1415 N CHERRY AVE
CHICAGO, IL 60642
(312) 281-6900
DMDII.ORG
DMDII@UILABS.ORG

DMDII
A UI LABS Collaboration
DIGITIZING AMERICAN MANUFACTURING

DMDII PROJECT CLOSEOUT REPORT

Integration of AVM iFAB Tools for Industrial Use	
Project Team Lead	Penn State Applied Research Laboratory
Principal Investigator	Christopher Ligetti, cxl300@arl.psu.edu
Project Designation	14-01-09
UI LABS Contract Number	0220150008
Project Participants	Penn State Applied Research Laboratory, Oshkosh Corporation, aPriori, PTC
DMDII Funding Value	$1,268,857
Project Team Cost Share	$1,278,579
Award Date	December 17, 2015
Completion Date	August 17, 2017
Award Date	December 17, 2015

This project was completed under the Cooperative Agreement W31P4Q-14-2-0001, between U.S. Army - Army Contracting Command - Redstone and UI LABS on behalf of the Digital Manufacturing and Design Innovation Institute. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Department of the Army.

TABLE OF CONTENTS

FIGURES

TABLES

I. EXECUTIVE SUMMARY

When developing a new product, it is critical to understand its manufacturability and cost. Having the ability to obtain this information early in the development process makes the process much more efficient. In nearly all products developed today, cost is a major customer requirement affecting the overall design. In order to effectively study tradeoffs between different technologies or features, a designer must know the cost of the component, subsystem, or full system that is being studied. Today's computer aided engineering (CAE) tools lack the ability to efficiently take a 3D design and roll up the cost of a complex system or subsystem or provide manufacturing feedback during the early stages of product development. Having manufacturability and cost feedback during design will lead to a more efficient and effective design process and realization of products faster.

In traditional design practices with current CAE tools, a designer has the ability to take a 3D design into separate cost analysis software for detailed cost analysis on that component. Furthermore, an organization's specific manufacturing cost/constraints can be integrated into the costing CAE tool such that the cost analysis output is reflective of the organizations manufacturing capabilities. While these CAE tools are effective at analyzing cost on a single component and allow for cost conscious development of that component – they are limited in their ability to perform a cost analysis of a subsystem or an entire system. Additionally, the current CAE tools require manual input of manufacturing data into the costing software, which is lost when the design is modified in subsequent iterations. There is a need to embed the manufacturing costing inputs into the CAD model such that, when the cost analysis is launched from the 3D model, the manufacturing inputs into the costing tool are automatically populated and retained in future iterations.

In current practice, the ability to conduct detailed manufacturing analyses, particularly detailed cost estimation at various levels of detail (i.e., part, sub-system, system, and full product), requires human-in-the-loop interaction. Reducing or eliminating this human interaction during the design process by having continuous manufacturability and cost feedback will reduce the number and magnitude of engineering change requests later in the product development process. Integrating this feedback into the designer's CAE toolset will increase the effectiveness of the tools.

The objective of this project was to mature and implement the DARPA Adaptive Vehicle Make (AVM) iFAB) manufacturing analysis software for an industrial design/manufacturing organization and demonstrate the technology for a product under development. The Manufacturing Analysis System (MAS) is a powerful software architecture that provides model-based, automated analysis of designs with the ultimate goal of providing accurate cost estimates at various levels of a product's bill of material. The MAS combines streamlined manufacturing data specification within the CAD/design environment, detailed should cost estimation using a variety of analysis models/tools, and product cost management, and visual feedback in a web-based environment to support a diverse product development team.

The benefits of the MAS technology directly target key requirements identified by Oshkosh, the industrial partner on the project and target implementation organization. These requirements included more streamlined methods for manufacturing data specification, design submission for automated manufacturing and cost analysis without leaving the CAD/design environment, cost rollups at multiple levels within a product's BOM, and product cost tracking throughout its design phase. A Use Case was conducted in which Oshkosh used the MAS tools for a product currently under development and compared key metrics against conventional manufacturing analysis and cost management methods.

The use of the MAS resulted in an 89-97% reduction in effort to generate manufacturing data input, a 95-98% reduction in manufacturing analysis and cost estimation time, and an 81-97% reduction in total duration in achieving product cost rollups. Additional anecdotal feedback from Oshkosh indicated that designers greatly appreciated the simplicity of the input methods in MAAT within Creo, the MAS project cost visibility was extremely valuable for a multi-functional product development team, and the PLM integration greatly improved efficiency and reduced data management risks. While not demonstrated during the Use Case because the test product had not entered the manufacturing phase, it is anticipated that implementation of the MAS will decrease the number of ECRs by providing designers product cost feedback early in the design process and guidance on where design change should be explored to reduce overall product cost.

The MAS was developed such that medium to large design/manufacturing OEMs with capable IT organizations could implement the technology without the need for commercialization or extensive support. This is made possible with a very detailed system specification and installation instructions, which is provided as a deliverable for this project. The web-based System Specification Document includes a detailed system overview and description of system elements, system/network diagrams, server setup recommendations and instructions, recommended/required specifications (hardware and software), and implementation tests. At the completion of this project, the IT organization at Oshkosh was reviewing the System Specification Document and developing internal requirements for implementation, perhaps before the end of 2017. The System Specification Document and MAS source code is also available to DMDII members in accordance with the terms of the Membership Agreement, where additional MAS implementation may be explored in 2018.

II. PROJECT OVERVIEW

a. Industry Problem

Cost is a major customer requirement affecting nearly all design and manufacturing products. Having the ability to obtain cost and manufacturability information early in the product development process makes the process much more efficient. However, in order to effectively study tradeoffs between different technologies or features, a designer must know the cost of the component, subsystem, or full system that is being studied. Designers currently use computer aided engineering (CAE) tools to perform manufacturability and cost analyses, but they lack an integrated capability to efficiently take a 3D design and roll up the cost of a complex system or subsystem and do not provide manufacturing feedback during early stages of product development. Having manufacturability and cost feedback during design will lead to a more efficient and effective design process and realization of products faster by identifying and eliminating costly design decisions.

In traditional design practices with current CAE tools, a designer has the ability to take a 3D design into separate cost analysis software for detailed cost analysis on that component. Furthermore, an organization's specific manufacturing cost/constraints can be integrated into the costing CAE tool such that the cost analysis output is reflective of the organizations manufacturing capabilities. While these CAE tools are effective at analyzing cost on single component and allow for cost conscious development of that component – they are limited in their ability to perform a cost analysis of a subsystem or an entire system. Additionally, the current CAE tools require manual input of manufacturing data into the costing software, which is then lost when the design is modified in subsequent iterations. There is a need to embed the manufacturing costing inputs into the CAD model such that, when the cost analysis is launched from the 3D model, the manufacturing inputs into the costing tool are automatically populated and retained in future iterations.

In current practice, the ability to conduct detailed manufacturing analyses, particularly detailed cost estimation at various levels of detail (i.e., part, sub-system, system, and full product), requires human-in-the-loop interaction. When and where that human manufacturing engineering interaction occurs depends on an organization's standard practices. In some organizations the manual manufacturing analyses and cost estimates occur after engineering release of the design, while in other organizations the manual assessment and feedback occurs during the design process before engineering release.

These two human-in-the-loop scenarios are further illustrated in Figure 1 and Figure 2. In Figure 1, design engineers complete and release a design that satisfies the design requirements, which are often heavily driven by engineering and performance requirements and not by manufacturability or cost. A manual manufacturing analysis of the complete design is then required prior to manufacturing release. It is during this time where many Engineering Change Requests (ECRs) are introduced based on non-manufacturable designs or excessive predicted costs, thus requiring re-engineering, delaying the manufacturing release, and increasing product development costs. ECRs are inevitably introduced after manufacturing commences due to unforeseen circumstances (e.g., requirements change, quality failures, etc.), and therefore, the human-in-the-loop interaction is repeated for re-engineered designs prior to release back to manufacturing, further delaying product development.

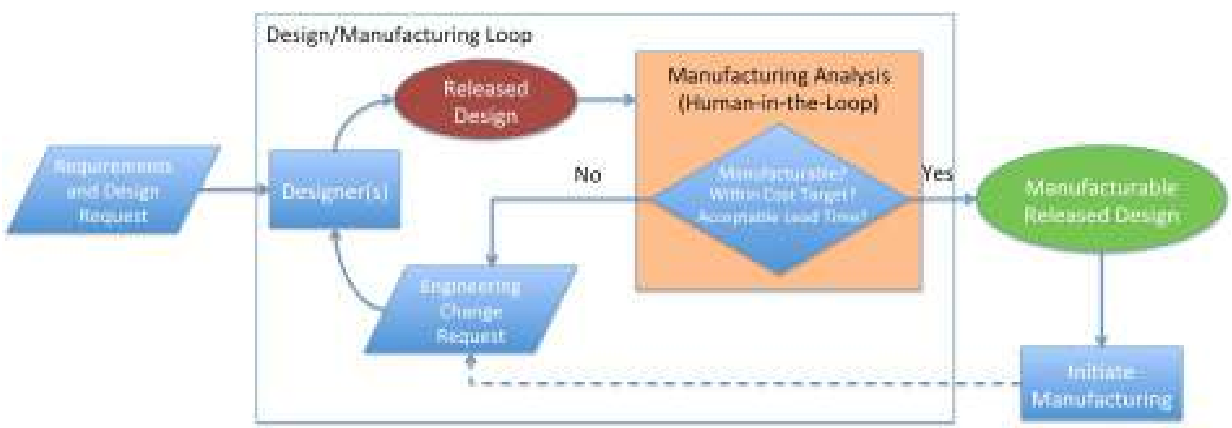


Figure 1. Design / Manufacturing interaction (Current State Scenario 1 – human-in-the-loop manufacturing analysis after design release)

In Figure 2, the second design/manufacturing loop requiring human intervention, the manual manufacturing assessment is conducted *during* design, prior to the release of the design for manufacturing. While this scenario may potentially reduce the number of ECRs resulting after manufacturing release, there is an increased risk of manufacturing delay. As with the scenario described in Figure 1, ECRs will inevitably occur after manufacturing commences, but the manufacturing assessment during re-design reduces the likelihood of ECRs after re-release to manufacturing.

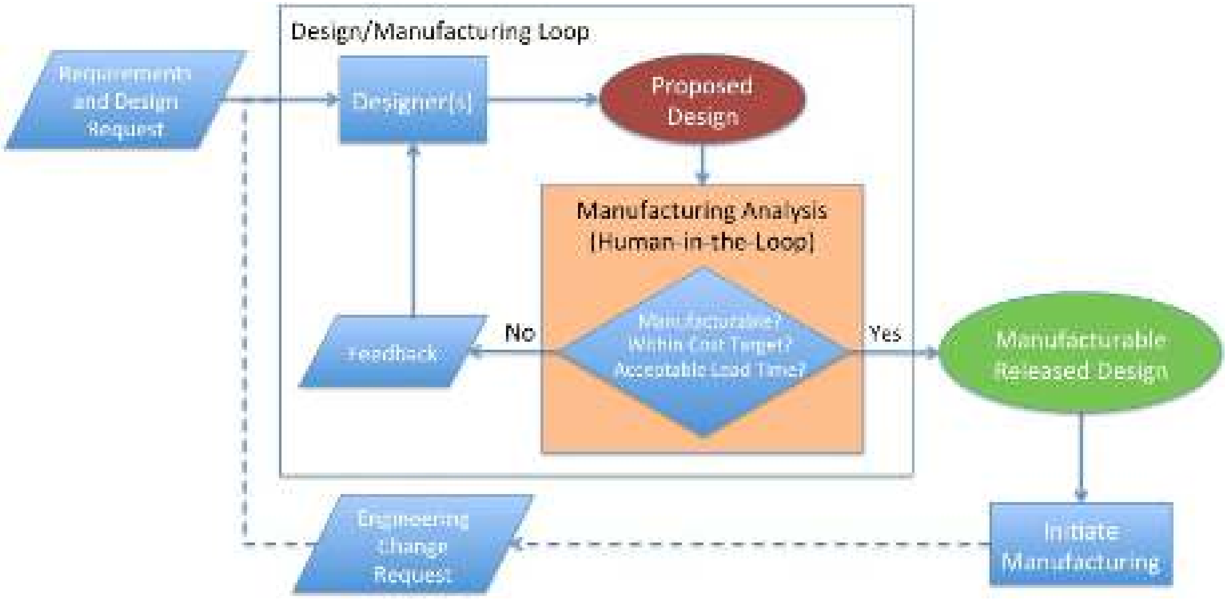


Figure 2. Design / Manufacturing interaction (Current State Scenario 2 – human-in-the-loop manufacturing analysis throughout design)

Having continuous manufacturability and cost feedback during design will reduce the number and magnitude of engineering change requests later in the product development process. Integrating this feedback into the designer's CAE toolset will increase the effectiveness of the tools.

b. Project Scope and Objectives

The objective of this project was to mature and implement the DARPA Adaptive Vehicle Make (AVM)-Instant Foundry Adaptive through Bits (iFAB) manufacturing analysis software for Oshkosh and demonstrate the technology on a product line from Oshkosh's Access business unit, JLG. The software architecture extended the iFAB information architecture developed under the AVM program, which was part of a larger tool chain aimed at the reduction of product development time from design concept through production run manufacturing. The iFAB design assist and manufacturing analysis software tools were successfully demonstrated during the AVM program by a wide test group through the FANG (Fast Adaptable Next Generation ground vehicle) design challenge focused on drivetrain and mobility design for armored combat vehicles and the FANG Gamma Testing phase focused on hull structure and survivability. Oshkosh was a contracted participant in Gamma Testing and expressed strong interest at the completion of the exercise in the automated manufacturing analysis and cost estimating capabilities of the AVM iFAB tool chain. In addition, Oshkosh expressed interest in the design assist tools introduced in Gamma Testing that were embedded in PTC Creo Parametric, the primary 3D design tool used by Oshkosh throughout their various business units.

This project sought to replace the traditional design/manufacturing loops shown in Figure 1 and Figure 2, where the reliance on human-in-the-loop manufacturing engineering assessment results in long delays in product development and increased product development costs.

The Manufacturing Analysis System (MAS) and methodology used in this project transitions two advanced technology concepts from AVM, as shown in Figure 3. The introduction of *CAD system-embedded Design Assist Tools* (DATs) enables designers to supplement their designs with sufficient manufacturing data required for downstream manufacturing assessment. This includes the MAAT (Manufacturing Analysis Augmentation Tool), which enables submission of design data packages for manufacturing assessment. The developed system also replaces the human-in-the-loop manufacturing engineering analysis with a suite of automated *Manufacturing Analysis Tools* that provide rapid feedback in the form of manufacturability checks and cost roll-ups at part and assembly levels.

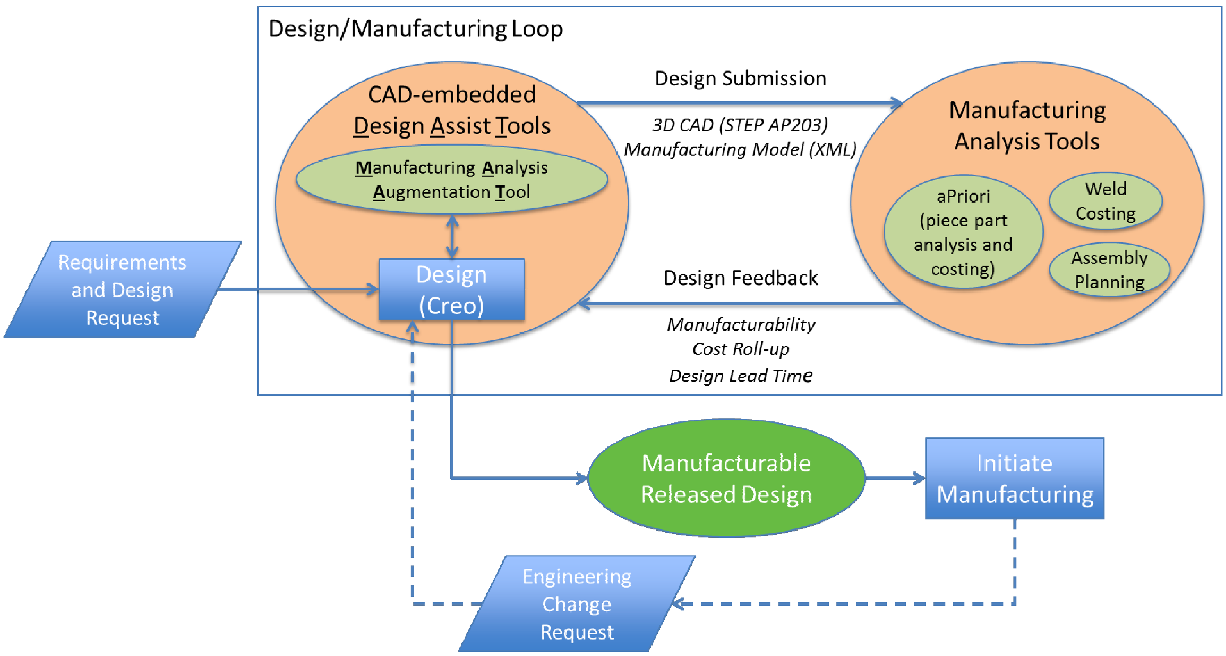


Figure 3. Projected Design / Manufacturing Interaction with Manufacturing Analysis System

c. Technical Approach

The purpose of this project was to expand and demonstrate a manufacturing analysis software system, as illustrated in Figure 3, at Oshkosh, the primary OEM supporting the project. We accomplished this by executing eight major tasks, as described in the following sections.

Evaluation of Current State iFAB Manufacturing Analysis Tools

While Oshkosh participated in the AVM Gamma testing and thus, were exposed to the iFAB Manufacturing Analysis Tools, further development of the tools through the completion of the AVM program warranted additional testing and evaluation. At the commencement of the project, Oshkosh was provided final software revisions of HuDAT (Hull Design-for-Manufacturability Assist Tool) and MAAT (Manufacturing Analysis Augmentation Tool) as well as access accounts to the ARL Penn State Manufacturing Analysis Portal (MAP), a web-based system where users can access design submissions and view detailed HTML-based results.

Oshkosh conducted extensive testing of the tools on a mix of CAD parts and assemblies provided by ARL Penn State as well as internal product designs. The testing included use of the Creo CAD plugins (MAAT and HuDAT) to conduct manufacturing data specification, submission of designs for manufacturing and cost analysis to the MAS, and observation of manufacturing feedback in the plugins and web portal.

While Oshkosh documented several benefits of the legacy MAS, including the improved efficiency of product costing exercises and simplified method for manufacturing data specification, the ultimate outcome of the task was a set of requirements for further MAS development that would be required to facilitate implementation at an industrial site such as Oshkosh. The primary requirements included extension of the MAAT Creo plugin and MAS web application and were identified as follows:

- *Streamline Data Inputs* – manufacturing data should be automatically populated based on CAD part parameters, or pull information from previous analyses
- *CAD Model Interaction* – users to interact with the active Creo CAD part/assembly while MAAT is open to interrogate model for relevant manufacturing information
- *Manufacturing Location* – users need to be able to select the manufacturing location for a fabricated part and receive cost feedback based on that geographic region
- *Analysis/Result Feedback* – additional cost feedback details are needed to help designers and engineers understand the cost drivers associated with a particular design
- *Product Cost Rollups* – the legacy MAS provided cost estimates at the lowest-level piece parts and the top-level assembly. Additional cost feedback at multiple levels of detail in a product structure is required
- *Enhanced Output Visualization* – a bill-of-material (BOM) based summary of manufacturing cost results is preferred to support multi-functional user groups
- *Cost Tracking* – the ability to track and visualize product costs over time as a product design is maturing is critical to understand the impact of design change on manufacturing cost
- *Revised Cost Models* – legacy welding and coatings cost models should be modified to accept more readily available manufacturing data

While Oshkosh's feedback on HuDAT was ultimately positive, it was determined that no further development would be pursued in this project for two primary reasons: 1) HuDAT's primary benefit of supporting automated design of monocoque, hull structures has limited use outside of the large, armored combat vehicle domain, and 2) significant development would be required to merge solid weld geometry with mating plates to generate a valid mesh for FEA – basic FEA tools embedded in commercial CAD packages can be used instead.

Manufacturing Analysis System Modification

Based on the requirements defined from Oshkosh's initial tool evaluation, the bulk of the modifications to the legacy iFAB tools centered on the extension of the MAAT Creo plugin and the re-development of the MAS web application, the central repository for product/project cost information. These tools are described in more detail in the next two sections.

Manufacturing Analysis Augmentation Tool (MAAT)

The MAAT tool, shown in Figure 4, is a plugin application developed using the PTC Creo Development Toolkit that can be added to a user's PTC Creo environment. Its primary purpose is to guide designers in specifying the minimum set of manufacturing information required for the MAS to conduct a detailed cost analysis. MAAT can be instantiated on any level of the CAD product structure (part or assembly). When run, a geometric reasoning algorithm is executed to not only identify the piece parts in the design, but also identify the assembly seams, defined as part-to-part, part-to-assembly, or assembly-to-assembly interferences that often imply a manufacturing joining operation.

