Applying Concurrent Engineering to the Joint Australian Engineering Satellite (JAESat)

Mr. Cameron Boyd
Australian Space Research Institute Ltd (ASRI)
PO Box 3890, Manuka ACT 2603, Australia
Tel: +61 (0)7 3366 0498
Mob: +61 (0)409 110 106
E-Mail: cboyd@asri.org.au

A/Prof. Werner Enderle
School of Electrical and Electronic Systems Engineering
Queensland University of Technology
GPO Box 2434, Brisbane QLD 4001, Australia
Tel: +61 (0)7 3864 9132
E-Mail: w.enderle@qut.edu.au

Mr. Shaun Wilson
Australian Space Research Institute Ltd (ASRI)
PO Box 3890, Manuka ACT 2603, Australia
Tel: +61 (0)2 6239 4288
Mob: +61 (0)438 394 288
E-Mail: swilson@asri.org.au

A systems engineering approach is required for the development of the JAESat micro-satellite under joint development by the Queensland University of Technology (QUT) and the Australian Space Research Institute (ASRI). This paper reviews concurrent engineering approaches and micro-satellite development. It outlines a recommended approach based on the history and complexity of the project, the benefits of current information technology norms, and industry experience, including modeling the system and business processes, project management techniques, and the development of a concurrent engineering environment that makes use of systems engineering and project management tools.

Keywords: concurrent engineering, systems engineering, small satellites, standards.

Introduction

The JAESat micro-satellite concept is based on the Radio Amateur Satellite Corporation of North America (AMSAT-NA) micro-satellite concept first realized in 1990 through the launch of four micro-satellites[1]. The JAESat Project originated with the Australian Space Research Institute Ltd (ASRI), a voluntary space technology educational and research organisation, and the Queensland University of Technology (QUT) in early 1997.

The aim of the project was to provide 'hands-on' space technology experience to students, industry and the general public. Shortly thereafter, the Cooperative Research Centre for Satellite Systems (CRCSS) became involved as a sponsoring agency. The University of New South Wales (UNSW) and the University of Queensland (UQ) also became involved to

[1] King, J. A., McGwier, R., Price, H., White, J. (1990), 'The In-Orbit Performance of Four Microsat Spacecraft', *Proceedings of the 4th Annual AIAA/USU Conference on Small Satellites*, 1990.

provide technical expertise and effort from both staff and students. Due to cost and schedule over runs with the FedSat Project in 2000, the CRCSS was forced to withdraw funding support from the project. As a result of this, the project required rescoping and rescheduling in an effort to ensure that the project could be completed.

The satellite has been designed and sub-systems prototyped through student research primarily at QUT. ASRI financially assisted the development of a solar cell manufacture capability in exchange for required solar cells at UNSW and contributed industry professionals to act as team leaders and engineers for the project. ASRI was unable to rescope and reschedule the project to meet the required deadlines for the launch service provider. QUT was approached to take back management of the project to completion. This management change means the project has come full circle. This paper draws from multi-disciplinary approaches to recommend a solution to the systems engineering for the JAESat micro-satellite project.

Background

Over the period of the JAESat Project, the team has consisted at any one time, of some 15 to 30 people from a range of backgrounds and educational levels. At most 5 to 10 people were collocated in the same city. As personnel were mostly undergraduate students, the quality of industry professionals could not be expected. However, in contrast to the industry professionals, the students had deadlines that they could dedicate themselves to, with lulls in activity only over holiday periods. The project required low costs, with a total project cost in 2002 of approximately $280,000 including donations of in-kind contributions.

During this period, the ASRI Project System was established to assist projects with project management and systems engineering. The joint experience of the industry professional membership base suggested the adoption a set of standards to assist in project management, systems engineering and product assurance. ASRI recommended the adoption of the European Cooperation for Space Standardization (ECSS) for the primary reason that they were readily available and that ASRI could contribute to their development. ASRI's Australian Lunar-Mars Investigation and Technical Evaluation (ALUMINATE) Project developed a prototype virtual project office based on the ECSS and the first draft of a tailoring process for the ECSS. The ASRI Project System was only partially implemented on the JAESat Project. As the project was already underway when the decision was made to adopt the ECSS, this resulted in substantial rework and loss of knowledge within the project.

Originally intended to run for two years, the project is now in its 6th year. Lack of clear requirements, constant redesign, and scope-creep caused partial iterations through the project system as different team members moved through the project. As participants never felt that they owned the project, each time a new team took over the development of a sub-system, they placed the old sub-system aside and started to work on a completely new sub-system. Information leakage, caused by a lack of change and knowledge management processes, resulted in a limited number of documents at varying stages of design and very few referenced each other.

With the hand-over of the project management to QUT, a workable systems engineering solution is being implemented to successfully complete the project to launch within the required timeframe of two years. Most of the team will be collocated at QUT with a mix of undergraduate and graduate students contributing. There will be at least 12 full-time personnel on the project in the first year, where the first AMSAT-NA micro-satellites had 16 – 24 industry professionals over two years. Student deadlines still provide a useful project milestone marker. The project still requires a low-cost budget, but can draw from some of the pre-existing work from previous iterations to lessen the cost. This paper draws from industry experience as a software quality assurance coordinator in the development of a project system for an automated broadband telecommunications connection management solution, as a technical consultant and trainer for an enterprise resource planning information and knowledge management system, and as a researcher with ASRI in the development of the ASRI Project System and Virtual Project Office (VPO)[2].

European Cooperation for Space Standardization

The ECSS is supported by a number of European aerospace organisations, including among others the British National Space Centre (BNSC), the Centre National d'Etudes Spatiales (CNES), Deutsches Zentrum fur Luft- und Raumfahrt (DLR), the European Space Agency (ESA), and the Canadian Space Agency (CSA). This group aims to develop a coherent set of standards to improve the efficiency of space projects, divided into three streams of project management, product assurance, and engineering. In many cases this has been the combination of several different European standards on the same topic into one standard document. In other cases, the existing standards were updated or specialized for space systems. Where no standards exist, the ECSS working groups develop them.

