Update on the Model Based Systems Engineering on the Europa Mission Concept Study

Todd Bayer, Seung Chung, Bjorn Cole, Brian Cooke, Frank Dekens, Chris Delp, I. Gontijo, Dave Wagner
Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Dr.
Pasadena, CA 91109
818-354- 5810
Todd.J.Bayer@jpl.nasa.gov

Abstract— **In May 2012 the Europa study team delivered to NASA the final reports on three distinct concepts for exploring Europa on a limited budget. The depth and quality of these reports have been widely praised by independent reviewers as well as by our sponsor. The application of Model Based Systems Engineering (MBSE) techniques is credited with enabling the team to study three quite different mission concepts for the resources normally sufficient to study only one or two. The Europa MBSE infusion itself has been awarded the NASA Systems Engineering Excellence Award in 2012. The Europa team is now preparing for its Mission Concept Review and has reaffirmed and strengthened the MBSE application. Significant new capabilities have been completed, most importantly the Powered Equipment List (PEL) and the computation of scenario-based power and energy margins. This paper provides an update on the continued successful application of MBSE in the dynamic environment of early mission formulation, the significant new results produced and several additional lessons learned in the process.**

Table of Contents

1. Introduction 1
2. Updated Results 1
3. Lessons Learned 9
4. Future Work 11
5. Conclusions 11
6. References 11
7. Biographies 12

1. Introduction

This paper updates and extends the March 2012 report on the application of Model Based Systems Engineering (MBSE) on the Europa Mission Concept Study. [1] In that original paper we described how the nascent system model helped the Europa pre-project team ride out and recover from near-cancellation in the wake of changing science priorities and drastic budget cuts within NASA. At the time of that paper we were nearing completion of several NASA-directed studies on how the Europa mission could be accomplished within the lower budget limits that were expected in the foreseeable future.

In May 2012 the Europa study team delivered to NASA these final reports, which described three distinct concepts for affordable exploration of Europa: a Europa Lander, a Jovian Orbiter with repeated Europa flybys, and a Europa Orbiter. The depth and quality of these reports have been widely praised by independent reviewers as well as by our sponsor. The application of MBSE techniques is credited with enabling the team to study three quite different mission concepts for the resources normally sufficient to study only one or two. The Europa MBSE infusion itself has been recognized by NASA as advancing the state of the art of Systems Engineering: the effort was awarded the NASA Systems Engineering Excellence Award in 2012.

At the direction of our sponsor, the Europa team is now preparing for its Mission Concept Review. The MBSE effort has been reaffirmed and strengthened. Significant new capabilities have been completed, most importantly the Powered Equipment List (PEL) and the computation of scenario-based power and energy margins. More detailed descriptions of some of the developments reported here are provided elsewhere. [2]

In the next section we provide an update on the modeling results and new developments since last year, including new patterns for modeling power and energy margins. New lessons learned are discussed in section 3 and future work in section 4.

2. Updated Results

Mission and Flight System Descriptions

We have now extended the SysML modeling effort to include both the mission and the flight system, thus allowing the model to be interrogated in various flexible ways. Here we describe some of the work involved and the infrastructure created to support such model analysis. The Europa study concept has defined a series of Viewpoints for addressing concerns related to Flight System Mass, Mass Margin, Bill of Materials and Deployment Mass. These are key products that address the concerns of building a spacecraft that fits within the capability of the launch vehicle. Sophisticated models to capture Power and energy consumption were developed as well and will be described in what follows.

View &Viewpoint Software Platform

978-1-4577-0557-1/12/$26.00 ©2013 IEEE

In order for modeling to be effective in this study, it is necessary to have a software environment that can support collaboration between Systems Engineers as well as the Domain experts. The diagram “Model Based Engineering Environment” in Fig. 1 describes the target state of the software platform we envision to support the modeling effort. The corner stone of this MBSE effort is the repository with its Application Programming Interface (API), which is the authoritative source of engineering information. Different modeling clients collaborate through this repository as engineers build models that describe and analyze the system. Note that the key point of this environment is to enable effective and consistent communication between engineering disciplines.

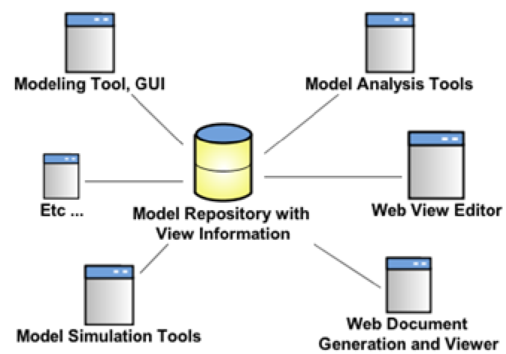


Figure 1 - The targeted Model Based Engineering Environment under construction.

In this Europa study we are using (and improving) a first version of this environment along with many other JPL projects. This version is architected around “Viewpoints and Views” as described in the ISO42010 standard, to capture document descriptions and to perform system integration. The capabilities of this environment focus primarily on capturing models with a mix of SysML modeling and web applications. Currently we have tools that produce automated Bill Of Materials (BOM) as well as several views of Mass tables from the model. The tables produced can be edited online or used for analysis with other software platforms such as Mathematica, Excel or Maple.

Work Breakdown Structure and Bill of Materials Views

An important part of the Europa modeling approach is the recognition that systems engineering touches the project as well as the product. The team anticipates a time when the model will be extended to capture requirements and verification methods. The tracking of a verification chain can be significantly enhanced by considering delivery responsibilities as the system is integrated. At this stage of development, the usefulness of the WBS to the study is to organize responsibility for providing current resource estimates.

The Europa study describes the project organization in terms of Work Packages that supply components. A Work Package is an element of required Work as illustrated in the “WBS Clipper Flight System” Work Breakdown Structure

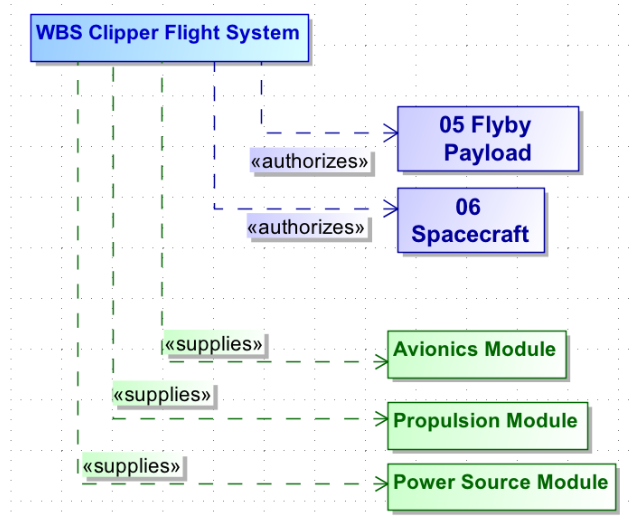


Figure 2 - Example work breakdown structure view.

in Fig. 2. Work Packages are hierarchically organized based on the authority structure of hardware/software implementation. For the formulation phase of the lifecycle (where the Europa study is currently), it is sufficient to identify the work in terms of what product will be supplied by each Work Package. These rules define the Work Breakdown Structure Viewpoint that addresses the project organizational mapping to the component products that need

Table 1 - Bill of Materials Table View

Work Package	Component Deployment	# Units	Mass (kg)
WBS Flight System (FS)			*Flight System Mass*
Payload			*Payload Mass*
SWIRS			*SWIRS Mass*
	SWIRS Sensor	*1*	*Component Mass*
Spacecraft			*Spacecraft Mass*
Telecom			*Telecom Mass*
	High Gain Antenna	*1*	*Component Mass*
GNC			*GNC Mass*
	Sun Sensor	*3*	*Component Mass*

to be delivered. These models capture the authority structure and supply relationships between the Work Packages and the individual components of the system and the whole system itself.

