, CECs evolving their approach to incorporate advances in MBSE are more successful in reducing time and cost needed to meet study goals. This paper surveys space mission CECs that are in the middle of this evolution, and the authors share their experiences in order to promote discussion within the community.

¹ Member of Technical Staff, Systems Engineering Division, curtis.iwata@aero.org, AIAA Member.

² NASA LaRC EDS Coordinator, AMA Systems Engineering Group, s.infeld@ama-inc.com, Senior AIAA Member

³ Integrated Design Center Manager, Applied Engineering & Technology Directorate, Code 500.

⁴ COMPASS Team Deputy, Mission Architecture and Analysis Branch, M.S. 162-2.

⁵ RAL Space CDF Software Developer, christina.mcquirk@stfc.ac.uk.

⁶ Systems Engineer, RAL Space CDF.

⁷ Systems Engineer, Advanced Design Engineering Group, jonathan.murphy@jpl.nasa.gov

⁸ Systems Engineer, Project Systems Engineering & Formulation Section, AIAA Member

⁹ Systems Engineer, Advanced Design Engineering Group.

I. Introduction

A Concurrent Engineering Centers (CEC) is an organization of people, tools, and facilities with a specific goal of rapidly generating and maturing engineering designs. Team of experts are assembled and given a design goal with a limited timeline of anywhere from hours to several days, and during this time, the team may generate one or more concepts. The members are purposefully co-located physically or virtually, rather than working from their separate offices. This co-location enables rapid design iterations by improving the speed and quality of communication between the team members, which consequently shortens the decision-making cycle. The tools and facilities are designed to support this process, such as networked computer terminals, lower fidelity analysis tools that are often linked together, large displays and projectors, white boards, and large reconfigurable meeting spaces.

A CEC can be used to generate, analyze, and refine concepts in an efficient manner. They typically serve the early part of a system's life cycle where there is large design freedom and uncertainty. By bringing together people with different backgrounds, the environment is more conducive to forming new ideas, resulting in a diversity of concepts. The question that drives any specific CEC session is complex enough that inputs from various experts are needed to create a feasible design. Rapid iteration is a key enabler that allows a design to be matured and refined very quickly. The end product of a CEC session can be a list of diverse concepts, a more detailed design, a set of requirements to go into a request for proposal, trade study results, or an independent evaluation of a concept.

One of the tools that is heavily employed by CECs is the use of models, which is the simplified abstraction of the system being designed. The use of models is not new to engineering. For example, the use of jargon implies the presence of a technical framework or model, and the terminology is an important tool to communicate the details of a system so that every engineer has the same mental model. A sketch on the back of an envelope or a scaled wind-tunnel aircraft are also examples of models.

CECs have adapted models to drive discussion and analysis by formalizing its application and by developing support tools to create and use it. The sophistication of these tools have increased over time, and the development has benefited from advances in model-based systems engineering (MBSE). The International Council on Systems Engineering (INCOSE) defines MBSE as "the formalized application of modeling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases" (Ref. 1).

CECs use models to help facilitate the design process and promote understanding of the system and key tradeoffs by providing a framework. Often, the model is a set of spreadsheets that represents aspects of the system; for example for a satellite, this would often include the payload, power, communications, propulsion, and attitude control subsystems. For most specifications of these subsystems, default values suffice. Then, the constraints of the problem, such as cost or schedule, will drive the team to compromise on certain objectives and previous assumptions, and due to the complexity of the design, it may not be obvious how changes in one subsystem affect the others. This is where the model is useful in highlighting the potential tradeoffs.

Software tools have been developed to improve the use of the models, and the variety tools and their sophistication have increased as well. Spreadsheet models can be as simple as a one-page spreadsheet or as complex as one that has a database and is accessed by multiple users from different geographical locations at the same time. This sharing of information using linked spreadsheets via computer networks enables quick iteration despite the larger and more complex models. Research is also done on developing estimating relations from historical data for parameters such as mass, cost, and power, which are important for creating credible solutions.

Due to the similarity in tools and purpose, advances in the systems engineering (SE) discipline, such as tools for MBSE, are making their way into CEC's software tools and infrastructure. SE encompasses the entire life cycle of a system while CECs are tailored to serve the early portion of the design, and Heinz Stoeber has further suggested that CECs are indeed the first integrated step of MBSE because it is the first setting where different types and sources of technical information are synthesized into a system model (Ref. 2). To describe a system with precision so that there is no ambiguity, description frameworks and languages such as the Systems Modeling Language (SysML) (Ref. 3) have been developed. One of the benefits of establishing these standards is the emergence of software ecosystems which enhance the tool capabilities to extend beyond capturing and describing the system, such as the ability to run simulations, to query the model for information, and to perform checks and validation. Because CEC activities are heavily software-based and the concept of "modeling" are similar, some of the developments in MBSE are being incorporated to CECs and vice versa.

