SANDIA REPORT

SAND2010-3242

Unlimited Release

Printed June 2010

Advanced Engineering Environment FY09/10 Pilot Project

Alan R. Pomplun, Edward G. Dutra, Grant W. Kiba, Jane Ann Lamph, Abraham L. Sego

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multi-program laboratory operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U. S. Department of Energy's National Nuclear Security Administration under Contract DE-AC04-94AL85000.

Approved for public release; further dissemination unlimited.

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from

U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: <http://www.osti.gov/bridge>

Available to the public from

U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd.
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online order: <http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online>

SAND2010-3242
Unlimited Release
Printed June 2010

Advanced Engineering Environment FY09/10 Pilot Project

Alan R. Pomplun, Edward G. Dutra, Grant W. Kiba, Jane Ann Lamph, Abraham L. Sego
Sandia National Laboratories
P.O. Box 969
Livermore, CA 94551-0969

Abstract

The Advanced Engineering Environment (AEE) project identifies emerging engineering environment tools and assesses their value to Sandia National Laboratories and our partners in the Nuclear Security Enterprise (NSE) by testing them in our design environment. This project accomplished several pilot activities, including: the preliminary definition of an engineering bill of materials (BOM) based product structure in the Windchill PDMLink 9.0 application; an evaluation of Mentor Graphics Data Management System (DMS) application for electrical computer-aided design (ECAD) library administration; and implementation and documentation of a Windchill 9.1 application upgrade. The project also supported the migration of legacy data from existing corporate product lifecycle management systems into new classified and unclassified Windchill PDMLink 9.0 systems. The project included two infrastructure modernization efforts: the replacement of two aging AEE development servers for reliable platforms for ongoing AEE project work; and the replacement of four critical application and license servers that support design and engineering work at the Sandia National Laboratories/California site.

This page intentionally left blank.

Acknowledgements

The authors express their thanks to the many people whose input contributed to the success of this project:

Project Management: Chuck Oien, Alan Pomplun.

AEE Windchill Infrastructure Modernization:

Sandia/CA: Leslie Brown, Robert Dankiewicz, Larry Navarro.

Parametric Technology Corporation: Ralph Goodwin, Will Hutcheson, Brian Johnson.

Windchill 9.1 Implementation Pilot:

Sandia/CA: Terry Bersie, Leslie Brown, Robert Dankiewicz, Grant Kiba, Jane Ann Lamph, Aaron Machado, Larry Navarro.

Sandia/NM: Abraham Sego.

Parametric Technology Corporation: Rusty Abernathy, Ralph Goodwin, Will Hutcheson, Brian Johnson.

Product Structure Pilot:

Sandia/CA: Art Hayes, Don Herron, Grant Kiba, Jane Ann Lamph, Aaron Machado, Randy Van Cleave.

Sandia/NM: Anne Barnes, Richard Chavez-Hatton, Jason Ritter, Tim Meeks, Robert Parks, Alice Starcher.

LLNL: Benjamin Grover, Rey Santos.

LANL: Gene Kilmer, Christopher Scully.

Parametric Technology Corporation: Rusty Abernathy, D. Jay Hunt, Will Hutcheson, Brian Johnson, Lisa Monk, Walt Morrow.

Mentor Graphics ECAD Library Administration Tool Pilot:

Sandia/CA: Edward Dutra, Jane Ann Lamph.

Sandia/NM: Lorraine Ashford, Terry Burrows, Andrew Clark, Fran Current, Lilia Martinez, Scott Pinnick.

KCP: Kevin Baughn, John Galichia, Brad Grantham, Linda Wormsley

Mentor Graphics Corporation: Della Fenton, Brett Graves, André Mosley.

Oasis Sales Corporation: Ed Lord, Bonnie Williams.

MCAD Data Migration:

Sandia/CA: Edward Allen, Terry Bersie, Grant Kiba, Aaron Machado, Catherine Otto, Michael Tootle, Martin Wong.

Sandia/NM: Jack Martinez, Abe Sego, Phillip Turner, Jay Smith, Tom Cox, Jason Ritter

Parametric Technology Corporation: Robert Dillon, Mayank Parikh, Chris Stelse, JP Zyblut.

Modernization of the Sandia/CA CADs Infrastructure for Design and Engineering Applications:

Sandia/CA: Leslie Brown, Robert Dankiewicz, Larry Navarro.

Sandia/CA Administrative and Purchasing Support: John Beitia, Donna Blevins, Angela Griffin, Bonnie Ladd, Pamela Williams.