The diagram in Fig. 2 is a view of the Flight System showing that the Payload and Spacecraft Work Packages are authorized by the Flight System to start work (and charge corresponding account numbers) to supply components to the flight system. This is a fragment of the contents of our model for the purposes of this example. This part of the model also provides a structure attaching different properties and behavior related to constraints that the individual areas of work have to operate within.

This viewpoint is also used to analyze and roll-up the mass per Work Package as illustrated in the "Bill of Materials Table View" in Table 1. Individual component masses are added (taking into account their multiplicity and deployment) producing a work package mass such as the "GNC mass" in the table. Work Package masses can also be added (e.g., GNC + Telecom + others not shown), to give the mass of work packages at a higher level in the structure (Spacecraft in this case). This process is repeated until the total mass for every work package is obtained and the mass roll-up goes all the way up to the "WBS Flight System" mass in the table. The diagram "Mass Constraint Patterns" in Fig. 3 illustrates the SysML constraints used to calculate mass and mass margins.

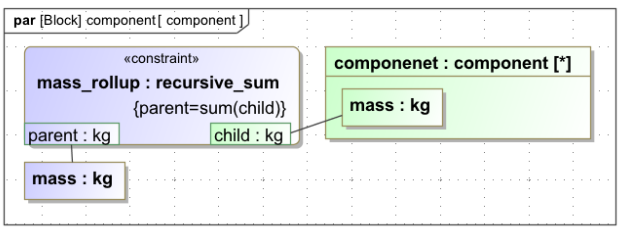


Figure 3 - Mass constraint patterns for mass roll-up.

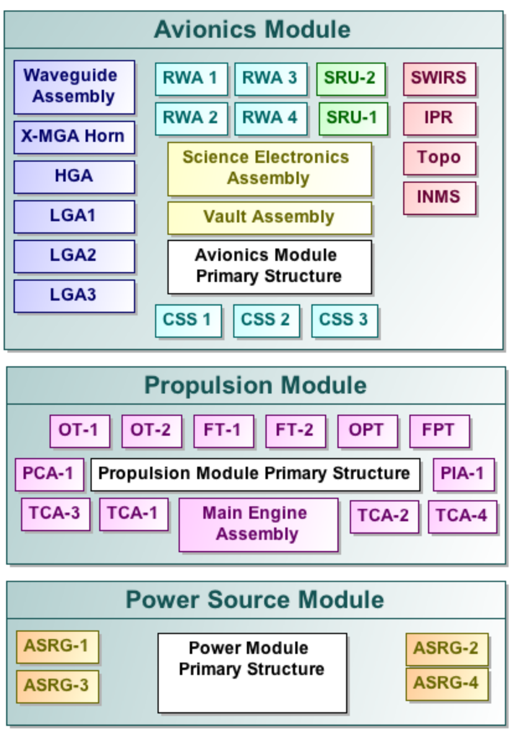


Figure 4 - Flight System Block Diagram.

Deployment Views

The "Flight System Block Diagram" in Fig. 4 illustrates the deployment of components within the modular architecture being considered in this study. This specific model is used by the Deployment Table Viewpoint to analyze and roll-up the mass of composite components as explained above.

Interfaces

As described before, one of the benefits of the MBSE approach is that we are able to generate a consistent set of systems engineering products from a single-source-of-truth, which we capture as models specified in SysML. [3] Block Definition Diagrams (BDD) for example are created as well as Internal Block Diagrams (IBD) to describe the internal components of a system and their interconnections (connectors in SysML). An example [4] of an IBD created in the Europa Mission study is shown in Figs. 5-a and b, where the color of connecting lines represents different connection (or connector) types. Blue connections for example represent data, red connections represent power; etc.

While this approach has been sufficient thus far for the Europa mission concept study, we have started defining new patterns for representing more detailed connection specifications. Fig. 6-a depicts the current approach to specifying connections while Fig. 6-b depicts our new approach, which allows the specification of the interfaces of parts using SysML ports. In this example we show 28V power and 1553 data interfaces. Fig. 6-c shows an example of how to define 1553 bus interface types. While only 1553 bus interface types are shown in Fig. 6, we have defined a wide range of interfaces including data, power and radio. It is expected that this list will expand to include also thermal and mechanical interfaces. These interface definitions are important, as they allow consistency checks to ensure that the connectors between two parts are valid in terms of their interface types and the number of available interfaces. In the spirit of ensuring consistency through a single-source-of-truth, this approach also allows us to be consistent when specifying the interfaces of the same part type.

Technical Margins

A year ago we reported using the "characterization pattern" to associate mass properties to elements of the model and then extract those characterizations into a mass equipment list report in which the masses are summed.[1] We also described our intent to try and use similar characterizations to describe power loads and data rates needed to assess other technical resource balance questions.