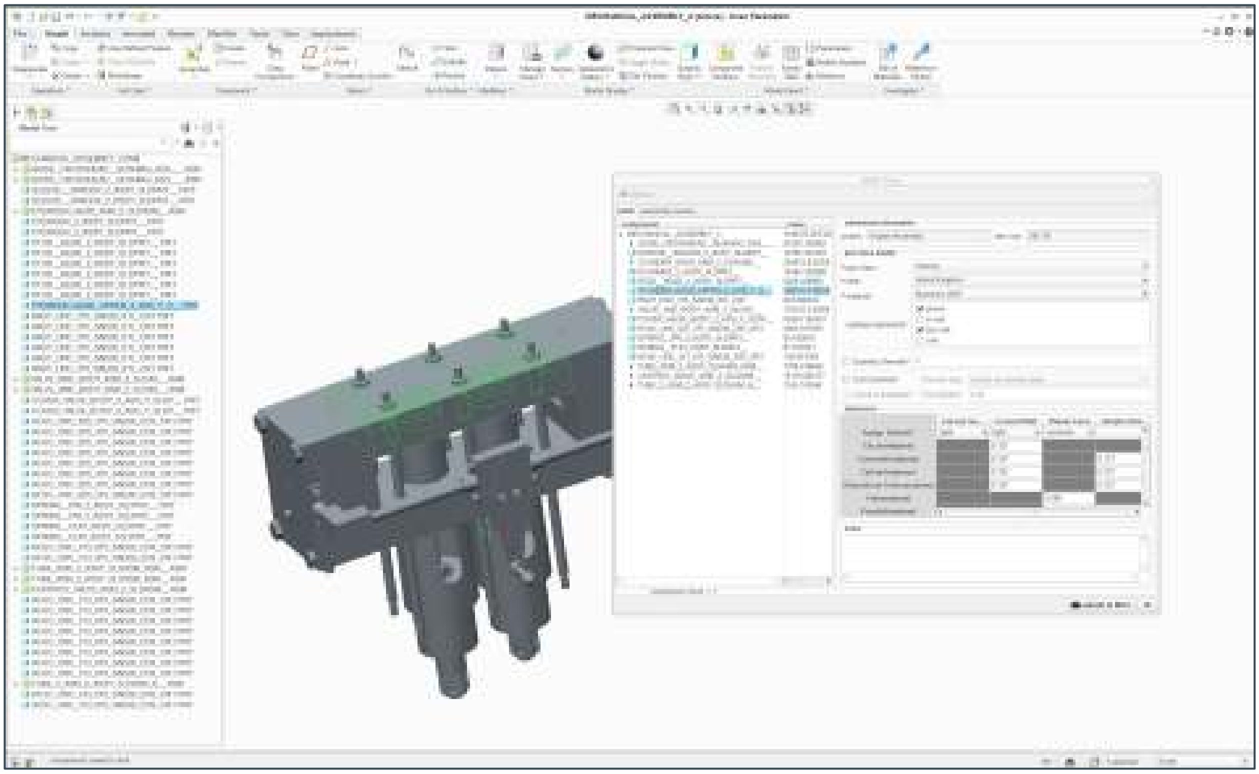


Figure 4. MAAT Plugin in PTC Creo

The MAAT user interface is organized into two primary views: Parts and Assemblies. When focused on parts, the user is presented with an expandable product-tree view of unique parts identified in the assembly. Each part requires manufacturing data specification prior to submission for cost analysis. Users can specify the following information for selected parts (* denoted required for MAS submission):

Part Class* – select from Casting, Machined, Pipe/bar/Tube, Plastic Molding, Plate/Sheet, or Purchased. If an assembly is selected in the product tree, the user must specify if the assembly is a roll-up (i.e., cost will be an aggregate of lower level parts and subassemblies), fabricated (total cost will included lower level piece part costs as well as assembly costs determined by the MAS), or purchased

VPE* – (VPE refers to Virtual Production Environment, an aPriori-associated cost model; see next section). Select the geographic region in which the part is assumed to be manufactured

Material* – select from pre-configured material types based on selected part class; automatically populated based on CAD material parameter if MAAT properly configured

Coating Requirement – select (multiple possible) from primer, e-coat, top-coat, carc

Quantity Override – select and complete to ignore the annual product quantity associated with the cost project

Cost Override – select and complete to use a manually-entered cost rather than allowing the MAS to analyze and cost a part

Tolerances – input feature-based tolerance values for variance tolerance types; automatically populate by selecting tolerance policies (tight, moderate, loose) for feature type based on MAAT configuration

When selecting the Assembly Seams view in MAAT, users are presented with a tree-structure of all subassemblies in the design and the associated assembly seams for each subassembly. When selecting an assembly seam, the user can specify either a mechanical joint class, a welded joint class, or incidental contact. The following information for each joint class is specified below (* denoted required for MAS submission):

Mechanical
- Fastening Method* – select from bolted, bolted (blind), machine screw, press fit, snap fit, crimp/clamp fit
- Fastener quantity* – specify the required number of fasteners required for each instance of the mechanical assembly seam

Welded
- Weld Type* – select from seam, stitch
- Bead Length – enter the bead length for stitch weld type
- Pitch – enter the pitch for stitch weld type
- Joint Length* – enter the total joint length for the welded assembly seam
- Two-Sided?* – check if the weld is two-sided
- Joint Type* – select from fillet or groove
- Weld Size* – enter the size of the weld (assumed to be leg length for fillet, depth for groove)
- Weld Material* – select the base material of the parts being welded

The MAAT plugin allows users to log in and select an associated cost project, retrieve manufacturing data from previous submissions to that cost project, submit additional analysis requests to the MAS, and change the associated cost project. This is all made possible through HTTP requests to the MAS web application (discussed in the next section) using a REST (Representational State Transfer) API. Design submissions include the 3D CAD geometry (in STEP AP203 format) and associated manufacturing models (i.e., manufacturing data specification) in xml format. The MAS is able to parse these submission packages and initiate the required analyses based on the manufacturing data specification.

Manufacturing Analysis System Web Application

A web-based application was developed as a central repository for project cost information. The Ruby on Rails (Rails) framework establishes a connection between the designer community using the MAAT application in Creo and the backend MAS analysis software that performs the actual manufacturing analyses. The web application supports user groups with varying roles, including designers that are consistently using MAAT to submit designs for cost analysis and tracking cost trends based as designs mature and chief engineers/managers that are primarily interested in project cost roll-ups at various levels of detail in the BOM. The MAS web application includes a built-in user management capability that enables organizations to add authenticated users and adjust their site permissions based on their role. Additional permissions may be set for users for specific projects.

The MAS web application enables organizations to set up and manage multiple *cost projects*. A cost project is assumed to be associated with a particular product under development. A user with sufficient permissions, e.g. chief engineer, using the web application to start a cost project, as shown in Figure 5.

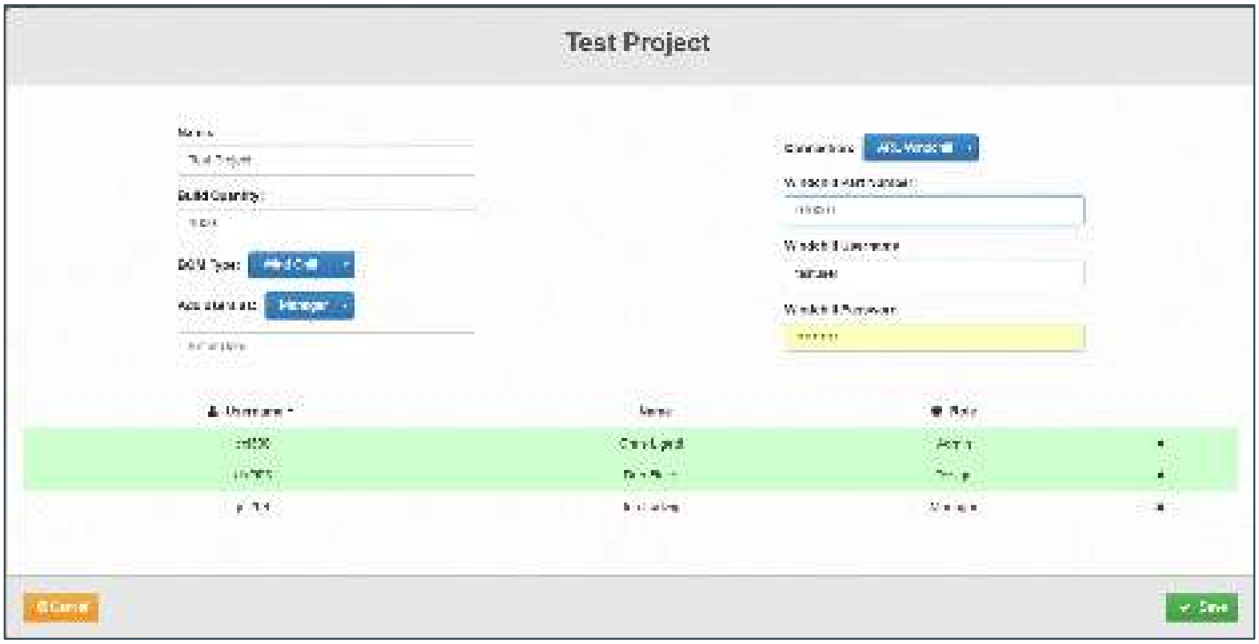


Figure 5. Cost Project Startup using the MAS Web Application

The initialization of a cost project requires the definition of a name, build quantity (annual volume), and BOM type. When selecting Windchill BOM type, the MAS web application interfaces directly with the specified Windchill (PLM system) server to extract BOM/product structure information (the PLM system interface is discussed in greater detail later in this report). The Stand Alone BOM type enables users to select an Excel file that contains BOM/product structure information in a standardized format. Finally, individual users can be added to cost projects with specified roles (guest, designer, manager).

After a cost project is initialized in the MAS web application, designs can be submitted from MAAT for analysis. Results from these analyses can be viewed in multiple ways using the web application. Product costs from the latest MAS submissions, from piece parts up through roll-up assemblies are presented in an expandable product tree, where users are shown the part name, part number, revision/version, target cost (user editable), current cost, weight, quantity (subassembly total and overall product total), user and date (latest submission). An example of the product tree cost project summary is displayed in Figure 6.

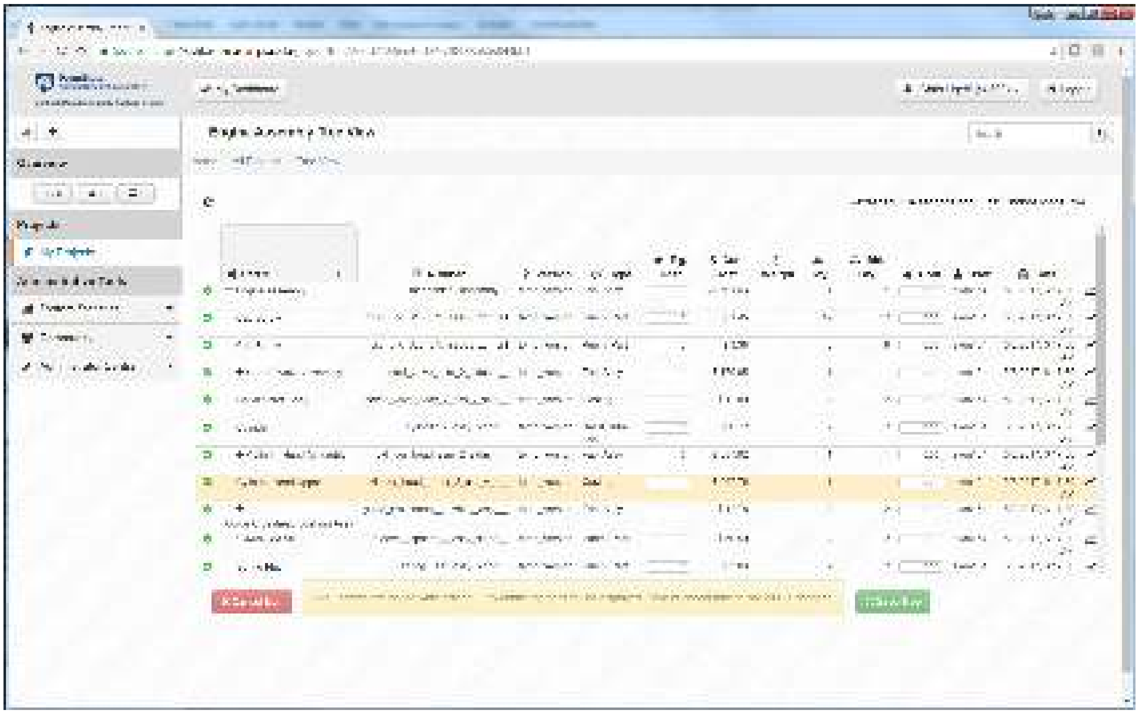


Figure 6. MAS Web Application – Cost Project Summary in Product Tree

Users can also obtain product cost information for individual parts or assemblies in the product tree. Part cost details include a breakdown of costs into the following categories: labor, material, overhead, setup, investment, and other miscellaneous costs. Assembly cost details include a breakdown of costs into purchased part, manufactured part, assembly, overhead, and material handling costs. Figure 7 and Figure 8 show examples of product cost details for a piece part and assembly, respectively.

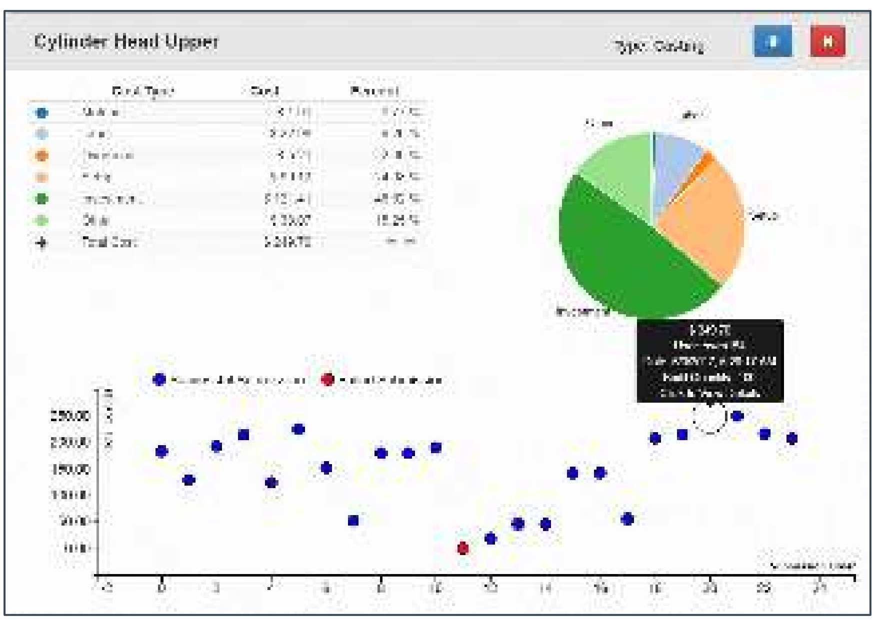


Figure 7. Product Cost Details (Part)

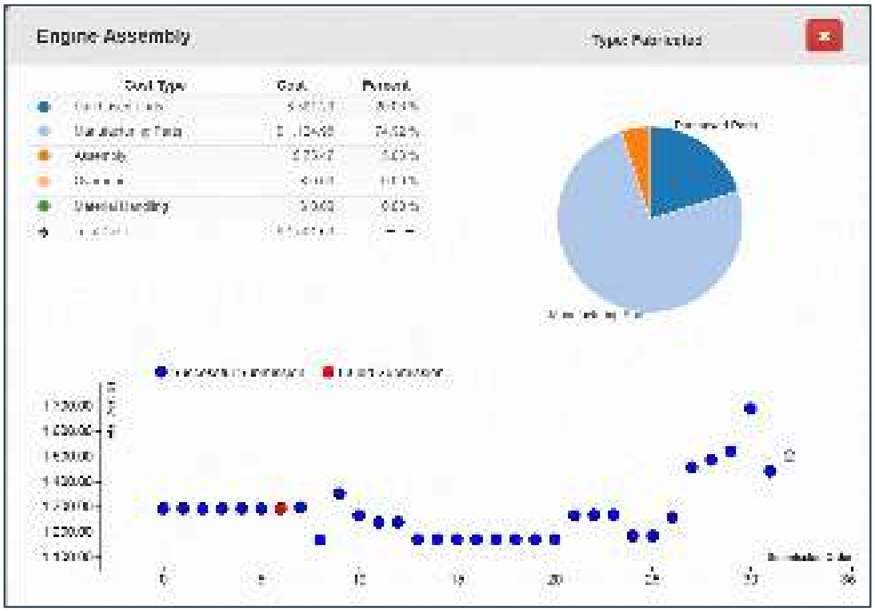


Figure 8. Product Cost Details (Assembly)

The product cost details for both parts and assemblies also include an interactive cost tracking chart, where, for each submission result on the chart, users can obtain the cost breakdown information, the submission user and date, and manufacturing details associated with that design submission (parts only), as shown in Figure 9.

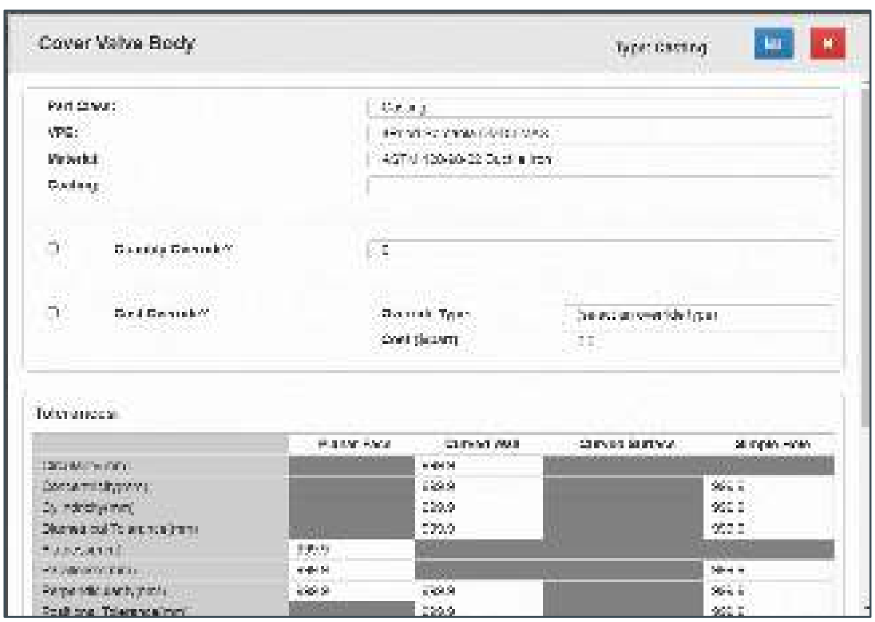


Figure 9. Manufacturing Data for Selected Part Submission in MAS Web Application

Configure Manufacturing Models

Several methods are used in the legacy iFAB Manufacturing Analysis System to support automated manufacturing analysis and cost estimation. These included custom-developed, configurable models for assembly analysis and weld cost estimation as well as implementation of commercial applications for piece part cost estimation.

aPriori software is the primary cost estimation analysis module used in the iFAB Manufacturing Analysis System for machined parts, castings, plate/sheet parts, and bar/tubes parts (including structural extrusions). As part of the AVM iFAB Foundry program, aPriori application engineers led the development of Virtual Production Environments (VPEs) for several organizations in the iFAB Foundry manufacturing network.

Oshkosh conventionally uses aPriori for limited component-level cost estimating. In that implementation, baseline VPEs provided by aPriori are employed for piece part costing through aPriori's client application user interface. This limits Oshkosh's ability to conduct automated manufacturing assessments for complex products, and a small number of aPriori subject matter experts further hinders widespread use within the organization.

The embedded cost models associated with the aPriori baseline VPEs (*Regional Data Libraries*) are all identical, which significantly limited the amount of cost model reconfiguration required. Specifically, the common baseline cost model (referred to as the *Logic VPE)* was modified to accept feature-specific tolerance inputs (*Process Setup Options*) for the original four part classes (machined, casting, plate/sheet, bar/tube), in additional to plastic molding, for bulk load analysis (automated, thin client implementation). The application of the Logic VPE to multiple Regional Data Libraries, thus resulting in a library of geographic region VPEs, was made possible through an overlay process. This process essentially enabled the creation of 12 VPEs to be used for the project, but could be extended to additional existing or future-developed regional data libraries to expand the VPE library. The VPE overlay structure is summarized in Figure 10. More detailed documentation of the aPriori VPE development can be found in Appendix a.

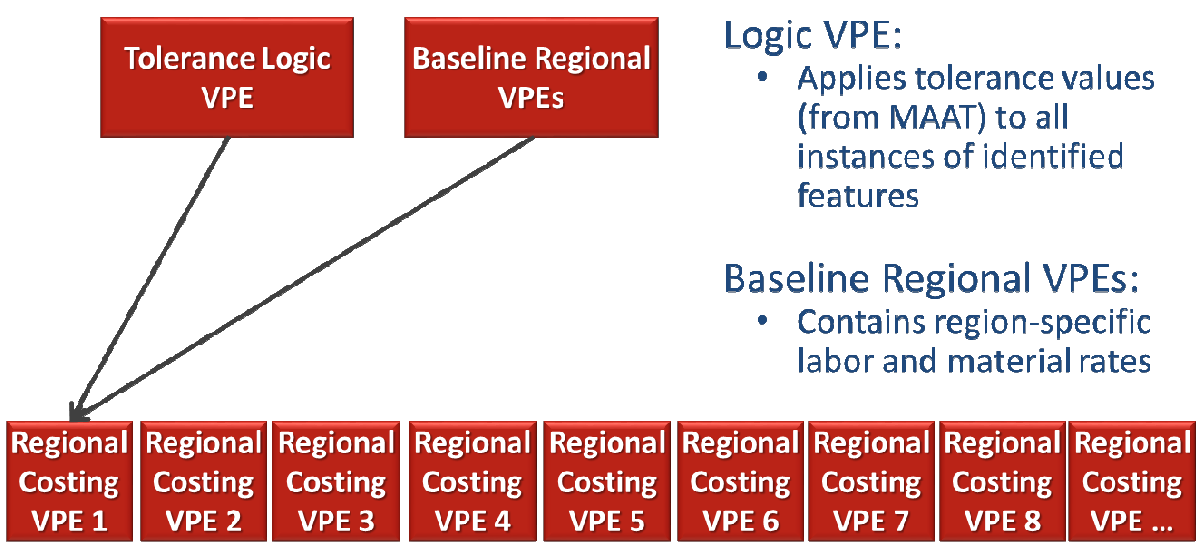


Figure 10. aPriori Virtual Production Environment Overlay Structure

In addition to the aPriori VPE development, further development of some of the custom-developed cost models were required. This included an overhaul of how coating costs were considered, allowing for multiple coating options to be specified, and implementing surface area-based coating cost models proposed by Oshkosh. In addition, the legacy weld cost model, which was heavily dependent on minimum part thickness as a design input, was modified to accept weld size as the primary input specification in addition to weld type (fillet or groove) and weld length.

Integration with Component Model Library

Many components in new product designs fall under the “purchased” part category, including catalog/commodity COTS items, as well as previously procured, vendor-fabricated components. Organizations such as Oshkosh often use a wide range of applications and databases to manage prior purchase of such components to support product cost management, however these tools are often disparate and disconnected from product cost methods.

Because no two organizations will employ the same purchased part management tools/databases, a standardized database schema was developed for existing parts that could be used for both COTS and vendor-fabricated components and integrate with the MAS. This data schema and associated *component model library* does not need to be as data-rich as the legacy AVM component model library (which contains a collection of engineering models for each component in addition to the manufacturing models that contain cost and lead time information). Instead, the intent was to establish the minimum set of information that would be required to support cost analyses using the MAS. A secondary objective in establishing the existing parts database schema was to simplify any processes that would need to be developed to extract this information from multiple disparate data systems (out of scope for this effort). The resulting component model library schema is shown in Table 1.

Table 1. Existing Parts Database Schema

existing_parts		
Primary Key	part_number	VARCHAR(255)
	component_number	VARCHAR(255)
	vendor	VARCHAR(255)
	source	VARCHAR(255)
	price	NUMBER
	lead_time	NUMBER
	notes	VARCHAR(255)

The primary benefit of access to existing part cost data is realized during manufacturing data specification using the MAAT Creo plugin. Without access to this information, users would be required to manually enter cost data for every purchased component, which could lead to inconsistencies between multiple users. Consequently, the MAAT application and MAS web application REST API were both modified to query this information when a user pulls existing manufacturing data for an existing cost project. Users are notified of existing part data in the MAAT user interface and still have the option to override purchased part costs in cases where they believe the existing part cost information is inaccurate or outdated.

Integration with Product Lifecycle Management System

Most industrial organizations that design large commercial and defense products use commercial product lifecycle management (PLM) systems that work very well with their CAD design package. Such is the case at Oshkosh, primarily a PTC Creo user that also uses PTC Windchill for product lifecycle management. A direct interface between the MAS and PLM supports the generation of cost project setup (i.e., BOM/product structure import) and management as designs mature over the product development cycle.