Contents

Summary	13
Acronyms	19
1.0 Introduction	21
1.1. Advanced Engineering Environment Concept	21
1.2. Background	21
1.2.1. Windchill 9.0 Tool Suite	22
1.3. Objectives and Scope	23
1.3.1. AEE Windchill Infrastructure Modernization	24
1.3.2. Windchill 9.1 Pilot	24
1.3.3. PDMLink Product Structure Pilot	24
1.3.4. ECAD Library Administration Tool Evaluation Pilot	25
1.3.5. MCAD Data Migration	26
1.3.6. Modernization of the Sandia/CA CADS Infrastructure for Nuclear Weapons Applications	26
2.0 Windchill Server Modernization	27
2.1. Summary	27
2.2. Background	27
2.3. Development Server Implementation	28
2.3.1. System Installation	28
2.3.2. System Configuration	29
2.3.3. Classified Development Server Implementation	31
2.4. Training Server Implementation	31
2.5. Conclusion/Recommendations	32
3.0 Windchill 9.1 Implementation Pilot	33
3.1. Summary	33
3.2. Background	34
3.3. Windchill 9.1 Implementation	34
3.4. Windchill 9.1 Validation	35
3.5. Additional Windchill 9.1 Testing	36
3.6. Conclusion	37
4.0 Product Structure Pilot	39
4.1. Summary	39
4.2. Background	41
4.3. Alignment with PRIDE Product Structure Effort	42
4.4. Product Structure Work at Other NSE Sites	44
4.5. Initial Product Structure Development Work	45

4.5.1. Using View Versions in PDMLink	46
4.5.2. VMWare Image for Development Work	49
4.5.3. Additional PDMLink Information	50
4.6. Sandia Design Definition Business Process Analysis	51
4.6.1. Current Design/Product Definition Business Process	52
4.6.2. Future Design/Product Definition Business Process	52
4.7. Product Structure Use Case Implementation.....	52
4.7.1. Use Case Definition	53
4.7.2. Use Case Development Details	56
4.7.3. Use Case Deployment	58
4.7.4. Suffix Changes	63
4.7.5. Additional Use Cases	63
4.8. Development of the ML Report	68
4.8.1. ML Requirements	68
4.8.2. ML Report Generation	69
4.9. Issue and Gap Analysis	74
4.9.1. Design Related Process Issues/Gaps:.....	74
4.9.2. Product Structure and Design Release Issues/Gaps:.....	75
4.9.3. Documents Issues/Gaps:	76
4.9.4. Material List Report Issues/Gaps:.....	76
4.9.5. Practices and Processes Issues/Gaps:.....	78
4.9.6. Additional Issues/Gaps:	78
4.10. Conclusions and Recommendations.....	78
4.10.1.Recommendations	79
4.10.2.Topics for Further Discussion and Definition.....	80
4.10.3.Suggested Additional Use Cases.....	81
4.10.4.Project Documentation.....	81
5.0 Mentor Graphics ECAD Library Administration Tool Pilot	83
5.1. Summary	83
5.2. Mentor Graphics Data Management System	84
5.3. DMS Installation and Configuration	87
5.3.1. Installation.....	87
5.3.2. Setting User and Group Account Objects (Access Controls)	89
5.3.3. Creating a Library Specification	89
5.3.4. Library Upload into DMS	90
5.3.5. Setting Element Release Status.....	90
5.3.6. Uploading Additional Information to a DMS Component	91
5.3.7. Creating a Production Library in DMS.....	92
5.4. Testing.....	93
5.4.1. Check-in/Check-out of Parts and Part Elements.....	93

5.4.2. Creating New Library Parts	95
5.4.3. Part and Element Searches	96
5.4.4. Updating an External Library from a DMS Production Library List.....	96
5.4.5. Designer Testing	97
5.4.6. Data Export	97
5.4.7. Printing.....	97
5.5. Library Administration Process	98
5.5.1. Current Library Administration Process	98
5.5.2. New Library Administration Process Utilizing DMS.....	101
5.6. Evaluation Results and Recommendations	101
5.6.1. DMS Library Administration Evaluation	101
5.6.2. DMS Designer Evaluation	102
5.6.3. Comparison to Other Applications	102
5.6.4. Recommendations for DMS Improvement	102
5.6.5. Lessons Learned.....	105
5.6.6. Next Steps	106
6.0 MCAD Data Migration Project.....	107
6.1. Summary	107
6.2. Objective and Scope.....	107
6.3. Introduction	108
6.4. Data Migration	109
6.4.1. MatrixOne Data Extraction: Rehearsal 1	110
6.4.2. MatrixOne Data Extraction: Rehearsal 2	111
6.4.3. Windchill 8.0 Data Extraction	112
6.5. Conclusion.....	113
7.0 Modernization of the Sandia/CA CAD Infrastructure for Design and Engineering Applications.....	115
7.1. Summary	115
7.2. Implementation.....	115
7.3. Server Virtualization	115
7.4. Conclusion.....	116
8.0 References.....	119
Appendix A: AEE Server Specifications	121
A.1 Three-tier Development Servers.....	121
A.3 Monolithic Training Server	122
Appendix B: Sandia Design Definition Current Workflow Process Diagrams	123
Appendix C: Future Design Definition Workflow Process in Windchill PDMLink 9.0.....	133

