Engineering Data Re-Use at JPL: Promise and Perils[1]

Craig E. Peterson
Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, CA 91109

818-393-2481
Craig.Peterson@jpl.nasa.gov

William M. Heinrichs
Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, CA 91109

818-354-1256
wmh@pop.jpl.nasa.gov

Abstract—In last year's IEEE Aerospace Conference Proceedings, a new approach to sharing engineering data using a Jet Propulsion Laboratory (JPL) developed Product Attribute Database (PAD) was described [1]. Since its development the PAD has been used on several projects at JPL with mixed results. The lessons learned in its implementation have wider implications for engineering processes and the associated data that is produced and consumed during a project's life cycle. In addition to the normal resistance of personnel to adopt a new tool who's benefits are not yet quantified in practice, the need to spend substantial effort early on to establish and promulgate a common understanding of the data has impeded acceptance and use of the PAD. The limitations of the original user interface and the lack of existing project data (either in template or detailed form) has also played a role in delaying widespread user acceptance.

Despite these drawbacks, the initial use of the PAD has highlighted misunderstandings regarding definition of engineering data and has led to numerous fruitful discussions among engineering disciplines. Even apparently intuitively obvious data such as the "total flight system mass" or the spacecraft mechanical configuration coordinate system have required refinement to eliminate the natural ambiguity that engineers routinely accommodate, but that leads to confusion when automated modeling tools are the creators and consumers of the data. These discussions have led to increased insight into engineering processes, particularly those using the model based design approach that is now being implemented at JPL.

TABLE OF CONTENTS

1. INTRODUCTION

PAD Overview

The PAD system [1] has been designed to be an integrating tool in information systems that support development projects. As Figure 1 shows, the PAD database supports interfaces to several classes of tools including the primary graphic user interface (GUI) known as the Product Attribute Conversation Tool (PACT). While the PAD is built on the Oracle™ database management system, the PACT was originally developed in Visual Basic for Applications (VBA) in Microsoft Excel™ and has recently been implemented in JAVA™, allowing it to run on almost all platforms. The JAVA™ implementation has also been integrated with a set of HTML pages for Web browser based access. Currently PAD supports interfaces to the DOORS™ requirements documentation system, simple systems engineering and accounting tools, high-end engineering models developed in Statemate, Matlab, or other modeling environments, a Project Data Management System (PDMS), a Flight Systems Testbed, and a Mission Data System (MDS) software development environment.

The PAD data structure allows for the creation of a product hierarchy or product breakdown structure (PBS), similar to a work breakdown structure, but product oriented. The PBS is the logical representation of the physical products produced in during a project's lifecycle and can include hardware, software, documentation, and even mission events during operations. An abbreviated example is shown in Table 1.

Products in the PBS can then be assigned Parameters, which is a combination of an Attribute and a State. Attributes are the measurable characteristics of the product and must include the appropriate units in its specification. States are required as a qualifier of the attribute, since attributes may be time or activity dependent. A time dependent attribute might be the total output power in watts. When applied to a Solar Array, it is important to distinguish between the "beginning of life (BOL)" and the "end of life (EOL) states for that attribute. Attributes without state variations use the "ALL" state. Examples of parameters are shown in Table 2.