The Windchill PLM interface was established by leveraging Windchill's Info*Engine and SoapUI to create a custom task to extract BOM information for a user-specified assembly. The MAS web application cost project startup/management UI was modified to allow users to select the Windchill BOM import option, select the Windchill server from a list of configured options, specify the top-level Windchill part number, and provide Windchill authentication information to create the MAS cost project. An option to update existing cost projects with the Windchill BOM was also included to support product structure change over time without the need to recreate a cost project.

The Windchill interface was developed as a one-way pull to populate the manufacturability and costing analysis. While it initially appeared beneficial to expand the MAS/PLM interface to manage cost data directly in Windchill, essentially introducing a two-way interface requirement, this was determined to be a potential risk to implementing organizations and was determined to be out of scope for this effort. However, it is highly likely that the developed interface could be extended to support such a requirement in a future effort.

Use Case for Existing Product Line

To demonstrate the benefits of the Manufacturing Analysis System for automated feedback and multi-level cost roll up during design, the project team proposed a use case, where an existing product line chosen by Oshkosh was assessed using the updated MAS, including the MAAT Creo plugin and web application. The intent of the use case was to demonstrate the benefits of using such a system including the reduction in effort to obtain manufacturability feedback and cost estimates at multiple BOM levels in the MAS web application and by establishing streamlined manufacturing data specification methods without requiring design engineers to leave their primary design environment (PTC Creo).

Oshkosh selected the Use Case product, which was an existing product from their Access business unit, JLG. The product BOM consisted of 1773 total parts and 526 unique part numbers. The JLG product was made up primarily of plate/sheet parts, bar/tube parts, fabricated assemblies (i.e., weldments), mechanical assemblies, and purchased COTS components. A large portion (79%) of the product BOM was evaluated by the MAS during the Use Case; the vast majority of unevaluated parts (97%) were COTS/hardware, and nearly all (97%) of Oshkosh-designed parts were evaluated. A total of 9 Oshkosh users participated in the Use Case.

At the time the Use Case commenced, the JLG product was in the late Design/Prototype phase, which somewhat limited Oshkosh's ability to conduct a side-by-side comparison of conventional cost management techniques with the MAS methods. However, Oshkosh was still able to establish comparisons of product cost management using conventional methods including aPriori standalone (i.e., thick client implementation with aPriori subject matter experts), Costimator (a competing commercial cost estimating tool that requires significant manual intervention and is limited in supporting manufacturing part classes), and vendor quotations. All conventional cost management methods require significant manual input for BOM-based, cost roll-up summarization, and Oshkosh reported that no standardized process/tool exists to facilitate cost roll-ups until the product enters the prototype phase of development. In comparison, the MAS supports manufacturing analysis and cost estimation from conceptual design through detailed design and prototype phase.

While specific metrics and comparisons to conventional cost management are discussed in greater detail in the KPIs and Metrics section, Oshkosh noted the following observations:

- Use of MAS resulted in savings as compared to conventional cost estimation/rollup methods
  - Manufacturing Data Specification – 89-97% time savings
  - Analysis Time - 95-98% time savings (use of aPriori in standalone mode results in equivalent analysis time)
  - Total Cost Rollup Efficiency – 81-97% time savings
- Additional Benefits
  - Simplicity of input methods within design environment (Creo)
  - Rollup capability provides better project cost visibility to multi-functional product development team
  - Integration with PLM (Windchill) improves efficiency

Oshkosh also documented specific feedback for both the MAAT Creo plugin as well as the MAS web application. This feedback is summarized in Table 2 and Table 3, respectively:

Table 2. MAAT Use Case Summary

Benefits	Suggested improvements
• Inputs are quick and intuitive • Some inputs can be automatically pulled from Creo parameters • No need to leave Creo environment to run MAAT • Cost estimates can be done earlier in the design process when fewer details are available • Part quantities are calculated automatically from BOM so designers do not have to guess on annual usage • MAAT does not appear to slow down Creo when running in background • Similar parts can be classified simultaneously with multi-select • Easy to cost large assemblies/subsystems	• Error messages are sometimes unintuitive • It can be difficult to identify which weld seams have been added since MAAT welds are not shown on the model • Calculated weld lengths are often incorrect (but can be manually corrected) • Analysis and feedback are less detailed/informative than alternative methods

Table 3. MAS Web Application Use Case Summary

Benefits	Suggested improvements
• Cost rollups can be viewed without access to Creo, Windchill or other software • Cost rollups are configurable • Rollup information includes other useful data such as number of piece parts, unique part numbers, and percentage of manufactured vs. purchased parts • Cost data includes time history of each part/assembly/subsystem/product • Target costs can be defined at any level and compared to estimated costs • User roles are defined per project and allow different functionality/permissions for different roles • Easy to update the bill of materials directly from Windchill	• Add confidence range rollup functionality (column already exists) • Allow users to customize views (column order) • Add filtering to columns • Allow inputs for labor and overhead at project level • Add visual comparison of estimated cost to target cost (either in table or timeline graph) • Show product cost in "My Projects" page • Add calculation for percentage of BOM costed • Allow project information (EAU, permissions) to be edited from tree view • Fix minor formatting issues for Internet Explorer

Implementation of the MAS is discussed in subsequent sections of this report, however, Oshkosh specifically documented some implementation risks as a direct outcome of the Use Case. These risks, listed below, are addressed in the TECH TRANSITION PLAN section:

Compatibility with future software versions
- o Creo 4.0
- o Windchill 11.0
- o aPriori 2017

Cost of hardware and software licenses
Ongoing tool maintenance
- o Updates for Creo 5.0+, Windchill 12+, aPriori

Future additional functionality
- o Common component library creation and maintenance
- o Integration with ERP

Overall, the Use Case conducted by Oshkosh demonstrated that the technology developed in this project has many benefits for a medium to large organization. A significant time savings was observed in generating cost estimates and product cost rollups, and this capability could be extended earlier in the design process with the available tools like MAAT and the MAS web application. The MAAT inputs and interface with the Creo design environment will enable more widespread use than traditional tools. The cost rollup capability provides more visibility throughout all levels of the project and drives commitment to continuously monitoring cost. Finally, the MAS integration with Windchill makes BOM updates more efficient and continuous. As a result, Oshkosh has expressed interest in implementing the MAS technology and is currently investigating the return on investment of the cost to implement versus the anticipated benefits.

System Specification for Implementation

The MAS provides an automated manufacturing assessment and cost estimating capability by interfacing with two commercial engineering applications: PTC Creo and aPriori. Where these tools exist at an industrial organization such as Oshkosh, additional commercialization is not required. Instead, this project focused on the development of a comprehensive System Specification Document with detailed installation instructions for the MAS. The expectation is that with this detailed documentation, an industrial organization such as Oshkosh would be able to implement and support the MAS technology with minimal, if any, outside support.

The System Specification Document includes detailed descriptions and/or instructions for the following elements:

- System overview
  - System/Network diagrams
  - System element descriptions (Web Client, Creo Client, Main server)
  - MAS Internal Server details
- Recommended Specifications
  - Main MAS Internal Server
  - Worker machines
- Setup instructions (Windows-based and Linux-based)
  - Network File System
  - Java Configuration
  - Apache Install
  - Web Application Dependencies Installation and Configuration
  - Key and Certificate Setup
  - Postgres Install and Setup
  - Rails Install and Configuration
  - Apache Configuration
  - Tomcat Installation and Configuration
  - Task Manager
  - Worker
  - Configure Process Permissions
  - Testing
- Miscellaneous
  - Recommended Specifications (OS, RAM and hardware)
  - Tested Machines
  - MAAT Installation Instructions
  - Miscellaneous Instructions
  - Windchill API Configuration
  - Development Configuration and Instructions
  - Rails Extras

The System Specification Document was developed in a web-based (HTML) format for simplified navigation. The documentation is not described in detail in this report, but instead was provided as a deliverable to DMDII and should be available to all DMDII members in accordance with the terms of the Membership Agreement.

d. Anticipated Benefits

Implementation of the Manufacturing Analysis System directly results in reduced product development time, aligned with the DARPA AVM objective of five-fold reduction. The projected benefits stem from the reduction of manual labor required to create, modify, and analyze a design in addition to the reduction of redesign efforts that occur due to ineffective design analysis tools.

The primary benefit is achieved by reducing the amount of manual labor required to provide detailed cost roll-ups and manufacturing analyses that occur either during design or after design release. The MAS, particularly its primary user tools, MAAT and the MAS web application, enable rapid manufacturing assessment, with emphasis on detailed product cost rollups, during the design process, providing rich feedback that improves the design prior to manufacturing release. The Use case also revealed a more definitive quantitative measure of this benefit by comparing current product development methods with those provided by the proposed software system. This is discussed further in Section III.

A further benefit of the automated manufacturing analysis capability is the potential reduction in the number of Engineering Change Requests (ECRs) that traditionally result when non-manufacturable designs are released to production. These ECRs can have a detrimental effect on product completion time and prove costly in requiring re-engineering to mitigate the identified issue. The manufacturing release of designs that have received a thorough manufacturing analysis, with clearly understood cost impact, will help to reduce these types of ECRs.

Finally, accurate cost roll-up at this level of detail is essential for a manufacturer to remain competitive; however, the excessive effort in manually generating this information inhibits the organization from conducting this analysis very often. The MAS's ability to provide detailed cost roll-ups at various levels of detail, including system and subsystem, replaces manual efforts of doing so in traditional product development, allowing for daily or weekly cost roll-up predictions to guide the design process.

III. KPI'S AND METRICS

While no formal goal-oriented metrics were defined at the commencement of this project, the Use Case conducted by Oshkosh revealed three metrics and validated improvements when using the MAS as compared to conventional methods of product cost management at medium/large design-to-manufacture organizations such as Oshkosh. Oshkosh compared the use of the MAS (including MAAT and the web application) to conventional cost analysis tools (aPriori standalone, Costimator, and traditional vendor quotation) and measured performance on the following metrics, which are described in the following subsequent sections: 1) Level of Effort, 2) Time to Receive Feedback, and 3) Total Effort Required (to obtain cost rollups). Another metric that was considered for evaluation when using the MAS was the number of engineering change requests (ERs), however, this metric cannot be fully explored until the candidate product enters the manufacturing phase of product development, which did not commence during the period of performance of this project.

a. Level of Effort

Level of Effort referred to the time required to specify the necessary manufacturing data to conduct manufacturing assessment and cost estimation. In total, 361 unique parts and assemblies were identified in the use case requiring manufacturing data specification. As shown in Figure 11, there was a significant reduction in manufacturing data specification time (measured in hours) using the MAS as compared to aPriori standalone (89% faster), Costimator (91% faster), and Quotation (97% faster).

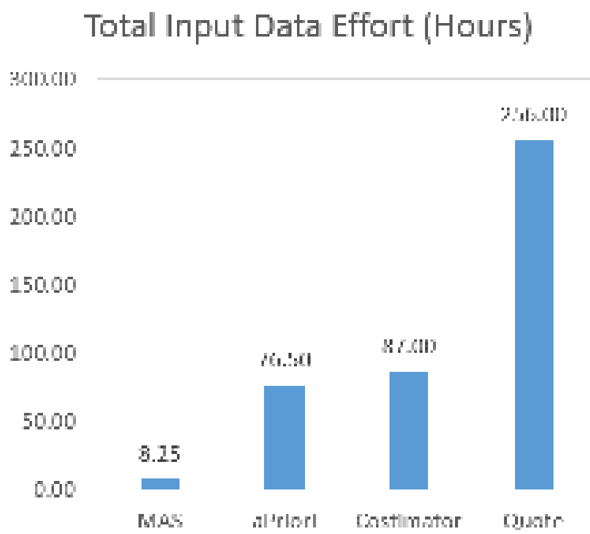


Figure 11. Comparison of Manufacturing Data Specification Level of Effort

b. Time to Receive Feedback

Time to receive feedback was measured as the time between the submission of designs and completion of analysis. As shown in Figure 12, there was a significant reduction in cost estimate wait time (measured in hours) using the MAS as compared Costimator (95% faster) and Quotation (98% faster).

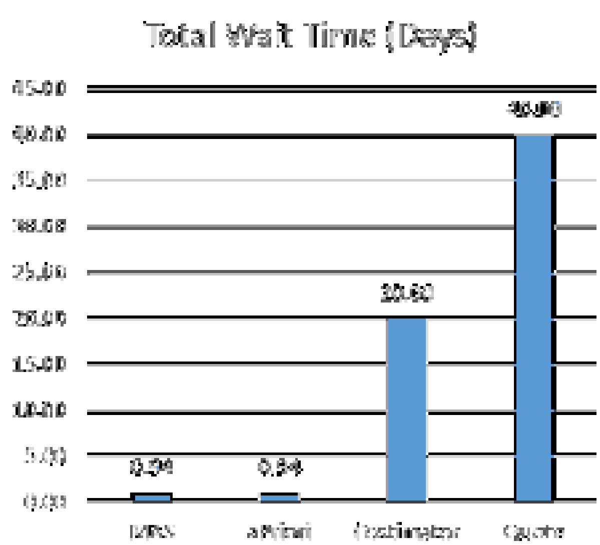


Figure 12. Comparison of Manufacturing Analysis and Cost Estimation Wait Time

The analysis time when comparing the MAS and aPriori standalone was identical, however this observation was expected since analysis time is heavily dominated by the piece part manufacturing assessment, and the MAS and aPriori standalone essentially use the same analysis method.

c. Total Effort Required

Total Effort Required combines the manufacturing data specification and analysis wait time metrics. It is a more appropriate metric for considering the fact that the MAS not only speeds up manufacturing data specification time and analysis wait time, but also automates the cost roll-up, which was a primary project objective. As shown in Figure 13, there was a significant reduction in total effort using the MAS as compared to aPriori standalone (81% faster), Costimator (94% faster), and Quotation (97% faster).

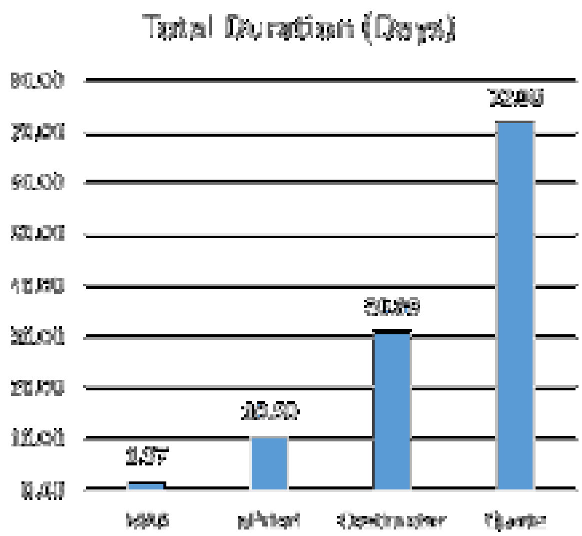


Figure 13. Comparison of Total Effort Required

IV. TECHNOLOGY OUTCOMES

a. System Overview

The system is comprised of a Main Central Server that is tightly integrated with a database to store jobs and the results of analysis runs. The Main Central Server handles all external requests which include those from web clients and Creo clients. Connected to the Main Central Server are Worker Machines that are used to process the jobs, i.e. perform the analyses. Jobs are passed to the Worker machines from the Main Central Server where they are processed and results are returned back and stored in the Central Database. An email server is also connected to the Main Central Server to send automated communication about account information. A Windchill Server holds configuration managed designs and cost information and is connected to the system store design information needed to perform the cost analyses. Figure 14 shows the high level system components and the relevant connections.

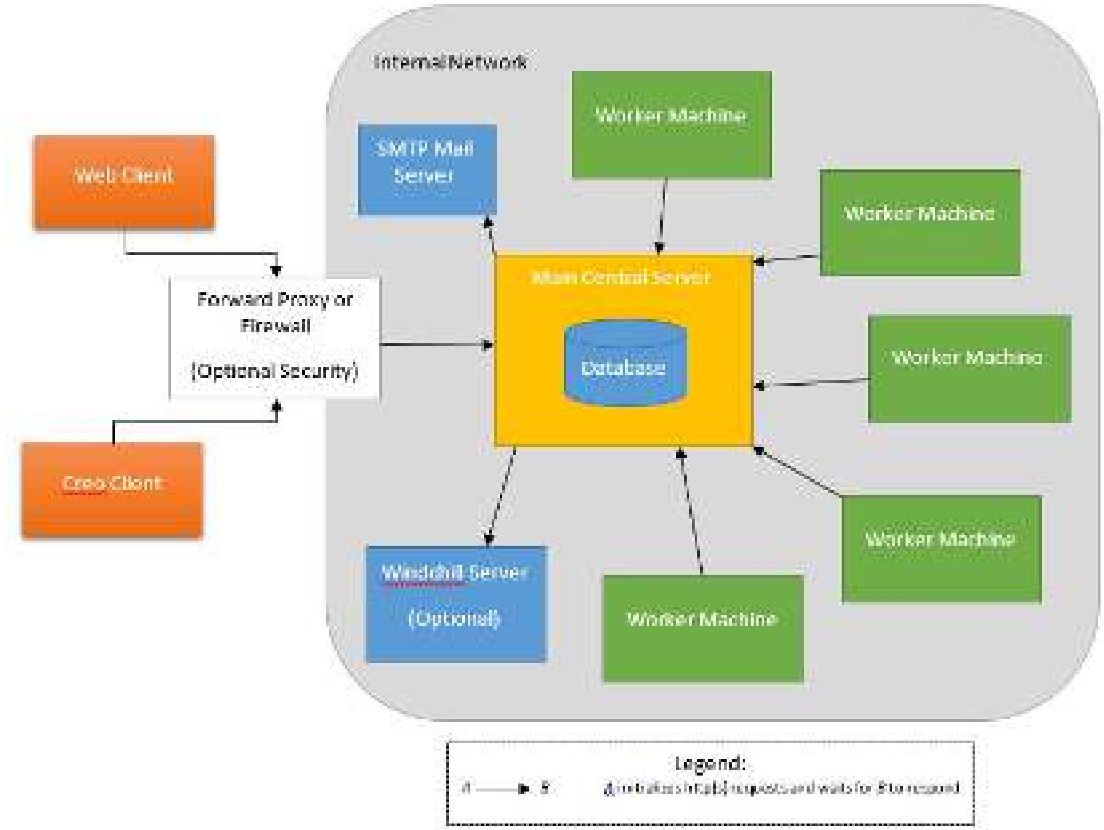


Figure 14. Manufacturing Analysis System Overview

b. System Requirements

There are several specific requirements and/or recommendations that should be met in order for an organization to implement the MAS. These include required commercial software and recommendations for hardware and software for the main MAS server and distributed worker machines.

Required Commercial Software Licenses:

PTC Creo 2.0
PTC Windchill 10 (only required for PLM integration)
aPriori 2016

Recommended Specifications

Main Server

The following specifications pertain to the main Manufacturing Analysis System internal server.

<u>Operating System</u>

An Ubuntu Server with long term support (LTS) is the recommended operating system for the main server. Any Linux distribution is preferred over windows because ruby is not native to windows and struggles for compatibility. Additionally, Phusion's Passenger is not windows compatible. Passenger is an excellent apache module that helps the rails application server handle more traffic in a stable manner. The rails application is able to handle simultaneous traffic much more efficiently than apache alone.

Using windows is possible but this configuration cannot handle multiple requests simultaneously. The windows versioncan process only one request at a time while the rest are queued and processed sequentially. Since Ruby is not native to windows the installation of all of the software required on a Windows system can be challenging. Cygwin is used to allow ruby to function on Windows by providing the missing OS commands that ruby uses. This works but is less stable since software updates may break the system.

<u>RAM</u>

Recommended: >= 32 GB
Recommended Minimum: 16 GB
Bare Minimum: 4GB

The development systems did not suffer from any major performance issues in the low traffic environment observed during the execution of this project, but most of the memory was in use at all times. Memory was always the bottleneck in the development systems during the analysis period.

Using 32 GB or more is ideal. In a low traffic (<8 users) setting 16 GB would be adequate. The development system only used 16 GB on the main server. The bare minimum is 4GB. It is not recommended to try to use a machine with 4 GB of RAM as the main server, but it may be possible in a very low traffic (1 user) setting. Do not install the software on the machine if it has less than 4GB of RAM.

<u>CPU:</u>

Recommended: >= 8 Core @ ~3.30 GHz
Recommended Minimum: >= 4 Core @ ~3.30 GHz
Bare Minimum: 1 Core @ 2.30 GHz

<u>Disk Space</u>

Recommended: >= 0.5 TB of free space
Recommended Minimum: 80 GB of free space
Bare Minimum: 50 GB of free space

The system will use a significant amount of disk space over time as the logs and submissions are all saved and retained until removed by a system admin. It is recommended to have excess disk space if available.

Worker Machines

The following specifications pertain to the repeatable Worker Machines which execute the core manufacturing assessment and cost estimation analyses and enable parallel processing.

<u>Operating System (OS)</u>

Windows Server 2012 or newer is recommended. Any Windows Server after 2008 is sufficient. Windows 7 or newer could also be used, but Windows Server is preferred.

<u>RAM</u>

a. Recommended: >= 32 GB

b. Recommended Minimum: 12 GB
c. Bare Minimum: 8 GB

As with the main server, memory was the bottleneck of the system. Using 32 GB leaves room for growth and should handle multiple simultaneous submissions easily. In a low traffic setting (<8 users) 12 GB would be adequate. The development system only used 12 GB on each worker machine.

These specs assume each worker machine will run 5 worker instances. More machines with lower specs and less instances could be substituted. It is recommended to run 25 worker instances total with 5 instances on 5 machines.

CPU

Recommended: >= 8 Core @ ~3.30 GHz
Recommended Minimum: >= 4 Core @ ~3.30 GHz
Bare Minimum: 4 Core @ 2.30 GHz

The bare minimum was the lowest tested specs for a worker instance.

Disk Space

Recommended: >= 0.5 TB of free space
Recommended Minimum: 60 GB of free space
Bare Minimum: 40 GB of free space

The system will use up disk space over time. It is recommended to have excess if available.

Virtualization

If this system will be deployed on a virtualization server or cluster, we recommend following the same guidelines specified in the test machines section of the MAS System Specification Document.

c. System Architecture

Figure 15 shows the Manufacturability Analysis System Architecture. The system is comprised of client applications/interfaces, and Analysis Server, and Worker Machines. The main server is comprised of an Apache server, Tomcat Servlet, Task Manager Instance, Network File System, and a PostgreSQL database. The Worker Machines are Windows virtual machines that host an API with the Main Server and the aPriori application. The Interface Clients include the Creo Client (MAAT) and the Web Client. For more details please see the System Specification Document delivered to DMDII under separate cover.