Figures

Figure 1.	Advanced Engineering Environment collaboration toolset.	23
Figure 2.	PRIDE Product Structure “Define a Part” use case diagram.	43
Figure 3.	System level use case diagram.	54
Figure 4.	Use Case 1 process diagram.	55
Figure 5.	Use Case 2 process diagram.	55
Figure 6.	Use Case 3 process diagram.	56
Figure 7.	Use Case 4 process diagram.	56
Figure 8.	Presentation graphic for Use Case 1, step 2.	59
Figure 9.	Presentation graphic for Use Case 2, step 3.	60
Figure 10.	Presentation graphic for Use Case 3, step 3.	61
Figure 11.	Presentation graphic for Use Case 4, steps 7 and 8.	62
Figure 12.	A “Manufacturing” view was created to support manufacturing data links for the 900011-00 revision “A” Well part.	64
Figure 13.	The WTPart trace code attribute was set to “Serial Number” by using the “Edit Common Attributes” action.	65
Figure 14.	WTParts represent the instances of a family table library part.	66
Figure 15.	The Sandia ML report generated in PDMLink.	72
Figure 16.	The ML report with tabulation.	73
Figure 17.	Typical DMS desktop configuration.	86
Figure 18.	NSE DMS landscape.	88
Figure 19.	Typical status visual cues.	94
Figure 20.	The Checkout Hierarchy Selection table.	95
Figure 21.	Comparison of the current library administration process and the new DMS library management process.	99
Figure 22.	The datasheet selection tab.	104
Figure 23.	Current design definition process overview.	123
Figure 24.	Current process Step 1: Design effort.	124
Figure 25.	Current process Step 2: Final design.	125
Figure 26.	Current process Step 2: Mechanical design.	126
Figure 27.	Current process Step 2: Electrical design.	127
Figure 28.	Current process Step 3: Supporting documentation.	128
Figure 29.	Current process Step 4: Compare Information in the Sandia EBOM system.	129
Figure 30.	Current process Step 4: Material list.	130
Figure 31.	Current process step 5: Store/release design.	131
Figure 32.	Future process overview.	133
Figure 33.	Future process Step 1: Final design (same as current process).	134

Figure 34.	Future process Step 2: Mechanical design overview.	135
Figure 35.	Future process Step 2: Mechanical design for legacy designs with an existing CAD model.	136
Figure 36.	Future process Step 2: Mechanical design for legacy designs with no existing CAD model.	137
Figure 37.	Future process Step 3: Part structure.	138
Figure 38.	Future process Step 4: WTParts.	139
Figure 39.	Future process Step 5: Supporting documentation (same as current process).	140
Figure 40.	Future process Step 5: Material list (same as current process).	141
Figure 41.	Future process Step 6: Finalize design.	142

This page intentionally left blank.

Summary

The Advanced Engineering Environment (AEE) defines a model for an engineering design and communications system that will enhance project collaboration throughout the Nuclear Security Enterprise (NSE). The AEE will contain a comprehensive set of engineering design and analysis tools with integrated data repositories which are linked together and made accessible through a computer network. This approach facilitates concurrent engineering by enabling the seamless, workflow driven, parallel flow of digital models and information across a secure distributed networking environment with the enforcement of need-to-know user access.

The AEE Pilot is a multi-year project that provides Sandia National Laboratories with an opportunity to explore new design and engineering tools to benefit nuclear weapons (NW) design and engineering activities at Sandia and across the NSE. The team partners with NW project teams within and outside of Sandia to implement these tools in innovative ways to improve design and engineering processes and to move toward the goal of a model-based engineering environment.

This FY09/10 AEE project builds upon work that was completed during the FY08/09 AEE project. The following is a summary of the project's six focus areas and the work which was completed:

1. **Modernize the AEE Windchill infrastructure:** A first priority for this AEE project was the modernization of the development servers that are used for AEE project activities. The AEE team purchased two new three-tier server systems to replace the aging systems that had been in use since the FY06 AEE project. The team installed, configured and validated the Windchill 9.0 application on the new systems in preparation for other FY09/10 project activities. The servers were initially configured to PRS12005 requirements¹, with additional changes to mirror the Sandia corporate Windchill 9.0 production servers. The team's intent was to duplicate the configuration of the production servers so that all of the AEE pilot development work and the resident data could be directly transferable to those systems.

The AEE team contracted Parametric Technology Corporation (PTC) to provide Windchill installation support on the new servers. As was expected, no significant issues were encountered during the installation and validation process. Upon the completion of the Windchill application testing and validation, the servers were put to use for AEE project activities.