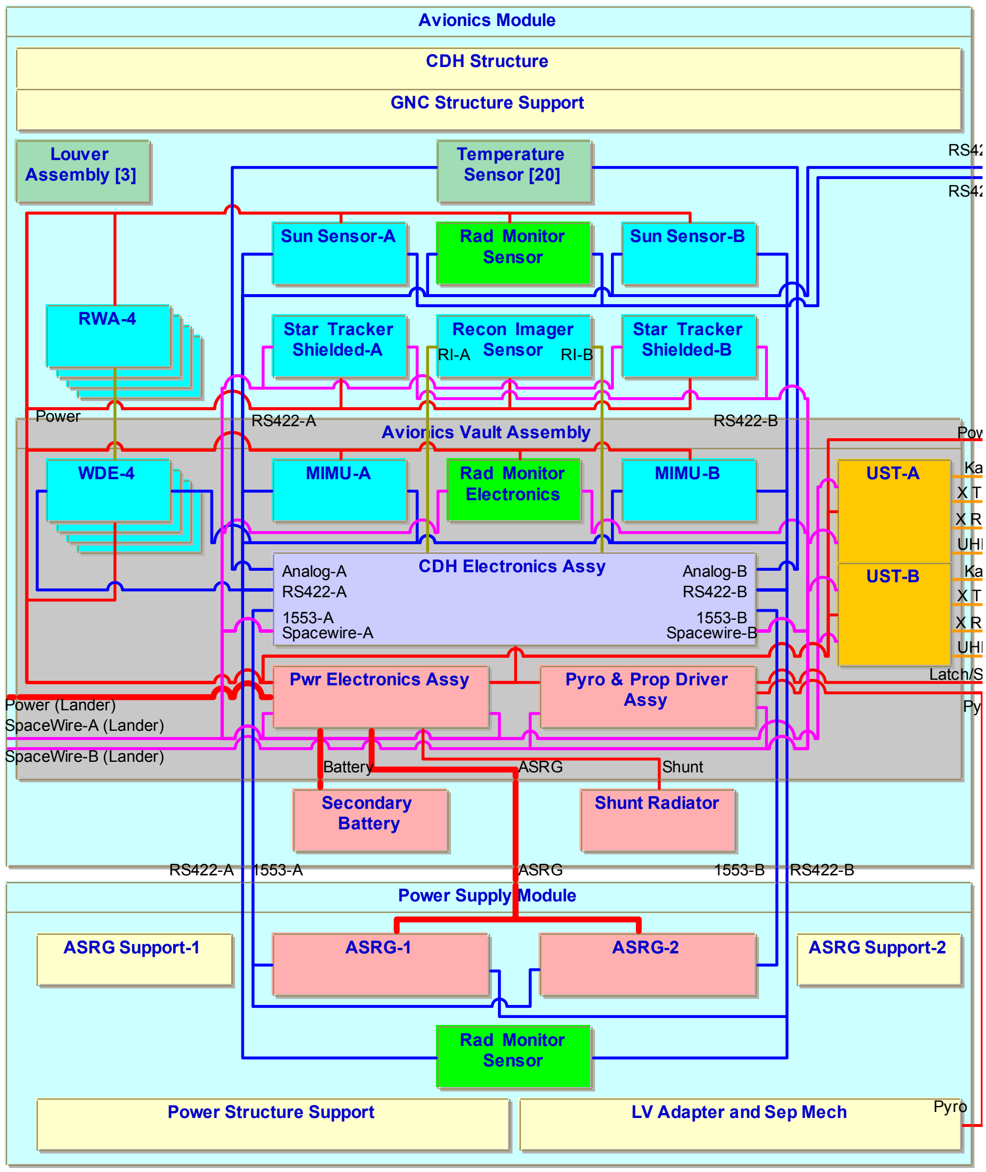


Figure 5-a - 'Carrier element' IBD, from the Europa Lander Mission study in ref. [4], page D-98.

While it is true that system mass changes over time through the separation of launch related parts or propellant depletion, the mass balance analysis only requires consideration of a few discrete configurations. However, electrical energy and data are produced, stored and consumed according to activity plans intended to achieve mission goals. Understanding the achievability of these behaviors requires some level of resource simulation over time to ensure that operational safety or design margins are enforced at all times. Thus, where analysis of mass characterizations only required the ability to extract and sum the masses of system components, the energy and data analyses require the ability to model the scenarios and the scenarios' associations to individual component energy or data characterizations, and to export those models into an environment in which they can be executed as a simulation model.

A great deal of effort went into deciding how to model the use of dynamic states such as power in the spacecraft. Many factors were taken into consideration, most prominently the maximal use of "vanilla" SysML, simplicity, scalability and the ability to accommodate multiple methods of calculation.

It is important to have flexibility to describe the amount of power consumed by a device via flat specification or by parametric relationships between multiple parameters or devices. It is also important to make sure that any values tied to a given behavior segment (a state in a State Machine or action in an Activity) are properly connected to that segment.

The approach taken was to have a special characterization block that contained current best estimate, contingency, and maximum expected values. These values could then be

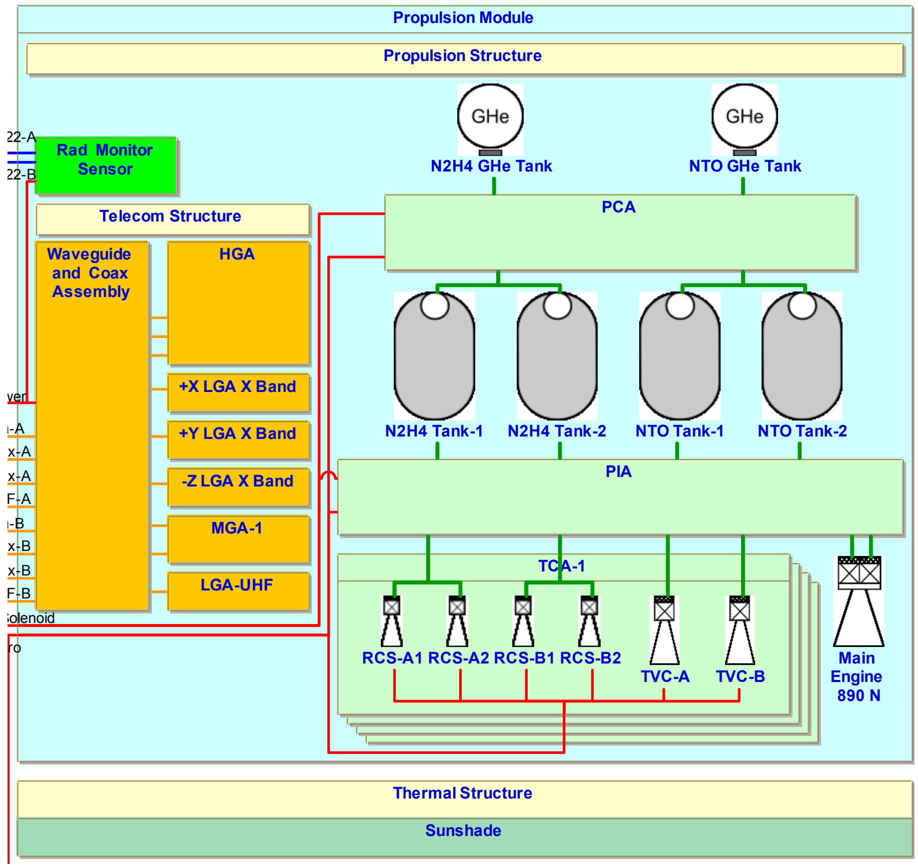
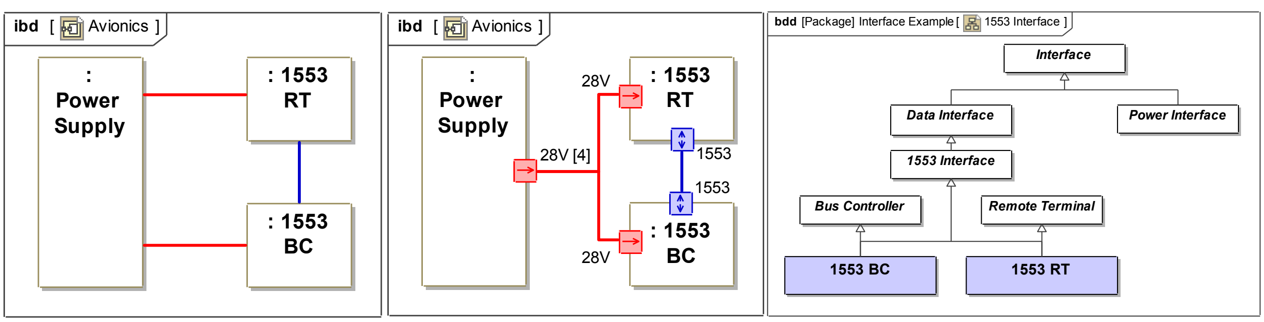