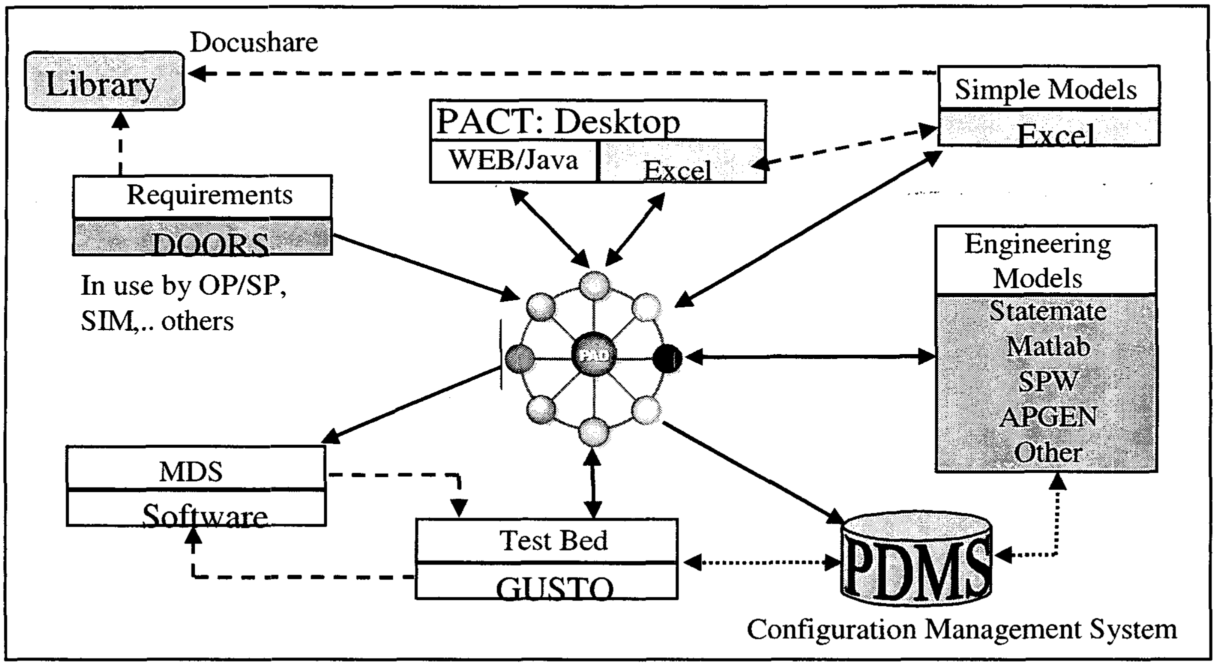


Figure 1 – PAD Interfaces

In the PAD, parameters can have different versions, based on the source of the value associated with it, so a Value Name field is included specify the different versions. Common versions of PAD parameters would include Vendor Specification, Current Best Estimate (CBE), Allocation, and Requirement. For versions that are designated Requirements, the PAD has an additional field to indicate whether the requirement value is a maximum, minimum, or nominal value. This field is used for specialized searches of the PAD that identify values that either exceed or fail to meet requirements. Examples of parameter versions are shown in Table 2.

It is the combination of these five fields, product, attribute (with units), state, value name, and (in the case of requirements) requirement type, that uniquely specify a value in the PAD. The PACT provides the ability to search and retrieve values based on any or all of these fields. Once a set of data is retrieved, the PACT provides the ability to output the results as a tab-delimited text file, making it easy to incorporate in spreadsheets or documents. The tables in this paper were constructed from actual PACT output.

As shown in Figure 1, the PAD also supports interfaces to other software tools, primarily through a custom C language application programming interface (C-API). The C-API allows these tools to read from and write values to the PAD. VBA add-ins for Excel™ are also supported which allow easy integration of the PAD with spreadsheet analysis tools. Additional VBA functions provide the capability to upload new PBS branches (or an entire PBS) into the PAD in one operation, or create large numbers of parameters and versions by constructing the Cartesian product of sets of products, attributes, states, and value names.

Table 1 – Example PBS

Project		
	Project Plan	
	Outreach Products	
	Launch Vehicle	
		shroud
		booster
	Spacecraft	
		Structures and Mechanisms
		Propulsion
		Power
		Thermal
		Telecom
		CDS
		ACS
	Instrument	
		Enclosure
		Electronics
		Sensor
	Science	
		Science Plans
		Science Products
	System Engineering	
		Project Requirements Doc.
		Analyses
	Mission Design	
		Mission Plan
		Navigation Plan

Table 2 – Example Parameter Versions

PRODUCT	ATTRIBUTE	STATE	VALUE NAME	TYPE
ARPS	Mass, Dry (kg)	All	Baseline	None
ARPS	Mass, Dry (kg)	All	CBE	None
ARPS	Power Available (W)	EOL	Baseline	Min
ARPS	Power Available (W)	EOL	CBE	Min
ARPS	Quantity (#)	All	Baseline	None
Battery	Mass Uncertainty, Dry (%)	All	CBE	None
Battery	Mass, Dry (kg)	All	CBE	None
Battery	Mass, Dry (kg)	All	Requirement	Max
Battery	Quantity (#)	All	CBE	None

Security of data is an important concern and is implemented in the PAD by establishing ownership of both products and values. Product owners can alter the PBS branches below the product they own. Value owners can write new or updated values for the parameter versions they own. Oracle™ logon ID's and passwords are the basis for this security implementation.