II. History, Evolution, and Specialties of CECs

Many CECs have been founded, and while all share a set of basic capabilities, each one also has its unique strength that it has developed in response to specific sponsor and customer needs. For example, Team X at NASA

Jet Propulsion Laboratory (JPL) specializes in maturing interplanetary mission concepts in preparation for Preliminary Design Review. On the other hand, The Aerospace Corporation has the U.S. Air Force's Space and Missile Systems Center as the primary customer, and so Aerospace's Concept Design Center (CDC), while it shares heritage with JPL's Team X, is specialized in designing Earth-orbiting satellites and constellations, which is then used to inform the requirements generation process for acquisition.

This paper surveys the following six CECs in the space domain and highlights the context in which these centers have evolved:

- The Aerospace Corporation – Concept Design Center (CDC)
- NASA Jet Propulsion Laboratory (JPL) – Product Design Center (PDC)
- NASA Goddard Space Flight Center (GSFC) – Integrated Design Center (IDC)
- NASA Glenn Research Center (GRC) – COMPASS
- Rutherford Appleton Laboratory (RAL) Space – Concurrent Design Facility (CDF)
- NASA Langley Research Center (LaRC) – Engineering Design Studio (EDS)

For each of the centers, the following areas are highlighted: History and Purpose, Tools and Processes, Team Composition and Funding, and MBSE Integration.

A. The Aerospace Corporation – Concept Design Center (CDC)

The origins of the CDC can be traced to the Concurrent Engineering Methodology (CEM) that was developed in 1993. Based on Aerospace's experience in implementing concurrent engineering methods, JPL contracted Aerospace to build processes and tools for its PDC, and out of this effort, the Distributed CEM architecture was developed. The success of the CEM and its tools at JPL paved the way for the creation of Aerospace's CDC in 1997. (Please see Refs. 4-6 for more information.)

The main customers are government decision makers, and the CDC studies are primarily conducted to support requirements analysis for Request for Proposals (RFP) and to assess the responses to these RFPs. The typical result of a CDC session is a potential Government Reference Architecture rather than a proposal.

1. Tools and Processes

The core of the CDC is a set of interconnected Microsoft Excel spreadsheets that represent the various systems and subsystems of the space, ground, and launch segments. There are different versions of the tool depending on the necessary fidelity, and they range from a single spreadsheet to a collection of workbooks that are linked to a SQL database. There is ongoing research and development into enhancing the capabilities such as improving the underlying heuristics and adding the option to include risk considerations into the design.

2. Team Composition and Funding

The CDC is managed by the MBSE Office (formerly the CDC Office), and it is responsible for the CDC's development (tools, processes, infrastructure, and skills), administration, and operation. The CDC maintains the following six teams that are tailored to specific types of systems:

- System Architecture Team
- Space Segment Team
- Ground Segment Team
- Communications Payload Team
- Electro-Optical Payload Team, and
- Human Spaceflight Team (currently in development)

The MBSE Office itself is minimally staffed, and as of 2015, it is managed by two full-time engineers. When there is a CDC session, the team is assembled with subject matter experts from the Engineering and Technology Group (ETG), and like most CECs, the studies are paid by the customers on an as-needed basis. In terms of overall number of MBSE experts within Aerospace, it is small but growing, and they interface through various projects and through the MBSE Community of Interest (CoI) at Aerospace. The members each have different technical backgrounds, report to different line management, and serve different customers, and this provides the CoI with a diversity of perspectives, which is useful in enumerating the needs, concerns, and use cases for MBSE.

3. MBSE Integration

The Aerospace Corporation has been investigating how SysML can enhance the current Excel- and database-based models. Several pilot projects have been conducted over the past several years to link the CDC tools to SysML tools, and one of the compelling use cases is to output a simple model that can serve as the starting point of a life-cycle model, one which can grow and evolve with the different phases of the design.

B. Jet Propulsion Laboratory (JPL) Product Design Center (PDC)

JPL's original concurrent engineering team, "Team X," was started in 1995, and it has conducted well over a thousand studies (for background see Ref. 9). It is now one of several concurrent engineering teams overseen by JPL's Innovation Foundry (Ref. 10). Taken as a group, these teams cover the mission concept design process from very early brainstorming sessions to costed point designs

1. Tools and Processes

Team X uses a set of linked Excel spreadsheets (about 20), each of which analyzes a single subsystem or design aspect, including mission design, risk, cost, all the major spacecraft subsystems, ground segment, and systems engineering. The spreadsheets exchange parameters by means of a central relational database, using a fixed parameter list of thousands of parameters. The number and complexity of the spreadsheets has grown over the years, and they incorporate numerous macros and dynamic user interface elements.

Team Xc, the CubeSat and SmallSat concurrent engineering team, uses its own custom Excel spreadsheets as the frontend GUI and accompanying MATLAB or Python scripts as the backend models, all of which are integrated using Phoenix ModelCenter.