2. **Pilot an application upgrade of Windchill 9.0 to the 9.1 release:** Staff members at several NSE sites were concerned that the implementation of a Windchill 9.0 application update to release 9.1 would have complications. The AEE project team piloted a Windchill 9.1 update on an AEE development server to determine the level of effort required and to identify any potential problems associated with the update. The software installation, testing and performance tuning were completed successfully and relatively quickly with support from a PTC consultant. All data residing on the server remained

¹ NWC Product Realization Standard – PDMLink Standard Installation and Configurations, November 2007.

fully accessible after the completion of the update. The results of this pilot were published² and distributed to the NSE sites.

3. **Pilot a product structure definition in Windchill PDMLink 9.0:** The Product Realization Integrated Digital Enterprise (PRIDE) Product Structure project goal is to deliver a single set of Product Structure Specifications that can be referenced and aligned to by PRIDE implementation projects across the NSE³. To support that effort, the AEE completed a pilot to define an engineering bill of materials (BOM) based product structure for a mechanical product within the Windchill PDMLink 9.0 product lifecycle management (PLM) application. The main thrust of the pilot was the implementation of four use cases that represent the typical lifecycle of a NW component design at Sandia, including the creation of the product definition and the design revisions and suffixes as the design is modified. The team used a simple pen model as the product for the use cases to avoid unnecessary complications. This pen model is being used by the PRIDE Product Structure team and other NSE sites to enable direct comparisons of their product structure work.

The AEE team began their effort by meeting with staff from Sandia/NM, LANL and LLNL to understand the PDMLink product structure work staff at those sites had already completed, and to understand where they were headed with future efforts. The AEE team also worked with the Sandia PRIDE Product Structure team to ensure that the work on this pilot would be directly beneficial to the PRIDE effort.

As a foundation for the use case development and implementation, the AEE team worked with a process architect from PTC to conduct a high-level business process analysis. The goals were to define Sandia's current design definition processes, propose a future process using the PDMLink 9.0 tool, and identify any significant issues or gaps between those processes and the tool capabilities. The deliverables included high-level process flow diagrams for both the current and proposed future product design definition workflows and a set of process issues and tool gaps that need further discussion and resolution.

In parallel, the AEE team worked with a PTC solution architect to design and implement the four use cases in PDMLink. In summary, a product structure was created for a simplified three-part cartridge PRIDE pen model by using its CAD model structure as the basis for creating a WTParts structure in PDMLink. WTParts are PDMLink objects that represent the elements of a design. Additional WTParts were then created for support drawings, specifications and the bill of materials. This was a hybrid between a model-centric and parts-centric approach to creating the product structure. The use cases also demonstrated how to handle revisions for minor design changes and rolling the suffix for major changes. Because the primary goal of this pilot was to demonstrate a BOM-based product structure, the PTC solution architect developed a PDMLink report generator to create a part defining material list (ML) report to represent the product structure. The ML report contains most of the information contained in a current Sandia ML report although it is dissimilar in appearance.

² *Advanced Engineering Environment Project Windchill 9.1 Implementation Pilot Summary Report*, July 2009.

³ *PRIDE Product Structure Specifications: Design Definition*, June 2009, page 5.

The AEE team's product structure work is documented in a number of presentations and reports in addition to this project report. The team has shared its work with PRIDE team members at Sandia/NM, LANL and LLNL through a series of presentations. They also hosted a hands-on workshop for the Sandia System Bill of Materials (SBOM) team who will continue the development of the product structure definition in PDMLink.

- 4. Mentor Graphics ECAD library administration tool pilot:** The AEE team piloted the implementation of Mentor Graphics Data Management System (DMS) software as a solution for resolving NSE multi-site shared ECAD library administration problems. This was a partnership effort between Sandia/CA, Sandia/NM and Honeywell FM&T (KCP). The team's objective for the pilot was to determine the value added to the design agency (DA) and production agency (PA) ECAD library administration processes through the use of the DMS tool.

The AEE team evaluated only the DMS library administration tool in this pilot. There are additional data management modules available for DMS that might be worthy of evaluation in the future. DMS is Oracle-based mid-ware, and with additional modules, it can provide project lifecycle maintenance capabilities, including a link between the ECAD engineering community and a PDM system (such as PDMLink).

The AEE team used a Sandia/NM Windows virtual server on the unclassified Sandia Restricted Network (SRN) as the DMS server for this project. DMS was installed with minimal adjustments to the out-of-the-box (OOTB) configuration to insure error-free functionality in the NSE environment. Because KCP is not allowed to have direct write access to a Sandia server, a crypto-card account was set up for the KCP librarian and a Sandia/CA computer was identified for KCP remote login.

