

Paper 254-26

# Cycle Down to Launch: Streamlining Production at Lockheed Martin

Dave Jennings, Lockheed Martin  
Radhika Kulkarni, SAS Institute Inc., Cary, NC

## Abstract

Lockheed Martin's Astronautics Division uses the SAS® System's scheduling and project management capabilities as the foundation for several Web-based solutions that monitor the performance of their missile production systems. These solutions have proven invaluable in helping Lockheed Martin personnel keep tabs on the work completed and work remaining for these launch vehicles.

Recently, Lockheed Martin Astronautics has undertaken an ambitious program to dramatically reduce its cycle time, the elapsed time between starting construction of a missile and delivering it for launch. Again, the SAS System plays a central role by providing flexible scheduling methods that make it easy to test the effects of changes in the product structure and details. We present a brief overview of these solutions and describe some of the methods used in the Cycle Time Reduction project.

## Introduction

The Space Launch Systems group of Lockheed Martin's Astronautics Division in Denver is responsible for the design, fabrication, and delivery of the Titan and Atlas missiles. Each of these programs has its own managers, contractors, and customers, requiring the coordination of several areas such as engineering, procurement, manufacturing, testing, etc. The program managers need an effective way to monitor performance of the production systems used in building these missiles. The Decision Support Systems group has built several Web-based applications to allow users access to production data and to enable managers to make effective decisions regarding the status of the project.

This paper describes some of these applications.

First, we provide some background information about the projects along with the questions that are typically asked by the decision makers. Next, we briefly describe some of the data used by the applications and the underlying architecture. Some of the key components of these decision support systems are: a data warehouse containing all the relevant production data, the scheduling procedures in SAS/OR® software, and SAS/IntrNet® software, which provides the technology to create Web-based applications that can be deployed to all users.

A thumbnail sketch of several of the applications is followed by a more detailed description of one particular application used in the Cycle Time Reduction project. This example illustrates how the framework of the SAS environment allows rapid development of Web-based applications, enabling multiple users to easily access analytical tools via a simple Web browser.

## Background

The Astronautics Division in Denver makes a small number of launch vehicles at a time. However, each launch vehicle is expensive to make, consists of about 50,000 parts, and has a long, complex fabrication cycle. During the production cycle of each missile, the program managers need to track the current progress and be able to react to any glitch in the process. They need ready answers to questions such as:

- Where is each missile in the build cycle?
- When will each missile be completed?
- How will any changes affect the schedule?
- What can be done to improve the schedule?

The Space Launch Decision Support Systems Group at Denver is responsible for making the information from the operational systems available to middle- and upper-level decision makers. This information may

come from several locations and systems. The goal is to streamline the data collection process, transform the operational data for consistency, perform the required analysis to answer the earlier questions, and make the data, as well as the results of the analysis, readily available. The DSS Group built systems to meet the division's needs by publishing reports to the Web (Rerecich 1998).

Several of these applications use the scheduling methods in SAS/OR to solve the business problems. These methods use production data from a variety of legacy systems that are gathered into a data warehouse, described briefly in the next section.

### Data Warehouse

Monitoring schedule status is the main focus of all the production-operations systems. The principal challenges lie in accessing and transforming data so that they are consistent. The main source of the production data is a legacy MRP system that contains Bill-of Material (BOM) data that describe the entire build process. These data include information about all the tasks that need to be performed from the time an order is received for a launch vehicle until it is launched. The section on the Cycle Time Reduction project contains more information about the BOM data. Other diverse data sources contain supporting information that is also used in various reports and analyses.

#### Decision Support System Data Integration Strategy

##### Top Level Data Flow

Utilizing existing data and structures to provide analysis, overview and simulations, without disturbing the transaction systems or "re-inventing" data.