2. Team Composition and Funding

The Innovation Foundry is tasked with nurturing concepts from their early inception, and it operates the following three concurrent engineering teams:

- Team X
- A-Team
- Team Xc

The A-Team facilitates very early brainstorming and concept exploration (Ref. 11), and Team Xc is a smaller concurrent engineering team that is focused on CubeSats and SmallSats (Ref. 12). All three of these teams operate similarly: they are employed as a service by internal and external customers, rather than being a mandated part of the design process; and for any given study, they will draw their membership from across JPL's line organizations. In the case of Team X (Refs. 8 and 9) this means drawing from a pool of two or three trained engineers for each of the domain chairs, and in the case of A-Team, this might mean calling in an expert from anywhere within JPL. In all of these cases, a single study will generally last for one to three one-day or half-day sessions. Additionally, the Innovation Foundry oversees proposal teams for NASA competed missions, which last for several months, and these teams do not operate as concurrent engineering teams.

3. MBSE Integration

The first pilot for MBSE directly in Team X was to try to use SysML directly in a study (Ref. 13). Conversations with other concurrent engineering experts and experiments at JPL indicated that a naked, unfiltered SysML editing tool would not have sufficient productivity to keep up in a Team X session. However, it was thought that a customized instance of MagicDraw tuned to an individual chair's needs might achieve the needed level of productivity. The experiments yielded many interesting results, but it did not show a clear path to adoption that integrated the SysML tool well with the engineering analyses required to complete a study. While this did not rule out SysML-based modeling in a concurrent engineering context, it did show that the tooling of 2012 was still insufficiently mature to make this an easy transition.

Along the way, a few key observations about SysML and the senior engineering staff that support work in Team X were made. First, the most natural modeling diagrams are those that mimic PowerPoint: the Internal Block Diagram (boxes and lines with the slide itself as an implicit context) and the Activity Diagram (basically a flowchart). In addition, specialized iconography was extremely helpful in overcoming initial resistance in having our technical chair attempting to use the tool. Also, it is hard for current graphical modeling tools to match the "quality" of Microsoft Office in terms of user control and graphical polish. And, the benefits of system modeling are not obvious to subsystem engineers (and may be very hard to make obvious as these benefits accrue at the integrated system level), so any introduction of new technologies must be cost-minimal or -neutral.

Finally, there is another high-level observation about user prototypes in a high-speed environment like A-Team or Team X: it is not a single effort. Once the team is engaged, engagement must be regular and iterative with signs that the team's input is highly respected. Anything less will lead to initial cooperation from earnest professionals, but the interest and support will decline if the direction does not appear to be sufficient to make the needed results in user interaction quality.

These observations are a big part of what led to the next generation approach. The interest was in leveraging structured data models to achieve improvements to the concurrent engineering environment. But it became clear that making these improvements useable to the team would involve a good deal of custom development.

To that end, JPL's Innovation Foundry is currently undertaking a software development effort to expand and upgrade its concurrent engineering tools and to take advantage of modern MBSE practices. This new capability is intended for use throughout the early formulation process by the A-Team, Team X, Team Xc, and proposal teams. The goal is to have tools for study management, an environment for modeling and analysis in collaborative engineering sessions, and several databases for studies, hardware, designs, and analysis models. It will allow the incorporation of analyses in a range of languages and fidelities, as well as being backwards-compatible with Team X's existing Excel workbooks. For the design definition and parameter exchange functions, the implementation will use a narrow subset of SysML vocabulary, sufficient to allow architectural variation and to describe the design at an appropriate level of maturity. This will also allow export to a SysML model which can be handed off to later stages in the design process.

C. NASA Goddard Space Flight Center (GSFC) – Integrated Design Center (IDC)

The IDC is an environment that facilitates multi-disciplinary, concurrent, collaborative, space system engineering design and analysis activities to enable rapid development of science instrumentation, mission, and architecture concepts.

The IDC was created as the Integrated Mission Design Center (IMDC) in 1997 to address the need to perform conceptual engineering, test the feasibility of competitive proposals, and provide credibility of the conceptual design. Two years later in 1999, the Instrument Synthesis and Analysis Lab (ISAL) was created to provide the same capabilities for competitive instrument concepts. The two labs were combined under an overarching organization, the IDC, in 2001. The lab names were updated in 2010 to the Mission Design Lab (MDL) and Instrument Design Lab (IDL). In 2012 a third lab capability was created, the Architecture Design Lab (ADL), to fill the need for additional flexibility with broad types of instrument and mission architecture studies. To date, over 640 studies have been completed by the IDC. While most are associated with Earth Science (atmospheric, ocean, land, ice, and vegetation) and Space Science (heliophysics, geophysical, planetary, and astrophysics) Earth-orbiting missions, the IDC has also studied instruments and missions to the Moon, planets, comets, and asteroids; communications satellites and systems; satellite servicing concepts; and International Space Station and other human-related missions. The IDC studies can range from short, broad architectural concepts to multi-week, detailed concept developments, as well as focused technical reviews and assessments.