The current version of the NSE library was uploaded into a new DMS library specification and the following capabilities were evaluated:

- Real-time library updates from multiple sites to a DMS library specification.
- Overall multi-site library management to avoid conflicts and duplication of effort by librarians at different sites.
- Formal library check-in/check-out process to track modifications to existing parts and part elements.
- Permissions and constraints to restrict library editing and need-to-know (NTK) to appropriate individuals.
- Ability to export and synchronize production libraries outside of DMS.
- Ability to instantiate parts into a schematic from DMS.
- Ability to access DMS component data from within a DxDesigner schematic.
- Ability to extract reports and spreadsheets from DMS queries.

On the whole, DMS worked as expected, although the AEE team did identify several areas for improvement. The biggest concern was not a DMS tool problem but a networking “bottleneck”. Sandia/CA was seeing process durations 10 times those of equivalent exercises that were run at Sandia/NM. Investigation into this Sandia/CA bottleneck is under way.

One significant advantage with DMS is that KCP can upload their own parts to the master library in real time via remote login to a Sandia computer. Under the current non-DMS process, library updates have to be performed by overwriting the entire master library with a new version. KCP is not allowed direct write access to Sandia servers, so they cannot overwrite the master library. To date, SNL/CA has been updating this library for KCP. One drawback to the KCP remote login process is that it ties up an SNL/CA Library Manager license.

5. **Support the Sandia MCAD PDMLink data migration project:** Sandia is transitioning to a new electronic enterprise PLM environment using the Windchill PDMLink 9.0 application. As part of this transition, legacy mechanical computer-aided design (MCAD) data must be moved from Sandia’s existing PLM systems into new Windchill 9.0 systems in both classified and unclassified environments. The AEE project contributed to this effort by providing funding to support the Sandia Data Migration team’s ongoing efforts to refine and deploy a data migration plan. The AEE funding was used only for Windchill consultant costs and the funding period was approximately two months in duration.

Project activities included preliminary testing and evaluation for moving legacy MCAD data from the unclassified MatrixOne PLM system into a Windchill PDMLink 9.0 development system, followed by a full migration of the legacy data. This process required the use of an intermediary application, Pro/INTRALINK, to transition the data between systems. The project team also completed testing and a full migration of legacy MCAD data from the unclassified Windchill 8.0 production system into the Windchill 9.0 development system, again with the use of the intermediary system.

A number of significant problems were encountered with the data extractions from both legacy systems. There was a higher than expected number of errors in the extracted configurations, primarily due to external references within the data itself that could not be resolved. The extraction scripts were modified to pull as much data from the legacy environments as possible to ensure that configuration problems were minimized. At the conclusion of the AEE-funded portion of the data migration effort, the data migration team had reached a decision point: Do they continue to try to refine the data extraction scripts to attempt to reduce configuration errors, or do they move forward and migrate the data that extracted and loaded without error with a plan to manually address the configurations with errors on an as-needed basis? It was determined that further refinement of scripts would not yield significant configuration improvements as the errors were within the CAD data itself. Therefore, the team proceeded with the migration of all data that correctly loaded and they planned to manually correct and load configurations with errors.

6. **Modernize the Sandia/CA CAD infrastructure for design and engineering applications:** The Sandia/CA CAD IT Services team, Org. 8945, manages a combined forty-eight CAD infrastructure servers on the classified and unclassified networks to support the site's design and engineering activities. Eight of these servers directly support Parametric Technology Corporation (PTC) software applications. These aging servers, ranging from three to nine years old, are all out of service warranty and represent a significant risk to the site's NW design and engineering capabilities. AEE provided the funding to replace the four most critical of these systems and to install, test and validate the PTC applications on the new systems.

This page intentionally left blank.

Acronyms

AEE - Advanced Engineering Environment
BOM - bill of materials
CAD - computer-aided design
CM - configuration management
COE – common operating environment
COTS - commercial off-the-shelf (product)
DA - design agency
DOE - Department of Energy
DMS – Data Management System
EBOM – electronic bill of materials (a view version used in Windchill PDMLink)
EBOM - Electronic Bill of Materials (a software application used by Sandia)
ECAD - electrical computer-aided design
eCIS – enterprise Component Information System
EDIF - electronic design interchange format
EPM – enterprise product model
FY - fiscal year
GB – gigabyte
GHz – gigahertz
GUI - graphical user interface
HTML - hypertext markup language
HTTP - hypertext transfer protocol
HTTPS - hypertext transfer protocol-secure
IP - intellectual property
IMS - image management system
JPEG - Joint Photographic Experts Group
KCP - Kansas City Plant
LANL – Los Alamos National Laboratory
LLNL - Lawrence Livermore National Laboratory
MBE - model-based engineering
MBIT - Model-Based Integration Tools
MCAD - mechanical computer aided design
MHz – megahertz
NNSA - National Nuclear Security Administration
NSE – Nuclear Security Enterprise
NTK - need-to-know
NW - nuclear weapon
NWC – Nuclear Weapons Complex
OIR - object integration rule
OS - operating system
OOTB - out-of-the-box
OUO - official use only
PA - production agency
PC - personal computer
PCB - printed circuit board