Best practice guidelines have been established for use of the PAD and are included in the PAD training. The primary purpose of best practices is determining what should and should not go into a PAD. It is equally important to know what should not go into a PAD because too much extraneous information can make the system confusing and difficult to use. Examples of best practices are:

- Services, support, and project phases are not products.
- Use Specific (Not Generic) Product Names:-e.g., Battery Control Electronics, Not Electronics.
- Ensure that the children of a product form the parent when taken together.
- Don't confuse the PBS with the document tree or the org-chart.
- Always specify units for attributes.
- Use comments to provide context – remember, the engineer or manager querying the PAD may not have your background.

More examples of best practices can be found in [1].

Putting the PAD into Practice

Version 1 of the PAD, PACT, and associated interfaces were made available to JPL projects last spring. Training courses were held and numerous conversations occurred between project staff members and the PAD development team. Project PADs were established and populated, initially by PAD developers in consultation with project system engineers. Also, as one of the capabilities developed for the Develop New Products (DNP) Implementation Project, the PAD and its interfaces were central to the success of an end to end demonstration of DNP tools and processes held in May of 1999. This demonstration was referred to as the Integrated Capabilities for Engineers (ICE) demonstration and was intended to illustrate how a project might use the entire suite of DNP tools and processes to support project engineering activities prior to a Preliminary Design Review (PDR).

The results of these activities were more divergent than anticipated. The following paragraphs provide more detail on where the PAD lived up to its promise, and where the perils of inserting new technology into a existing product development culture limited its success.

2. PAD AND THE ICE DEMONSTRATION

The Integrated Capabilities for Engineering (ICE) Team was given the assignment of taking several tools and processes that had been developed largely in isolation, and integrating them into a convincing demonstration of their capability to produce products required prior to a standard Systems PDR. This assignment was all the more challenging due to the short time (less than 8 weeks) allocated to it. The PAD (as it was designed to do) was the central integration point for the various tools and the interfaces had been verified and used in limited cases prior to the start of this exercise. However, a convincing demonstration in this context meant more than just showing that bits could be passed into and out of the PAD. The demonstration needed sufficient verisimilitude to convince the review board that these tools and processes were largely ready for prime time use on projects. This meant constructing and populating a PAD with realistic project data. Over 1000 parameter versions were included in the PAD used for the demonstration. Fortunately, the PAD for the Europa Orbiter mission had been developed to the point where the ICE team could make a copy and only make those alterations needed to support specific demonstration activities.

Difficulties arose in the initial discussions among the members of the demo team when it was discovered that subtle differences in assumptions existed between various members of the team. The most glaring example was the disagreement over the meaning of the parameter version "Flight System – Mass(kg) – All – CBE". After the expected debate over dry mass versus wet (loaded with propellants and pressurants) mass, it was discovered that another wrinkle existed. While the total flight system wet mass at launch is the upper bound for attitude control algorithms, it does not represent the total mass that the launch vehicle must carry into space. The difference is the mass of the launch vehicle adapter, which mates the flight system to the launch vehicle, but separates from the flight system when it is released. As a result, what was usually thought of as one parameter version had to be separated into three distinct versions in order to avoid confusion.

These types of discussion arose frequently in the initial integration effort. Fortunately, an attitude of teamwork prevailed, so no team member was alienated as a result of this process, despite the enormous pressure on the team.