1. Tools and Processes

Both the MDL and IDL use systems engineering integration software to assist in the gathering, integration, and display of subsystem engineering parameters. For example, the software creates a systems roll-up of mass and power that can be ported into a Master Equipment List (MEL). The MDL utilizes a tool that was developed in-house over 10 years ago, and it is continually reconfigured to keep up with current operating platforms. The IDL has been using an off-the-shelf tool for the past 7 years. Unfortunately, commercial support has been discontinued, and the IDL has frozen its operating environment to guarantee reliable performance. Both labs rely heavily on these products and are actively investigating the next generation of collaborative systems engineering tools.

At the subsystem level, each engineer utilizes discipline specific design and analysis tools (e.g. SolidWorks, ZEMAX, STK, and MATLAB). These tools are the same tools that the engineers use in spaceflight systems development. This provides credibility in the engineering results and also produces study products that are useful to the follow-on development activities.

The Lab Leads and Systems Engineers coordinate the day-to-day study activities and implement systems engineering processes within the concurrent and collaborative environment. These processes allow for rapid development, evaluation, and iteration of the concept design with continuous interaction with the customer team to ensure convergence towards the study objectives.

2. Team Composition and Funding

The IDC is managed at the Code 500/Applied Engineering & Technology Directorate (AETD) level and is primarily staffed with engineers spanning all Divisions within AETD:

- 540/Mechanical Systems Division
- 550/Instrument Systems & Technology Division
- 560/Electrical Engineering Division
- 580/Software Engineering Division
- 590/Mission engineering and Systems Analysis Division

In addition, participants from other GSFC Directorates contribute to the conceptual design in areas such as science (Code 600), reliability (Code 300), and cost estimating (Code 100). All of the experienced study engineers

apply their knowledge of current and evolving technology in their respective disciplines to quickly establish the best approach to meet the customer's study requirements. The customer team is also a critical part of the collaborative design process, providing continuous input and immediate feedback on the evolving design.

The IDC study funding comes from Code 100's New Opportunities Office (NOO) and Office of the Chief Technologist (OCT). These offices are responsible for assembling the proposal teams to address emerging HQ instrument, mission, and technology AOs, as well as internal technology development activities. Additionally, we provide our services directly to in-house GSFC flight programs and projects, HQ programs such as ESSP, and for external NASA centers and industry.

3. MBSE Integration and Challenges

The IDC has been improving subsystem tools to expand the types of missions that can be investigated as well as improving systems tool outputs so that they can be ported into proposals (e.g. MEL format and eventually the Heritage appendix templates). These products are critical to the development of the basic project schedule and parametric cost model.

There are clearly limitations to maintaining let alone evolving towards modern MBSE tools with the system modeling software used now. Previously, the MDL conducted a process improvement study to document all interfaces between subsystem engineers during a typical weeklong mission concept study. The MDL is using the results of the study to update the requirements for the next generation of collaborative systems engineering tools. The IDC will also follow a future Department of Defense (DoD) and GSFC effort using MBSE through a 12-to-18-month sounding-rocket mission to gain a better understanding of how MBSE can best be utilized for rapid design and be included as the architecture for the next generation collaborative design tools. These two separate activities will better align the IDC with industry's current MBSE efforts.

D. NASA Glenn Research Center – COMPASS Team

The COMPASS team is a multidisciplinary, concurrent engineering group whose primary purpose is to perform integrated systems analysis and concept designs of spacecraft and other engineering systems. It was established at NASA Glenn Research Center (GRC) in 2006 to meet the need for rapid mission analysis and multi-disciplinary systems design for in-space and human missions. The team represents a logical extension of GRC's long history of design and analysis of space systems concepts and missions.

The main focus of the COMPASS lab has been to provide technology assessments of programs and projects for NASA, other agencies, and private industry. With a focus on concept designs that serve as design reference missions (DRM) for different technologies, the COMPASS team provides technology developers options for technology use and insight into technology investment. Several of these technology-focused concept designs have been for technology demonstrator missions bound for flight. Some, like the SCAN test bed (originally called CONNECT when COMPASS did its independent concept assessment) and the Asteroid Redirect Robotic Mission (ARRM) (originally called Fetch when COMPASS did its independent feasibility assessment) have gone on to fly and become a \$1.25 billion program, respectively.

The process of assembling what has become the COMPASS team took multiple years and went through various starts and stops. The convergence of several factors gave birth to the COMPASS team including the following:

1. The right people,
2. The appropriate tools,
3. A supportive meeting space, and
4. A challenging and sufficient customer base

More on COMPASS' history and origins can be found in Ref. 9.

1. Tools and Processes

The COMPASS team's primary tool used to perform concurrent engineering and concept design is a database and data transfer tool known as GLIDE (GLObal Integrated Design Environment). GLIDE is a client-server software application, which was purpose-built to mitigate issues associated with real-time data sharing in concurrent engineering environments and to facilitate discipline-to-discipline interaction between multiple engineers and researchers. Accessing a MySQL database on the back end, COMPASS' application of GLIDE uses Excel as the user interface to interact with the engineers and their tools as they perform the rapid concept design during the standard COMPASS sessions, which can last 2 to 3 weeks.