In 2000, ASRI adopted the ECSS as the basis for the ASRI Project System. Several standards systems were investigated, such as military (MIL-STD), International Standards Organisation (ISO), and Institute of Electrical and Electronic Engineers (IEEE) standards. Additional specialized standards were also investigated. Many of the investigated standards were found to be inputs into the ECSS development process. For this reason, and as the ECSS standards are coherent, with respect to both English and standard units, and available for use on the Internet, it was selected. Furthermore, the ECSS has been implemented or assessed against numerous satellite or satellite technology projects, for example:

- The INTErnational Gamma-Ray Astrophysics Laboratory (INTEGRAL) is an astrophysical observatory providing detailed images and high-resolution spectrometry of gamma-ray sources. It was developed by ESA as a part of the Long Term Scientific Plan (Horizon 2000) and was launched on 17 October 2002. The software process implemented for this successful project was used as a basis for a case study to validate the ECSS Software process defined in ECSS-E-40.
- The Gravity and Ocean Current Explorer (GOCE) aims to accurately measure the Earth's gravitational field and study ocean currents. It is the main satellite of the ESA's Earth Observation Program and used the Concurrent Development Environment (CDE). The CDE provides power budgets, attitude and control simulators, avionics and operational databases, and a variety of thermal models. This project tailored the ECSS Verification process defined in ECSS-E-10-02A to develop test plans and to define interfaces for subcontractors. It also implemented the overall ECSS Systems Engineering process as defined in ECSS-E-10A, the ECSS Software process defined in ECSS-E-40, and the ECSS Packet Utilisation Standard defined in ECSS-E-70-41A. Software quality assurance was implemented as defined in ECSS-Q-80A.

[2] Boyd, C. (2001), 'ALUMINATE Project Update and the Virtual Project Office', *Proceedings of the 11th Australian Space Research Institute Conference*, Sydney, December 2001.

- The NEw GEneration Satellite ARchitecture (NEGESAR) project intends to develop a strategy to launch active and complex electronic equipment into space using quality COTS components. Specifically, it aims to provide a system that decreases the high vibration loads, increasing thermal efficiency and high-energy particle shielding, and increasing the performance to cost ratio such that non-space rated equipment can be used in space applications. This project uses the ECSS Space Environment standards as defined in ECSS-E-10-04A.
- The French-Brazilian Micro-satellite (FBM) project is a joint project between CNES and Brazil's National Institute for Space Research (INPE) to develop, launch, and operate a scientific and technology demonstration micro-satellite with experiments and sub-systems from both countries. This project is incrementally implementing the ECSS standards, specifically the ECSS Verification process defined in ECSS-E-10-02A, the ECSS Ground systems and operations standards defined in ECSS-E-70, and the ECSS Software process defined in ECSS-E-40. It is understood that this project has since ceased.

The process ASRI is currently working through is the tailoring of the ECSS, based on ECSS-M-00-02A. This involves the selection of a level where modifications will need to take place, then the modification of key standards to meet ASRI's requirements. These modifications are based on the size, cost, and duration of ASRI projects, the organizational complexity, the number of collaborating organizations, the nature of human resources within the organisation, the identified technical and project risks, the available funding, and additional constraints, such as specific Australian standards. The goal is to produce a set of modified standards that each ASRI project will tailor further for each specific project. In essence, this will be a "bare essentials" version of the ECSS specifically for ASRI and the development of a project system where individual projects within ASRI can make use of a standard project system, templates, and tools.

The JAESat Project immediately made use of the ECSS engineering documents, specifically the space environment models, and material and component lists, however, it did not tailor the ECSS to fit the project. Later, when ASRI established a "generic space project" system that was then expected to be tailored further for each individual project, it was not implemented for JAESat. It only drew limited lessons from the systems engineering or project management aspects of the system. These lessons were more to do with the establishment of information management norms, including systems engineering templates, project management and business processes, and selected standards for the project to adhere to. These issues are not trivial to managing a project, but they are not space project specific.

Project Management

The strangest thing about project management is that many practitioners seem to believe in a "perfect project", and that this "perfect project" is what they aim to produce with their preferred project implementation. There is no "perfect project". More fundamental to the success or failure of a project is the decision by the project team to either design a project implementation to get as close to the "perfect project" or to design a project implementation to get as far away from the worst project. Another approach that often works is to develop the project implementation most likely to happen. The belief that there is no such thing as a "perfect project" has been born from industry experience in software and aerospace fields. Interestingly, systems engineers do not believe in a perfect systems engineering approach, only in the systematic and logical application of scientific and engineering efforts to produce a product that meets the customers' requirements.

Each project manager has a preferred paradigm. These include such paradigms as evolutionary, rapid, waterfall, V, and spiral management. All paradigms have pros and cons that need to be weighed up with respect to the individual project methodology and systems engineering approach, the project tools and personnel, and the associated technology.

Originally, the JAESat Project has been focused on the end-point of manufacture, and has been a simple project paradigm "develop and test" loop for student academic requirements. This simple project paradigm is shown in Figure 1. Several implementations of the JAESat Project model used this paradigm, including lecturers managing student teams, student project managers managing the project team, and industry professionals managing the project team.

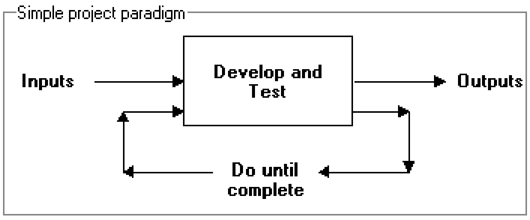

Figure 1. Simple project paradigm

When ASRI negotiated a free launch for the micro-satellite with an expected delivery date, the original end-point moved from manufacture to operations and disposal, which called for a more complicated project paradigm. Through a process of sporadic evolution, a waterfall paradigm was selected, shown in Figure 2.

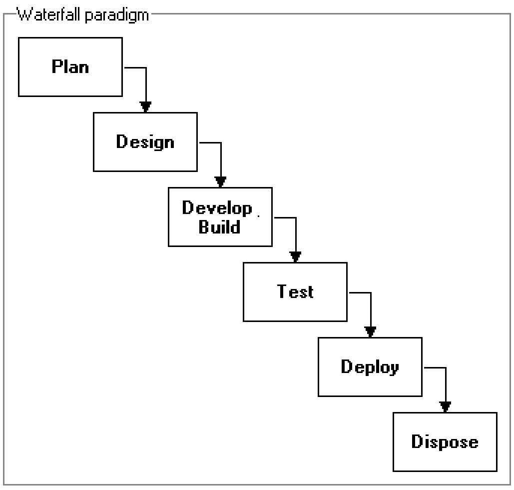

Figure 2. Waterfall paradigm

The Waterfall paradigm is among the easiest to understand and implement. It lends itself well to using a lifecycle approach, with the ability to add "kill points" and reviews at the precipice of each waterfall section. It defines requirements early in the project lifecycle and they are not expected to change. It also enables relatively easy documentation trails to be established, and quality assurance procedures to be implemented by associating the reviews and documentation with quality audits. Unfortunately, this paradigm can easily swamp the project with too many documents, or badly integrated documentation sets. Also, generally in a real project, there are feedback loops that re-iterate one or more stages of the paradigm. For example, there may be issues identified during testing that lead to re-development and/or re-design. The personnel are expected to be reasonably familiar with potential methodologies to solve a problem, but not necessarily the technical aspects of the solution.