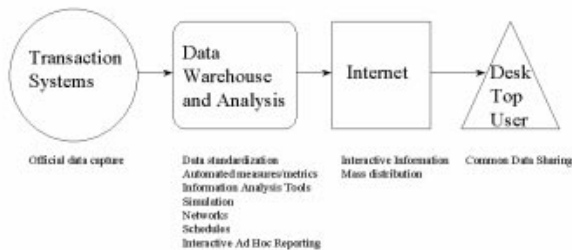


Figure 1. Data Flow

Information from all these diverse sources is combined to feed into a Data Warehouse. Batch jobs that run over the weekend refresh this data warehouse, from which further processing creates specific Data Marts that are used in the Web-based applications. Figure 1 illustrates the main ideas behind the data integration strategy in use at Lockheed Martin Astronautics. Having a common Data Warehouse ensures that all users have access to the same information

and that they are on the same page as far as the current status of the project is concerned. The next section describes the basic framework used in these applications.

### Architecture

All the Web-based applications use a common architecture (see Figure 2) described in the following steps:

- You access a Web page where you can choose from various selection criteria. These could be some simple options to control the output format, or they could be listboxes that allow you to select subsets of the data for analysis.
- Once you make all the selections, you submit a query that dispatches a request to the application server that runs the SAS programs. Sometimes, the SAS programs might access another SAS/SHARE® server that contains the data.
- The results of the analysis are streamed back to your Web browser. These results could be graphics or tabular output using the Output Delivery System (ODS). Several graphics procedures in SAS allow you to click on specific areas of the graph to drill down to successive Web pages. These Web pages can submit further SAS programs that are context sensitive (based on the selected ares) and display the next level of results.

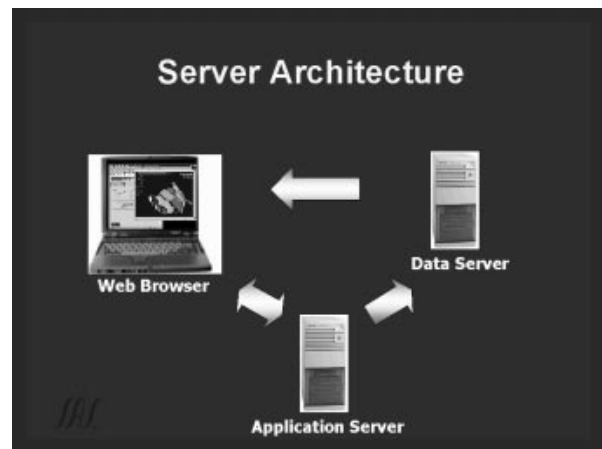


Figure 2. Server Architecture

### Sample Applications

The applications at Lockheed Martin are tailored to different groups having specific needs. For example, users are interested in the schedules, assembly structures, work remaining, load across factories, and resources available to support the load. In this section,

short sketches of some of the products currently in use are presented.

**Control-Point Flow**

This product uses product structure from the MRP system to generate high-level network diagrams. The original product was used to assist in researching questions in the Cycle Time Reduction project. Production engineers maintain the product structure in the MRP system by data entry screens. Periodically, mistakes are made resulting in broken structure. The Control Point Flow product enables the users to immediately identify breaks in the structure.

**Past-Due Reporting**

This product looks at current MRP orders and flags the ones that are past due with regard to some specific order status. This product then uses the CPM procedure to search for any “out of bed orders.” An “out of bed order” is one where the scheduled dates are not supported by lead times; it typically occurs when some manual intervention has been made to the MRP system (for example, a part planner manually changed a due date.) Planners use this report on a weekly basis to identify and work on any past due orders. This product is also used in generating charts for monthly business management reviews.

**MRPII Metrics**

Several MRP metrics are calculated by using the CPM procedure to simulate the MRP system to identify whether a work order passed or failed certain metric criteria. For example, the metric might require that an order should be completed no sooner than 10 days before its due date and no later than 3 days after that date. The CPM procedure enables you to calculate the threshold dates and determine if the order passed the criteria. SAS/GRAPH® software is used to create GIF files for displaying the metrics. Senior management uses these metric Web pages (updated weekly) to monitor manufacturing performance.

**Work-Remaining Reporting**

The work-remaining charts were originally developed for the Titan program and have been slightly modified for the Atlas program. Users access the system via a URL and can choose to monitor a project at multiple levels of detail and see the overall schedule of work remaining for each vehicle (see Figure 3). Using the Output Delivery System, each of the bars in the chart can be a link to another Web page that shows the detail on that particular vehicle. In turn, the resulting Gantt chart’s bars are links to other pages that provide further detail on the selected subassembly, and so on. In this way, users can drill down to information at any desired level of detail.