Figure 5-b - Propulsion module IBD, from the Europa Lander Mission study in ref. [4], page D-98.

connected via Parametric diagrams to other parts of the model. In addition, a fact of the UML metamodel was used in this behavior modeling: behaviors are metaclasses that are specializations of Class. Thus, the Block with our characterizations could be specialized into State Machines or Activities. These in turn can be defined as Submachines or Behaviors for State Machines or Activities that would describe the legal behavior of the device. One more clever use of SysML semantics is to promote the idea that a property on a behavior, can redefine the value of a property on a plain Block. The idea is that this redefinition occurs if and only if that behavioral type is active.



(a) (b) (c)

Figure 6 – Specifying connections in SysML. BC = Bus Controller, RT = Remote Terminal. (a) is an example of an IBD with block association. (b) is an example of an IBD with interfaces. (c) is an example of an interface model.

The dynamic redefinition pattern for a "Power Load Product" is seen in Fig. 7, where the appropriate states and state machines are colored in pink and green. Again, it can be seen that there is a separate block with a characterization that describes steady-state power values (the "Power Load Characterization"). This is then redefined by State Machines (in the "Power Load Behavior Characterization" block) holding values to be applied when the given state is active. Redefinition in this case is interpreted to apply only to properties that have changed – the name, role, and units of the value property of "steady-state power CBE" are unchanged. The only change is to the default value. The knowledge of when something is "active" is to be provided by the states.

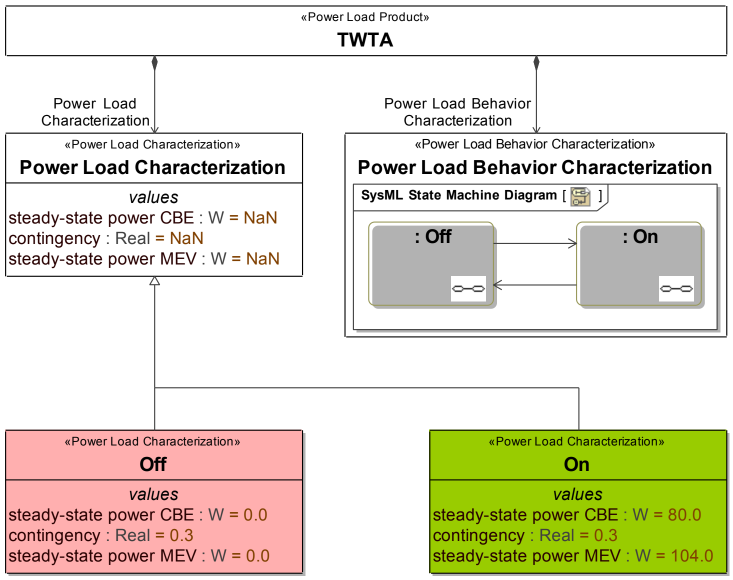


Figure 7 - Generic approach to dynamic redefinition of values for behavior.

There is some connection here to other approaches for describing parameters of states changing as they are entered. In a similar situation Conrad Bock [5] describes a way of having a person increment the number of times of being sick when a state is entered by creating a state instance, as shown in Fig. 8. His idea works somewhat with our diagram in Fig. 7 in that the UML specification describes Generalization as making all instances of a specific classifier also valid instances of the more general one. The Power Load Characterizations are akin to the specialized state class in that example. The actual timing of these states (On between t0 and t1) would be the state instance. More work needs to be done on these ideas to assure harmony, but the approach is promising.

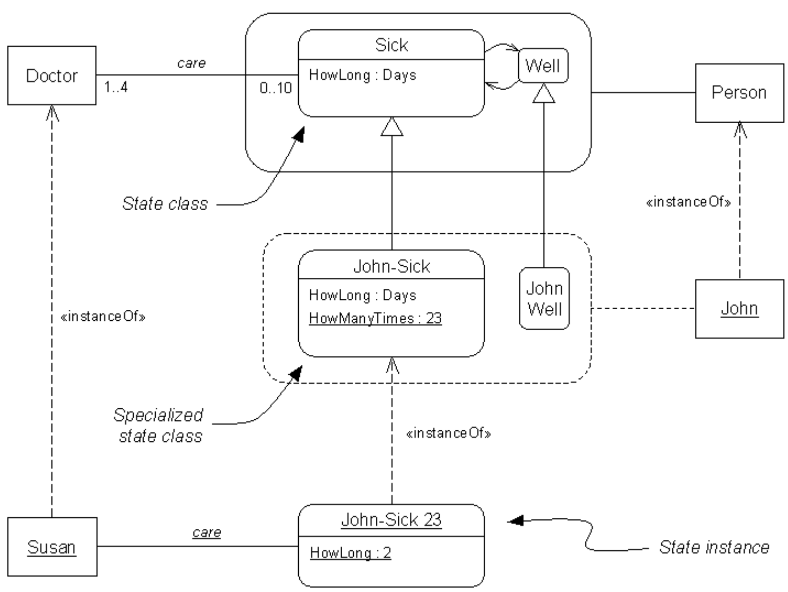


Figure 8 - Generation of state instances to account for parameter changes via behavior. From Ref. [5].

Determining when a given state in a State Machine should be active is the job of normal SysML Semantics. To drive mission-level and component-level scenarios employing state machines, Sequence Diagrams were used as shown in Fig. 9. Duration Constraints were applied to State Invariants to decide how long a given state should be exercised. This, combined with the characterization of power levels for a given state, was used to drive out various power analyses.

Looking at the power loads x time in a slightly different way, given power and energy characterizations, system-level power modes, and a scenario, we have sufficient information to perform power and energy margin analyses. With the scenario and the power load behavior, we are able to compute the power load profile. With the power load profile and the power source and energy storage behaviors we are able to compute the power source and energy storage profiles. These profiles allow us to understand the available power and energy margins.