As the JAESat Project is a joint university-industry project undertaken mostly by students, it requires deliverables for assessment. Prototypes have been developed for the design and develop "kill points" to assist with assessment, as shown in Figure 3. Unfortunately, student teams often find it easier to develop entire sub-systems after a handover than perfect a previous design. This is due to misunderstanding the scope of the system or sub-system concept, poor documentation, and potentially due to assessment requirements. It inevitably results in a new sub-system produced for each replacement student team, and ultimately both similar flaws and few lessons learned. This issue clearly identifies a management challenge to initiate appropriate and timely handovers to prevent this constant re-design.

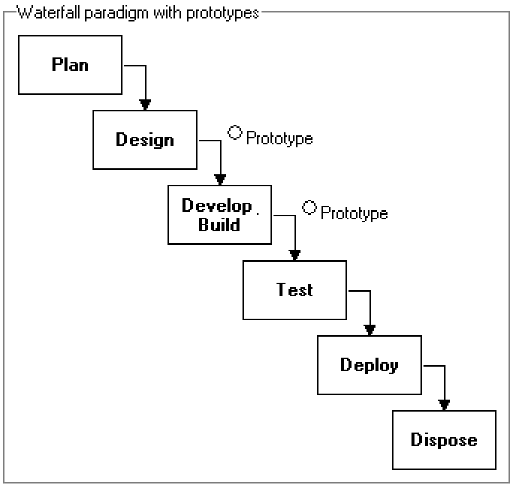

Figure 3. Waterfall paradigm with prototypes

Additional standards and methodologies can then be layered over these paradigms to produce the project model. In many cases these will be defined by the location the project is being developed and operated in, and the product being developed. For the JAESat Project, the ECSS were used as the basis for standards, with Australian standards overriding where necessary, for example, risk management, where the AS/NZS 4360:1999 – Risk Management[3] standard is used.

The current project paradigm, based on the waterfall paradigm, with attention to the phasing structures of the ECSS, is shown in Figure 4. In this diagram, feedback loops have been added to show where potential re-design or re-development can occur. Alternately, it allows the individual design and development teams to decide on the testing methodology and timing.

[3] Standards Australia (1999) 'AS/NZS 4360:1999 – Risk Management', 1999.

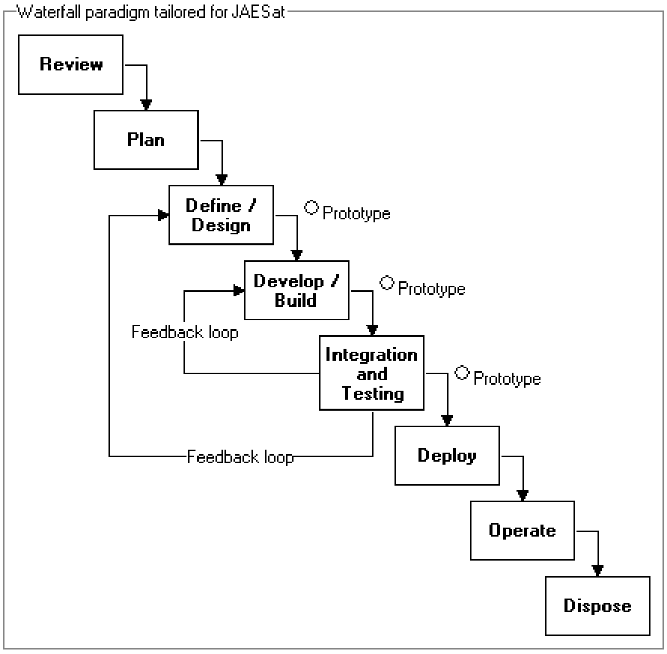

Figure 4. Waterfall paradigm tailored for JAESat

The phases defined in the ECSS are mapped clearly to the phases of the JAESat Project as shown in Table 1.

ECSS Phase[4]	JAESat Phase
0 – Mission Analysis/Needs Identification	0 – Review
A – Feasibility	A – Plan
B – Preliminary Definition	B – Define/Design
C – Detailed Definition	C – Develop/Build
D – Production and Ground Qualification Testing	D – Integration and Testing
E – Utilization	E – Deploy, Operate
F – Disposal	F – Dispose

Table 1. Comparison of ECSS and JAESat Project phases

From the project paradigm and lifecycle structure, project management disciplines such as business modeling, cost and schedule management, configuration management, human resources, information and documentation management, and integrated logistics support can be established. These disciplines are clearly identified in both ECSS and best practice literature, and are based on the structure pioneered by Rational Software for the Rational Unified Process software suite.

The level of management is also something that may require some flexibility when it comes to a satellite project. This is not only due to the number of contractors, or external agents, at a variety of levels that need to be integrated. Sometimes the technology transfer issues associated with procurement of equipment or sub-systems imposes an artificial level of management on specific areas of the project. These levels also mimic the levels defined by the

[4] European Cooperation for Space Standardization (1996) 'ECSS-M-30, Space Project Management – Project Phasing and Planning', ESTEC, Revision A, 19 April 1996.

ECSS for the systems engineering approach, which leads to a clear understanding for personnel where technology transfer issues may be of international interest.

These three dimensions, the lifecycle structure, the project management disciplines and the level of management, provide the basis for the project model as shown in Figure 5.