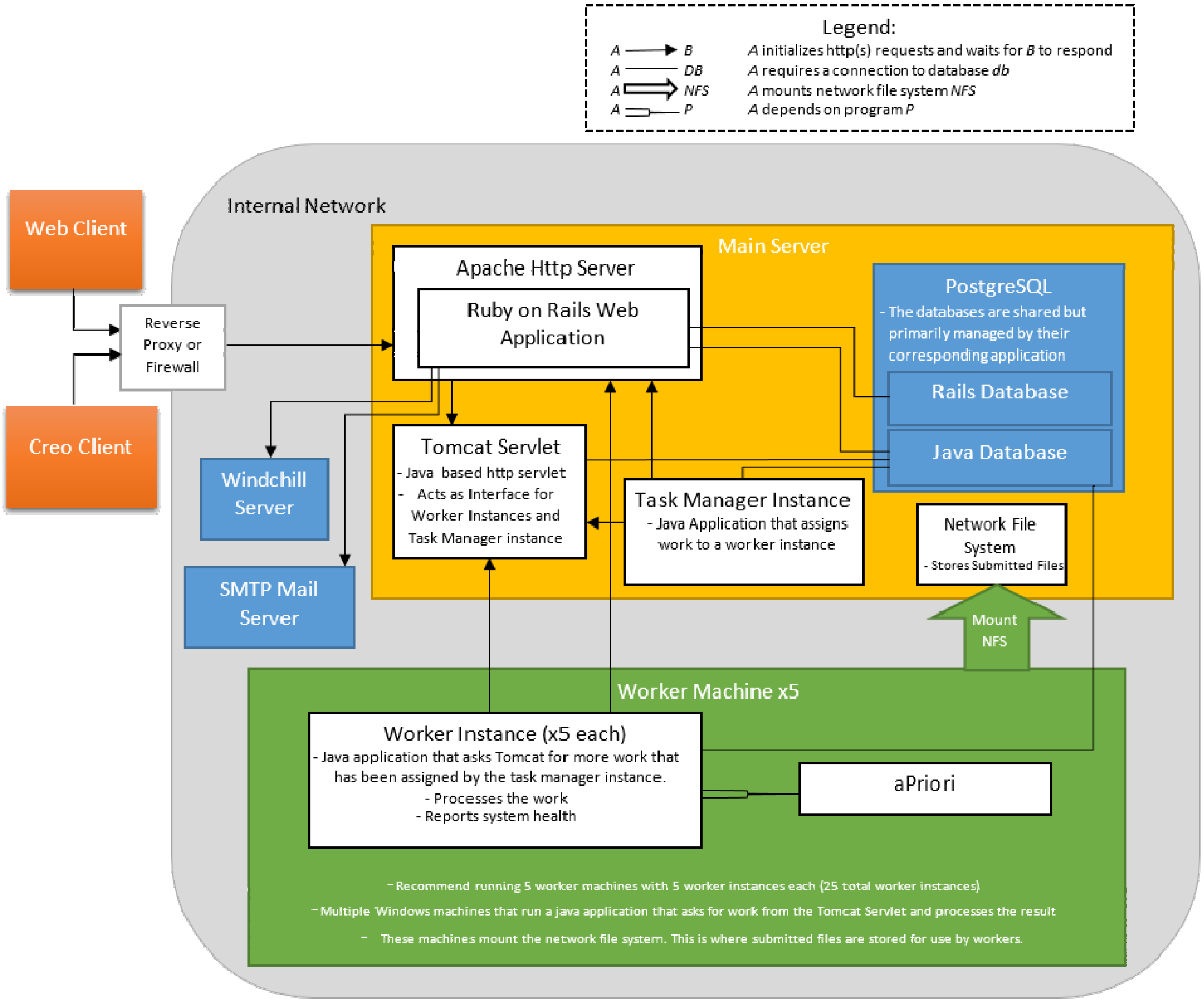


Figure 15. Manufacturing Analysis System Architecture

d. Software Development Documentation/Design Document

As discussed in section III-c, a comprehensive System Specification Document with detailed installation instructions were developed for the MAS to support industrial organizations to implement the MAS technology with minimal, if any, outside support. The System Specification Document was developed in a web-based (HTML) format for simplified navigation. The documentation was provided as a deliverable to DMDII and should be available to all DMDII members in accordance with the terms of the Membership Agreement upon request.

V. ACCESSING THE TECHNOLOGY

a. Background Intellectual Property

The Manufacturing Analysis System relies on three commercial software applications, PTC Creo, PTC Windchill (optional), and aPriori. These software packages are considered background intellectual property, where they are only available to implementing organizations with license agreements with the software vendors.

b. Project Intellectual Property

All Manufacturing Analysis System software developed in this project, as well as that previously developed under the DARPA AVM program, was done so under the open MIT open source license agreement. This includes the following:

Software interfaces to commercial software tools (aPriori, PTC Creo and Windchill)
Manufacturing Analysis Augmentation Tool
MAS Web Application
MAS decomposition and analysis dispatching software
Custom manufacturing/cost analysis models, including assembly sequencing, mechanical assembly, welded assembly, and paint/coatings

While the Penn State Applied Research Laboratory retains intellectual property rights of the developed MAS technology, all implementing organizations or can use or extend the MAS software based on the MIT open source license. Future modification of the MAS by ARL Penn State or other organizations is not required to be disseminated to other organizations and may be done so under a more restricted software license.

VI. INDUSTRY IMPACT & POTENTIAL

The Manufacturing Analysis System developed in this project is primarily intended for industrial OEMs that design and manufacture (both internally and through vendor outsourcing) large/complex products built at low-medium rates of production. This would include, but is not limited to heavy manufacturing industries (farm/construction equipment, DoD combat vehicles, etc.). Benefits to high-rate production environments such as automotive manufacturing is not anticipated. Also, benefits to the small and medium enterprise may be limited based on lack of design functions or inaccessibility of the required commercial software tools expected for the MAS.

VII. IMPLEMENTATION BARRIERS

The intent of this project was to develop a software system to support automated manufacturing analysis throughout the design phase for new and existing products. Commercialization of the technology is not being pursued due to the dependencies on other commercial software applications and an implementing organization's ability to install and support the tools based on open source licensing. However, as is the case with most large/complex software system implementations, there are potential risks and barriers to successful implementation. Oshkosh, the industrial partner participating in this project and the first potential implementation site, was instrumental in identifying these risks and barriers, which are summarized below in Table 4.

Table 4. Implementation Barriers

Barrier/Risk	Potential Mitigation
Installation/setup Difficulty	Refer to detailed System Specification Document (project deliverable)
Future Software Version Compatibility Creo 3.0+ Windchill 11.0+ aPriori 2017	Creo – initial testing of MAAT in Creo 3.0 was successful. Implementation in Creo 4.0 may require MAAT software revision due to updated development toolkit from PTC; software development support may be required Windchill – the custom task for BOM extraction developed in this project, leveraging the Info*Engine and SoapUI, may need to be modified with major revision to the Info*Engine framework. Support from PTC may be available with a current license agreement aPriori – virtual production environments (VPEs) have been proven to be upward compatible with new versions of aPriori. A change to the aPriori bulk load interface may require modification of the MAS software interface that invokes aPriori; software development support may be needed
MAS Software Use Difficulty	Refer to MAAT and MAS Web Application user instructions (project deliverable)
Required Commercial Software (PTC Creo, Windchill, and aPriori) Cost	MAS was specifically developed for organizations that either already had licenses for PTC applications and aPriori or would strongly consider purchasing based on the benefits of the technology. Commercial software costs cannot be accurately estimated because these must be negotiated between the industrial organization and the commercial vendors based on a number of factors, including number of users, frequency of use, etc.
Intellectual Property Rights	All MAS source code is open source (MIT) and comes with unlimited use rights. The license agreement also assumes that implementing organizations can extend/modify the MAS as needed.

VIII. FUTURE PLANS

While no specific development efforts have been defined, there are several areas for MAS expansion that should be considered for future development, including expansion of MAS analysis capabilities, development of the MAAT for other CAD applications, and implementation of other analysis modules specifically intended to provide manufacturability feedback beyond cost estimation.

a. MAS Expansion for Additional Manufacturing Processes

The current MAS is limited to piece part analysis for machined, cast, sheetmetal, bar/tube, and plastic molded parts. Implementation at other industrial OEMs may require support for analysis of forgings, investment casting, and additively manufactured parts. Many of these part types are at least partially supported by aPriori, and additional manufacturing model (VPE) configuration to make these cost models compatible with the MAS manufacturing data model will be needed. Further, analysis models for more complex assemblies can be developed for specific industries, such as electrical/wire harness assemblies, hydraulic assemblies, etc. All future analysis capabilities will likely result in modifications to the MAAT CAD plugin for manufacturing data specification and the MAS web applications for display of manufacturing analysis feedback.

b. MAAT Implementation for Other CAD Applications

The Manufacturing Analysis Augmentation Tool (MAAT), used for manufacturing data specification and design submission to the Manufacturing Analysis System, was developed for a single CAD application, PTC Creo. The reason Creo was selected was due to previous development in the DARPA AVM program, as well as regular use at Oshkosh, the industrial OEM partner to participate in this project. However, the MAAT plugin could be re-developed for other CAD applications such as Siemens NX or Dassault Systèmes SOLIDWORKS using their available API or development toolkits.

c. Additional Manufacturability Analysis Modules

The MAS architecture is flexible in that additional analysis modules can be developed and implemented to provide more comprehensive feedback to design engineers. Examples of such modules/tools include the ANA manufacturability analysis software (DMDII 14-01-07), which evaluates concept design CAD geometry, generates visual feedback of manufacturability, and computes specific manufacturability metric scores. Implementation of such modules may require minor data interface modifications to the MAAT, the added analysis software, and possibly the MAS web application.

IX. CONCLUSIONS

The objective of this project was to mature and implement the DARPA Adaptive Vehicle Make (AVM) iFAB) manufacturing analysis software for an industrial design/manufacturing organization and demonstrate the technology for a product under development. The Manufacturing Analysis System provides model-based, automated analysis of designs with the ultimate goal of providing accurate cost estimates at various levels with a product's bill of material. The MAS combines streamlined manufacturing data specification within the CAD/design environment (MAAT), detailed should cost estimation using a variety of analysis models/tools, and product cost management and visual feedback in a web-based environment (MAS Web Application) for a diverse product development team.

The benefits of the MAS technology directly target key requirements identified by Oshkosh, the industrial partner on the project and target implementation organization. These requirements included more streamlined methods for manufacturing data specification, design submission for automated manufacturing analysis without leaving the CAD/design environment, cost rollups at multiple levels within a product's BOM, and product cost tracking throughout its design phase. A Use Case was conducted in which Oshkosh used the MAS tools for a product under development and compared key metrics against conventional manufacturing analysis and cost management methods. The use of the MAS resulted in an 89-97% reduction in effort to generate manufacturing data input, a 95-98% reduction in manufacturing analysis and cost estimation time, and an 81-97% reduction in total duration in achieving product cost rollups. Additional anecdotal feedback from Oshkosh indicated that designers greatly appreciated the simplicity of the input methods in MAAT within Creo, the MAS project cost visibility was extremely valuable for a multi-functional product development team, and the PLM integration greatly improved efficiency and reduced data management risks. While not demonstrated during the Use Case because the test product had not entered the manufacturing phase, it is anticipated that implementation of the MAS will decrease the number of ECRs by providing designers product cost feedback early in the design process and guidance on where design change should be explored to reduce overall product cost.

The MAS was developed such that medium to large design/manufacturing OEMs with capable IT organizations could implement the technology without the need for commercialization or extensive support. This is made possible with a very detailed System Specification Document with installation instructions, which is provided as a deliverable for this project. The web-based System Specification Document includes a detailed system overview and description of system elements, system/network diagrams, server setup recommendations and instructions, recommended/required specifications (hardware and software), and implementation tests. At the completion of this project, the IT organization at Oshkosh was reviewing the system specification and developing internal requirements for implementation, perhaps before the end of 2017. The System Specification Document and MAS source code is also available to all DMDII members in accordance with the terms of the Membership Agreement and US government organizations, where additional MAS implementation may be explored in 2018.

X. APPENDICES

a. User Guides

Manufacturing Analysis Augmentation Tool (Creo Plugin)

Contents

1.0 Purpose

The Manufacturing Analysis Augmentation Tool (MAAT) is a plugin application developed using the PTC Creo Development Toolkit that can be added to a user's PTC Creo environment. Its primary purpose is to guide designers in specifying the minimum set of manufacturing information required for the Manufacturing Analysis System to conduct a detailed cost analysis. The MAAT can be instantiated on any level of the CAD product structure (part or assembly). When run, a geometric reasoning algorithm is executed to not only identify the piece parts in the design, but also identify the assembly seams, defined as part-to-part, part-to-assembly, or assembly-to-assembly interferences that often imply a manufacturing joining operation.

The MAAT plugin allows users to log in and select an associated cost project, retrieve manufacturing data from previous submissions to that cost project, submit additional analysis requests to the MAS, and change the associated cost project. This is all made possible through HTTP requests to the MAS web application (discussed in the next section) using a REST (Representational State Transfer) API.

2.0 Procedures

The instructions in this manual assume that the user has installed PTC Creo 2.0 and has a valid MAS user account with the appropriate permissions that allow access to cost projects and submissions to these projects.

2.1 Installation

Ensure that the user has sufficient privileges to install software on their computer. The minimum system requirements are:

> Windows 7, 32- or 64 bit
> 8 GB RAM (or more is recommended)
> Creo2 Parametric previously installed

Uninstall any previous versions of the MAAT, as shown in Figure 1.

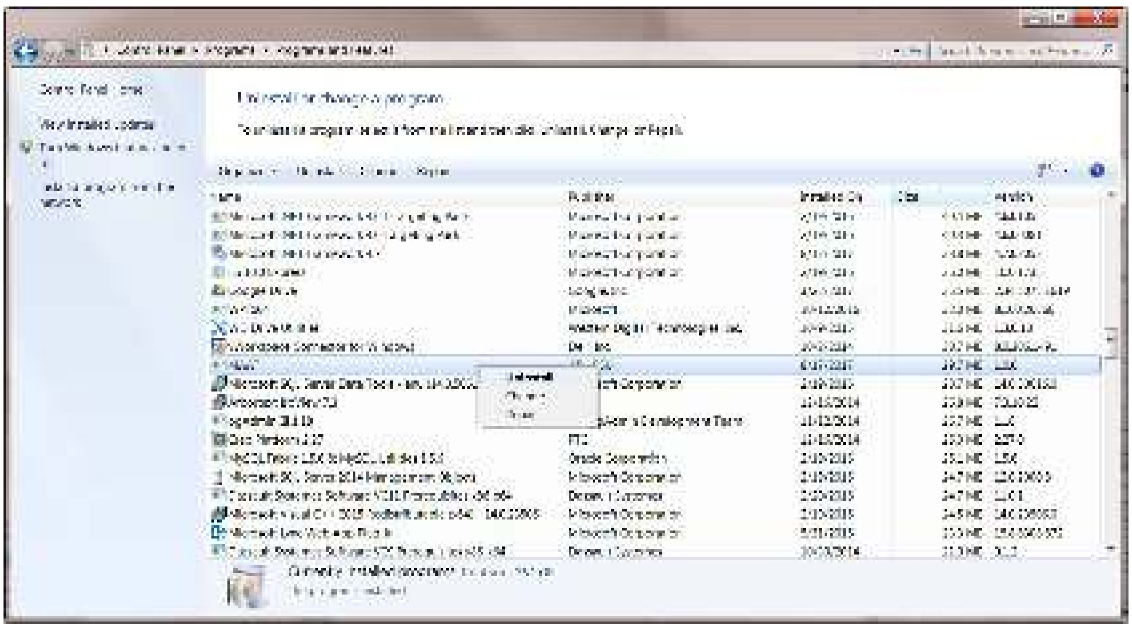


Figure 16. Uninstall any previous versions of the tools

Install the MAAT software by executing the maat-installer.msi (MAAT Installer), as shown in Figure 2.

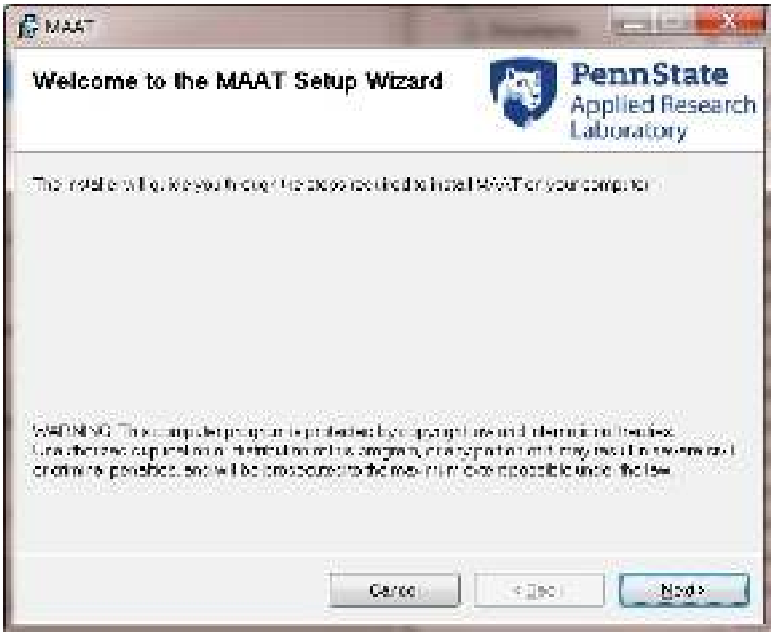


Figure 17. Run the maat-installer.msi

Specify the installation directory (recommended to be one where the user has both read and write access), as shown in Figure 3. Also specify whether MAAT should be made available for the current Windows user (*Just Me*) or all users (*Everyone*).

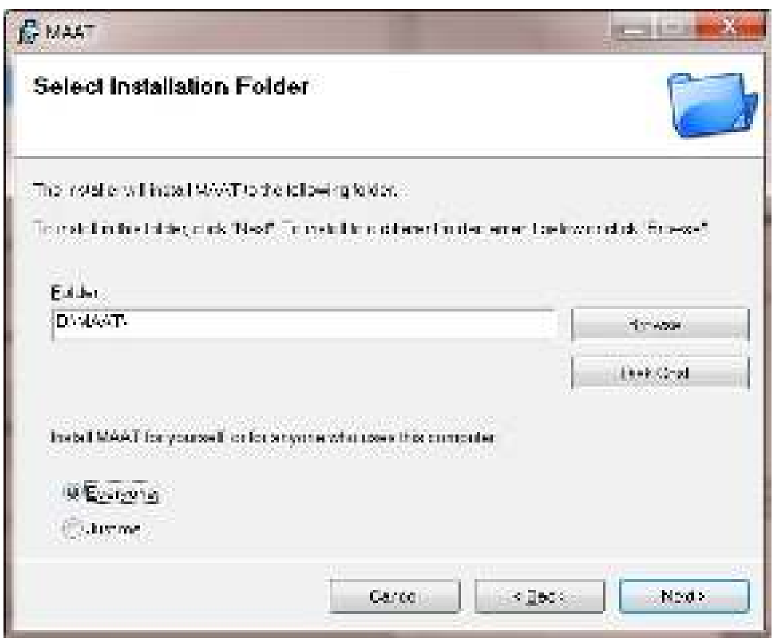


Figure 18. Specify installation directory

The resulting installation directory should look like the one shown in Figure 4.

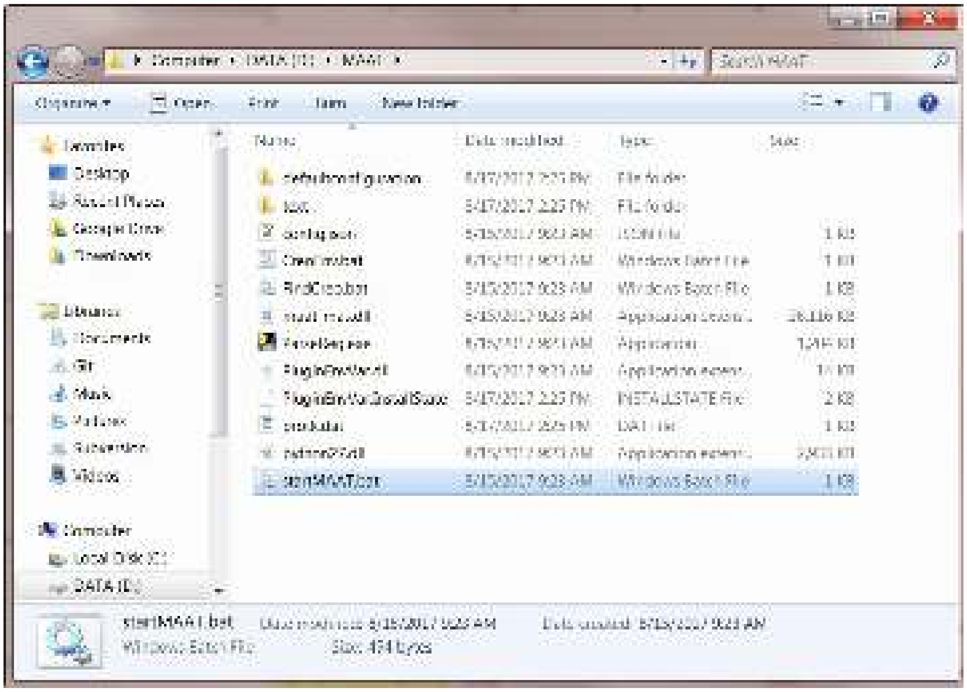


Figure 19. MAAT Installation Directory

2.2 Plugin Activation in Creo

After the MAAT has been installed, there are three different ways that it can be loaded into the Creo environment. The first method to launch the MAAT, is to run the ***startMAAT.bat*** file (highlighted in Figure 4). Creo Parametric 2.0 will be located and Creo will launch as normal, with the MAAT plugin loaded as an Auxiliary Application

The second method to start the MAAT is to open Creo Parametric as the user normally would and then enter the ToolsAuxiliary Application functions. In the *Auxiliary Applications* the user can select Register and select the protk.dat file that was installed in the MAAT Installation directory. ($maat_installdir\protk.dat). When prompted, select “mas-maat” and the select Start to add the plugin into the Creo environment.

The third method to launch the MAAT is to register the plugin into the user’s default Creo configuration. After opening Creo as the user normally would, select FileOptionsConfiguration Editor. Click Add… and specify the name as “protkdat”. The user will then be able to click Browse and select the protk.dat file that was installed in the MAAT Installation directory (see Figure 5).

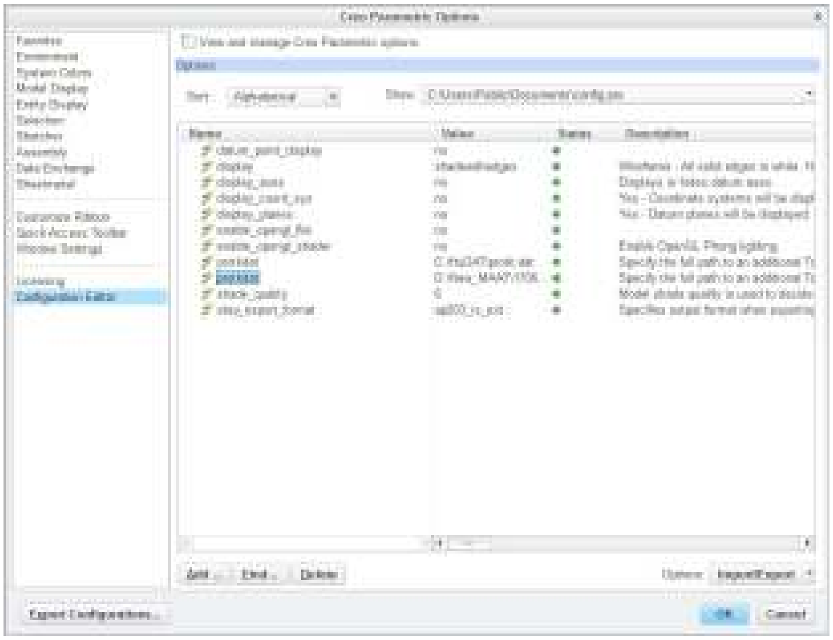


Figure 20. Add the MAAT to Default Configuration

Click Export Configurations… to save the configuration with the added MAAT plugin. Each time Creo is launched (from .\PTC\Creo 2.0\Parametric\bin\parametric.exe), the MAAT plugin will be active for that session. For third method to work, the user must edit the “exec_file” line in the protk.dat file (in the MAAT) installation directory such that the file path is absolute and not relative.