PDF - portable document format
PDM - product data management
PLM - product lifecycle management
PRIDE - Product Realization Integrated Digital Enterprise
Pro/E - Pro/ENGINEER
PRS - product realization standard
PTC - Parametric Technology Corporation
PWA - printed wiring assembly
PWB - printed wiring board
RAID - redundant array of independent disks
RAM - random access memory
REL – release
RMI - Requirements, Modernization and Integration
RPM - revolutions per minute
RU – rack unit
SCN - Sandia classified network
SCSI - small computer system interface
SME - subject matter expert
SNL - Sandia National Laboratories
SNL/CA - Sandia National Laboratories/California
SNL/NM – Sandia National Laboratories/New Mexico
SRN - Sandia restricted network
TIFF - tagged image file format
URL – uniform resource locator
WIP – work in progress
WTDdocument – Windchill technology document
WTPart – Windchill technology part

1.0 Introduction

1.1. Advanced Engineering Environment Concept

Sandia's research and engineering contributors have always been geographically dispersed, which makes it difficult to collaborate effectively. As part of the Nuclear Security Enterprise, Sandia, and the other NSE sites must modernize their engineering technology infrastructure through which this fundamental engineering data and scientific knowledge can be easily stored, retrieved, visualized and exchanged between Design Agencies and Production Agencies, regardless of location.

To achieve NNSA's vision of a more responsive enterprise, the NSE sites require an Advanced Engineering Environment (AEE). The AEE is the enabling technology of model-centric engineering. It is a computational and communications system that contains a comprehensive set of engineering design and analysis tools with integrated data repositories which are linked together and made accessible through a computer network. This approach facilitates concurrent model based engineering by enabling the seamless, parallel flow of digital models and information across a distributed networking environment. It provides the added benefit of real time engineering collaboration between multidisciplinary groups who are involved in a product's design and development process, which in turn results in higher quality products that are developed in less time and at a lower cost.

The AEE's long term goal is to provide the following benefits to the NSE:

- Accelerate the development of highly advanced scientific, research and engineering projects.
- Provide consistent, comprehensive engineering data to all authorized project participants.
- Provide a mechanism through which Sandia can collaborate on digital engineering data with other DOE/NNSA sites and partner universities to increase efficiencies, eliminate errors, and reduce cycle time and cost.
- Improve design intent understanding through enhanced visualization, leading to better and more timely decision-making.
- Facilitate work on projects with widely geographically distributed participants.
- Enhance knowledge retention by capture of engineering decision bases.
- Faster response to engineering issues, reducing project delays.

1.2. Background

This AEE FY09/10 project builds upon work that was completed during the FY08/09 AEE project⁴. The FY08/09 project had an emphasis on evaluating the capabilities of several of Windchill's individual design and collaboration tools to determine how they could be used in an integrated design environment. A major effort was the piloting of a desktop collaborative NW

⁴ *Advanced Engineering Environment Collaboration Project*, January 2009.

design and engineering process for both mechanical and electrical design using the ProductView, PDMLink, and ECAD Workgroup Manager applications.

For the FY09/10 project, AEE continued to focus on implementing the Windchill tool suite to support Sandia NW design and engineering work. Because Sandia is moving to Windchill as its corporate PLM for CAD data management, AEE's goal was to deploy and test the Windchill tools to understand and resolve potential problems ahead of this transition. The product structure and data migration pilots were two priorities for this project.

This AEE project also continued its investigation of ECAD tools to support joint NW design work between NSE sites. The FY08/09 AEE project included an evaluation of the PTC ECAD Workgroup Manager application for design collaboration. This AEE project conducted a pilot evaluation of the Mentor Graphics DMS application for multi-site ECAD library administration.

1.2.1. Windchill 9.0 Tool Suite

The Windchill 9.0 tool suite integrates PTC's applications for data management, MCAD, ECAD, visualization and analysis, intellectual property capture, and technical publication. Figure 1 represents the ideal AEE toolset where all of the tools are fully integrated and users have remote access to all data (where authorized). Although these tools are not all fully integrated at this time, PTC continues to work toward that goal.

Figure 1. Advanced Engineering Environment collaboration toolset.

1.3. Objectives and Scope

Implementing the Advanced Engineering Environment will require the full integration of tools, systems, and data across all of the NSE sites. The previous FY08/09 AEE project addressed only the informal in-process engineering activities associated with product lifecycle management (PLM) which is a front-end step in a model-based engineering (MBE) process. This project began to address the management of product definition and product structure and the formal processes of PLM during the design phase.