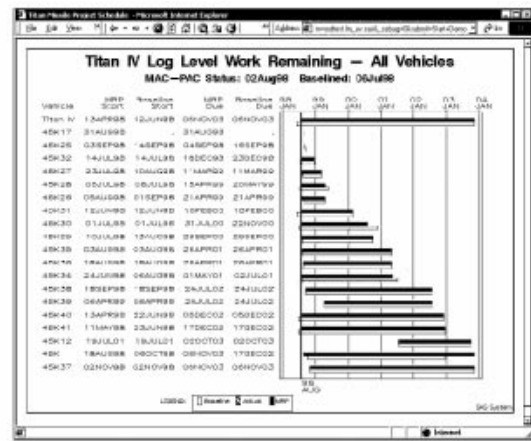


Figure 3. Production Schedule for the Titan IV Missiles

**Cycle-Time Reduction**

The Cycle Time Reduction product was developed to help reduce the build cycle of an Atlas launch vehicle. This product (and the software that supports it) is described in greater detail in the rest of the paper.

**Cycle Time Analysis**

The next few sections describe Cycle Time Analysis (CTA), software for calculating and displaying the critical path and all subsequent paths in project networks. The Critical Path Method is a standard way to determine the longest path (critical path) in an activity network. The length of the critical path determines the duration of the project. If the project network under consideration is the set of activities involved in the production of a given product, the project duration is referred to as the cycle time for the product. One of the main goals of manufacturing is to reduce this cycle time. In any cycle time reduction effort, the first target is to reduce the length of the critical path by focusing on the tasks that comprise this path. Quite often, this will lead to another path in the project cycle becoming the critical path. Thus, it is not enough to consider the critical path alone; all paths close to the critical path in length should also be considered in the effort to reduce the build cycle.

CTA helps in the Cycle Time Reduction effort by ordering all the paths in the build network and enabling you to display all the tasks in the *n*th longest path. The following sections define the business problem from the perspective of Lockheed Martin Astronautics, describe how you can use the CPM procedure to calculate not just the longest path, but any of the desired paths in the project network, and finally, describe a simple Web-based application that uses CTA.

## Business Problem

The Cycle Time Reduction product at Lockheed Martin Astronautics was developed to help reduce the build cycle of an Atlas launch vehicle. When the cycle-time reduction project was first begun, the product cycle time was approximately 48.5 months. This was the time span from the placement of the first order to launch. The goal was to reduce this cycle time to 24 months within one year and then to 10 months by the end of the third year. In order to achieve such a reduction it was necessary to look at all activities in the production process that contribute to any path that is longer than 24 months (Miller 1999).

## Software Solution in use at LMA

The Cycle Time Analysis (CTA) software is a set of macros used to solve the business problem described above. As part of the Cycle Time Reduction effort, project teams were established to reduce any paths in the build cycle that were deemed by management as too long. Weekly, these teams use a "View All Paths" Web page to order all the paths in the build cycle in decreasing order of path length. They can then use the Web page to generate a Gantt chart detailing the information on any selected path.

The input into the application is product structure data, namely the bill of materials (BOM). The BOM data for the Atlas Launch vehicles obtained from a legacy MRP System are used to maintain a SAS data warehouse containing all the production data for the missile. These data describe the activity network for building an Atlas missile and are loaded into SAS data sets on a weekly basis.

The analysis answers the following question: What is the  $n$ th longest path in the schedule? The activities on this path are the candidates for reducing the missile build time. A simple and fast algorithm that identifies the path has been built using off-the-shelf components, including the CPM procedure, in the SAS language. This algorithm is packaged as a SAS language macro, which can then be driven from a Web interface using the CGI approach and a SAS application server.

The following sections describe the details of the software solution for this problem. First, a brief description of the traditional approach to the problem is presented. Next, this problem is formulated as an application of the CPM procedure (SAS Institute Inc., 1999). A brief summary of the CPM procedure is presented highlighting the options that are used in the solution. Then, a simple example is used to illustrate the use of the CPM procedure to determine the  $n$ th longest path in the project network. Finally, a brief outline of the Web-based application is described.