2.3 MAAT Operation

To begin the manufacturing augmentation data process using the MAAT and submit jobs to the Manufacturing Analysis System (MAS), open any Creo part or assembly as the user normally would, as shown in Figure 6.

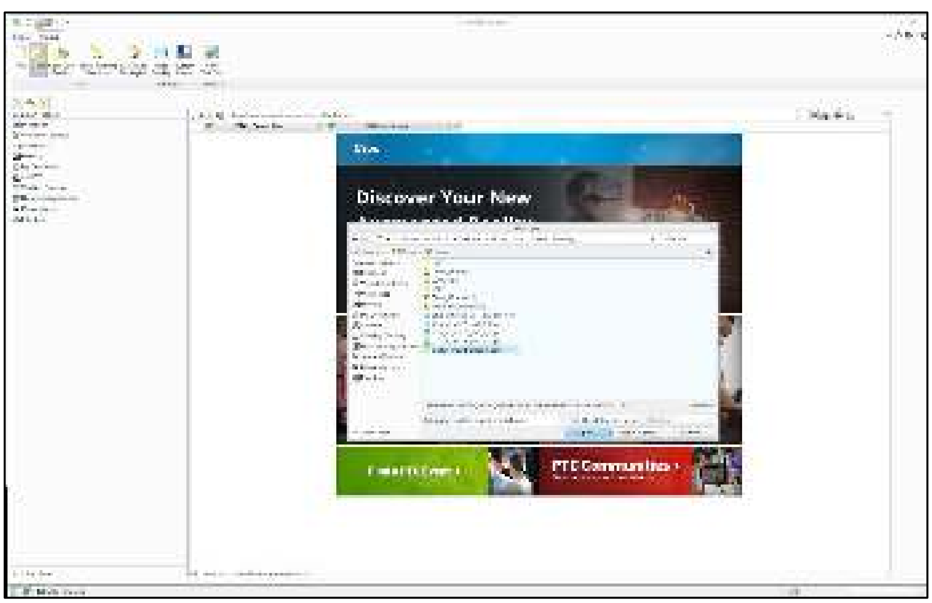


Figure 21. Open a Creo assembly

To begin manufacturing data augmentation process, select a part or assembly from the Model Tree, right-click a part or assembly and select *MAAT*.

The user will initially be prompted to provide login credentials. This is to allow for direct submission to the MAS for manufacturability assessment and cost estimation. Enter the user's MAS credentials and click the Login button, as shown in Figure 7[1]. **Note: the user must click on the login button to proceed.** When the user does this, the MAAT will analyze the assembly structure, identify the individual piece parts, and determine assembly seams (i.e., physical contacts between parts in the model tree). The main MAAT interface will then appear, as shown in Figure 9.

Figure 22. Login for the MAAT Plugin using MAS Credentials

After logging in, the user will be prompted to select a Cost Project to associate with (see Figure 8). The project list is provided based on the user's subscriptions to those projects in the MAS web application, typically handled by an administrator or project lead.

Figure 23. MAS Project Selection in the MAAT

Select the correct project (i.e., assembly). It is recommended to keep the "download latest from server" option checked. This will do two things:

1) Any manufacturing data submissions conducted by other users (and submitted to the MAS) will be pulled to save users from duplicate entry and inconsistent data specification
2) Cost data on existing parts (purchased COTS or vendor fabricated parts with past cost data associated with them stored in the MAS database) will be extracted, saving designers from having to look up purchased part cost information

If this is the first time the user is launching the MAAT on an assembly, they will also be prompted as shown in Figure 9. It is recommended this first time to select No to ensure that the MAAT identifies all assembly seams (i.e., part-to-part interfaces) for the current version of the assembly. Any seam data

[1]The user must click on the Login button to proceed. In addition, the End Point refers to the web application associated with the Manufacturing Analysis System. This URL can be modified in the config.json file in the MAAT installation directory (consultation with your IT organization is strongly recommended to confirm web application URL).

that exists from previous MAS submissions will be pulled into the MAAT session after the seam detection is run.

Figure 24. Run Seam Detection Prompt

For assemblies that have been previously specified in the MAAT tool and submitted to the MAS, the user may not want to have the MAAT conduct the assembly seam detection (this process can take large amounts of time for very large and complex assemblies). In particular, if the user is certain that there have been no design changes (i.e., product restructuring, new parts, etc.), then assembly seam detection can be avoided by selecting No for the prompt shown in Figure 10.

Figure 25. Design Change Prompt

The MAAT user interface is shown in Figure 11. There are two primary views in the MAAT, as indicated by the tabs across the top: Parts and Assembly Seams.

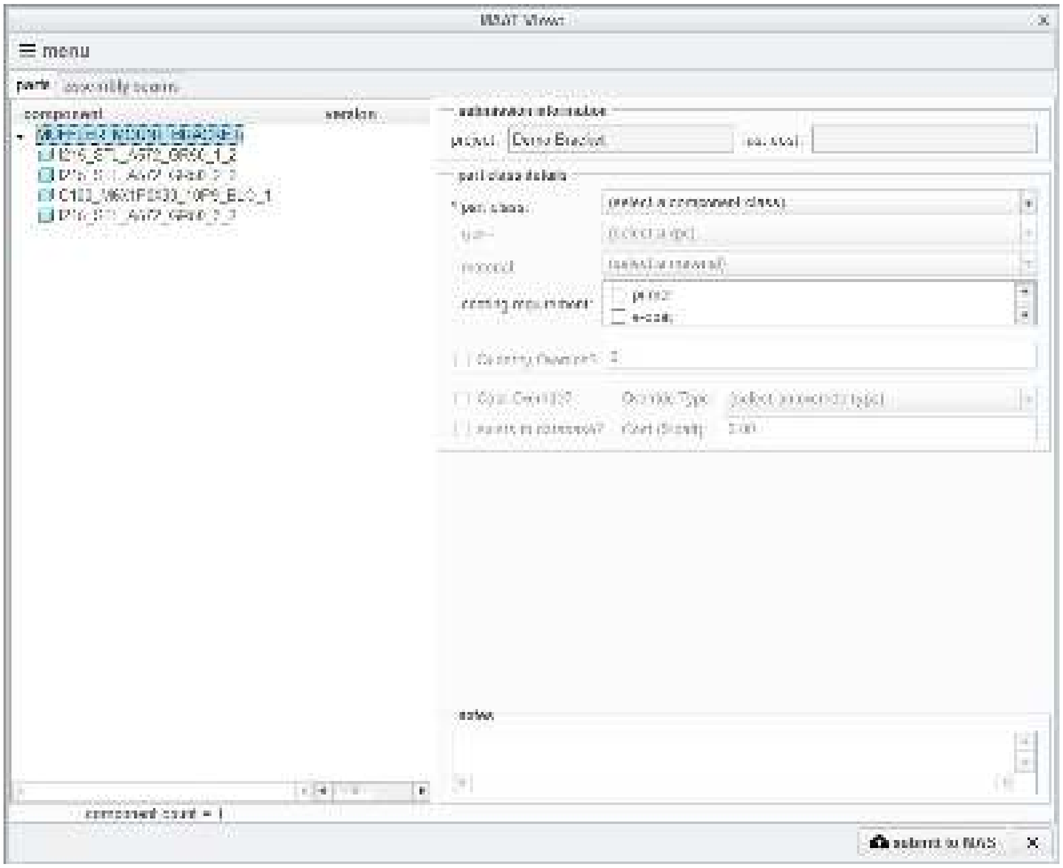


Figure 26. MAAT User Interface (example shown for assembly in Creo model tree)

2.3.1 Parts View

When in the Parts view, notice how the product structure closely mimics the model tree in the primary Creo view. This is illustrated for an assembly of medium complexity (i.e., multiple product structure layers) in Figure 12. The difference between the views is that duplicate parts in the Creo assembly tree will only show up one time in the MAAT Parts product structure. When selecting a part or assembly, all instances of that part or assembly are highlighted in the 3D model and the model tree. Selecting a part or assembly in the MAAT product structure will also display the component count at the bottom of the MAAT user interface.

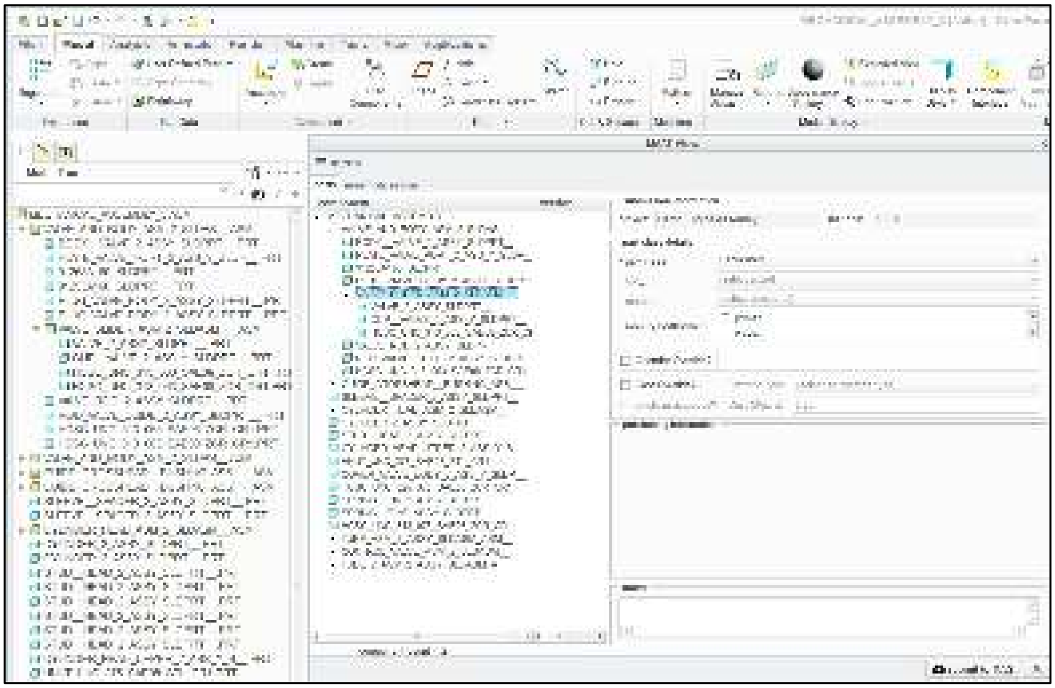


Figure 27. Comparison of the MAAT Product Structure in Parts View to Creo Assembly Tree

Users will need to provide manufacturing data specification for all parts and assemblies in the product structure. The required data is dependent on the type of part or assembly, as described in the following sections.

2.3.1.1 Assembly Data Specification in Parts View

In the parts view, users need to specify the *part class* for assembly from the following options that are available when the user selects the assembly in the MAAT user interface: Fabricated, Purchased, or Rollup. Each are described below:

> <u>Fabricated Assembly</u>
> Users will select Fabricated Assembly for assemblies that will have their assembly seams fully defined (discussed in section 2.4). The costing of these assemblies will include the rollup of piece part costs but also the cost associated with the assembly seam definition.
>
> <u>Purchased Assembly</u>
> Users will select Purchased Assembly for assemblies that do not need their lower level parts or assembly seams specified and analyzed. If selected, users will need to complete the "purchasing information" (only price (USD) is required for submission and analysis).
>
> <u>Rollup Assembly</u>
> Users will select Rollup Assembly for assemblies that will not require their assembly seams to be fully defined. The costing of these assemblies will include the rollup of all lower level piece part and assembly costs but will not include additional cost for the assembly processes.

2.3.1.2 Part Data Specification in Parts View

The bulk of manufacturing data specification in the Parts view is completed for piece parts. Piece parts are either defined as a fabricated part (Casting, Machined, Pipe/Bar/Tube, Plate/Sheet, or Plastic

Molding) or a purchased part. These part classes can be selected in the MAAT for a selected part. The required manufacturing data for each Part Class is described below.

Fabricated Parts – see Figure 13

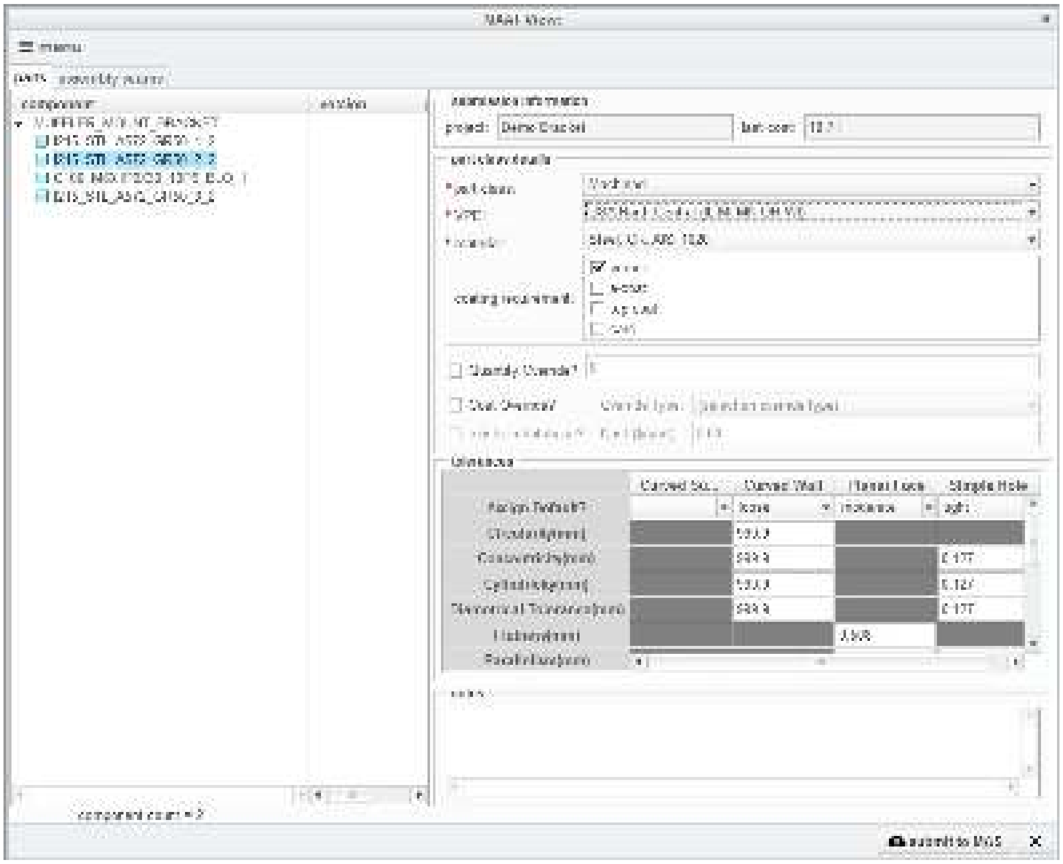


Figure 28. Fabricated Part Data Specification in the MAAT

The following manufacturing data fields are available for Fabricated Parts (* denotes required):

Part Class* – select from *Casting, Machined, Pipe/Bar/Tube, Plate/Sheet, or Plastic Molding*

VPE* (Virtual Production Environment) – select the geographically-based cost model to use when analyzing the part

Material* - select material from a list of available materials in the cost model. Material can be automatically populated if a Creo parameter mapping is established in the \bin\defaultconfiguration\maat_part_classes.xml file. For example:

<material id="4150425" displayName="SPEC,AISI 1018 CR BAR" aPrioriName="Steel, CR, AISI 1020" />

Where *material id* refers to the MATL parameter on the Creo part, display name refers to the material names listed in the MAAT material drop down, and aPrioriName refers to the material name in the cost model.

Coating Requirement – select from *primer, e-coat, top coat, carc,* or no coating. Parts can have multiple types of coating (e.g., primer & top coat).

Quantity Override – MAS cost projects manage the total quantity of a particular part in a BOM when estimating cost. For early concept design, where it is unknown where else a particular part may be used in the design, the designer may choose to override the quantity and explicitly select the number of the selected part to be fabricated. Estimated costs include amortized non-recurring engineering costs based on the total quantity (either derived from the total BOM or using the quantity override).

<u>Cost Override</u> – Designers can directly input a cost for a part if they prefer (rather than have the MAS estimate a cost). When selecting this option, the user must select the type of the cost override (manually costed, vendor quoted). If fabricated parts are contained in the Existing Parts database, and the user selected to pull manufacturing data from the server, the "exists in database" option will be selected and the retrieved part cost will be displayed.

<u>Tolerances</u> – users can specify general tolerances for parts depending on the selected part class. Possibly tolerance types include circularity, concentricity, cylindricity, diametrical tolerance, flatness, parallelism, perpendicularity, positional tolerance, roughness, runout, straightness, and threaded. Individual tolerances can be specified for the following part features: curved wall, curved surface, planar face, simple hole, and complex hole. When specifying a value for a tolerance type for a feature type, all detected features of that type will assume that tolerance value.

<u>Fabricated Parts – see Figure 14</u>

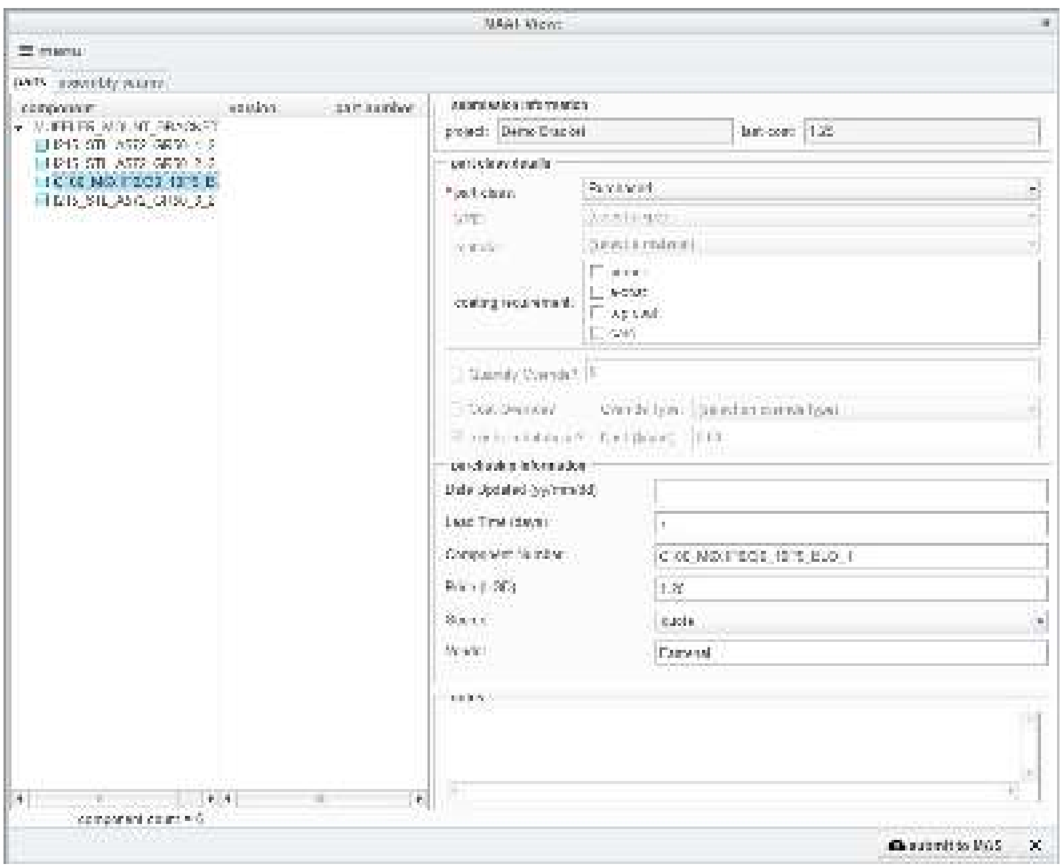


Figure 29. Purchased Part Data Specification in the MAAT

The following manufacturing data fields are available for Purchased Parts (* denotes required):

<u>Coating Requirement</u> – select from *primer, e-coat, top coat, carc,* or no coating.

<u>Date Updated</u> – *e*nter date that purchased part cost was obtained.

Lead Time – enter the number of days to obtain the purchased part
Component number – enter the part number for the purchased part. This could be a vendor's part number for a custom assembly or a catalog part number for a commercial off the shelf (COTS) item
Price* – enter the per part price.
Source – select from estimate, quote, or last cost.
Vendor – enter in the vendor for the purchased part.

2.3.2 Assembly Seams View

For fabricated assemblies, users are required to specify information about how parts are attached (e.g., mechanically, welded) in order for assembly time and associated cost to be calculated by the MAS. The MAAT software guides users by geometrically reasoning about a selected assembly and detecting the "assembly seams". Assembly seams are the part-to-part, part-to-assembly, and assembly-to-assembly interfaces that may imply a manufacturing joining operation, and are determined based on the CAD assembly product structure, which is assumed to be a manufacturing BOM.

When in the Assembly Seams view in the MAAT, the assembly product structure is again displayed on the left pane, but in this case, only assemblies are displayed. For assemblies with only parts at the next lowest level, the MAAT will only display a single assembly in the product structure. For more complex assemblies with multiple levels of subassemblies and parts, the MAAT will display the assembly structure only. See Figure 15 for an example of the MAAT detected assembly structure.