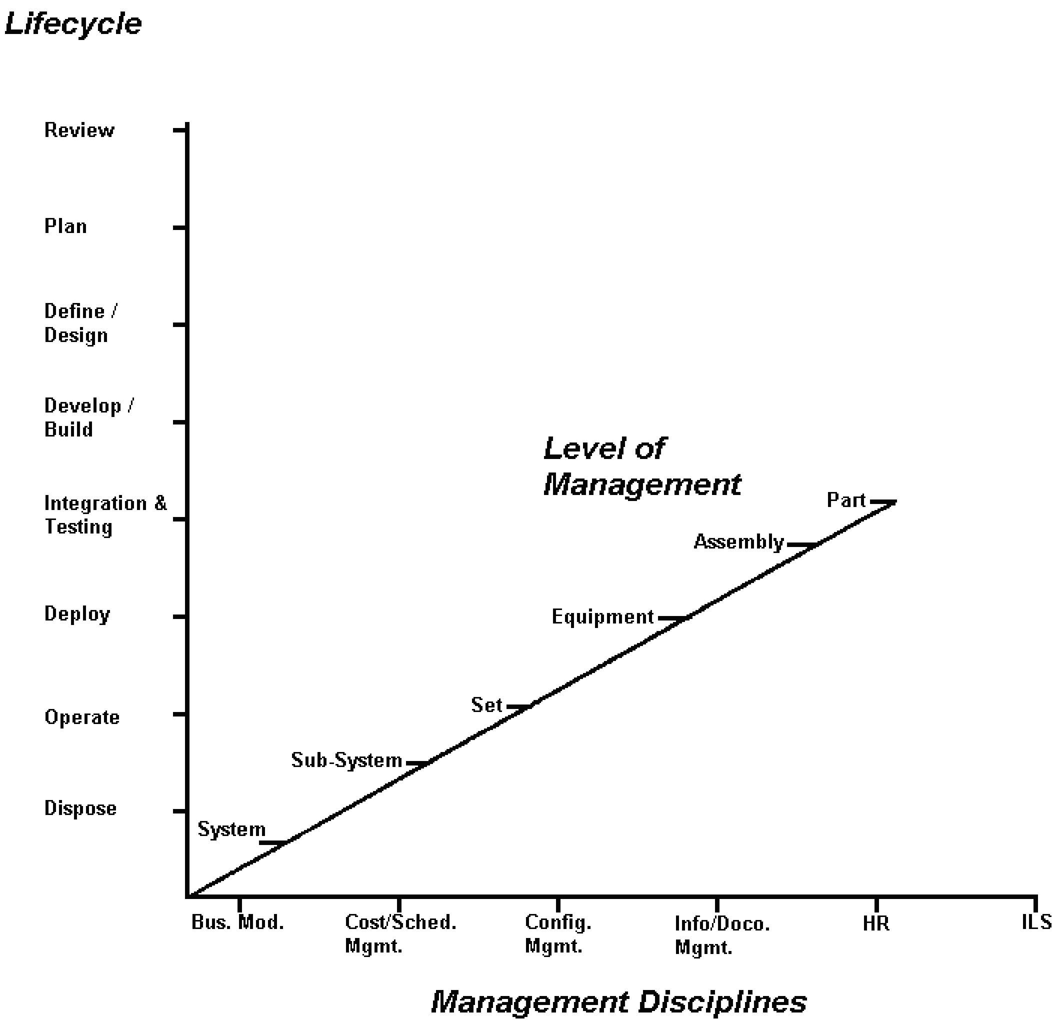

Figure 5. Project model domain

Although the project model domain can be described in such a way, there is no single approach that will result in a successful project – there are numerous. To clarify, there are a large number of project management and systems engineering solutions to the problem of building a satellite – there is no silver bullet. Instead of looking at the perfect project or worse project, in the past the JAESat Project has been focusing on the "most likely to happen" project. The current process has used this as a filter. Ultimately, there will always be a better way.

Systems Engineering

Systems engineering is the selective application of scientific, engineering, and management efforts to define and implement requirements defined by the customer into a system that meets these requirements. Systems engineering also aims to integrate technical parameters and constraints and ensure compatibility while integrating all technical disciplines and specialties into a single body of engineering knowledge. Once again, there are a myriad ways to apply systems engineering to the problem of building a satellite. Originally, the systems engineering approach had been implemented by a variety of people across several teams within the project. With the enormous contribution by a Ukrainian systems engineer, a full systems engineering approach was defined, however it was never endorsed nor adhered to, and work packages were not established based on this plan. The detailed high-level block diagrams of the system fell into the shadows of legend until only recently uncovered during a large-scale project review. The developed systems engineering approach matches very closely the current systems engineering model, described later.

In isolation to this work, the systems engineering model was established around a set of documents that ASRI created templates for, as shown in Figure 6. It merged the scientific and logical approach with a key deliverable at each step. The Project Proposal (PPR) was the management start point with the next step being the delivery of the Operational Concept Description (OCD).

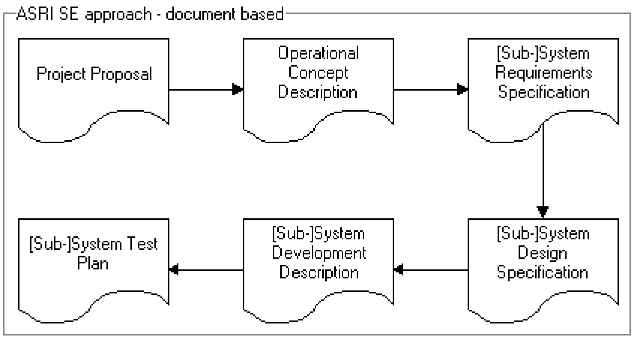

Figure 6. ASRI systems engineering approach

The OCD described the characteristics of the proposed space system from an operational perspective and intended to assist with the definition of overall system goals and long-range operations planning. It also provided guidance for the development of the system requirements, as specified in the [Sub-] System Requirements Specification (SRS). The SRS also defined the methods to be used to ensure that each requirement had been met and the system or sub-system external interfaces requirements.

The SRS was used as the immediate input for the [Sub-] System Design Specification (SDS) and later as an input for the [Sub-] System Test Plan (STP). The SDS specified the design solution requirements for the system or sub-system and the methods to be used to ensure that each requirement had been met. System or sub-system external interfaces related to the design solution were also included. The [Sub-] System Development Description (SDD) used the SDS to produce the final design solution, drawings, manufacturing method, assembly, test requirements, and other pertinent information for the flight production of the space system. Finally, the [Sub-] System Test Plan (STP) was developed from the SDD and the SRS for testing. At the time, there was no operations consideration, although this came later.

When JAESat began to use the first version of the tailored ECSS project system “off-the-shelf”, not only was the ASRI tailoring and infrastructure incomplete, but the project system had been designed such that individual projects would tailor further to map to the requirements of the individual projects and systems. This next iteration of tailoring did not occur for the JAESat Project. Only material lists and ground station procedures were used, with the systems engineering processes useful as background reading.