## Description of Solution

In a production environment the Bill of Materials describes the product structure and identifies in great detail all the components that go into each part of the finished product. The BOM tree for the Atlas Launch Vehicle contains the Product Launch as the final end product with all the parts as nodes in a tree. The final end product is called the root node of the tree. A node in the tree without any children is referred to as a leaf. You can determine the longest path in the tree by using any standard tree traversal routine that calculates the length of the path from any node in the tree to the root node. Once all the path lengths have been calculated, order the leaf nodes in increasing order of the path lengths. To determine the  $n$ th longest path, pick the  $n$ th node in this ordered list of leaf nodes; the path from this node to the root gives you the  $n$ th longest path.

The above problem can also be solved using the CPM procedure, or any simple algorithm that calculates the critical path in an activity network. The activities in a project are typically represented as an activity network. The Critical Path Method determines the longest path in the network. The length of this longest path is also the project duration. It is well known that the determination of the critical path is a polynomial algorithm proportional to the number of arcs in the network. In the current scenario, the problem is to determine the  $n$ th longest path in the network. The solution presented in this paper reduces the problem of finding the  $n$ th longest path to the problem of finding the critical path in a suitably modified project network.

The CPM procedure calculates the early and late start schedule for all the activities in a project. In addition to precedence constraints between the activities, you can specify alignment dates with specific alignment types (*SGE* for "Start Greater than or Equal to" or *FLE* for "Finish Less than or Equal to"). In the presence of alignment constraints, the critical path of the network may change. The CPM procedure determines the earliest start time and the latest finish time for all the activities in the network subject to both the precedence constraints and the alignment constraints. It is this feature of the CPM procedure that is used to determine the activities in the  $n$ th longest path in an activity network.

The data for the project network come from the Bill of Materials that describes the entire product structure. At each level of the product structure tree, all the components required to finish the parent part can be thought of as the predecessor activities to the parent activity. The lead-time for each component is the duration of the corresponding activity. Using this correspondence between a part in a BOM structure and

an activity in a project network, you can use the CPM procedure to determine the schedule for building the final product in the Bill of Material structure. In addition, you can determine the  $n$ th longest path in the build cycle using the simple sequence of steps described below.

- Step 1. The CPM procedure is invoked with a project start date.
- Step 2. The output data set is sorted by increasing late start time. This operation is equivalent to sorting the leaf nodes (or the project start activities) in the correct order; i.e., in decreasing order of path lengths.
- Step 3. Set an alignment constraint ( $SGE$  at late start time) on the  $n$ th leaf node in this ordered list. Invoke the CPM procedure with this modified project network (the original network augmented with the additional alignment constraint) and an earlier project start date.
- Step 4. The critical path obtained by the second invocation corresponds to the  $n$ th longest path in the original network. In other words, the critical path corresponding to the modified project network is the  $n$ th longest path in the original network. Extract the nodes on the desired path by selecting the nodes in the output data set with  $T\_FLOAT=0$ .

Note that steps 1 and 2 are independent of  $n$ , the index of the desired path. The data set produced in step 2 is used as the input data set for the CTA application. This data set can be determined *once* for a given network and used as input for Step 3 for any  $n$ .

### Example

A simple product structure illustrates the method described in the previous section.

Consider the Bill of Materials structure shown in Figure 4. There are 6 leaf items in this tree corresponding to 6 paths leading to the Final Product. Each node is labeled with the name and the lead-time for the part. The specified lead times are used as the durations of the activities to determine the production schedule for the Final Product. The CPM procedure is invoked with a start date of December 4, 2000. The resulting schedule is displayed in Figure 5. Note that the critical path goes from the leaf node Item H to the Final Product. A better picture of the path lengths through the tree can be obtained by drawing the schedule using a reversed tree structure with all the nodes in the tree positioned according to their late start times (see Figure 6).