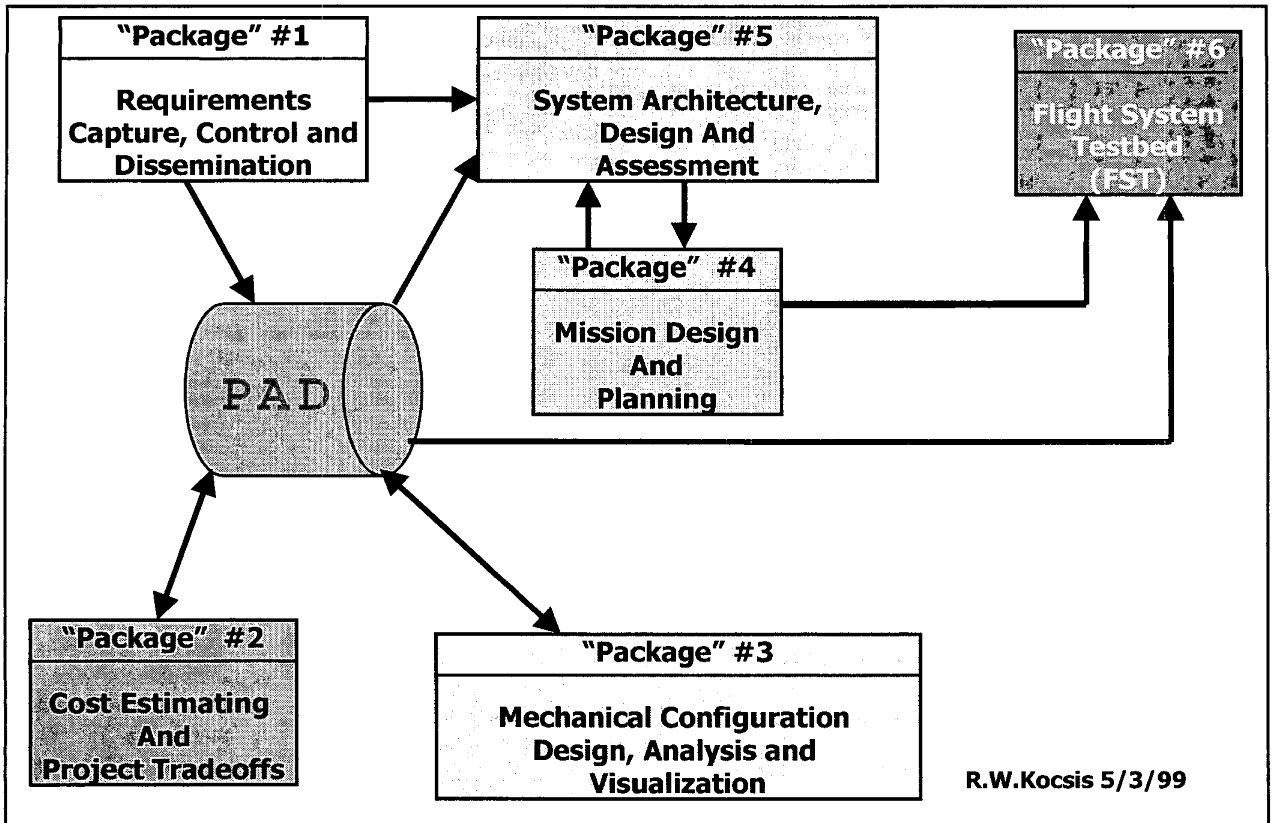


Figure 2 – ICE Demonstration Process Flow

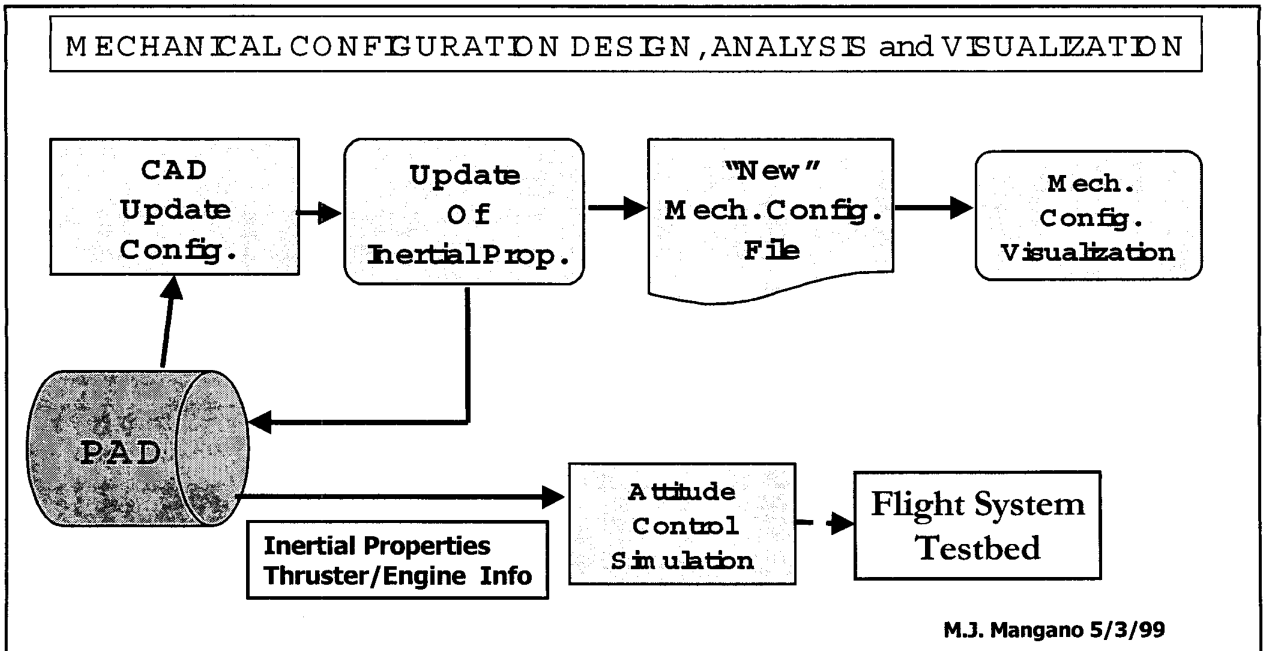


Figure 3 – Package #3 Detailed Flow

Once these issues were surfaced and resolved, the team was able to focus more on the process and the products of the process, rather than on the data and tools supporting it. As the demonstration took shape and the processes began to be run end to end, new subtleties appeared. One of these was the issue of process synchronization.

While the DNP process approach emphasizes concurrent engineering, the ICE demo made it clear that some activities must be performed sequentially, or confusion may result. Figure 2 shows the very high level view of the combination of tools and processes (referred to as packages) that formed the ICE demo. The packages were demonstrated in the same order as the package numbering. Figure 3 shows an expansion of the Mechanical Configuration Design, Analysis and Visualization (Package #3.) During the demonstration, a third power source is added to the flight system. This results in a modification to the requirements (Figure 2 – Package 1), the spacecraft dry mass and corresponding propellant requirements (Figure 2- Package 2) and the spacecraft configuration (Figure 2 – Package 3). Once the spacecraft configuration was updated, its mass properties (inertia tensors) could be recalculated and used for attitude control simulations. The ultimate destination was the Flight System Testbed (FST), an integrated set of software and hardware simulators that can replicate the behavior of a spacecraft.

When the mass properties for the spacecraft are initially calculated based on the addition of another power source, the resulting spacecraft is not well balanced between the center of gravity and the thrust moment arms for the attitude control thrusters. The usual procedure at this point is to perform some initial calculations