2. Team Composition and Funding

Having access to the team of experts in their disciplines is the key to a successful design team. To achieve an effective and productive concurrent design team requires the right mix both in skills and personalities. Assembling a

team of discipline experts is key to developing and validating successful technical designs. Tools are a good start, but it is the people that make the team successful. Figure 1 captures the essence that the COMPASS team is made up of experts who are matrixed in from discipline organizations, and it also depicts the interaction of the different disciplines throughout the process, which is facilitated using the tools in the COMPASS lab.

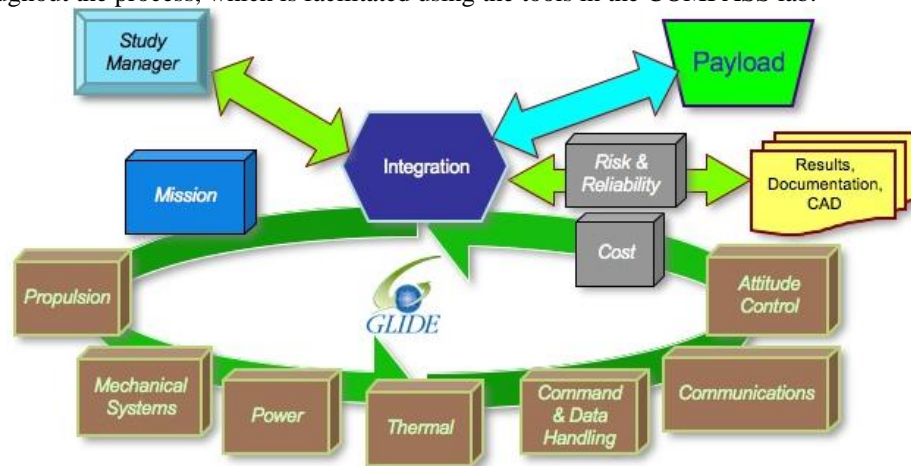


Figure 1. COMPASS team integration

3. MBSE Integration

Experimentation in using MBSE and NoMagic's MagicDraw to complement COMPASS designs has started as a systems engineering tool used for the ARRM (Asteroid Redirect Robotic Mission) program. Starting from COMPASS concept design products, the ARRM systems engineers have tailored an MBSE MagicDraw instance to model the concept of operations and vehicle interactions of the ARRM mission module and SEP (Solar Electric Propulsion) module. While this level of detail is applicable to an effort on the scale of a project, it has been found to be too detailed for the rapid two-week standard COMPASS design sessions.

Initially, the process for the COMPASS team would have allowed for generous pre- and post-session analysis activities. A concurrent design process was first assumed to consist of 1 to 4 weeks of pre-design session activities, several days for a design session, and finally, 1 to 4 weeks of post-session activities. As time has progressed and as COMPASS has become more in demand, the team has scheduled design sessions so that the pre- and post-design session activities overlap. The team typically is working three sessions at a time: the active session in the room, the documentation of the previous design study, and the setup of the next design study.

The ultimate goal of the COMPASS process for future inclusion of modern MBSE tools is to work with the systems engineers both within COMPASS and without and to standardize the output from a COMPASS study such that it can be easily implemented in MagicDraw and other SysML applications. Investigations into the level of use of MBSE during the 2 weeks of the COMPASS sessions are still ongoing, as is the development of version 3.0 the GLIDE application.

E. Rutherford Appleton Laboratory (RAL) Space – Concurrent Design Facility (CDF)

RAL Space's Concurrent Design Facility (CDF) was founded in 2011, and it is designed to host feasibility studies which output system architecture models.

4. Tools and Processes

Custom macro-based software is used with Microsoft Excel, Visio, and Project to develop models. Subsystems are modeled with custom tools, requirements, constraints, assumptions, and risks recorded in the system architecture model. Costing, staffing, and scheduling information can also be included. The costing is done using both internal and European Space Agency (ESA) costing Excel tools that are linked using custom code. Scheduling is incorporated by linking to Gantt charts in Microsoft Project. Several external tools are supported including AGI's System Tool Kit (STK), Siemens' Solid Edge (limited geometries), MATLAB, ANSYS, Agisoft's PhotoScan, Bayesian Evolutionary Analysis Sampling Trees (BEAST) Software, and ESATAN.

System models that are developed by the CDF can be used directly in the next phase of the project. For example in 2012, a feasibility study was performed on a low-cost pathfinder mission to provide early-warning capability of coronal mass ejection (CME) events that can impact the Earth. The concept was built around the UK-made Heliospheric Imager onboard NASA's Solar Terrestrial Relations Observatory (STEREO) spacecraft. The resulting

satellite concept was called HAGRID, which stands for "Heliospheric imaging for Assessment of Global and Regional Infrastructure Damage."

Current development efforts are focused on the pyCDF software, which will allow links from the specialist tools and the graphical design tools to the numerical values held in the CDF data exchange, and it will still use the more accessible Excel front-end. MBSE tools will also be integrated as part of the pyCDF effort.