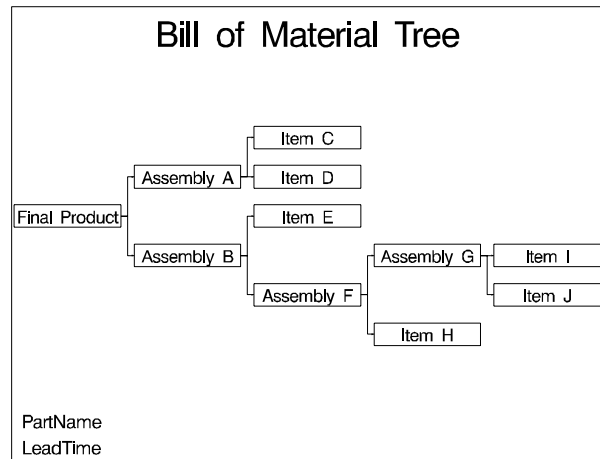


Figure 4. Product Structure

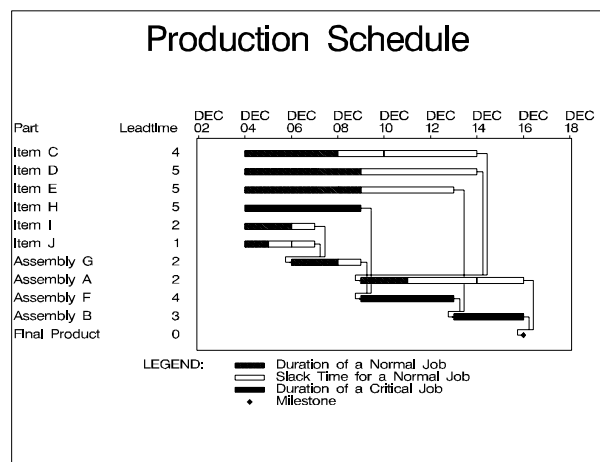


Figure 5. Production Schedule

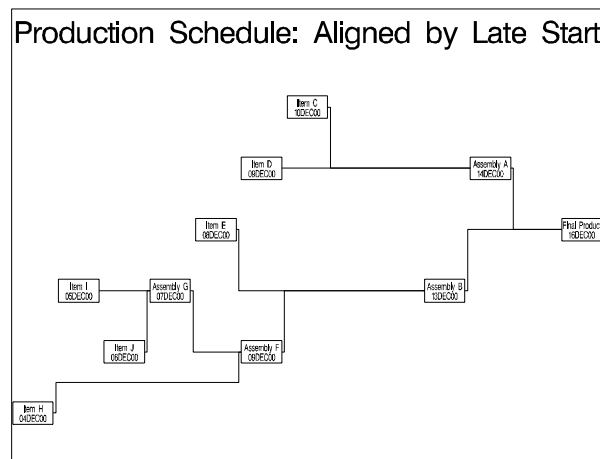


Figure 6. Network Diagram of Production Schedule

For example, the second longest path is from *Item I* -> *Assembly G* -> *Assembly F* -> *Assembly B* -> *Final Product*. The problem under consideration is how to

identify the nodes in this path using the CPM procedure.

The first invocation of the CPM procedure (with a project start date of December 4, 2000) produces an output data set, shown in Figure 7. The observations are sorted in increasing order of the Late Start times. A variable, Leaf, indicates if the node in the current observation is a leaf node. Also, the leaf nodes are associated with a path number variable, PathNumber. Since the network is hierarchical, there is only one path from any given node to the root of the tree. Thus, there is a one-to-one correspondence between the  $n$ th leaf node in this table and the  $n$ th longest path in the activity network. This is the input data set used by the macro lpath( $i$ ) to calculate the  $i$ th path.

Schedule Sorted by Path Number				
Part	Leaf	Path Number	L_START	L_FINISH
Item H	1	1	04DEC00	08DEC00
Item I	1	2	05DEC00	06DEC00
Item J	1	3	06DEC00	06DEC00
Assembly G	0	.	07DEC00	08DEC00
Item E	1	4	08DEC00	12DEC00
Item D	1	5	09DEC00	13DEC00
Assembly F	0	.	09DEC00	12DEC00
Item C	1	6	10DEC00	13DEC00
Assembly B	0	.	13DEC00	15DEC00
Assembly A	0	.	14DEC00	15DEC00
Final Product	0	.	16DEC00	16DEC00