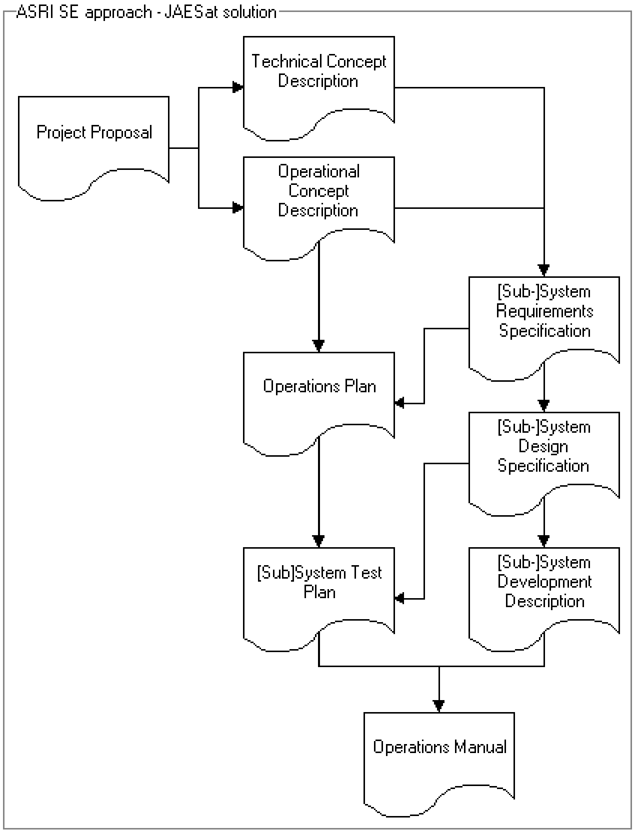

Figure 7. ASRI JAESat SE solution

As the ASRI-endorsed systems engineering approach was not being implemented on the JAESat Project, a new approach was pioneered within the project, as shown in Figure 7. It incorporated key deliverables for the satellite project plus the requirement for students to have assessable deliverables. It also built on the project infrastructure and templates that had already been developed. It was expected that the two key volumes produced would be the Master Level System Specification (MLSS) and the Master Level Design Documentation (MLDD). The MLSS would be composed of all SRS and SDS documents and arranged by level of decomposition. The MLDD would contain all SDD and STP documents. It was also noted that the structure adhered to the ECSS, with integration and control of the systems engineering coming from both project management and systems engineering management. Table 2 shows the link between a process-based and document-based approach, where the deliverable used as output of the previous process phase is used as the basis of the current process.

ECSS SE Process[5]	Deliverable(s) used in process
Integration and control	Project Proposal, Systems Engineering Management Plan (SEMP)
Requirements engineering	Technical Concept Description, Operational Concept Description.

[5] European Cooperation for Space Standardization (1996) ‘ECSS-E-00, Space Engineering – Policy and Principles’, ESTEC, Revision A, 19 April 1996, pp .18 – 21.

ECSS SE Process[5]	Deliverable(s) used in process
Analysis	[Sub-]System Requirements Specification
Design and configuration	[Sub-]System Design Specification, Operations Plan, [Sub-]System Development Description
Verification	[Sub-]System Test Plan

Table 2. Comparison of ECSS and JAESat Project phases

Unfortunately, by this point the project had been running for more than 5 years and core team personnel began to move away from the project intellectually. Documentation, materials and prototypes became lost and there was no configuration management or knowledge management to find the pieces or figure out whether they were of the same JAESat version or what to do with them.

ASRI began a review to determine if the JAESat Project should continue and the recommendation was that it should be terminated or rolled over into another micro-satellite project. In discussions, it was agreed to hand over all project material to QUT for them to complete the project with the bulk of the materials and design. The current SE model for the JAESat Project, as shown in Figure 8, is based almost entirely on the ECSS, but using the templates as developed by ASRI.

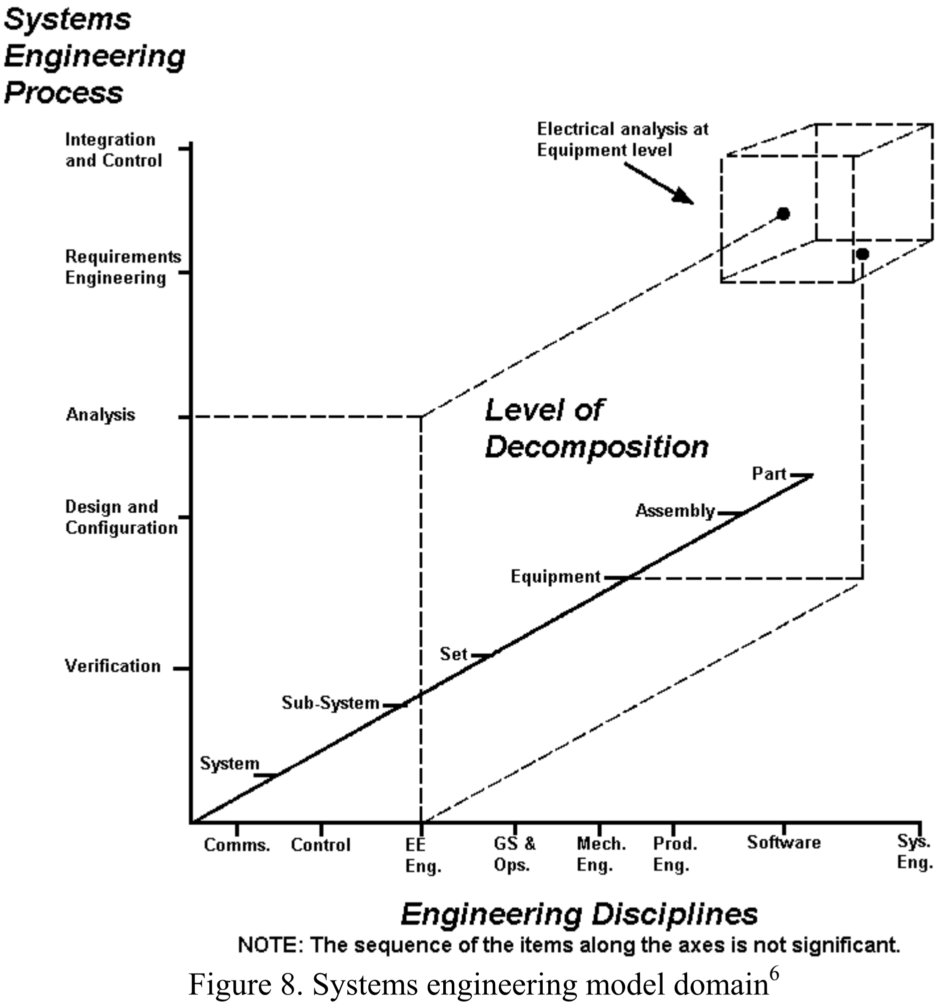

Figure 8. Systems engineering model domain[6]

[6] Adapted from European Cooperation for Space Standardization (1996) 'ECSS-E-00, Space Engineering – Policy and Principles', ESTEC, Revision A, 19 April 1996 p.21

Concurrent Engineering

Concurrent engineering combines the best practices of project management and the systematic approaches of systems engineering. Using the strengths of information technology, such as computer-aided design, electronic commerce, and platform-independent knowledge management systems, the product is designed concurrently with the related business processes. The idea behind this approach is to ensure that the developer considers all aspects of the development lifecycle concurrently, including requirements engineering, manufacture, support, and end-of-life. The key enabling skills for personnel include time management, communication, and suitable training to use both the system and the tools. Concurrent engineering enables the simultaneous coordinated engineering through the use of virtual product models to ensure changes are managed adequately.