to determine the mass and location of added ballast that will remedy this situation, and this was planned as part of the demonstration. But because the mass properties were developed using a CAD tool and the balancing calculations were performed on a spreadsheet, the intermediate (unbalanced) inertial values were written into the PAD where it could be picked up by the spreadsheet. During a demo dry run, the FST lead called up, wondering why he was unable to control the spacecraft during his simulated orbit insertion maneuver. This caused some real concern for the validity of the demo until it was determined that he had read the values from the database after the mass properties were updated for the third power source, but *before* the spacecraft had been balanced and the mass properties were updated again.

In one respect this was an important test of the validity of the tools being applied – an unbalanced spacecraft should be difficult to control. This lesson was not lost on the review board members who recognized that a negative test is frequently as important as a positive result. But it also illustrated a valuable lesson in integrating processes and tools with a central database, namely that tools cannot be applied blindly simply because data exists. It is the responsibility of the data consumer to insure that the data has reached the necessary maturity before using it.

3. PAD AND PROJECT USE

Two projects at JPL became initial users of the PAD. Both were in the pre-project (requirements definition) phase, which seemed to be a reasonable time to try and engage them in population and use of the PAD. Special training sessions were scheduled so that several people on each project would be trained together, followed by a separate session to try and work out both the product structure of their PADs and the needs for data sharing between project personnel. This was where the troubles began.