Figure 7. Schedule Sorted by Path Number

To determine the  $n$ th longest path in the network, add a time constraint to the project network, requiring the  $n$ th leaf node to start on or after its previously determined Late Start time. For example, to determine the second longest path in the network, set an alignment constraint of type SGE on the node, Item I, with the alignment date of 05DEC00. Next, invoke the CPM procedure with a project start date specified to be any date earlier than the previous start date. For instance, let the start date be December 1, 2000. Only the nodes that are on the second longest past will be critical in this invocation; all other nodes will have a non-zero float. To select the desired path, select only the nodes with T\_FLOAT=0. The project schedule corresponding to such an invocation of the CPM procedure is displayed in Figure 8. Note that only the nodes on the second longest path (of the unconstrained network) are critical (corresponding to T\_FLOAT=0). In other words, the desired nodes are Item I -> Assembly G -> Assembly F -> Assembly B -> Final Product.

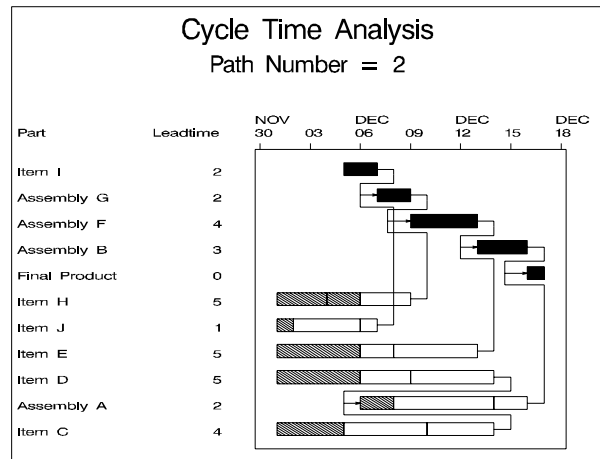


Figure 8. Determination of Second Longest Path

### The Lockheed Martin Application

Lockheed Martin Astronautics uses the method described in the previous sections to determine all the activities corresponding to any selected path number. The users have access to the project network via a Web page that allows them to select any number between 1 to the maximum number of paths in the tree. Once a path number is selected, a query is dynamically submitted to the SAS System that runs the macro, lpath( $i$ ). The result is displayed on the page in the form of a table or a Gantt chart. A sample page from the Lockheed Martin application is displayed in Figure 9.

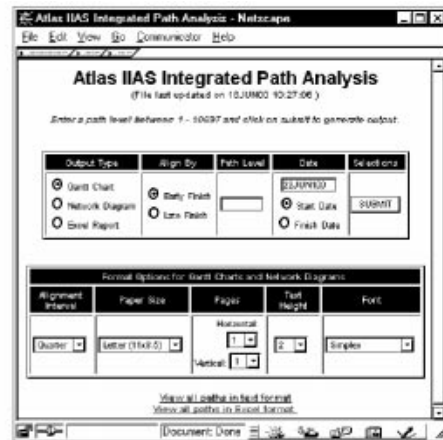


Figure 9. The Atlas II Cycle-Time Reduction Application

### The CTA Application

This section describes the Cycle Time Analysis Application, a slightly modified version of the Cycle Time Reduction product described in the previous section. The CTA application was built at SAS Institute as a prototype to illustrate the architecture described in Figure 2. This Web-based solution shows how you can build a decision support tool using SAS components. This prototype uses the Bill of Material data for the Atlas Launch Vehicle. The input to this application is the schedule data set, a157schd, which contains the schedule for the Atlas launch vehicle, A157. In addition to the early and late start schedules, as computed by the CPM procedure, this data set also contains two variables: leaf, indicating whether a node is a leaf node, and PathNumber, specifying the index number of the path length from the node to the final product in the BOM tree.