A basic requirement for the AEE project is that there will be no customization of the Windchill application and that it be used in its commercial out-of-the-box (OOTB) state. Specifically, no alteration of the application code base is allowed. However, configuration to adapt the applications to specific NSE business process needs is allowed.

This project had a number of different activities, each of which had its own defined objectives and scope. The following sections provide the detail of those activities.

1.3.1. AEE Windchill Infrastructure Modernization

The objectives for the AEE infrastructure modernization were:

- Replace aging development server hardware with new systems.
- Install and validate the Windchill 9.0 application on the new servers.
- Configure the servers to mirror the SNL/NM Windchill production servers so that results from AEE development work will be directly applicable to those systems.
- Perform a new Windchill 9.0 installation on an existing training server to be used for PDMLink training classes.

The scope of this effort included the replacement of the existing unclassified and classified AEE development servers with new hardware, and the installation, tuning and validating of the Windchill 9.0 application. The training server refresh was limited to the reinstallation of the Windchill 9.0 application to resolve issues that had been limiting its use. The hardware was relatively new and still under warranty, and did not require replacement.

1.3.2. Windchill 9.1 Pilot

The objectives for the Windchill 9.1 pilot were:

- Install the Windchill 9.1 application update on an existing AEE Windchill 9.0 development server which contained CAD data and document files.
- Determine the level of effort required to install the update and any complications that were encountered.
- Communicate the results to the NSE sites in response to concerns from their PDMLink teams that this update would be problematic when applied to their Windchill 9.0 systems.

The scope of this effort was limited to applying the Windchill 9.1 update to the AEE classified server only, completing a post-installation test plan to validate operation, and verifying that all CAD data and documents on the server could be accessed and had not been compromised. This pilot did not include the full testing and capabilities evaluation the Windchill 9.1 release.

1.3.3. PDMLink Product Structure Pilot

The primary objective for this pilot was to define an engineering BOM-based product structure for a mechanical product within the Windchill 9.0 PDMLink tool. This effort was intended to support the larger NSE PRIDE Product Structure effort by exercising a number of use cases that represent many of the activities in the Sandia design/product definition workflow. Project goals included the following:

- Work with NSE PRIDE Product Structure team members to understand their goals and to align AEE activities accordingly.
- Meet with staff at other NSE sites to understand and to learn from their efforts at implementing product structure in PDMLink.

- Flowchart the current Sandia design/product definition workflow and business processes as the basis for developing a proposed future process within the PDMLink application.
- Identify appropriate use cases for this pilot project.
- Implement the use cases in PDMLink.
- Identify business process issues and gaps between current processes and PDMLink OOTB capabilities.
- Communicate project results to the NSE sites.

The scope of the pilot was limited to the exercising of a modest number of use cases using a simple three-part pen cartridge assembly. This approach was intended to minimize complications that could potentially side-track the AEE team. The goal was to start simple and if time allowed, begin to exercise more complex real-world weapons models once the team had established a process for product structure definition.

The effort to define a proposed future design definition workflow process using PDMLink was limited to a high level analysis only. The intent was to provide a process to follow for the use case exercises and to identify process issues and tool gaps at this high level. The AEE team's goal was to provide a preliminary foundation that other project teams could use as a starting point for their own product structure work.

1.3.4. ECAD Library Administration Tool Evaluation Pilot

The objective of this pilot was to evaluate the Mentor Graphics Data Management System (DMS) software as a solution for resolving NSE multi-site shared ECAD library administration problems. The evaluation included exercising and documenting the following capabilities:

- Incorporation of real-time library updates from multiple sites to a DMS library specification.
- Management of a multi-site library to avoid conflicts and duplication of effort by librarians at different sites.
- Tracking of existing part modifications via a formal check-in/check-out and versioning process.
- Application of permissions and constraints to restrict library editing and need-to-know (NTK) access to appropriate individuals.
- Export and synchronization of production libraries outside of DMS.
- Instantiation of parts into a schematic directly from DMS.

Although DMS has a number of data management modules, the scope of this evaluation was limited to the library administration functionality only. The project included the evaluation of DMS at three project partner sites: Sandia/CA, Sandia/NM and KCP. The AEE team's evaluation includes lessons learned and recommendations for improving DMS.

1.3.5. MCAD Data Migration

The objective for this project was to support the migrating of legacy MCAD data from Sandia's Matrix MXPro and Windchill 8.0 product data management systems into a new PTC Windchill 9.0 PDMLink system. The Sandia Data Migration team had been working on this effort for some time and had already completed most of the required planning and infrastructure development work prior to AEE involvement. AEE provided funding to support the implementation phase of data extraction from the MXPro and Windchill 8.0 systems, and data loading into the intermediary Pro/INTRALINK application in preparation for migrating the data into the PDMLink system.