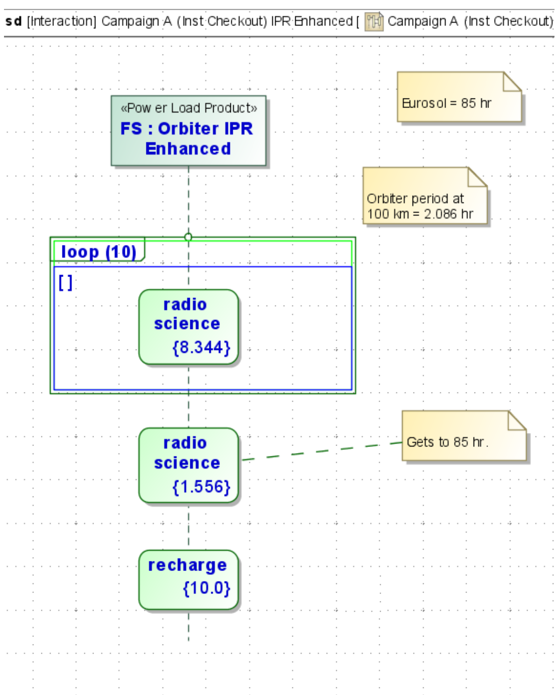


Figure 9 - Mission Concept of Operations scenario snippet defined by a Sequence Diagram. Numbers are fictitious.

First the system-level power modes are defined using SysML State Machines, then the State Machines are strung together as State Invariants within a SysML Sequence Diagram to form a scenario. Fig. 10-a depicts an orbit scenario for the Multiple-Flyby Mission from the Europa

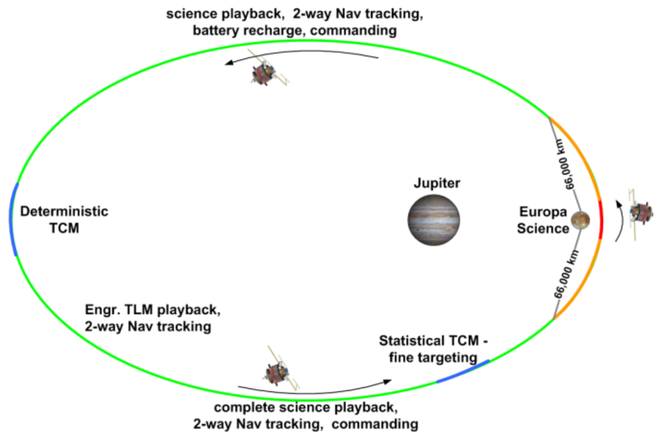


Fig. 10 – a: Orbit Scenario from the Multiple-Flyby Mission study in ref. [4], page C-122.

Habitability Mission concepts, ref. [4], page C-122. The varying colors along the orbit represent different phases, i.e. different system-level modes. This scenario is specified in a Sequence Diagram shown in Fig. 10-b whose State Invariants are colored according to 10-a. Then, SysML duration constraints are used to specify the time duration of each state invariant as shown in 10-c.

Using these model elements, a script was written to extract the scenarios and generate an executable simulation in Mathematica. The Mathematica simulation model had the form of a set of equations to solve for total power load over discrete intervals of time by simply summing up the individual loads identified in the scenario during each time interval. Battery charge can be extracted in the same way from the scenarios and plots of resource utilization are generated by the Mathematica script as show in Fig. 11.

We are also extending these patterns to include data production, storage, and transport (data balance), and finding ways to associate behaviors with model elements other than hardware components. Fig. 12 is an example of the initial data producer characterization. Unlike power loads, data can be produced by non-hardware elements (i.e., software), and data production can be either continuous (as measured by a constant sampling rate), or discrete (e.g., a camera taking discrete pictures, or an onboard data reduction process sub-sampling an image to produce a thumbnail product).

In order to process the SysML model of data production, storage and transport, we will be looking at using Modelica as the execution language. Like Mathematica, Modelica [6] is a declarative language based on equations so the system is described as a set of equations solving for an equal number

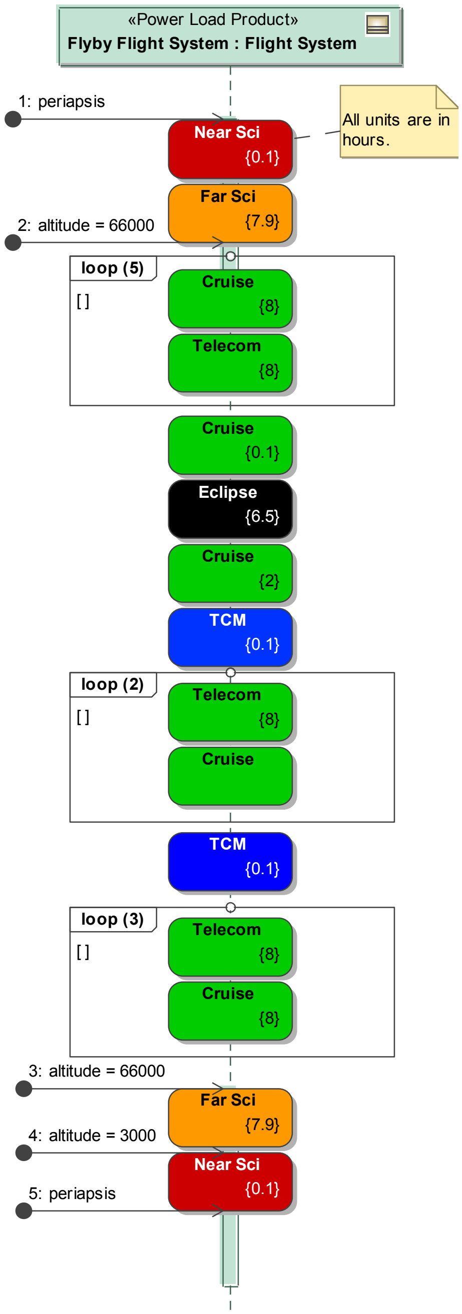


Fig. 10 – b: A SysML Sequence Diagram representing the Orbit Scenario from the Multiple-Flyby Mission study in ref. [4], page C-122. Numbers are fictitious.

of state variables. Modelica offers some extra advantages as it includes a number of language features to automatically solve and execute this system as a function of time, as time is a special variable in Modelica. Transforming the data model into Modelica for execution will simplify the model expression, making it more easily verifiable and reusable.

Science Margin

Science margin modeling and calculations were first demonstrated in the Kepler mission.[7] The Europa study recognizes that this is a valuable tool, amenable to modeling, despite the fact that in this mission the problem is much more complex. The Kepler mission consisted of a single instrument and a single science investigation, whereas here there are potentially up to nine different investigations. It is far from trivial to model and calculate the sensitivity of each science investigation to changes in mission architecture and engineering parameters. Nonlinear interactions where changes that can improve science margins for one investigation will detrimentally affect another instrument are beyond the modeling capability at present. Another problem also beyond the scope of this study is the optimization across the whole mission by changing engineering parameters to produce the highest science return for all science investigations simultaneously. Therefore only the simpler, linear case is being modeled, where engineering parameters are changed to optimize individual science instruments/investigations, while using engineering experience to make sure this does not produce unacceptable degradation in other investigations.