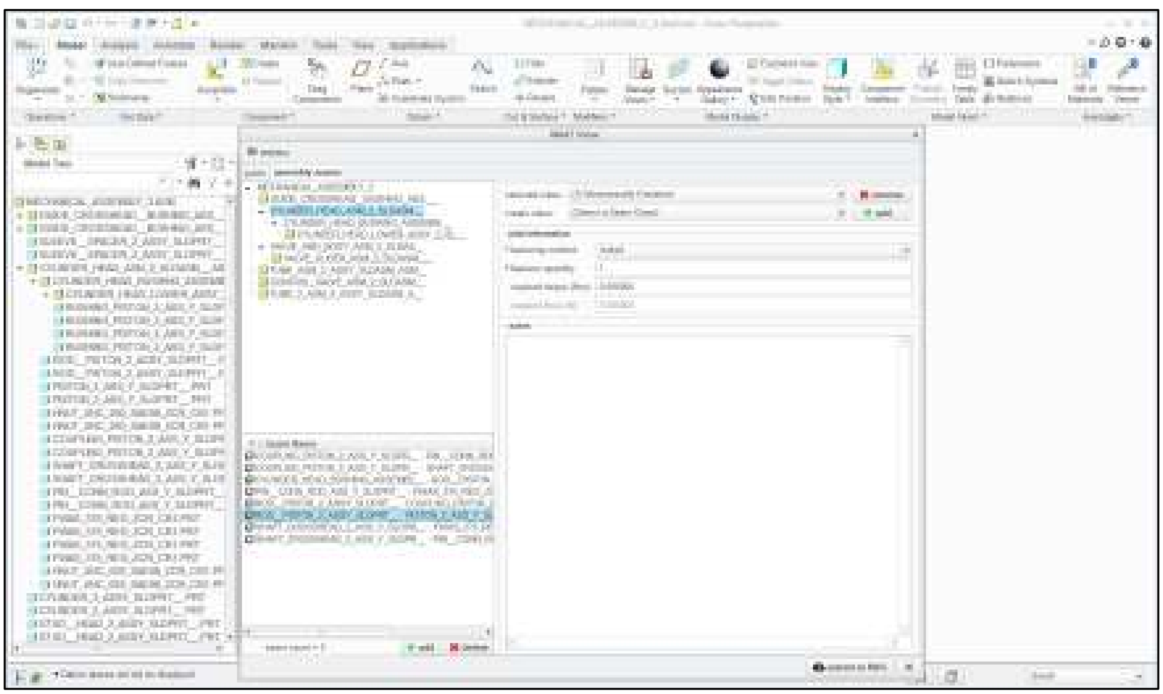


Figure 30. Comparison of the MAAT Product Structure in Assembly Seams View to Creo Assembly Tree

When selecting an assembly or subassembly from the MAAT product structure in the Assembly Seams view, the associated detected assembly seams are displayed in the lower left pane (displayed as *Part/Assembly1 – Part/Assembly2*). When selecting a specific assembly seam, the two parts (or assemblies) making up that seam are colored orange and blue in the 3D model (this occurs for all instances of that seam).

An assembly class (*selected class*) and associated manufacturing data needs to be provided for each assembly seam. To add an assembly class, select from Mechanically Fastened, Welded, or Incidental Contact next to *create class* and click *add*. The required manufacturing data is discussed below (* denotes required).

Mechanically Fastened

Fastening Method* - select from bolted, bolted (blind), machine screw, snap fit, press fit, crimp/clamp fit

Fastener Quantity* – specify the quantity of fasteners needed (for bolted, bolted (blind), machine screw). Note that when multiple of the same assembly seam are detected, the user should specify the quantity for one instance of that assembly seam.

Welded

Weld Type* – select from seam or stitch. If stitch selected, enter the bead length and pitch values

Joint Length* - specify the length of the welded joint

Two-Sided? – specify whether or not the weld is a single-sided joint or a double-sided joint

Joint Type* - select from fillet or groove

Weld Size* – select the size of the weld (assumed to be leg length for fillet, or weld depth for groove)

Weld Material* – select the base material of the parts making up the weld

Incidental Contact

There are cases when an assembly seam is detected, but there is really no joint mechanism that needs to be defined. For instance, a part may be in in contact with another part but the parts are not fastened, welded, or bonded to each other.

2.3.2 MAAT Main Menu Options

The MAAT plugin is a simple user interface with only one primary user menu. The options under the main menu are discussed below:

save – While manufacturing data submissions to the MAS stores manufacturing data specification on the MAS database, available for all users with permissions to retrieve this information, some users may select to locally save the MAAT manufacturing data specification before they are ready to submit to the MAS (e.g., manufacturing data specification for an assembly may be incomplete). A MAATData folder is created in the users Creo working directory (or the MAAT installation directory if the user does not have write permission to the working directory), where manufacturing data specification is saved to xml files for retrieval in future sessions with the MAAT.

mfg class - executes a validation check to ensure that the user has provided all required manufacturing data input for parts and assemblies in the Parts view.

assembly detail - executes a validation check to ensure that the user has specified an assembly class for every identified assembly seam and that all required manufacturing data input is specified.

change project – the user can change the project to which the current MAAT session is associated.

submit – submits the part or assembly (whichever was selected when running the MAAT) to the MAS for manufacturing analysis and cost estimation. Analysis results can be viewed in the MAS web application (discussed in a separate user guide). It should be noted that users can also submit to the MAS for analysis by clicking the *Submit to MAS* button, which is always available on the lower right of the main user interface.

logout – logs the user out from the MAS connection and closes the MAAT plugin.

isolate components – when selected, all components in the assembly will be hidden except the part or assembly selected in the product structure in the Parts view. This isolation also occurs for seams selected when in the Assembly Seams view.

2.3 MAAT Configurations

The MAAT tool can be configured to accommodate various organizations' design techniques, preferences, and manufacturing models (i.e., aPriori VPEs). These configurations are all saved in XML files (.\bin\defaultconfiguration\) that can be distributed to all users to ensure consistent use of MAAT. Each configuration is described in the next four sections.

2.3.1 MAAT Part Classes

The maat_part_classes.xml configuration file defines default values and displays for the MAAT for the fabricated part classes. Each part class specifies an *id*, which maps to the aPriori cost model process group name, and a *displayName*, which is what the user sees in MAAT. The user should only consider editing the *displayName* if needed. Editing the *id* will corrupt the submissions files to the MAS and prevent analysis.

```
<partClass id="Stock Machining" displayName="Machined">
```

Each part class has features defined. These are set based on what features are detected by the aPriori cost models and should not be edited.

```
<feature name="Planar Face">
```

Each feature then has a list of tolerance types that can be specified in the MAAT. For each tolerance type specifies a name (aPriori specific; do not edit), the MAAT *displayName*, unit, *defaultValue* (999.9 indicates that no specific tolerance will be applied to the feature when costing the part), *highValue* (the tolerance setting when the user selects "loose" in MAAT), *medValue* (the tolerance setting when the user selects "moderate" in the MAAT), and *lowValue* (the tolerance setting when the user selects "tight" in the MAAT). The user is encouraged to modify the default, low, med, and high values, but all other settings should be unchanged

```
<tolerance name="Stock Machining/Machining:planarFaceFlatnessSMValue" displayName="Flatness(mm)"
unit="mm" defaultValue="999.9" highValue="999.9" medValue="0.508" lowValue="0.127"/>
```

Each part class has a material filter specified, which indicates which Creo .prt parameter the MAAT application will reference when attempting to extract a specified material attribute. The user can modify this materialFilter setting.

```
<materialFilter>MATL</materialFilter>
```

The configuration then lists all available materials that can be specified for the part class. Each material specifies an *id* (which is what cross references the Creo part parameter specified in the materialFilter), the *displayName*, and the *aPrioriName* (material name in the aPriori cost models used by the MAS). The user can modify and add materials to the list, but should be careful to select an aPrioriName that they are certain exists in the aPriori cost model.

2.3.2 MAAT Seam Filters

The purpose of the maat_seam_filters.xml configuration file is to allow users to specify specific types of parts that should be ignored and not considered for the MAAT assembly detection. For example, users often find it cumbersome and not useful for seams to be detected between fasteners and other parts, or between fasteners and other fasteners (e.g., washer to a bolt or nut). The MAAT can ignore parts based on a Creo part parameter setting.

The user must add a parameter filter. This tells the MAAT which parameter to look for on each part when conducting assembly detection. The user can modify the default filter parameter. The user then can specify which values that parameter can contain for the part to be ignored for seam detection. There can be many *contains* values specified.

```
<filter parameter="PART_NAME">
        <contains>nut</contains>
        <contains>bolt</contains>
        <contains>washer</contains>
        <contains>screw</contains>
        <contains>fastener</contains>
        <contains>rivet</contains>
        <contains>pin, flat head clevis</contains>
        <contains>cotter pin</contains>
</filter>
```

The user can also set additional filter parameters, such as the following:

```
<filter parameter="IS_FASTENER">
        <contains>true</contains>
</filter>
```

2.3.3 MAAT Tree Parameters

The Parts and Assembly Seam product structure view in the MAAT plugin can display Creo part parameters as defined in the maat_tree_parameters.xml configuration file. Too add the parameter columns to the product structure views, the user needs to define the *displayName* (column header in the MAAT) and the *parameter* (the Creo part parameter).

```
<treeParams>
        <partTree>
                <treeParam>
                        <displayName>version</displayName>
```

```
                    <parameter>PTC_WM_VERSION</parameter>
                </treeParam>
                <treeParam>
                    <displayName>part number</displayName>
                    <parameter>PART_NUMBER</parameter>
                </treeParam>
            </partTree>
            <assemblyTree>
            </assemblyTree>
        </treeParams>
```

2.3.4 MAAT VPEs

The cost models used by the MAS to conduct manufacturing analysis and cost estimating are associated with a commercial software application called aPriori. Custom cost models, called Virtual Production Environments (VPEs) can be developed and specified when conducting a manufacturing analysis. The purpose of the maat_vpes.xml configuration file is to define the list of available aPriori VPEs that are compatible with the MAS. The user needs to define the *name* and the *displayName* in the configuration for each VPE.

```
<vpe name="aPriori USA DMDII MAS" displayName="USA" />
```

Manufacturing Analysis System Web Application – User Guide

3.0 Purpose

A web-based application was developed as a central repository for project cost information. The Ruby on Rails (Rails) framework establishes a connection between the designer community using the MAAT application in Creo and the backend Manufacturing Analysis System that performs the actual manufacturing analyses. The web application supports user groups with varying roles, including designers that are consistently using MAAT to submit designs for cost analysis and tracking cost trends based as designs mature and chief engineers/managers that are primarily interested in project cost roll-ups at various levels of detail in the BOM. The MAS web application includes a built-in user management capability that allows organizations to add authenticated users and adjust their site permissions based on their role. Additional permissions may be set for users for specific projects.

2.0 Procedures

The instructions in this manual assume that the MAS web application has been properly configured and instantiated following the procedures in the MAS system specification. Instructions are separated into Primary User Instructions, which includes the primary functions of the MAS web application, and Administrator Instructions, which include functions that not all users will use, including the ability to create MAS user accounts.

2.1 MAS Web Application Primary User Instructions

Primary User Instructions are presented based on key MAS web application functions, including Login and Main Application Views, Cost Project Startup, Cost Project Summary, and Part and Assembly Cost Details.

2.1.1 Login and Main Application Views

Using a web browser (Chrome or Firefox strongly recommended), go to the MAS web application URL provided by an administrator. For the purposes of this user guide, please assume the following URL: https://avm-osh-win-srv.arl.psu.edu. The user should enter in their username and password when prompted, as shown in Figure 1.

After logging in, the user will be able to click on the My Projects button on the left panel or the Projects button in the middle of the main page. Cost projects are discussed in greater detail in the section 2.1.3.

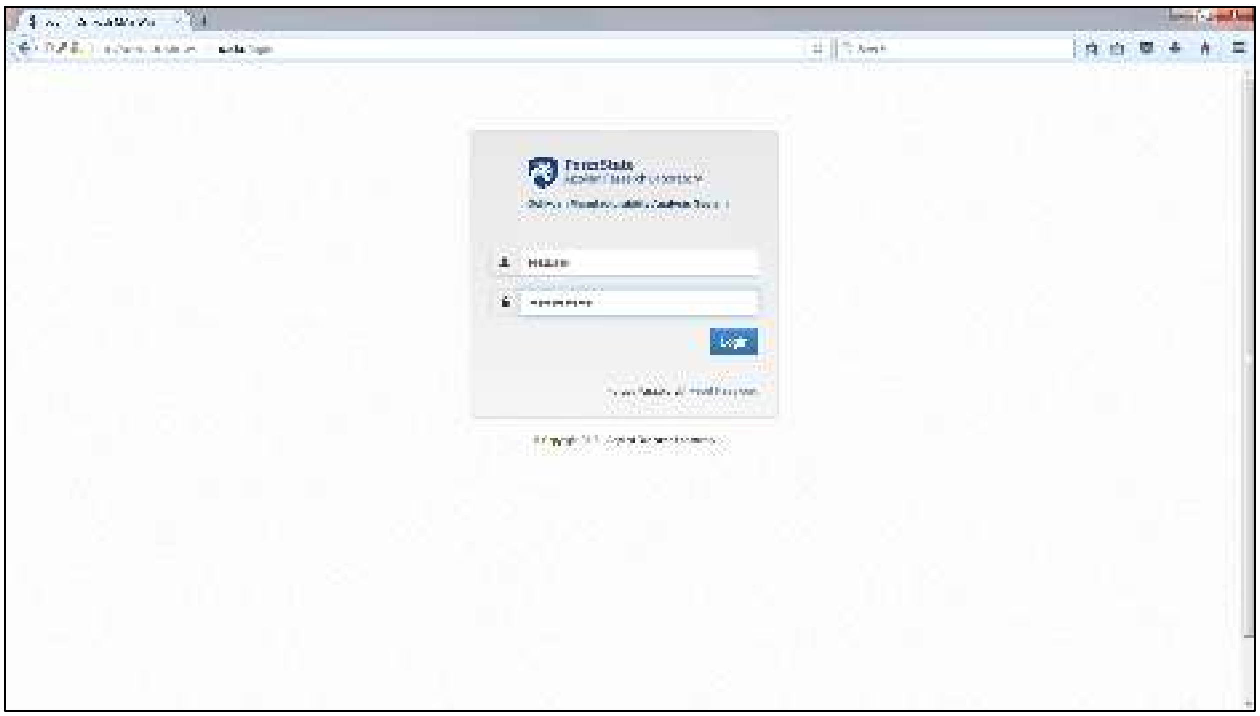


Figure 31. MAS Web Application Login

2.1.2 Cost Project Startup

The use of the MAS for manufacturing analysis (triggered by design submissions from the MAAT Creo plugin application) requires a Cost Project to be created. Only users with a MAS web application site role of Project Manager or Administrator can create cost projects (site roles are discussed further in section 2.2.1.2). Click on + New Project to start a new cost project. This will bring up the project startup form, as shown in Figure 2.

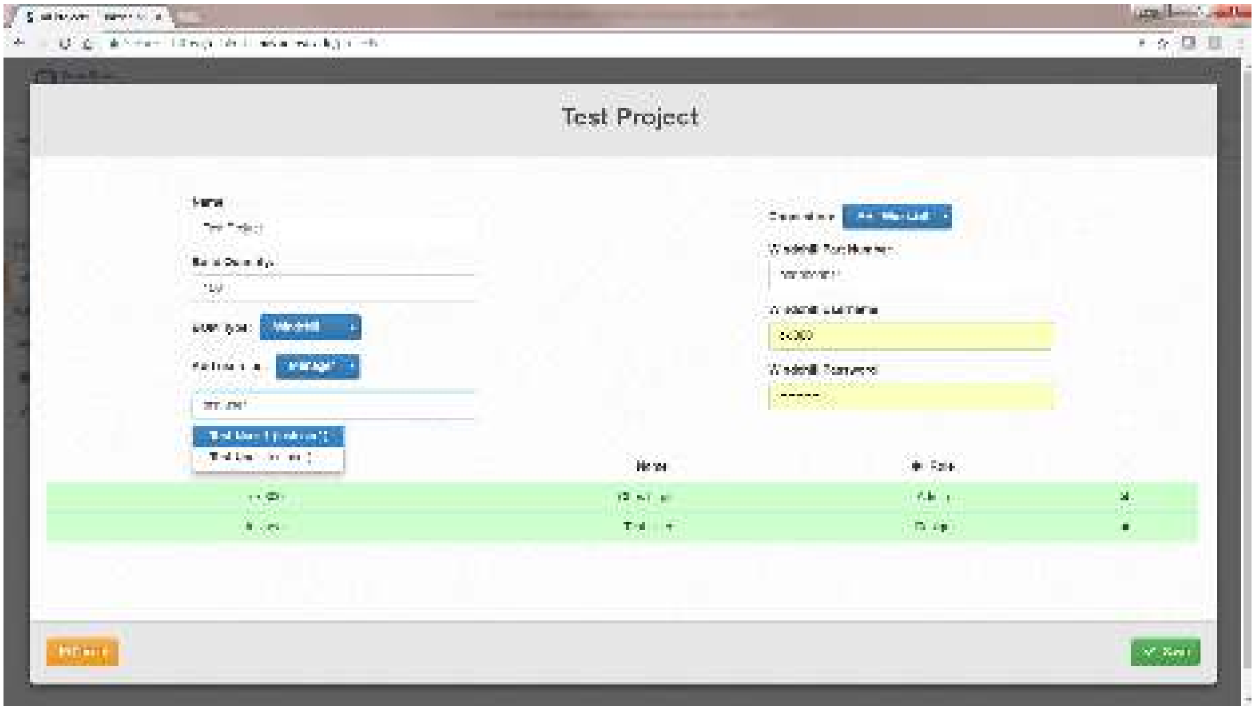


Figure 32. MAS Web Application Cost Project Startup

Complete the following information:

Name – the project name will be displayed in the MAS web app (Cost Projects view) as well as in the MAAT (Creo plugin) application

Build Quantity – this is assumed to be the annual quantity of the top level product for the project

BOM Type – select from Windchill or Standalone. When selecting Windchill, the user will be required to specify the Windchill server to connect to, enter in a Windchill part number for the top level assembly, and provide their Windchill user credentials. When selecting Standalone, the user will be required to browse for a BOM Excel file (this is discussed further below)

Add User As – users can be added as 1 of 4 different roles, as shown in Table 1 below. Select the preferred role for the user, and begin to type the user's name into the field to bring up potentially matching users. Select the user, and they will be added to the users table at the bottom of the Cost Project startup page.

Table 5. Cost Project Roles

	Can view project analysis	Can submit products	Can modify user project permissions	can edit project (e.g., build quantity, BOM update)	can select parts (i.e., include / exclude parts for cost rollups)	can delete project
Guest	X					
Design	X	x				
Manager	X	x	x	x	x	
Admin	x	X	X	X	X	X

Connection – Select from a list of preconfigured Windchill servers (only when Windchill is the specified BOM type)

Windchill Part Number – Enter the Windchill part number for the top level assembly for which the user wants to pull the BOM

Windchill Username/Password – Enter the Windchill user credentials to allow the MAS Web Application to interface with the Windchill server to pull the BOM information

+ Select File – Browse for the BOM Excel file. The BOM Excel file must be in the format of the sample BOM displayed in Table 2

Table 6. Standalone BOM Excel Format

Structure Level	Number	Name	Version	Quantity	Unit
0	valve_and_body_asm_2_sldas__	Valve and Body Assembly	G.2 (Design)	1	each
1	body__valve_2_assy_sldprt__	Body Valve	G.2 (Design)	1	each
1	plate_valve_port_2_ass_y_sldp__	Plate Valve Port	G.2 (Design)	1	each
1	91253a160_sldprt__	Cap Screw	G.2 (Design)	2	each
1	plug_valve_body_2_assy_sldprt__	Plug Valve	G.2 (Design)	2	each

Click Save to create the Cost Project. The project should now be visible in the Projects page as well as the Projects pages of any users that were added to the Cost Project, as shown in Figure 3. Cost Projects can be later edited by clicking the button in listed next to the project if the user has Manager or Admin access to the project.

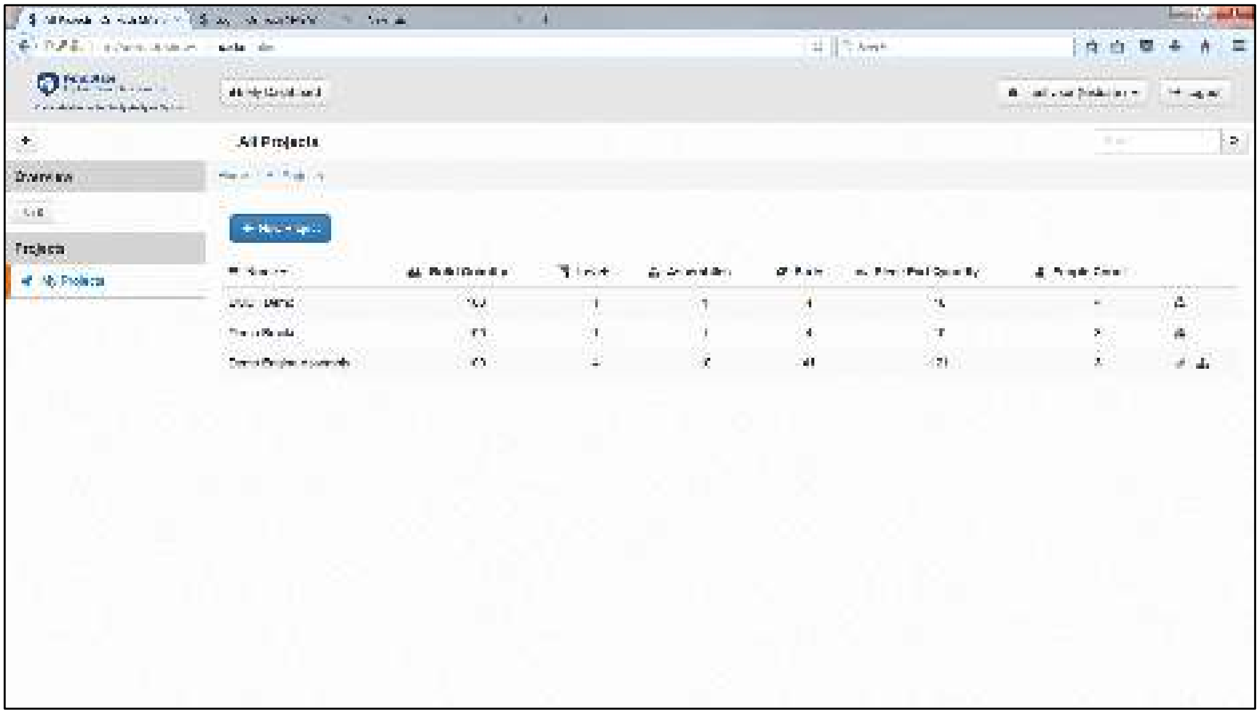


Figure 33. MAS Web Application Cost Projects View

Cost Project summary information is provided in the Projects view, as specified below:

Name – Cost Project name specified at project startup
Build Quantity – annual production quantity specified at project startup
Levels – number of levels in the assembly BOM
Assemblies – number of unique assemblies in the assembly BOM
Parts – number of unique parts in the assembly BOM
Piece Part Quantity – total number of piece parts in the assembly BOM, considering multiples of any part
People Count – number of MAS users added to the cost project.

2.1.3 Cost Project Summary

Click the button located next to the project to view the current analysis and cost results for that Cost Project. This will display a product structure view of top level assembly, as shown in Figure 4.