Concurrent engineering is not new, having been applied to manufacturing and design for more than 20 years with significant improvements. In an article in Business Week[7], metrics for these significant improvements are provided for the manufacturing and design of a variety of systems. As determined by the National Institute of Standards & Technology, the Thomas Group Inc., and the Institute for Defense Analyses, the following improvements can be attributed to concurrent engineering practices:

- 30% - 70% less development time
- 65% - 90% fewer engineering changes
- 20% - 90% less time to market
- 200% - 600% higher overall quality
- 20% - 110% higher white-collar productivity
- 5% - 50% higher dollar sales
- 20% - 120% higher return on assets[8]

On seeing these figures, the question arises: How is concurrent engineering implemented to provide these figures? There are some fundamental work practices that assist with attaining these benefits. They include working lean, having multi-disciplinary teams, using right-first-time methods, adopting parallel processing activities, and considering all constraints continuously – the traditional tenets of concurrent engineering.[9]

Working lean means that only value-adding development tasks are undertaken, and further, that prioritization occurs to get the “biggest bang for the buck”. Rapid information sharing is initiated when teams of multi-disciplinary personnel are assembled regarding technical issues, solutions, or tools. The use of right-first time methods includes the processes and technologies used to ensure relevant and correct information is available for personnel on request. By beginning sequential tasks just before the completion of a prior task, like a baton relay runner starting to run as the baton approaches, the project duration can be decreased while increasing the benefits gained during information handover. If all constraints are tabled from all teams

---

[7] Port, O., Schiller, Z., King, R. W. (1990), ‘A smarter way to manufacture: How ‘concurrent engineering’ can reinvigorate American industry’, *Business Week*, April 30, 1990, pp 64 – 69.

[8] Port, O., Schiller, Z., King, R. W. (1990), ‘A smarter way to manufacture: How ‘concurrent engineering’ can reinvigorate American industry’, *Business Week*, April 30, 1990, pp 64 – 65.

[9] Evans, S., Lettice, F., Smart, P. (1995), ‘A faster, cheaper and safer route to CE’, World class design to manufacture, Bradford, 1995.

.

within the project lifecycle early in the design process, it is reasonable to assume that there will be less change to the original requirements and more efficient integration between teams.

Examples of aerospace concurrent engineering implementations include the ESA/ESTEC Concurrent Design Facility, the concurrent engineering experiences of NASA with the very successful NEAR spacecraft, and the Aerospace Corporation's Concept Design Center.

The ESA/ESTEC Concurrent Design Facility (CDF) was established in 1998 as a concurrent engineering mission design environment to more effectively manage the organizations resources during early space mission design and to establish a knowledge management system. The reference group is composed of members of all space engineering disciplines and includes customers. Work is undertaken on a central server that provides discipline-specific models, and real-time assessment of trade-offs and options through a browser interface and MS Excel spreadsheets. A room has been established with each team member provided an allocated workspace and computing facilities. [10]

The US National Aeronautics and Space Administration (NASA) project team working on the Near Earth Asteroid Rendezvous (NEAR) spacecraft for NASA used standard systems engineering approaches throughout most of the project, however concurrent engineering practices were used where the sub-system requirements overlapped and where system-level requirements enabled two or more sub-systems attention. The flight software system engineer, the Mission Operations System software lead, the mission operations manager, and technical leads formed a reference group that discussed sub-system status and better communication was established between interacting teams. This provided lessons learned from past missions that were available through the combined multi-disciplinary nature of the teams. [11]

Conceptual design studies are undertaken by the Aerospace Corporation at the Concept Design Center (CDC), usually to evaluate the costs and benefits of new technologies or to teach space systems engineering. In conjunction with NASA's Jet Propulsion Laboratory (JPL), Aerospace developed the satellite system model by linking system and sub-system design spreadsheets, where changes to one sub-system would have immediate impact on the designs of another. More conceptual studies were undertaken in the same time due to the speed and efficiency of this solution.[12]

The reasons why concurrent engineering is being considered for the JAESat Project stem from the figures provided above. Concurrent engineering work practices can decrease development time, decrease engineering changes, increase in productivity, and increase overall quality. The decrease in time to market, and increase of dollar sales and return on investments is not of significant importance for a university or not-for-profit organization and so has not been considered as benefits as applied to JAESat. As the bulk of the project workforce are full-time students, a decreased development time frame is required to keep the team cohesive during the project and to keep the project within a "lust-to-dust" student capability. A decrease in engineering changes would result in less re-work and re-design, and

---

[10] Bandecchi, M. (2002), 'The ESA/ESTEC Concurrent Design Facility (CDF)', *ESA Workshop on Aerospace PDE*, April 2002.

[11] Landshor, J.A., Harvey, R.J., Marshall, M.H. (1994), 'Concurrent Engineering: Spacecraft and Mission Operations System Design', *Proceedings of SpaceOps*, 1994.

[12] Smith, P.L., Dawdy, A.B., Trafton, T.W., Novak, R.G., Presley, S.P. (2001), 'Concurrent Design at Aerospace', *Crosslink – The Aerospace Corporation magazine of advances in Aerospace technology*, Winter, 2001.

a more stable and reliable system. Also due to the student nature of the workforce, the quality of the product is not at an industry standard. With the ability to increase quality through engineering practices, this means that the student quality can be increased to meet or exceed industry quality standards.

With reference to the work practices and the example implementations described above, the lessons learned for the JAESat Project are to:

- Ensure the benefits gained during the concept design study phases continue into the other project phases.
- Establish a reference group, containing a representative of each team, to assist with communication and act as a change control board.
- Model the core system with design spreadsheets on a central server where modification of system data immediately notifies other team members of the changes.
- Interface to central servers via a browser through plug-in applications for the Microsoft suite (Word, Excel, Project, PowerPoint, Access) and PDF. Standard email and face-to-face meeting capabilities are assumed.

Tools

The tools a project selects to use are another important area for the success of any project, but for a concurrent engineering solution they are vital. If the best tools cannot be used by the team, then there is no point in having the best tools. Essentially, the bottleneck for the functionality of the tool is restricted by the personnel. It is always more beneficial to use a tool the team already knows how to use rather than participate in an often lengthy selection, procurement, and training regime for a new tool. This clearly pushes the cost of the project up. In-house training, if done properly, can alleviate this problem, especially where the team is co-located in the same room. However, ultimately, it is assumed that project personnel are familiar with information technology norms.

Information technology norms are usually based around an environment consisting of a web server for hosting of an integrated project web that includes all documentation and templates, email and meetings, workflows, costs and schedules. Project personnel would be expected to work on a client workstation linked to the central repository with standard operating environments (SOE) with spreadsheets, word processing, database, presentation, project management, email and browser facilities. The browser would be used as the primary navigation and interface tool for the project, acting as a launcher for applications as required through the use of plug-ins. If the plug-in is not available, the SOE can be modified to associate file types with an application.