Cost Estimation and Integration with Cost Models

The cost modeling tools are well established and so the integration with cost modeling has two steps. First, we export SysML model data to be used by the cost models, which are then run manually by the domain experts. These tools include the three most used at present: PRICE-H, SEER, and NICM. Once the cost results are returned they are entered into SysML (manually for now), wrap factors are applied for each discipline (systems engineering, management, etc) and the total cost roll-up is calculated using the SysML tool.

A trade was performed to select which tool within SysML would be used for cost roll-up, and we settled on trying two methods. One is the Cameo Simulation Toolkit and the other uses a custom Jython script. The cost calculations within SysML are straightforward, so no strong mathematical tool was needed, such as Matlab or Mathematica. The wrap factors are each invariably a percentage of sub-totals, and the roll up of the costs is a straightforward sum of the cost of sub-systems. A sample parametric diagram of the cost model is show in Fig. 13.

Both methods for cost roll-up work successfully and the final choice is in part a matter of preference. The Jython script is very easy to code and re-use on other mission options and no parametric diagrams are needed. This is useful, since several mission options are being costed. The only disadvantage is that it is brittle, with hard coded block names that have to be changed for each re-use. On the other hand, the Cameo Simulation tool kit provides much better feedback on what calculations are being performed, but each

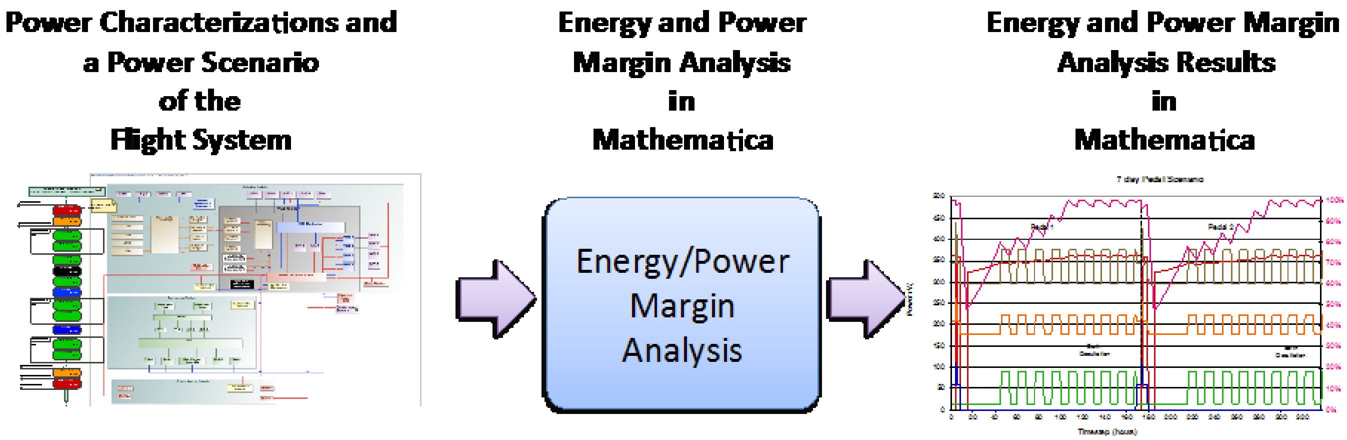


Figure 11 - Energy and power margin analysis process. Scenarios in Sequence Diagrams containing state machines are transformed into executable Mathematica code that produce plots of energy and power usage & margins.

time the equations are to be re-used for a different mission, a new set of parametric diagrams have to be created between the Work Breakdown Structure elements and the cost constraint equations. Although the equations are kept in a library, so that they don't have to be re-entered, this represents extra work for each re-use.

3. Lessons Learned

In our previous paper [1] we described 12 lessons we learned during the initial infusion of MBSE into the Europa study. With another year of work and reflection, we add these 7 additional lessons.

Leverage Learning with Synergistic Work.

With the limited pool of modeling talent available, we were tempted to ask for a full-time commitment from our modelers. But we knew there were other efforts where MBSE application was being tried and that these efforts would have a strong desire for the same personnel. We also believed that having the experts engaged in two or three modeling efforts would provide benefits that outweighed the lack of full time commitment. We have found this belief to

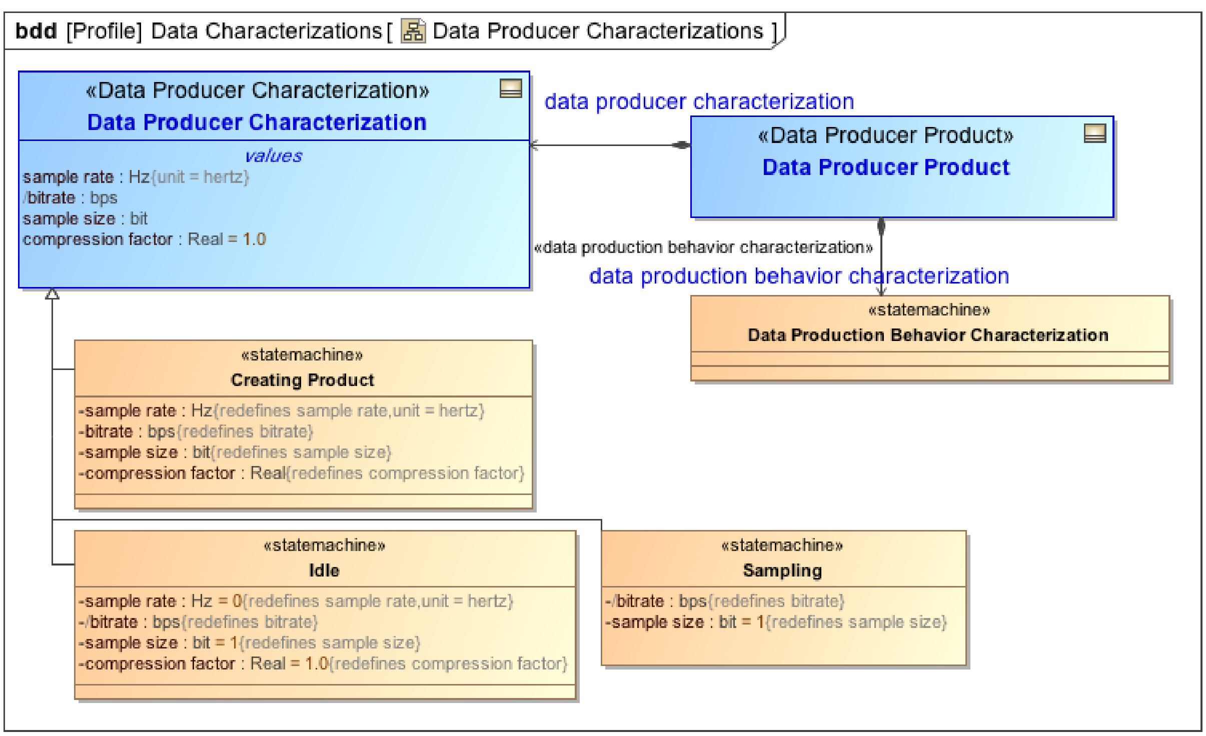


Figure 12 - Data Producer Characterization.

be fully validated: the learning that has been shared between the three efforts has been enormously beneficial for all, and has clearly accelerated the institutional infusion.