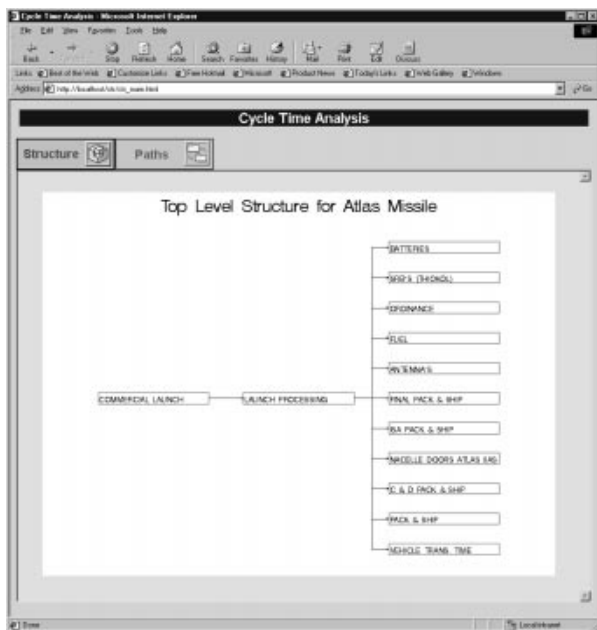


Figure 10. Cycle Time Analysis Application

The initial view of the CTA application is shown in Figure 10. The top-level structure of the BOM tree is displayed; double-clicking on any node in the display will display the next level of the tree corresponding to that node. Note that the diagrams are generated dynamically whenever any node is selected.

The Paths tab leads to a tabular display of all the leaf nodes in the BOM tree, sorted according to its corresponding path number (see Figure 11.) Selecting a given path number in this display will lead to a Gantt chart of the activities on the selected path. For example, selecting path number 2 will display the activities

on the second longest path (see Figure 12.) Note that the schedule corresponds to the early and start times in the input data set a157schd.



Figure 11. Sorted List of Paths

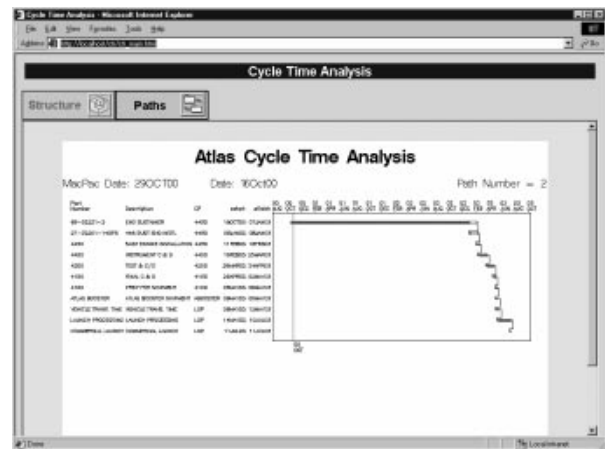


Figure 12. Second Longest Path

### Conclusions

This paper describes some of the Web-based SAS applications in use at Lockheed Martin Astronautics. These applications help monitor the production of the Atlas and Titan launch vehicles. A brief overview of the underlying architecture used in the solutions is followed by a closer look at one specific application. The Cycle Time Analysis application illustrates how you can use the Web to access the power of the analytical tools in the SAS System.

## Acknowledgments

We are grateful to Virginia Clark and Ed Hughes for valuable assistance in the preparation of this paper and to Yi Liao who helped develop the CTA Application at SAS Institute.

## References

Miller, Sandra K. (1999), "Cycle Down to Launch," 1999 APICS International Conference Proceedings, 1999.

Rerecich, Art (1998), "W-O-W: Data Warehouse, SAS/OR and the Web," *Proceedings of the Twenty-Third Annual SAS Users Group International Conference*, Cary, NC: SAS Institute Inc., 567-573.

SAS Institute Inc. (1999), *SAS/OR User's Guide: Project Management, Version 8*, Cary, NC: SAS Institute Inc.

SAS software, SAS/OR, SAS/IntrNet, and SAS/GRAPH are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.

## Contact Information

Radhika V. Kulkarni, SAS Institute Inc., SAS Campus Drive, Cary, NC 27513. Phone (919) 531-7692, FAX (919) 677-4444, Email Radhika.Kulkarni@sas.com.

Dave S. Jennings, Lockheed Martin, Mail Stop 5507, P.O. Box 179, Denver, CO 80201. Phone (303) 977-9588, Email Dave.S.Jennings@lmco.com.