5. Team Composition and Funding

The current RAL Space CDF team consists of two systems engineers and receives oversight from the systems group leader. A Systems Engineer, Project Manager, and applicable subsystem specialists are brought in for each study. As RAL Space specializes in instruments, testing, and calibration; most studies are in conjunction with another university or industrial partner. The RAL Space CDF has benefitted from core funding for the development and maintenance of the facility, but requires additional funding to support CDF studies.

6. MBSE Integration and Challenges

The RAL Space CDF uses thread definitions to follow calculations through the various excel spreadsheets. However, this setup is specific to each core trade, for example mass budget or power budget, and often evolves with each study.

S/C Total Dry Mass at Launch Budget CDF Thread (Mass Budget)
 Iss 1.0
 21 March 2011
 Approved

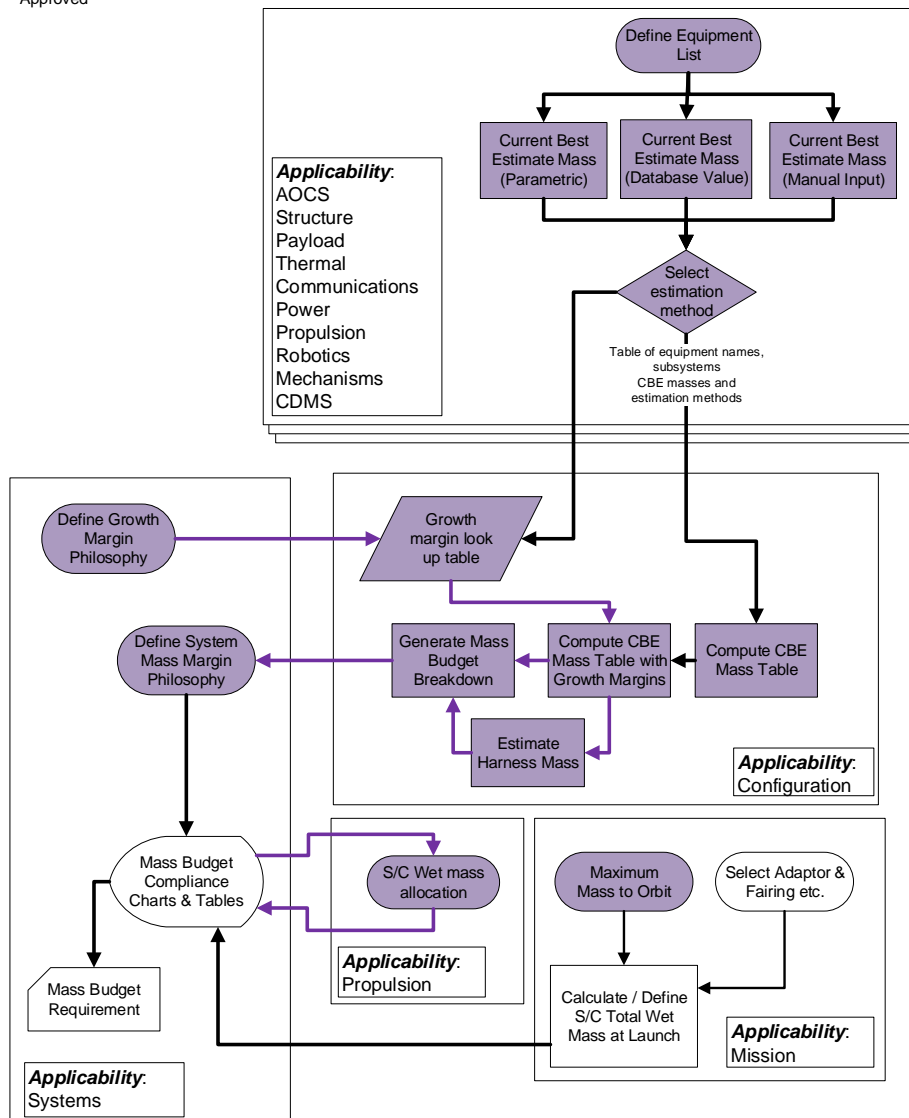


Figure 2. RAL Space CDF Spacecraft Dry Mass Roll-Up Flow Diagram

1. Tools and Processes

The EDS uses Microsoft OneNote¹⁰ as the working space for all disciplines in order to have a consistent system model. This links the system information to their source in the disciplines and to files that contain analysis and drawings. A template is loaded and tailored for each study, and this file often becomes the main EDS output after some post-session clean-up.

More linking between subsystems and an automatic roll-up of calculated numbers is being developed using Microsoft Excel Query to create an EDS team parameter tool. This uses a central Systems Integration Engineering (SIE) workbook that queries discipline workbooks. This is similar in functionality to GSFC's MDL Prime tool but without a formal database back end. Additionally, the workbooks are attached to the corresponding section in the OneNote Notebook, so the Notebook remains the single source of system data during the session as well as the report at the end of the session.