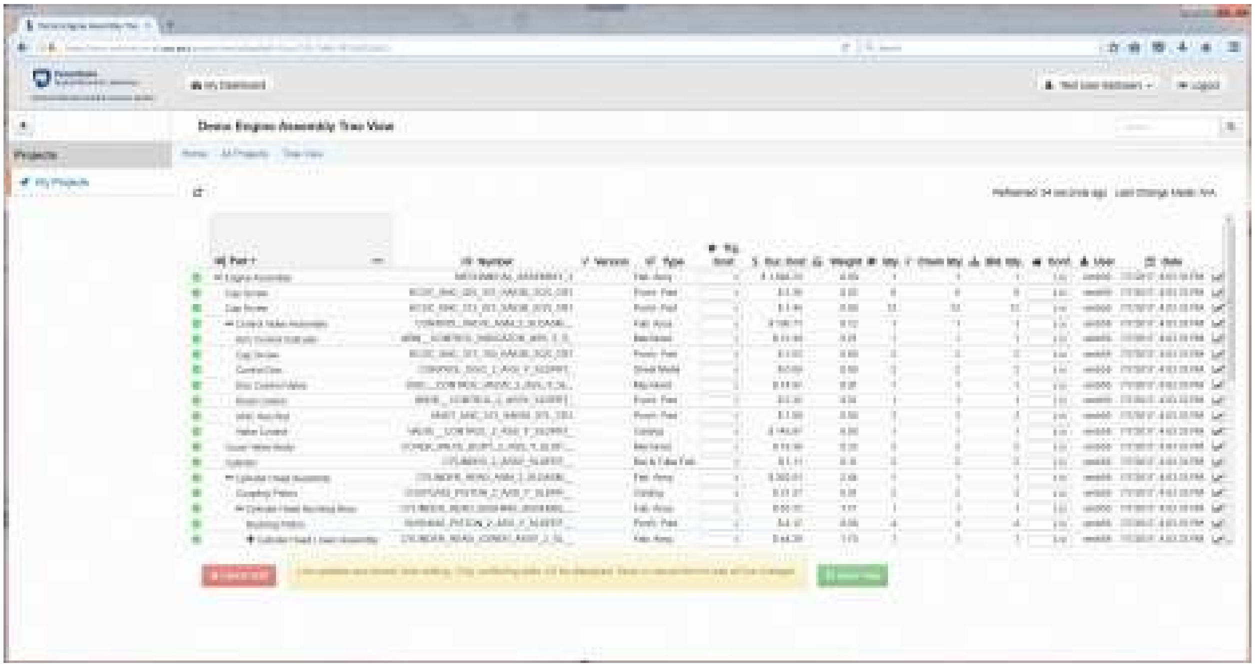


Figure 34. Cost Project Summary View

The following information is displayed in the summary product structure view:

Part – associated with the part name in the loaded BOM
Number – associated with the Creo Part number
Version – version of the Creo Part or Assembly when design was submitted to the MAS (only displayed if the default PTC_WM_VERSION parameter is populated in the Creo Model and was able to be retrieved and included with the design submission from MAAT)
Type – the type of part or assembly; maps to the part class specified in MAAT
Cur. Cost – result of the latest manufacturing analysis / cost estimate for the part or assembly.
Weight – if the PRO_MP_MASS parameter is set on the Creo Part, this mass will be displayed in the Cost Project Summary for each part. Assembly weights are simply a roll-up of lower level part and assembly weights.
Qty – the quantity of the particular part or assembly in its direct parent assembly
Build Qty – the total quantity of the particular part or assembly in the total BOM
Conf – users can manually add in a confidence in the cost analysis for each part or assembly
User – the username of the MAS user that last submitted the part or assembly to the MAS
Date – date of the last MAS submission for the part or assembly

2.1.4 Part and Assembly Cost Details

For any part or assembly in the Cost Project summary product structure, click on the button to display the cost details. Cost details vary based on the part class, as shown below.

<u>Assemblies (Rollup or Fabricated)</u>

When viewing assembly cost details, the user is presented with a cost breakdown by Purchased Parts, Manufactured Parts, and Assembly, as shown in Figure 5.

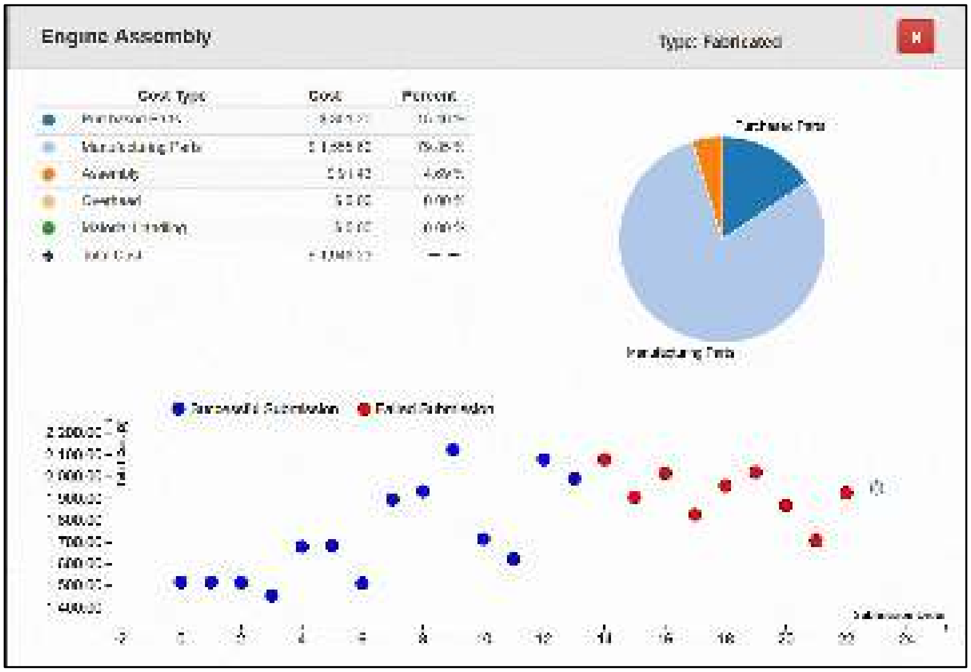


Figure 35. Assembly Cost Details and Tracking

<u>Parts (Fabricated)</u>

When viewing fabricated part cost details, the user is presented with a cost breakdown by Material, Labor, Overhead, Setup, Investment, and Other, as shown in Figure 6.

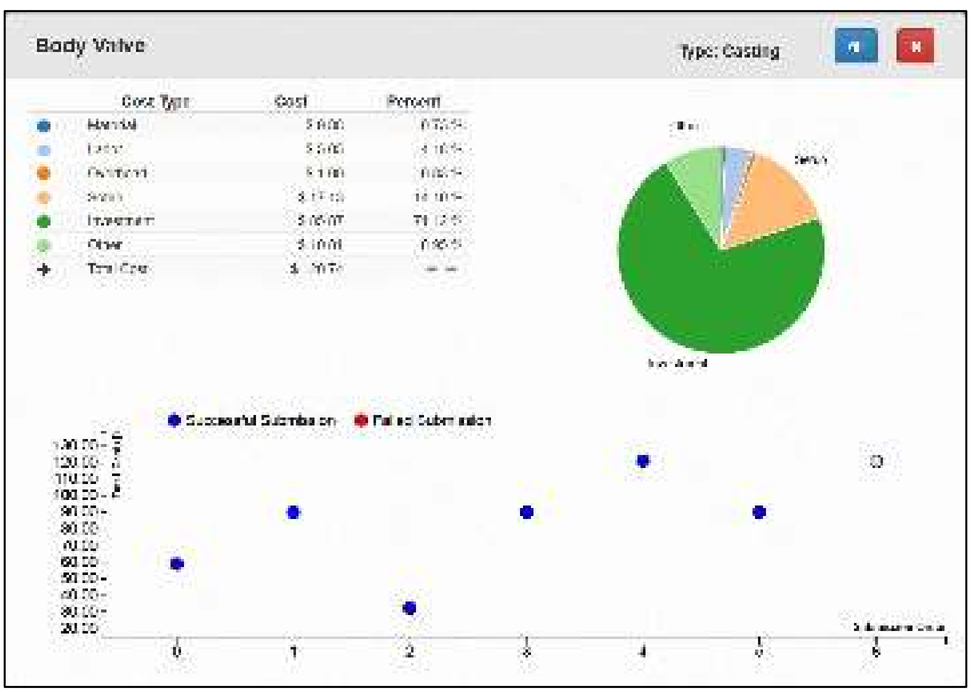


Figure 36. Fabricated Part Cost Details and Tracking

Also, as shown in Figure 5 and Figure 6, the cost details for each part and assembly include tracking of all past submissions. Hovering over points in the tracking graph shows the user and submission date for that cost result. The user can select a point in the graph and view the cost breakdown for the particular submission as well as the specified fabricated part class (for fabricated parts). Blue points indicate design submissions that were successfully analyzed by the MAS. Red points indicated design submissions that failed to complete analysis. Hovering over these points will provide some information as to why the part or assembly failed to cost.

Users can click the button to view the manufacturing data specification for fabricated parts. An example of this view is shown in Figure 7. Click the button to return to the cost details view for the part.

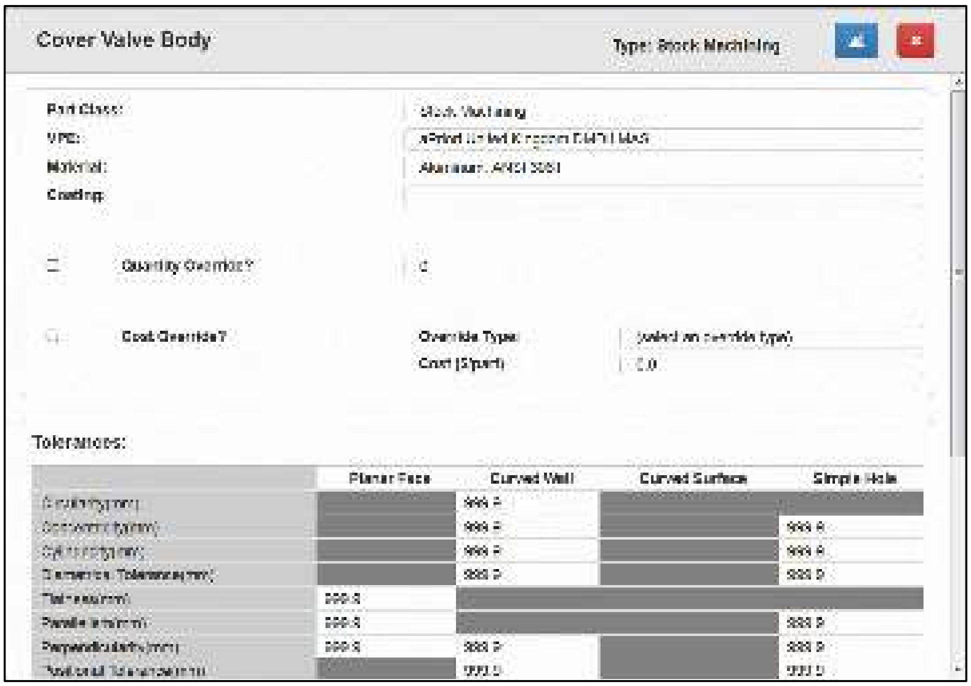


Figure 37. Manufacturing Data Specification for Selected Fabricated Part

<u>Parts (Purchased)</u>

Purchased part cost details only includes a tracking graph, and users can still view the manufacturing data specification by clicking the button.

2.2 MAS Web Application Administrator Instructions

The MAS Web Application administrators can perform a variety of other functions that Primary Users cannot access. These functions are categorized under Community and Administrator Central on the left pane of the MAS web application.

2.2.1 Community Admin Functions

The Community functions pertain to the MAS user accounts and web application site roles. These are described in more detail in the following sections.

2.2.1.1 Users

Clicking on Users will show a list of all registered users of the MAS, as shown in Figure 8.

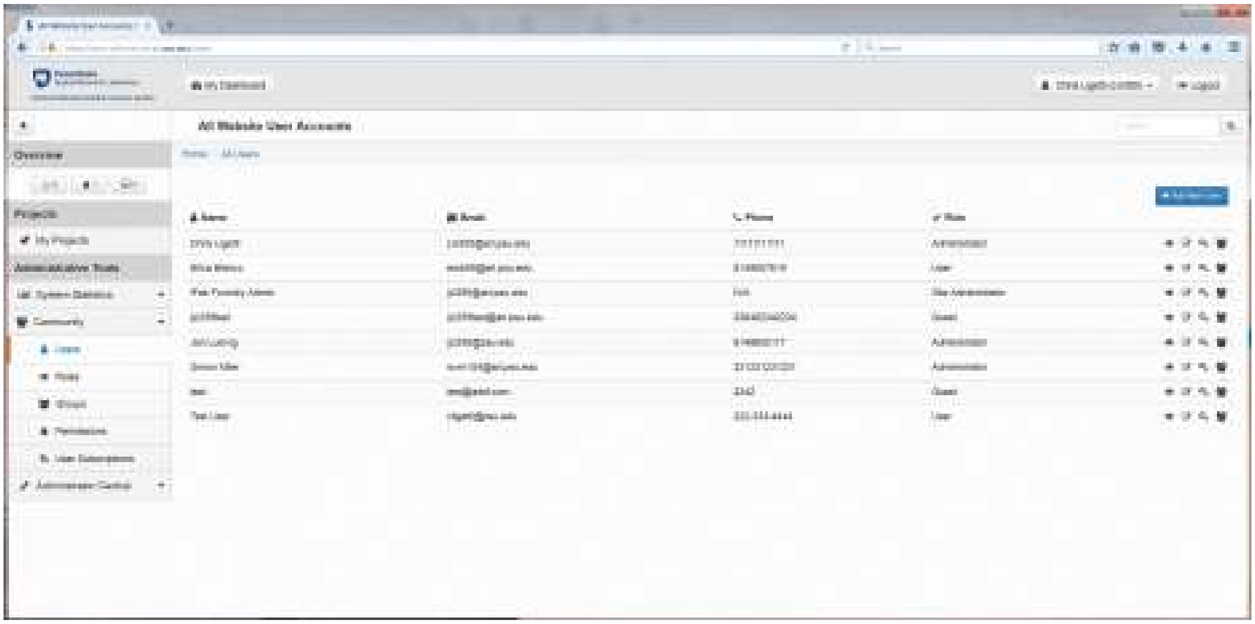


Figure 38. MAS User Accounts

New users can be added by clicking the + Add New User button. The administrator will be required to provide the username, Full Name, Email Address, Phone Number, Role (Guest, Project Manager, Administrator, User), Email Notification Subscription, and initial Password (users are prompted to change this at first log in).

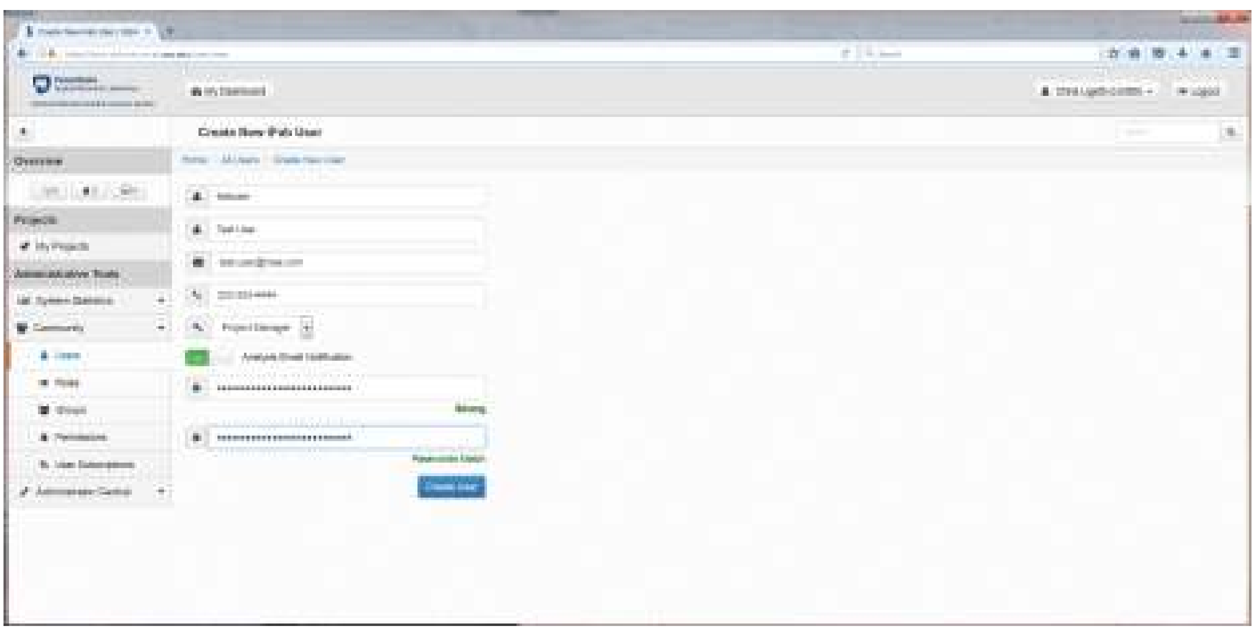


Figure 39. Add New MAS User Account

2.2.1.2 Roles

The Roles page displays the MAS site roles, number of users, and number of active sessions by those users. There are five roles, whose site permissions are summarized below:

1. Guest
   - Can view Cost Projects and cost details for projects they are members of
2. User
   - Can view Cost Projects and cost details for projects they are members of
   - Can authenticate using the API (i.e., can log in from MAAT and submit designs to projects to which they have been assigned)
3. Project Manager
   - Can view Cost Projects and cost details for projects they are members of
   - Can authenticate using the API (i.e., can log in from MAAT and submit designs to projects to which they have been assigned)
   - Can Add Cost Projects
4. Administrator
   - Can manage all projects regardless if they are a member or not
   - All site permissions (ie system log, cluster monitoring, etc)
5. Site Administrator
   - Same as Administrator, but there can only be one Site Administrator

2.2.2 MAS Web Application Administrator Central Instructions

The Administrator Central contain more system related functions, including Windchill server connection setup, subscriptions for system monitoring, cluster monitoring, system logs, and additional site settings. These are described in the following sections.

2.2.2.1 Windchill Connections

A Windchill Connection needs to be configured to allow the MAS web application to interface with an instance of PTC Windchill for BOM import during Cost Project startup. When in the Windchill Connections page, the user will see a listing of currently configured Windchill connections. The connection is really nothing more than a URL to the Windchill server. Multiple Windchill connections can be added (these will be available for the user to select from when creating a cost project). Click + New Windchill Connection and provide a Name, the Windchill server URL, and a Description. The connection will be tested when the user presses Save to ensure successful configuration.

2.2.2.2 Active Users

The Active Users page lists all users with active MAS web application sessions. The username and Role will be listed in addition to their time active, their login time, and the time their session will expire if they were not performing functions on the site (i.e., auto logoff).

2.2.2.3 Subscriptions

The MAS web application Site Administrator is automatically subscribed to a number of notifications that are sent via email. These subscriptions are summarized in Figure 10.

Figure 40. MAS Web Application Subscriptions

Additional users can be added to each subscription by clicking the button. The administrator can select a site Role which will list the users with that Role. Click the checkbox to add a user to the subscription and then click the Update Subscription Users button.

2.2.2.3 Cluster Monitoring

The Cluster Monitoring page allows administrators to monitor the status of the MAS server, machines/nodes, and individual workers. The Tomcat and Task Manager status (Figure 11) is most critical and should be monitored to ensure that the MAS is able to process manufacturing analysis jobs.

Figure 41. Cluster Monitoring – Tomcat and Task Manager Status

Cluster Monitoring page also displays job status (i.e., manufacturing analysis requests), as shown in Figure 11. Administrators should pay close attention to Queues or Unfinished tasks that do not appear to be resolving, as this may indicate a problem with the system.

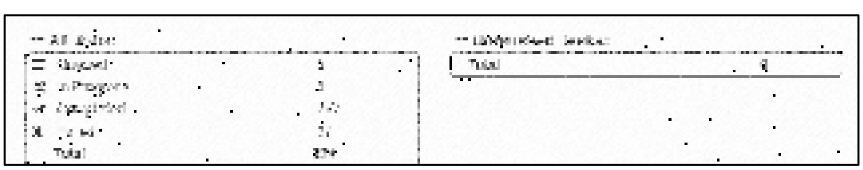


Figure 42. Cluster Monitoring – Job Status

Finally, administrators can monitor individual machine (i.e., node) status, including CPU History, Memory History, and individual Worker status, as displayed in Figure 13.

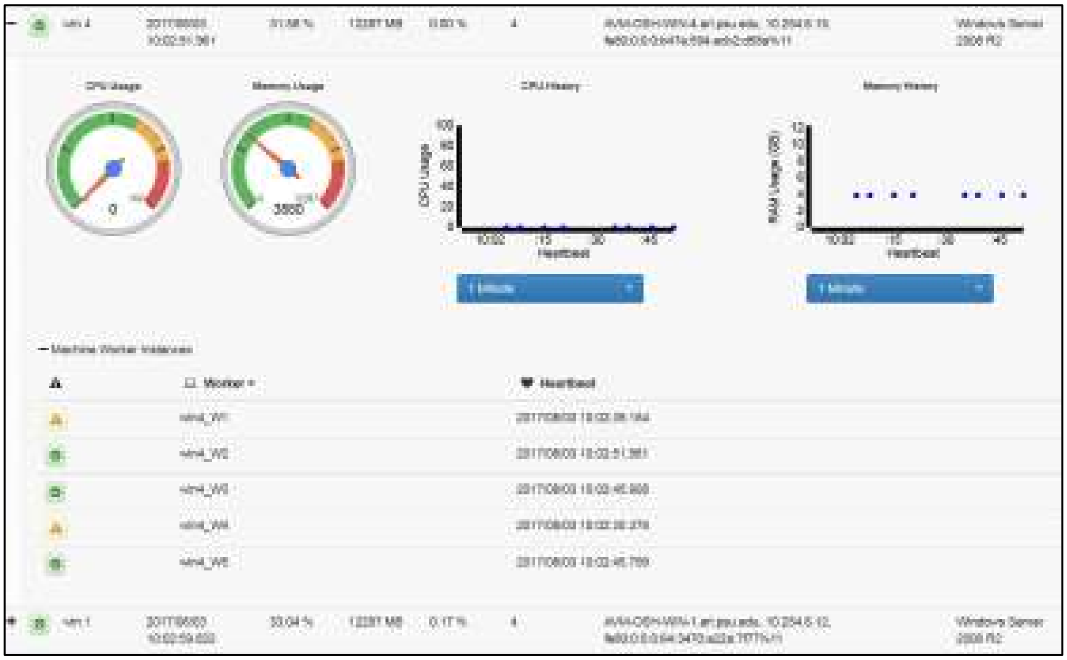


Figure 43. Cluster Monitoring – Worker Status

2.2.2.4 View System Logs

Administrators have access to several System Logs, as shown in Figure 14. Click the button to view specific logs.