Although the project system engineers (who had been our primary points of contact) were very enthusiastic and supportive of the PAD, the other engineers were more conservative in outlook, and saw the PAD as providing little or no value added to their activities, compared to the effort involved. Although many of the participants had younger engineers working for them that used automated modeling software, few of the engineers present during training used the tools themselves, so the integrating power of the PAD was not a strong selling point. In spite of numerous one on one sessions and even the development of small custom spreadsheet applications for various engineers, the projects' use of the PAD remains limited to only a small number of (largely) junior engineers that use it share data between modeling tools. Several conclusions can be drawn from this experience.

First, the amount of time required to initially set up a project PAD needed to be reduced. This led the PAD team to develop the templates described in the next section.

Second, if the use of the PAD was to extend beyond the more IT-savvy young engineers, there had to be relatively immediate pay-off. The advantages of using the PAD had to be easily recognized by those more concerned with managing engineering activities than performing them. If a project manager (as opposed to a project system engineer) could see the advantage of using the PAD, then the rest of the project personnel likely would, too, or at least be more receptive to the idea. The existence of a ready-made template would help in this area as well, but more would be better. This led the PAD team to consider attempting to archive data from previous projects, which could then be transferred to a new project's PAD. When completed, this could allow projects to rapidly pull together detailed information based on similar or identical systems. The following section describes these efforts in more detail.

4. TEMPLATES AND PAST PROJECT DATA

Common data definition and organization is essential for effective communication through information technology systems. The PAD is no exception. All tools and processes integrated with the PAD must have a common understanding of the data definition and organization. Failure to have a common understanding yields a communication breakdown potentially having disastrous and/or costly effects.

To facilitate projects' organization of PAD data the PAD Team has developed two products: PAD best practices and the PAD template. The PAD best practices are the result of detailed analysis of studies and military specifications in the data definition and organization fields[2]. PAD best practices are reduced to a straightforward format in the PAD training materials. The PAD template incorporates PAD best practices and models a fictitious skeletal space flight project.

The PAD template was built using the PAD best practices to verify which best practices work and which do not. Data for the template was mainly taken from four JPL TeamX[3] studies. These studies include a high-Earth orbiter, a planetary orbiter, a planetary lander, and a sample return mission. From these studies, it was determined what products are needed to complete the mission and what technical parameters are required to fully describe those products. The PAD template is also compatible with the standard project Product Breakdown Structure (PBS) that has been developed as part of the DNP Project Leadership Process, which will be used for all future projects at JPL.

The PAD template acts as a "scratch PAD" for newly formed projects to jump start data definition and organization. The template contains a well-defined PBS to the subsystem level (level III.) The template also provides parameters to technically describe the template products. These parameters incorporate common units, definition of attributes (dry vs. wet mass), based on the minimal data set required for spacecraft preliminary design.

A newly formed project can now incorporate a copy of the template in their PAD, thus providing a jump start in organizing and defining their project data.

The template was internally tested before release for general use. One test modeled a previously launched mission, Stardust. By populating a template with Stardust data, the PAD Team has been able to further refine the template by expanding the number of products and identifying parameters that better describe the products and those that do not.

Stardust is but the first of several missions that will be captured in the PAD. Metrics and best practices have been collected for establishing a PAD archive and a collaborative relationship has been established with both the JPL Archivist and the newly begun JPL Knowledge Management initiative. A PAD archive will eventually contain several previously flown missions – all based on the standard PAD template. This archive will serve as a knowledge base for current missions to understand both the products of the past and their needs for their particular mission. New PACT capabilities are being developed to copy all or portions of these missions into a project's PAD, thus providing an efficient mechanism for engineering data reuse and an obvious benefit to projects.

5. Future Plans

As noted in the previous section, more archives of existing project data are planned. These archives will be useful to not only to formal projects, but can support proposals for future projects as well. The PAD development team is considering forming a partnership with the JPL proposal center. The amount of data managed for a proposal is substantially less than that required for a flight project, and may make a more appropriate application for the current PAD. This could have two additional benefits. First, the accumulation of proposal PADs would make a handy data archive for future proposals to draw on. Second, winning proposals would already have a PAD set up and ready for their transition to a project basis, saving them the time and effort of starting anew.