Innovation is Bottoms-Up.

We didn't know what scripts or plugins or modeling patterns to develop before we started. We let the discovery of the need drive the solution. There was 'top down' innovation but not in the traditional sense of pre-ordained specifications: it consisted mainly of constant guidance during the modeling process to keep the effort focused on satisfying the end objectives.

Models Evolve.

The model needed in concept formulation is very different than the model needed in detailed design, or in operations. Models need to evolve and grow, and sometimes shrink. This should be the focus of model reuse along the project lifecycle. It also helps to answer the people who will suggest that building a detailed model of the last flown mission will help you formulate the next. It all goes back to the principle of modeling for a purpose, and not more. While the models may change, these changes can be evolutionary and cumulative as long as they are connected by a common set of ontologies and methodologies.

Automated Web-Based Model Reports are Critical

One of the issues faced by adopters of MBSE is that the default vendor offerings require a consumer or reviewer of model information to use the vendor tool. Because the tools (e.g., MagicDraw) have a significant learning curve, this can present an insurmountable hurdle to acceptance among non-modelers (i.e., management, sponsors and review board members).

As we developed the Europa models, we did not have a good resolution for this issue. As luck would have it however, a separate team at JPL, working on a separate problem (defining JPL's next generation ground system architecture), had already invested in and developed a solution. The other effort was willing to share, and the Europa study was thus able to receive for free, a fully-developed capability to output MagicDraw models into a web-based format.

This is another example of the benefits of synergy.

Get Outside Expertise

To misuse a quote from Isaac Newton, if Europa's MBSE application has gone further than others, it is because they stood on the shoulders of giants. From the very beginning, visionary managers at JPL brought in world-class outside

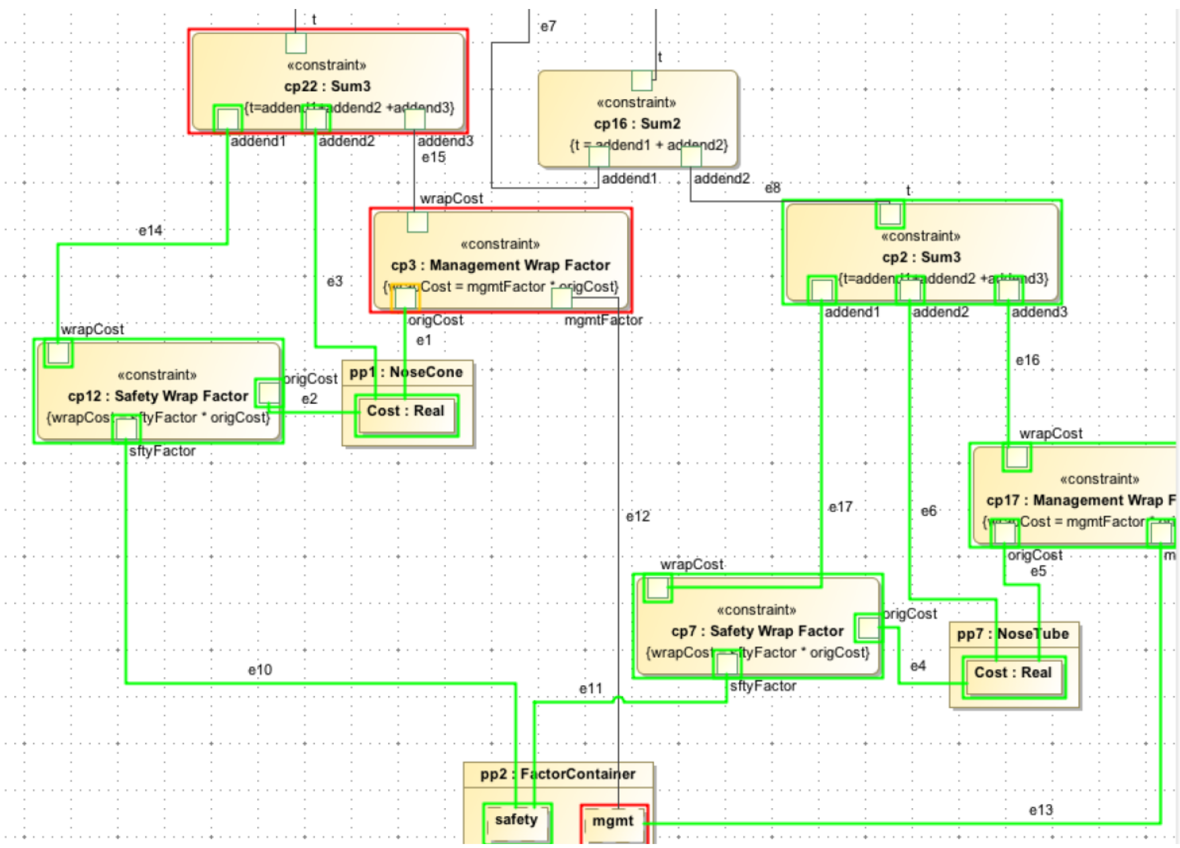


Figure 13 - Schematic diagram of a cost model used in the EHM study.

expertise to teach, advise, and guide our adoption of MBSE. These experts imbued our efforts with a maturity and credibility which would otherwise have been achieved only through slow, painful and expensive trial and error. In addition, because some of these experts have also been in positions of executive leadership, they were able to help executive leadership at JPL understand the value proposition for MBSE. So outside expertise has proven invaluable both for the quality of the infusion itself, as well as the institutional support for the infusion.

Peer Pressure Pays

The outside experts mentioned above could convey to our executive leadership their informed assessment of the state of the industry in a way that we practitioners could not. Their assessments are far more authoritative than ours could be, so that when they warn of JPL falling behind and becoming less competitive if it does not proactively engage with MBSE, the message is compelling and believable. In this way we have found that peer pressure pays in terms of building institutional support for MBSE.

Likewise we have found peer pressure within JPL to be an effective driver of infusion. IMCE has organized several lab-wide opportunities for emerging MBSE-based efforts to showcase their work and share lessons learned. The obvious benefit of course has been to cross-fertilize and share learning across many efforts. The additional benefit is the spirit of healthy competition which has been fostered.