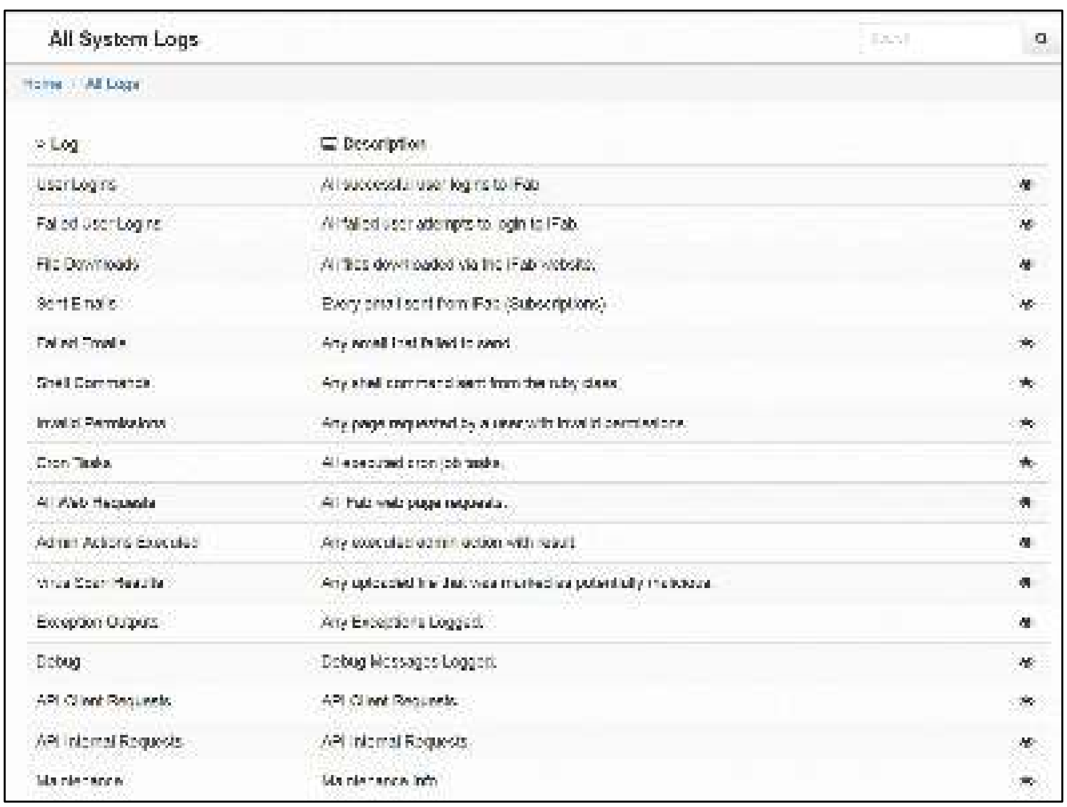


Figure 44. MAS Web Application System Logs

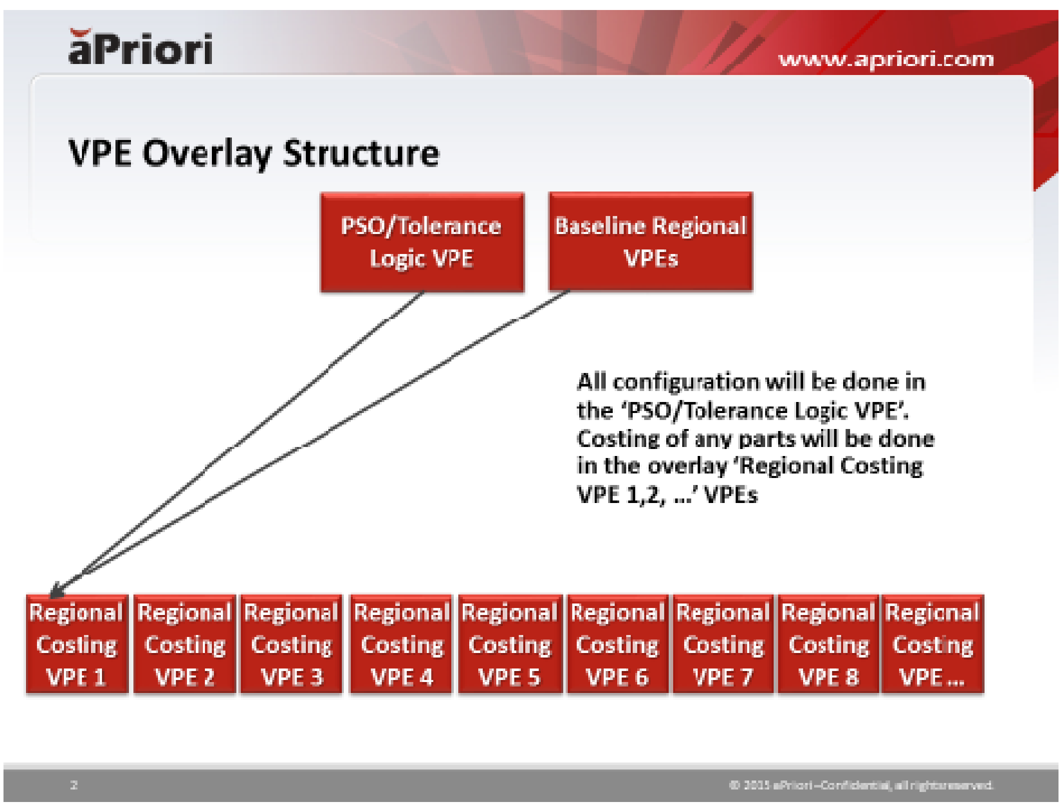
2.2.2.5 Site Settings

There are currently two MAS web application site setting available to change:

- Site Icon (.ico) – this will display in the web browser tab next to the page name
- Site Logo (.png) – this image will appear on the MAS web application log in page as well as the main site page after users log in.

Each setting can be changed by clicking + Select File and browsing for a new file of the appropriate type. Click Save to apply the change.

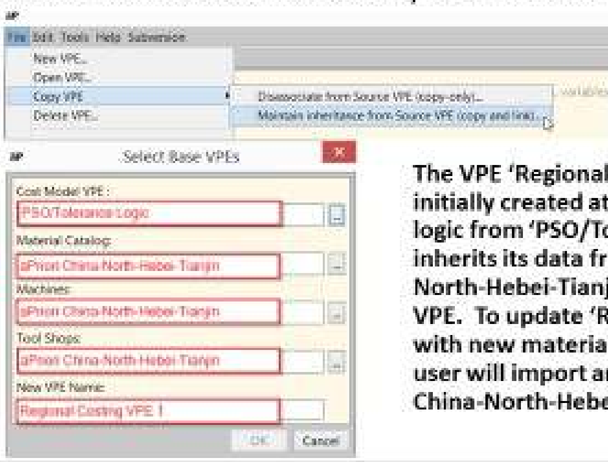
b. aPriori VPE Development Documentation



aPriori www.apriori.com

VPE Overlay Structure (cont.)

- Each overlay VPE (e.g. Regional Costing VPE 1) is created once in the VPE Manager
- Update RDLs: Use the VPE Import and overwrite the existing RDL
- PSU ARL will create these overlay VPEs with aPriori support



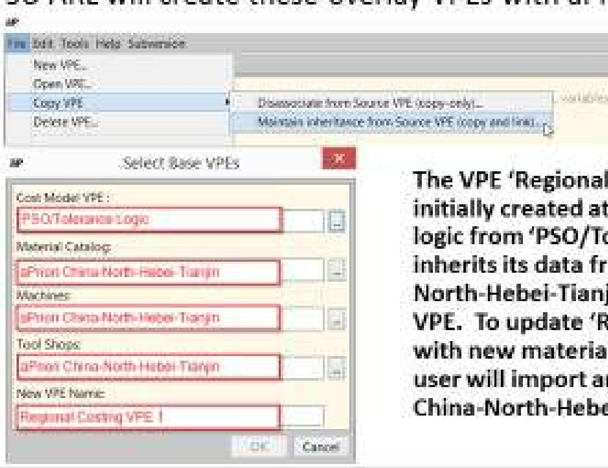
The VPE 'Regional Costing VPE 1' is initially created at left. It inherits its cost logic from 'PSO/Tolerance Logic' VPE and inherits its data from the aPriori China-North-Hebei-Tianjin regional data library VPE. To update 'Regional Costing VPE 1' with new material and labor rates the user will import and overwrite the aPriori China-North-Hebei-Tianjin VPE

5

aPriori www.apriori.com

VPE Overlay Structure (cont.)

- Each overlay VPE (e.g. Regional Costing VPE 1) is created once in the VPE Manager
- Update RDLs: Use the VPE Import and overwrite the existing RDL
- PSU ARL will create these overlay VPEs with aPriori support



The VPE 'Regional Costing VPE 1' is initially created at left. It inherits its cost logic from 'PSO/Tolerance Logic' VPE and inherits its data from the aPriori China-North-Hebei-Tianjin regional data library VPE. To update 'Regional Costing VPE 1' with new material and labor rates the user will import and overwrite the aPriori China-North-Hebei-Tianjin VPE

5 © 2015 aPriori – Confidential, all rights reserved.

aPriori www.apriori.com

Sheet Metal (Plate)

- These 29 GCD/Tolerance pairs will be created as PSOs at the Sheet Metal node

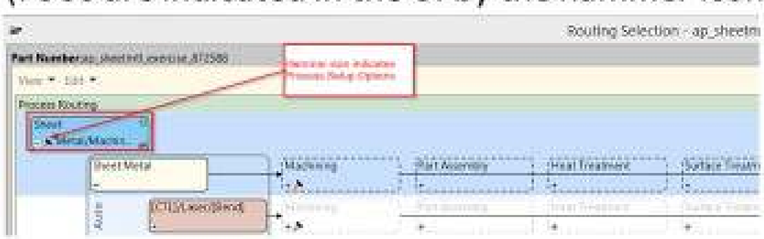
GCD	Planar Face	Curved Wall	Curved Surface	Simple Hole	Complex Hole
Circularity		X			
Concentricity		X		X	
Cylindricity		X		X	
Diam tolerance		X		X	
Flatness	X				
Parallelism	X			X	
Perpendicularity	X	X		X	
Position tolerance		X		X	
Tolerance/profile of surface	X	X	X		X
Roughness	X	X	X	X	X
Runout		X	X	X	
Straightness		X			
Symmetry					
Threading(T/F)				X	

5 © 2015 aPriori - Confidential, all rights reserved.

aPriori www.apriori.com

Sheet Metal (Plate) (cont.)

- All 29 PSOs will be entered at the Sheet Metal node (PSOs are indicated in the UI by the hammer icon)



- Placement of all PSOs at the root node allows for export/import of the PSOs from a central location and ensures that any required machining processes are in scope

6 © 2015 aPriori - Confidential, all rights reserved.

aPriori www.apriori.com

Sheet Metal (Plate) (cont.)

- All 29 PSOs are editable from the Bulk Costing module using the Inputs File spreadsheet:

X	Y	Z	AA	AB	AC	AD	AE
sourceModel	sourceModelScenario	Stock.MachiningsimpleHoleThreadsSMValue	Stock.MachiningsimpleHoleDiametricTolerancesSMValue	Stock.MachiningsimpleHolePositionalTolerancesSMValue	Stock.MachiningplanarFaceTolerancesSMValue	Stock.MachiningsimpleHoleSurfaceFinishSMValue	
		5	0.05	0.1	0.347	1.5	

- The necessary hooks to catch the PSOs will be centralized as much as possible within library files that are shared across different CSL files
  - This serves to facilitate exporting and importing this work into a previously configured VPE

7

aPriori www.apriori.com

Bar & Tube

- These 13 GCD/Tolerance pairs will be created as PSOs at the Bar & Tube node
- Planar Face tolerances will only be applied to planar face children of End GCDs

GCD	Planar Face	Simple Hole	Complex Hole
Diam tolerance		X	
Flatness	X		
Parallelism	X	X	
Perpendicularity	X	X	
Position tolerance		X	
Tolerance/profile of surface	X		X
Roughness	X	X	
Runout		X	
Straightness			
Symmetry			
Threading (T/F)		X	

8

aPriori www.apriori.com

Stock Machining

- These 27 GCD/Tolerance pairs will be created as PSOs at the Stock Machining node

GCD	Planar Face	Curved Wall	Curved Surface	Simple Hole
Circularity		X		
Concentricity		X		X
Cylindricity		X		X
Diam tolerance		X		X
Flatness	X			
Parallelism	X			X
Perpendicularity	X	X		X
Position tolerance		X		X
Tolerance/profile of surface	X	X	X	
Roughness	X	X	X	X
Runout		X	X	X
Straightness		X		
Symmetry				
Threading (T/F)				X

aPriori www.apriori.com

Casting

- These 27 GCD/Tolerance pairs will be created as PSOs at the Casting node

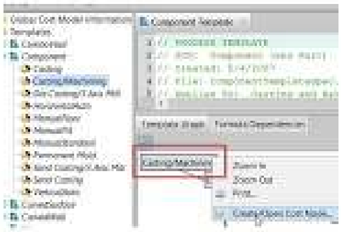
GCD	Planar Face	Curved Wall	Curved Surface	Simple Hole
Circularity		X		
Concentricity		X		X
Cylindricity		X		X
Diam tolerance		X		X
Flatness	X			
Parallelism	X			X
Perpendicularity	X	X		X
Position tolerance		X		X
Tolerance/profile of surface	X	X	X	
Roughness	X	X	X	X
Runout		X	X	X
Straightness		X		
Symmetry				
Threading (T/F)				X

© 2015 aPriori – Confidential, all rights reserved.

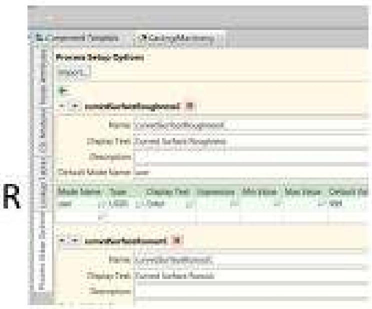
aPriori www.apriori.com

Process Setup Options (PSOs)

- PSOs were created within the CMWB at the root node By right mouse clicking on the root node, one can open the cost node:



- PSOs were created as Type USER



aPriori www.apriori.com

Code location

- Within each configured process group, the most critical concentration of code is found in the libGtolProcessCapabilities file accessible in the CMWB through Global Cost Model Information → Globally Available CSL/Lookup Tables → Library CSL
  - Each primary process group except Stock Machining locally pulls the PSOs through this library file. libGtolProcessCapabilities in machining is unique in that it globally pulls all PSOs set in primary process groups
- The bulk of customized code begins toward the end with the comment '//Begin of Custom Additions for PSU ARL'
- The following steps are processed successively:
  1. All available PSO values are pulled:

```
curvedWallCircularity = (setup.curvedWallCircularityC if setup.curvedWallCircularityC != null
                         0 otherwise)
```

aPriori www.apriori.com

Code location (continued)

2. Next, all available PSOs are checked to determine which have been actively set:

```
IsCurvedWallCircularityTolSet = (true if (curvedWallCircularity < 999 and curvedWallCircularity > 0)
                                 false otherwise)
```

3. The information from step 2 is fed into a function called IsAnyTolValuePSOSet(gcd). This function serves as a pruning rule for performance reasons and is used in ComputeAllGtolCompensationFactor(gcd, process).

```
* Function: IsAnyTolValuePSOSet
* Purpose: Test whether a specific gcd/tolerance has been set via a PSO
* Checks all tolerances that may exist for a specific gcd type
*/
IsAnyTolValuePSOSet(gcd) = (true if isCurvedSurface(gcd) and (IsCurvedSurfaceRoughnessTolSet or IsCurvedSurfaceRunoutTolSet
                            true if isCurvedWall(gcd) and (IsCurvedWallCircularityTolSet or IsCurvedWallConcentricityTolSet
                                or IsCurvedWallDiametricToleranceTolSet or IsCurvedWallPerpendicularityTolSet or IsCurvedW
```


aPriori www.apriori.com

Code location (continued)

4. The information from step 2 is also fed into a function called IsGcdGtolValuePSOSet(gcd, gtolCategory). The arguments for the function include whatever gcd the cost engine is currently processing as well as the geometric tolerance category. This function serves as a switch to determine if a tolerance was set via a PSO or through the UI in the GCD window and is used in the function, GetGcdGtolValue(gcd, gtolCategory, process):

```
GetGcdGtolValue(gcd, gtolCategory, process) =  ( gcd[gtolCategory] if IsGcdGtolValueSet(gcd, gtolCategory) == true
                                                 getPSOValue(gcd, gtolCategory) if IsGcdGtolValuePSOSet(gcd, gtolCategory) == true
                                                 -1 otherwise )
```

aPriori www.apriori.com

Code location (continued)

5. GetGcdGtolValue(gcd, gtolCategory, process) is fed into IsProcessGtolCapable(gcd, gtolCategory, process). This function determines whether a tolerance set on a given GCD is capable of being achieved by a given process
6. The best achievable tolerances by a given process are maintained in tblGtolProcessCapabilities, which is available through Globally Available CSL/Lookup Tables → Lookup Tables (note: for this configuration this table was not modified in any way).

aPriori www.apriori.com

Code location (continued)

7. If the desired tolerance cannot be set within the primary process group then a routing rule set on a particular GCD and process will fail and steps 1 through 6 will be repeated within the Machining process group in its libGtolProcessCapabilities.csl file
   - For example, within Sheet Metal, in the Laser Cut process, the operation called Laser Cutting is used to create the GCD 'SimpleHole'. Double clicking on Laser Cutting, one will find a routingRule file. If the tolerance on the simple hole cannot be achieved by laser cutting, the routing rule will fail, and any feasibly machining operation will be considered

aPriori www.apriori.com

Code location (continued)

8. Within many routing rule files is the Boolean rule: GtolValueOk. This rule evaluates the function IsProcessAllGtolCapable(gcd, process)

9. The GCD Template routing is evaluated through all possible sequences and if the routing rule in step 8 fails, it will advance to the machining node, opening up the entire Machining process group:

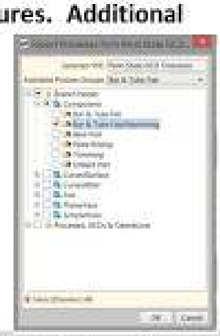
10. Steps 1-9 are repeated within the Machining process group. If the GCD feature's tolerance is achievable in Machining, then secondary machining will be included in the operation sequence, otherwise the GCD will fail all operation sequences

aPriori www.apriori.com

Process to merge work into a configured VPE

- **Notes:**
  - **The PSO/Tolerance VPE will be referred to as the source VPE and the previously configured VPE will be referred to as the target VPE**
  - **It is recommended when merging VPEs that the source and target process groups be of the same cost model version. Additional difficulties arise when, for example, Sheet Metal CMV 115 attempts to merge into CMV 50. Older CMVs often do not support geometric tolerances found in libGtolProcessCapabilities.csl and simply following the below steps will likely cause costing failures. Additional steps would need to be taken to upgrade a CMV.**

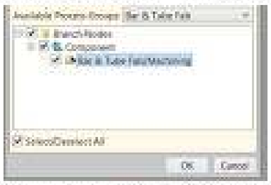
1. Within the CMWB navigate to File → Export → Processes. Select the Source VPE and the Bar & Tube Fab process group and export only the root node (Bar & Tube Fab/Machining) to a location on your desktop. Repeat for Casting/Machining, Stock Machining/Machining, and Sheet Metal/Machining



aPriori www.apriori.com

Process to Merge work into a configured VPE

2. In the target VPE, navigate to File → Import → Processes from the appropriate process group, then import to the target VPE the exported process root node from the source VPE. Repeat for the other primary process groups. This step imports in all PSOs for a given process group:



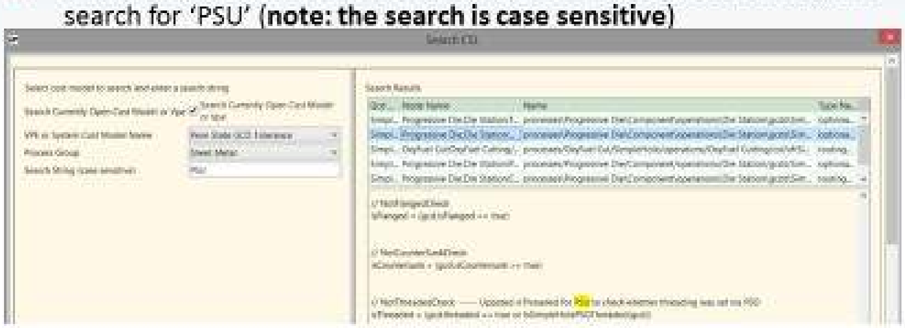
3. In the **machining** process group of the target VPE navigate to libGtolProcessCapabilities.csl. If this file is an inherited object, disinherit it and copy and paste the contents of libGtolProcessCapabilities.csl from the source VPE. If it a disinherited object, use of a text comparison software program such as Beyond Compare may be required in order to properly merge. Otherwise, visually compare the two library files and once satisfied, copy and paste the contents of libGtolProcessCapabilities.csl directly from the source to the target.

19

aPriori www.apriori.com

Process to Merge work into a configured VPE

4. In the CMWB of the source VPE, navigate to Tools → Search CSL, and search for 'PSU' (**note: the search is case sensitive**)



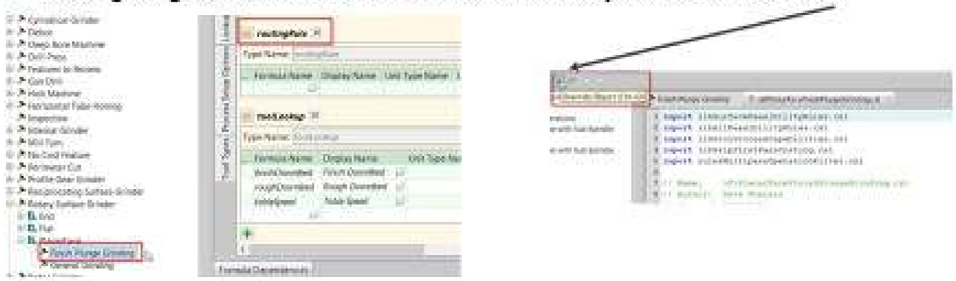
5. On the right hand side shows all the files where the string 'PSU' occurs. Clicking on each file shows the text of each file and points toward where code was modified. Each of these files in the target will have to be modified. The file type will either be an optionalOperation or a routingRule.

20

Process to Merge work into a configured VPE

6. Example: In the machining process group, searching for 'PSU', the first entry in the Search CSL dialog has the following entries: 'Gcd Name = PlanarFace'; 'Node Name = Rotary Surface Grinder:Finish Plunge Grinding//PlanarFace'; 'Name = processes\Rotary Surface Grinder\PlanarFace\operations\Finish Plunge Grinding\csl\ofrPlanarFaceFinishPlungeGrinding.csl'; 'Type Name = routingRule'

- These entries indicate the file path of where to modify the changes in the CMWB of the target VPE: Within the Processes, GCDs & Operations → Rotary Surface Grinder → Planar Face → Finish Plunge Grinding → Routing Rule
- Clicking the green icon allows one to disinherit an object and edit the code:



Process to Merge work into a configured VPE

7. Compare the modified code in the source to the target and change only code relevant to tolerances immediately adjacent to the string 'PSU':

Source VPE (desired code):

Target VPE (initial code):

```
Rule GtolValueOk: (gcd.isAnyToleranceSpecified and IsProcessAllGtolCapable(gcd, 'Surface Grinding'))
Message GtolValueOk: GtolValueOkMsg
```

Target VPE (final/desired code). Ensure that the file imports libGtolProcessCapabilities and that the modified code matches the code from the source VPE

aPriori

www.apriori.com

Process to Merge work into a configured VPE

8. Repeat steps 4-7 for all files identified by 'PSU' string
9. Note that in order to test a primary process group such as Sheet Metal, it is required that the Machining process group first be modified. Because of the logic of the routing sequences, if a tolerance is not capable of being achieved in a primary process group, the code will look to the secondary process group of Machining. If the necessary modifications to Machining are not in place, there will be errors when costing parts and using the PSOs in Sheet Metal