Additional plans include providing the capability for JPL's Advanced Product Development Team (Team X) to capture the results of their studies in a specially designed PAD. This would allow them to archive and retrieve the results of their studies, enabling them to "pick up" where they left off with study revisions or to perform variations on the study, without losing their baseline concept. These archives could also be used downstream by the project being studied, insuring re-use of the study data during project development.

Another possible customer of the PAD is the Mission Data System development team that is developing the new end to end information system for use on JPL flight projects. Discussions have just begun in this are but show promise. It is only natural that software developers would recognize the utility of a project PAD.

Additional features are planned for the PACT. They include new security controls that allow collaborative use of a PAD by system and subsystem contractors off-site. (A mode of project operation that is becoming increasingly common.) These controls will allow restrictions to be placed on access to specified products and parameter versions, thus allowing competing companies to collaborate without fear of disclosing proprietary information.

Until more data has been archived and the additional features delivered, the PAD team will continue to work with projects at a grass roots level. Those engineers who are

---

[2] Mil-Hdbk-881 Appendix F: Space Systems Work Breakdown Structure and Definition, for example.

[3] JPL TeamX is a team of 15+ engineers who rapidly prototype a mission concept into a feasibility study. Products of TeamX include estimated total mission costs, mock up of a spacecraft, and identification of potential trouble spots.

working with modeling tools and recognize the utility of an integrating database will be the primary route to demonstrate the PAD's value to project management.

6. CONCLUSIONS

It is always difficult to know *a priori* the best way to introduce a new tool into an existing culture. The experience with the ICE demonstration made it clear that the PAD has great utility when all the members of a team are fully engaged in using it. It is also clear from experience with flight projects that the majority of engineers need to see more immediate value from a new tool if they are to become advocates of its use. This has led to a two pronged approach. First, of finding ways to provide immediate value through access to various kinds of archived data, and second, to build on support from those engineers whose need to share model data allows them to recognize the value of the PAD in data capture and re-use.

7. ACKNOWLEDGEMENTS

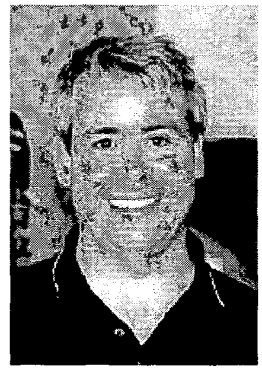
The authors would like to thank the other members of the ICE demonstration team, particularly the ICE demo lead, Robert Kocsis, mechanical systems lead Michael Mangano, attitude control modeler David Henriquez, Flight System Testbed lead Steve Allen, and system modeler George Cancro. Also deserving our thanks for their many hours of engagement are Richard Stoller and Curt Henry. Our thanks to the rest of the PAD development team, Thomas Clymer, our DBA, Tammy Roust, VBA developer and PAD integrator for the Project Trades Model, and Stephanie Sharo, who populated the PAD template and the Stardust Mission archive. Finally, our thanks to Dr. Joel Sercel, whose leadership brought the PAD to life.

8. REFERENCES

[1] Joel C. Sercel, Thomas F. Clymer, William M. Heinrichs, "The Product Attributes Database (PAD): First of a New Class of Productivity Tools for Product Development". *1999 IEEE Aerospace Conference Proceedings,* March, 1999

9. BIOGRAPHIES

Craig Peterson *is a Senior Engineer of the staff in the Jet Propulsion Laboratory's Mission Architecture Development Group specializing in developing new and innovative mission concepts in support of NASA's Space Science Enterprise. His current activities involve working with projects to assist them in implementing and using their project PAD. He is also the Program Architect for the Solar System Exploration Technology Development Program and as such is a user of the PAD. Peterson holds a BA in Mathematics and Physics from Gustavus Adolphus College and has pursued post-graduate studies in applied mathematics and engineering at UC San Diego, UCLA, and the California Institute of Technology.*



William Heinrichs *is an Associate member of the information technology staff in the Jet Propulsion Laboratory's Engineering Economics & Costing group. Most of his recent work specializes in relational database projects. His current activities involve managing the PAD team, acting as the technical lead for the PAD's interfaces, and programming the PAD's main graphical user interface (PACT). Heinrichs holds a BS in computer science and an MBA, both from Rensselaer Polytechnic Institute..*

The research described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.