2. Team Composition and Funding

There is a pool of scientists and engineers that are being trained (as of summer 2015) on a formalized study process, and they are helping to refine the collaboration tools. From this pool and the customer team, a team is created as needed for each conceptual phase study to produce the requested deliverables. There is no cost to the customer because EDS team pool members are authorized to charge their regular projects for their time (part of NASA's "whitespace policy"). When a contractor is needed for a specialty or lack of availability of civil servants, this labor is charged to the customer project.

3. MBSE Integration and Challenges

At the EDS, SysML is now being used during design sessions for CubeSat mission customers to do some of the design work in parallel and to have as a project resource afterwards. The INCOSE CubeSat Reference Model in MagicDraw is used as a framework, and known information about a system is populated as part of the pre-work activity. This typically includes the mission objectives, top-level requirements and constraints, concept of operations (as the top-level activity diagram), and logical architecture, and these objects are linked together as much as possible. For example, an action in the concept of operations "satisfies" a requirement and is associated with the element of the architecture that performs it. During the session, the EDS modeler works with a member of the project team to add more details to the model as decisions are made. This member starts taking ownership of the SysML model and becomes the modeler for that project going forward.

Ultimately, the EDS team parameter tool will be integrated with the SysML model so that the parameters are pulled into the SysML model rather than into the SIE workbook so that the model becomes the SIE's main tool instead of needing an additional EDS modeler. However, it is hard to envision the modeling moving fast enough for the session when there is a more complicated and less pre-defined system than a CubeSat, or one that is beyond the early concept phase.

III. Discussion

Based on a survey of six CECs, it is evident that these centers are using models to support their concurrent engineering activities, and each CEC is continuing to evolve and improve their tools and processes. Many of the models are developed as spreadsheet tools, which offer a simple way to contain data, to input formulas and relationships, and to reconfigure the setup to adapt to new studies. Over time, these spreadsheet models have been upgraded to communicate to a central database so that multiple users can access the information at the same time, and this has also allowed the tools to accommodate larger models with higher fidelity and increased complexity. These linked spreadsheets are highly effective when honing in on a feasible design.

Some of the development efforts within CECs are focused on incorporating modern MBSE standards and tools, namely SysML and software tools that support it. More upfront effort is necessary to build a SysML model, but there are many benefits to using these tools because they offer capabilities such as generating up-to-date documents and diagrams, checking for model consistency, and tracing requirements. Because SysML was designed to capture all aspects of a system and its life cycle, people have found the language to be too rich and too time consuming to use in the conceptual design process. To tackle this challenge, JPL's Innovation Foundry, for example, is developing a limited set of SysML objects and relationships and custom software so that it is simple enough to use in studies. As this new MBSE paradigm becomes adopted in later stages of the life cycle such as in development and

¹⁰ Trade names and trademarks are used in this report for identification only. Their usage does not constitute an endorsement, either express or implied, by the authors or by the National Aeronautics and Space Administration.

production, it becomes increasingly desirable for CECs to develop the initial model so that it can be used in the later stages.

As CEC participants and customers make that paradigm shift to using a single system model, there needs to be a time-efficient way to present the model information at various levels, and this information needs to be easy to understand. This is especially important because study results need to be disseminated externally to funding bodies, managers, and subsystem engineers. Remembering that the purpose of CECs is to evaluate concepts and to provide a robust early design or highlight areas where more development is needed for the concept to be viable, pre-defined ways to view model information such as having a template for difference audience “viewpoints” in a SysML model, such as those in the Department of Defense Architecture Framework (DoDAF). All of this aids in focusing research and development efforts by giving the wider space community visibility of the concept readiness level.

As with anything new, there are many challenges that need to be overcome in addition to those already mentioned. Organizing and facilitating a CEC session already requires a broad range of experience and knowledge of different subsystems, and modeling is an additional skill that needs to be taught to future CEC engineers. The current SysML software tools are not well-suited for agile conceptual-design studies, but they are slowly being improved as the user community is learning its benefits and defining how they want to use it. SysML itself is also evolving, and the manner in which space systems are expressed is also still being explored and refined, so it will be an iterative process as best practices emerge. Development of good custom software is resource intensive, and many of the advanced use cases require experienced programmers, which appears to be something that the aerospace industry lacks as a whole. Finally, the MBSE paradigm shift is also a cultural shift, and the CECs and its customers alike will need to learn to communicate using the model and not with reports, presentations, and other document artifacts.

IV. Conclusion

Engineering design activities, including those conducted in Concurrent Engineering Centers, use models to aid in communicating and developing systems, and the models and their uses have become more sophisticated over time. These models can be abstract and semantic, and in this case, they serve as a conceptual framework. Others are numerical and explicitly define relationships, such as heuristic equations for sizing. Many of the current generation of CECs heavily use linked spreadsheets (or workbooks), and the different subsystems spreadsheets, such as propulsion and power, imply that there is structural meaning to how the information is organized. Modeling languages such as SysML try to formally bridge the semantic model, which is typically formed and held in one's mind, with the data and parametric models, which are stored in the software tools. All of the surveyed CECs are evaluating or actively incorporating modern MBSE tools so that it can output a SysML model at the end of study because of the added benefits of implementing MBSE and because of the growing adoption of SysML for MBSE implementation across the entire design life cycle.