This environment is identified as the concurrent engineering environment, and can be organized into ten categories:

- Client software requirements
- Communication tools
- Document management
- CAD features
- Other document features
- Construction workflow applications
- Time control
- Project information
- Links to other services

- Security[13]

A reasonable first pass at the concurrent engineering environment solution for the JAESat Project has been made in Table 3 by taking the categories, the lessons learned from the concurrent engineering examples, and information technology norms into account. Users should be able to use a web browser to access the project information, with a search facility, if required. Project files, including CAD files, should be viewable and tracked modifications should be able to be made. Additional online tools will be developed, such as the online meeting minutes tool, to assist with project activities.

Concurrent engineering environment categories	JAESat solution
Client software requirements	Based around the use of a web browser to access solution services or features.
Communication tools	Based around the use of email, automated notification to users of changes to the system, and other communication tools, such as real-time chat, meetings, video or teleconferencing.
Document management	Based around a conventional fixed folder tree structure with a full-text and filename search capability. Backup via CD-ROM available. Availability of data and changes should be as close to real-time as possible with either a daily email notification summary. Version management should also be implemented.
CAD features	CAD features must ensure the team can view and make tracked changes with as part of SOE. CAD design tools may be specific for individual domains.
Other document features	Users should be able to view and make tracked changes to common files, such as MS Word, MS Excel, MS Access, MS Project, MS PowerPoint, and Adobe PDF files.
Construction workflow applications	Issues and change orders may be automated such that a change can be managed.
Time control	Each system user requires a modifiable calendar as well as a scheduler tool that enables time management.
Project information	Contact information for the team must be kept current and daily field reports will be required to determine update notifications for users once a day. An offline communication log will need to be established for other communications, such as telephone calls or letters. An online meeting minutes tool will be established to assist the development and increase the speed of completed minutes.
Links to other services	Printing services are often in high demand.
Security	The solution will have variable document access control regulating user views and rights. Information backup, service reliability and scanning for viruses should all be covered by the host.

Table 3. CE environment categories and the JAESat solution

Many of the tools themselves have already been defined either as an industry standard, defined early and now legacy, or as the QUT engineering SOE. The standard JAESat Project

[13] Lakka, A., Sulankivi, K., Luedke, M. (2001), 'Features and Capabilities in the Concurrent Engineering Environment', *ProCE Intermediate Report*, Technical Research Centre of Finland, March 2001.

tools are defined in Table 4. Other tools that may be relevant to investigate include cost model, trade-off, and risk analysis tools.

Tool	Function
Internet Explorer	Internet Explorer provides browser functionality and many plug-ins for associated applications and is usually part of any windows-based standard operating environment.
Email client	Email clients such as Outlook, Netscape, Pine, or Eudora provide email functionality.
Microsoft Project	MS Project is a standard tool for project management and provides functions such as scheduling and tracking tasks, communication of this schedule information via email, notification of status updates, the ability to develop and generate reports, and the management of information on project resources, budget and costs.
Microsoft Office	MS Office is a standard tool for word processing.
Microsoft Excel	MS Office is a standard tool for spreadsheet applications, including financial and design planning.
Microsoft PowerPoint	MS PowerPoint is a standard tool for developing and presenting presentations.
Concurrent Versioning System (CVS)	Concurrent Versioning System is a tool for managing configuration of documents and files.
ArgoUML or ProxyDesigner	These tools are freeware Unified Modeling Language (UML) viewers.
Protel	This tool assists in the design of circuit boards.
AutoCAD	This tool assists in the design of the structure.
DreamWeaver	This tool assists in the design and maintenance of the project web.

Table 4. JAESat CE environment tools

These tools could be placed on a personal computer for use when required, however, this will not help the personnel perform their tasks, and does not keep strict control over the tasking during the project. All these tools can be integrated through a "project web". Examples of project web can be found in Rational Unified Process and can be developed with Lotus Notes. Rational Unified Process uses an intranet-like website, with incorporated workflow and templates, and assists team training through a web-based project infrastructure with access to standard tools. Lotus Notes is usually thought of as simply an email client, however, it can also be used for substantially more, including team rooms, and web-based process and knowledge management.

Another example, the ASRI Virtual Project Office (VPO), provides an online environment that supplies structure, workflow, process management, and templates, including specified tools for development activities and project calendars and schedules. It is developed as an HTML environment using server-side scripts to provide automation. Based on free-ware or open-source tools, the VPO makes use of a recent tool called Jabber. Jabber is an open XML protocol for electronic messaging that is written in XML and Perl. It provides a facility similar to email, but also includes messaging and meeting tools. The meeting tools have been developed such that virtual meetings are held online with minutes automatically generated from the text the attendees type, and action points are automatically followed up with nagging emails.

The concurrent engineering environment for the JAESat Project is being established based on the ASRI VPO, but will extend the solutions autonomous capability. It will be run from a web server with a scripting area for the development of CGI scripts, and ultimately incorporate automated processes and a component database for quick high-level assessment of design solutions.

Conclusion

The intended approach for the JAESat Project is through the adoption of systems engineering and project management models that incorporate the correct level of resolution required and ensuring these models are available to, and understandable by, the project personnel. The evolution of the project management and systems engineering models have progressed such that reviews and planning is progressing. Once final sign-off on the project handover has occurred, the project will move into requirements engineering in parallel with conceptual design. These phases are not expected to take long as the project has been progressing for some time. Although they are considered “rubber-stamp exercises”, they are being performed thoroughly and logically to ensure traceability and verifiability.

The JAESat Project has progressed through three distinct stages of systems engineering maturity, and is currently in the fourth. The first was where no systems engineering approach was endorsed across the entire project. The second was the document-based approach, with the third stage extending this to include some project-wide definition of systems engineering techniques. These stages can be marked by lessons learnt.

During the first stage, the project was affected by the lack of a traditional customer or client and the difficulties integrating sub-systems in the absence of a common systems engineering approach. Not having the clear distinction between customer and project team has meant that many “customer” requirements were developed in the heads of team members and never made it to paper. It was often difficult to write these issues down when the same person documenting them was the recipient of the document. This has accounted for the bulk of the scope-creep during the project. Over time and with attrition of personnel, some of these customer requirements were lost and some design concepts have remained to this day.

The second stage saw the development of a document-based approach, using a variety of available standards, including MIL-STD, IEEE, ISO, and ECSS. By linking the documents, a default lifecycle approach was established, but there was no configuration management. The deliverables had not been incorporated into student assessments and both the students and the volunteers then working on the project found they were swamped with requests for documents they could not deliver. The second stage also found the project struggling to communicate with distributed teams across Australian educational institutions and with international teams. A central repository was established to house the documents and it was accessed via the internet. Although this resolved some communication issues, those relating to lack of adequate documentation were still crippling. The implementation of an immature tailored ECSS project system meant that templates were being developed concurrently to development activities. Often documentation and analysis was postponed for release to the internet because the templates were not ready. Reverse requirements engineering was undertaken to define the satellite system requirements based on the ECSS.