4. FUTURE WORK

We will support the team in getting to and through the Mission Concept Review. We will produce a significant fraction of the required documents from the system model. We will work toward integration of other key models with the SysML model, including STK, AFT, and the mechanical model. We will build capabilities to support the team in Phase A and beyond, including providing the capability to develop and manage requirements in the system model.

5. CONCLUSIONS

The system model we developed for the Europa study has become the recognized authoritative source for a key and growing set of real engineering products including physical composition, mass, power, and system behavior. Our use of system modeling has contributed directly to the recognized high quality of our mission concept, including: increased stability of the concept description, increased reliability of the key technical resource estimates, and increased agility in the face of changing sponsor needs and priorities.

If these studies result in an approved mission, then that team will have much more useful information to draw on than the traditional study would have provided: our use of MBSE will pass to the project team a much more durable, rich and extensible body of information from which to start.

But regardless of the fate of the Europa study, our progress is already serving as a positive example and providing a powerful springboard to the next projects adopting MBSE – both within JPL and throughout NASA.

This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

6. REFERENCES

[1] - T. J. Bayer, S. Chung, B. Cole, B. Cooke, F. Dekens, C. Delp, I. Gontijo, K. Lewis, M. Moshir, R. Rasmussen, D. Wagner, “Model based systems engineering on the Europa mission concept study”, IEEE Aerospace Conference Proceedings, 05, 22, 2012.

[2] - Seung H. Chung, Todd J. Bayer, Bjorn Cole, Brian Cooke, Frank Dekens, Christopher Delp, Doris Lam, “Model-Based Systems Engineering Approach to Managing Mass Margin”, in Proceedings of the 5th International Workshop on Systems & Concurrent Engineering for Space Applications (SECESA), Lisbon, Portugal, October, 2012

[3] - T. J. Bayer, S. Chung, B. Cole, B. Cook, F. Dekens, C. Delp, I. Gontijo, K. Lewis, M. Moshir, R. Rasmussen, and D. Wagner, “Early formulation model-centric engineering on NASA’s Europa mission concept study” in *Proceedings of 22nd Annual International Symposium (IS2012)*, Roma, Italy, July 9–12 2012.

[4] - Europa Study 2012 Full Report. National Aeronautics and Space Administration, JPL D-71990, May 1 2012, publicly available here: http://solarsystem.nasa.gov/europa/2012study.cfm.

[5] - Conrad Bock, “A more object-oriented state machine”, Journal of Object-Oriented Programming, Vol. 12, No. 8, 2000.

[6] - Modelica language website, www.modelica.org.

[7] - R. Duren, et. al., “Systems Engineering for the Kepler Mission: A Search for Terrestrial Planets”, Aerospace Conference, 2006 IEEE.

7. Biographies

Todd J. Bayer *is a Principal Engineer in the Systems and Software Division at the Jet Propulsion Laboratory. He is currently the Flight System Engineer for the Europa Habitability Mission Studies. Most recently he was the Assistant Manager for Flight Projects of JPL's Systems and Software Division. He received his B.S. in Physics in 1984 from the Massachusetts Institute of Technology. He started his career as a project officer in the US Air Force at Space Division in El Segundo, California. Following his military service, he joined the staff of JPL in 1989. He has participated in the development and operations of several missions including Mars Observer, Cassini, Deep Space 1, and Mars Reconnaissance Orbiter, for which he was the Flight System Engineer for development and Chief Engineer during flight operations. During a leave of absence from JPL, he worked as a systems engineer on the European next generation weather satellite at EUMETSAT in Darmstadt, Germany.*

Brian Cooke *received a B.S. in Engineering Science and Mechanics from Virginia Tech in 1995. He has been with JPL for more than 15 years and is the recipient of three NASA Exceptional Achievement Medals. He is currently the Project System Engineer for the Europa Habitability Mission helping to develop and plan NASA's exploration of this intriguing Jovian moon. He has previously served as the Kepler Project System Engineer, Dawn Project V&V Engineer and the GALEX Instrument I&T Manager.*

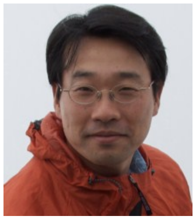
Bjorn Cole *is a systems engineer in the Mission Systems Concepts section of the Jet Propulsion Laboratory. His research interests are in the fields of design space exploration, visualization, multidisciplinary analysis and optimization, concept formulation, architectural design methods, technology planning, and more recently, model-based systems engineering. His most recent body of work concerns the infusion of systems modeling as a data structure into multidisciplinary analysis and architectural characterization. He earned his Ph.D. and M.S. degrees in Aerospace Engineering at the Georgia Institute of Technology and his B.S. in Aeronautics and Astronautics at the University of Washington.*

Frank Dekens *is a Systems Engineer in the Instruments and Science Data Systems division at Jet Propulsion Laboratory. He is a member of the project systems engineering team on the Europa Habitability Mission. Prior to this, he was the Instrument Systems Engineer for the Space Interferometry Mission. Frank received his BS and Ph.D. in physics, both form the University of California, Irvine.*

Christopher Delp *is the Systems Architect for Ops Revitalization task in MGSS. He is also a member of of the Flight Software Systems Engineering and Architectures Group at the Jet Propulsion Laboratory. His interests include software and systems architecture, applications of model-based systems engineering , and real-time embedded software engineering. He earned his M.S. and B.S. degrees from the U of A in Systems Engineering.*

Ivair Gontijo *is a member of the systems engineering team on the Europa Habitability Mission. Prior to this he lead the design and manufacturing of the RF packaging for the radar that landed the Curiosity rover on Mars. His area of technical expertise is optoelectronics and RF devices. In the 90s he worked on III-V semiconductor materials, semiconductor lasers and optical waveguides. For the past 15 years he led process Engineering and Packaging groups in the fiber optics industry, developing transmission systems at data rates up to 40GB/s. He has a BSc in Physics and an MSc in Optics from UFMG in Brazil and a PhD in Electrical Engineering from Glasgow University, UK.*

David Wagner *is a software system engineer and architect in the Flight Software Applications and Data Management group at JPL and was a principal developer of the Mission Data System in 2000-2006. Since then he has continued to apply MDS technology and State Analysis in several applications. He is currently a member of the project system engineering team on the Europa Habitability Mission formulation project. He has a BS in Aerospace Engineering from the University of Cincinnati, and MS in Aerospace Engineering from the University of Southern California.*



Seung Chung *Dr. Seung Chung is the Technical Group Supervisor of the Modeling and Verification Group at the Jet Propulsion Laboratory. His group is responsible for developing and maintaining the tools used to verify and validate spacecraft command sequences and for developing the spacecraft models and the operational rules that the tools verify and validate against. He also holds leadership roles in architecting model-based approaches to systems engineering for both ground and flight systems, including Europa mission concept studies. He received his Ph.D. in Autonomy from the Massachusetts Institute of Technology.*