Appendix

A. Resources for MBSE Development in CECs

Two resources have been helpful as a source of ideas, support, and connections in the writing of this paper, and they have also been instructive in introducing MBSE principles into the procedures in the redevelopment of the LaRC EDS. For anyone working on this subject, please consider joining these groups to learn from its members and to share new experiences:

- INCOSE's Model-Based Conceptual Design Working Group¹¹
- NASA/AIAA Concurrent Engineering Working Group¹²

¹¹ Contact David Harvey, MBCD chair: david.harvey@shoalgroup.com

¹² Contact Daniel Nigg, CEWG chair: Daniel.A.Nigg@aero.org

Acknowledgments

The authors thank and acknowledge the following contributors of ideas presented in this paper:

Daniel Nigg
Chair of the NASA/AIAA Concurrent Engineering Working Group
Systems Director, Programmatic Assessments, Advanced Studies & Analysis
Former Director, Concept Design Center (CDC), Systems Engineering Division
The Aerospace Corporation

Jared Lang
Senior Member of Technical Staff
Model Based Systems Engineering Office
The Aerospace Corporation

Kris Romig
GSFC's MBSE Coordinator

Mark Steiner
GSFC's Mission Design Lab Team Lead

Frank Kirchman
GSFC's Mission Design Lab Systems Engineer

Tammy Brown
GSFC's Instrument Design Lab Team Lead

References

- ¹ *INCOSE SE Vision 2020*, INCOSE, 2007, INCOSE-TP-2004-004-02.
- ² Stoweiver, H., "Model based Systems Engineering (MBSE) Missing Link In the digital Enterprise Strategy?" *INCOSE Lost Angeles International Workshop 2014*, INCOSE, URL: http://www.omgwiki.org/MBSE/lib/exe/fetch.php?media=mbse:01-iw14-mbse_workshop-keynote-missing_link_in_the_digital_enterprise-a-stoweiver.pdf [cited 26 May 2015]. 2014.
- ³ Object Management Group, "OMG SysML," URL: <http://www.omgsysml.org/> [cited 26 May 2015].
- ⁴ Aguilar, J. A., Dawdy, A. B., and Law, G. W., "The Aerospace Corporation's Concept Design Center," *8th Annual International Symposium of the International Council on Systems Engineering*. Vol. 2, INCOSE, 1998.
- ⁵ Aguilar, J. A. and Dawdy, A. B., "Scope vs. Detail: The Teams of the Concept Design Center," *Aerospace Conference Proceedings*, Vol. 1, IEEE, 2000.
- ⁶ Noguchi, R. A. and Nigg, D. A., "Application of Model Based Systems Engineering to Early Life Cycle Systems Engineering Processes in Government Acquisition," *INCOSE Insight*, INCOSE, 2014.
- ⁷ Ziemer, J., Ervin, J., and Lang, J., "Exploring Mission Concepts with the JPL Innovation Foundry A-Team", *AIAA Space Conferences Proceedings*, San Diego, California, September 10-12, 2013
- ⁸ Sherwood, B., Adler, M., and Wheeler, R., "The New Team X: Concurrent Engineering Advancement at JPL," Presentation at the 58th International Astronautical Congress, Hyderabad, India, September 24-28, 2007.
- ⁹ Wall, S. D., "Reinventing the Design Process: Teams and Models." AIAA. 1999.
- ¹⁰ Sherwood, B. and McCleese, D., "JPL Innovation Foundry," *Acta Astronautica*, Vol. 89, 2013, pp. 236-247.
- ¹¹ Ziemer, J. K., Ervin, J., and Lang, J., "Exploring Mission Concepts with the JPL Innovation Foundry A-Team." AIAA Space 2013 Conference and Exposition. 2013.
- ¹² Zarifian, P. et al., "Team Xc: JPL's Collaborative Design Team for Exploring CubeSat, NanoSat, and SmallSat-based Mission Concepts," IEEE Aerospace Conference, 2015 (to be submitted).
- ¹³ Cole, B., Dubos, G., Banazadeh, P., Reh, J., Case, K., Wang, Y., Jones, S., and Picha, F., "Domain-specific languages and diagram customization for a concurrent engineering environment," IEEE Aerospace Conference, March 2013, pp.1-12.
- ¹⁴ McGuire, M. L., Oleson, S. R., and Sarver-Verhey, T. R., "Concurrent Mission and Systems Design at NASA Glenn Research Center: The Origins of the COMPASS Team," *AIAA Space 2011 Conference and Exposition*, NASA Glenn Research Center, Cleveland, 2011, NASA/TM-2012-217283, AIAA Paper 2011-7240.
- ¹⁵ Gough, K. M., Allen B. D., and Amundsen, R. M., "Collaborative Mission Design at NASA Langley Research Center," *1st Annual Space Systems Engineering Conference*, Georgia Institute of Technology, Atlanta, 2005, GT-SSEC.A.3.