The third stage extended the development of the document-based ASRI systems engineering approach and project system to align with the ECSS. This period, spanning more than two years, ended when QUT re-acquired project management control. In this stage, many templates were completed, but few were used "in anger". The document-based systems engineering solution was aligned with the ECSS systems engineering process such that the documents were integrated as deliverables for phase kill-points. The project had lost most of the long-term personnel and even high-level documentation was difficult to complete and formally review. Although the documentation set and approach where very useful, the project had progressed for too long without progress and had become choked with documentation. Even new or active teams were dragged into the culture of apathy because of the requirement for key sub-system data that was not being provided. Those documents that were produced where using previous documentation that had not been configuration managed, so solutions that were proposed could not be integrated, or were radically different to the current operational concept.

In summary, the systems engineering approach using concurrent engineering techniques is a completely new approach for the JAESat Project, and uses the previous approaches as inputs. The following lessons have been learned regarding the systems engineering approach for the JAESat Project:

- A common systems engineering approach is being established with set phases and templates available before they are required for use. The approach is based on the ECSS and key deliverables are assessable.
- To ensure information is available on request, a concurrent engineering environment is being established using the internet as the base operating environment.
- To ensure that requirements are defined early in the process, including customer or client requirements, requirements engineering is to be complete by October.
- The central repository requires configuration management to be implemented so integration problems are minimised. This is currently underway.
- It is preferable to co-locate the entire team in the same city, and at regular meetings, so technical communication can be progressed in a multi-disciplinary forum.
- The project must be kept within the two year "lust-to-dust" timeframe, allowing for defined and structured phasing.

The current technical status of the JAESat Project is on-hold pending the signing of the agreement between QUT and ASRI for the negotiation of roles and responsibilities of the joint partners, but this is expected to be completed at the time of this paper. The first draft of the JAESat VPO has been published and is under review by team personnel, including templates, workflow for the first two phases, and work packages. The legacy documentation has been stored on the server for team access, and configuration management for high-level documents are being put in place to roll-out the new documentation set. Once the agreement has been signed, material and components will be sent to QUT for testing, re-design and further development, as required.

The project is expected to start preliminary definition according to the new approach, and based on the project review, with the view to defining a set of requirements by October. In 2004, integrated student teams will complete most of the development and manufacturing. The satellite should be ready for transportation, after final manufacturing and several months of integration and testing, by August 2005. By adopting concurrent engineering practices, specifically with respect to the concurrent engineering environment, the project will make use of the multi-disciplinary teams already established and the high level of information technology capability. The systems engineering and project management models are based on a tailored subset of ECSS standards implemented as an intranet project web that integrates required project information and tools.

References

Bandecchi, M., Melton, B., Ongaro, F. (1999) 'Concurrent Engineering Applied to Space Mission Assessment and Design', *ESA Bulletin 99*, September 1999.
Bandecchi, M. (2002), 'The ESA/ESTEC Concurrent Design Facility (CDF)', *ESA Workshop on Aerospace PDE*, April 2002.
Boyd, C. (2001), 'ALUMINATE Project Update and the Virtual Project Office', *Proceedings of the 11th Australian Space Research Institute Conference*, Sydney, December 2001.
European Cooperation for Space Standardization (1996) 'ECSS-M-30, Space Project Management – Project Phasing and Planning', ESTEC, Revision A, 19 April 1996.
European Cooperation for Space Standardization (1996) 'ECSS-E-00, Space Engineering – Policy and Principles', ESTEC, Revision A, 19 April 1996.
Evans, S., Lettice, F., Smart, P. (1995), 'A faster, cheaper and safer route to CE', *World class design to manufacture*, Bradford, 1995.
Standards Australia (1999) 'AS/NZS 4360:1999 – Risk Management', 1999.
King, J. A., McGwier, R., Price, H., White, J. (1990), 'The In-Orbit Performance of Four Microsat Spacecraft', *Proceedings of the 4th Annual AIAA/USU Conference on Small Satellites*, 1990.
Lakka, A., Sulankivi, K., Luedke, M. (2001), 'Features and Capabilities in the Concurrent Engineering Environment', *ProCE Intermediate Report*, Technical Research Centre of Finland, March 2001.
Landshor, J.A., Harvey, R.J., Marshall, M.H. (1994), 'Concurrent Engineering: Spacecraft and Mission Operations System Design', *Proceedings of SpaceOps*, 1994.
Port, O., Schiller, Z., King, R. W. (1990), 'A smarter way to manufacture: How 'concurrent engineering' can reinvigorate American industry', *Business Week*, April 30, 1990, pp 64 – 69.
Smith, P.L., Dawdy, A.B., Trafton, T.W., Novak, R.G., Presley, S.P. (2001), 'Concurrent Design at Aerospace', *Crosslink – The Aerospace Corporation magazine of advances in Aerospace technology*, Winter, 2001.

Author bios

Mr. Cameron Boyd (ASRI)
Mr. Cameron Boyd holds a Bachelor of Arts with a double major in Cognitive Science and is currently completing a research masters on the implications of the Missile Technology Control Regime (MTCR) on Australian space development (due for completion December 2003). He has also been trained in quality assurance techniques, including Rational Unified Process, Capability Maturity Model and ISO9000/1. He has demonstrated the use of these methodologies in the development and maintenance of enterprise resource planning and broadband telecommunication software as technical writer and trainer, technical consultant, and software quality assurance officer. Mr. Boyd is currently employed with Aerospace Concepts Pty Ltd as a consultant, providing project management and systems engineering services.

A/Prof. Werner Enderle (QUT)

Werner Enderle received his Ph.D. in 1998 from the Technical University of Berlin. He joined the German Aerospace Center (DLR) in 1992 and since 1994 he worked for the German Space Operations Center (GSOC) in the space flight dynamics division. He was specialized in spacecraft orbit and attitude determination based on GPS. Since 1998 until 2000 he was responsible for the DLR space born GPS receiver development. Between 2000 and 2001 he was joining the Galileo Support Team (GAST), which was supporting the European Commission (EC) in the context of the design studies for the European Global Navigation Satellite System (Galileo). Since 2001 he is an Associate Professor for Aerospace Avionics at the Queensland University of Technology in Brisbane, Australia.

Mr. Shaun Wilson (ASRI)
Shaun Wilson is a professional engineer with qualifications and experience in aerospace engineering and in information systems design and management. He is a graduate of the Australian Defence Force Academy in Canberra, the Royal Air Force College at Cranwell in the United Kingdom and three civilian universities. Having worked as a consultant for a number of years, Shaun is now principal of Canberra-based aerospace and systems engineering house, Aerospace Concepts Pty Ltd. His voluntary work for ASRI has included time as a director and as manager and team member of ASRI satellite and rocket projects.