The scope of this project was limited to funding the PTC Windchill consultants who worked with the Sandia Data Migration team. Sandia staff was funded from a different source. AEE provided a fixed budget over a two month period to enable the Data Migration team to cover a gap in their project funding so that they could continue their critical work uninterrupted.

1.3.6. Modernization of the Sandia/CA CADS Infrastructure for Nuclear Weapons Applications

The objective of this project was to upgrade the at-risk infrastructure that supports software applications for NW design and engineering activities at the Sandia/CA site. This infrastructure consists of numerous aging application, print and license servers that are all out of warranty and in need of replacement.

The scope of this project was limited to replacing four critical servers that provide PTC software applications and licenses to the Sandia/CA site. These servers are essential for supporting NW project work as Sandia transitions to PDMLink as its production PLM system.

2.0 Windchill Server Modernization

2.1. Summary

A first priority for this AEE project was the modernization of the Windchill development servers that are used for AEE project activities. The AEE team purchased two new three-tier server systems to replace the aging systems that had been in use since the FY06 AEE project. The team installed, configured and validated the Windchill 9.0 application on the new systems in preparation for other FY09 project activities. The servers were initially configured to PRS12005 requirements⁵, with additional changes to mirror the Sandia corporate Windchill 9.0 production servers. The team's intent was to duplicate the configuration of the production servers so that all of the AEE pilot development work could be directly transferred to those systems.

The AEE team contracted with Parametric Technology Corporation (PTC) to provide Windchill installation support on the new servers. As was expected, no significant issues were encountered during the installation and validation process. The AEE team had worked with PTC to install Windchill 9.0 the previous year on the servers that are being replaced and this was not a new process for them. The Windchill installation procedure is documented in a separate report: "Windchill PDMLink 9.0 M070 Installation and Configuration for Sandia National Laboratories."

Upon the completion of the Windchill application testing and validation, the servers were put to use for AEE project activities, primarily the product structure development pilot, which is documented in another section of this report. They are also being used for a continuation of the B83 project work that was initiated in the FY08/09 AEE project.

The AEE team also completed the installation and configuration of Windchill 9.0 on a training server that will be used to train Sandia staff on the use of the Windchill tool suite. This server had been physically installed during the AEE FY08/09 project, but problems with the system's configuration had limited its use. The Windchill application was completely reinstalled and configured to match the installations on the two AEE development servers. After testing and validation, the server was used for the first in a series of Sandia/CA staff PDMLink training sessions.

2.2. Background

The AEE project had purchased and deployed two three-tier Windchill 8.0 development systems at the Sandia/CA site during the AEE FY06 project. Those servers were upgraded to the Windchill 9.0 application release during the AEE FY08/09 project. However, they were now out of warranty and were in need of replacement with improved systems to support the activities planned for the AEE FY09/10 project.

Although the older AEE development servers had been generally configured to PRS12005 criteria, they did not exactly match the configurations used on the Sandia corporate Windchill production servers. As a result, the knowledge gained from previous AEE pilot projects that are deployed on these servers may not have been directly transferrable to the production Windchill

⁵ NWC Product Realization Standard – PDMLink Standard Installation and Configurations, November 2007.

systems. As the AEE team has learned from past experience, differences in the configuration settings between PDMLink systems can result in significantly different outcomes on those systems. The new servers were configured to match the Sandia corporate Windchill production systems to increase their value as learning and development platforms.

2.3. Development Server Implementation

The existing unclassified and classified AEE development servers are three-tier systems that each consist of an Oracle database server, a Windchill applications server, and an Arbortext server. The new servers are comparable three-tier systems with updated hardware (refer to Appendix A). All of these systems are based on the Windows operating system (OS). The new systems use the 64 bit Windows 2003 Enterprise Server OS, while the old systems are 32 bit. To ensure flexibility to meet the current AEE requirements as well as potential future uses, the new servers have the capability for hard drive expansion or connection to a storage area network (SAN), should they be converted from development servers to full production systems.

Because the Arbortext application is currently not in use at Sandia and would not be used for any of the pilot activities during this AEE project, the AEE team decided to deploy only the new Oracle and Windchill servers at this time. The Arbortext servers are physically installed, but they were not configured and tested with the Arbortext application. The team decided to wait to deploy them until a need arose, rather than deploying them and paying the software maintenance fee while they went unused. For any casual use, the Arbortext application is still available on the two older AEE servers, which have not yet been decommissioned.

2.3.1. System Installation

The AEE team contracted with PTC to provide support for the installation, testing and verification of Windchill 9.0, maintenance build M070, on the new servers. The PTC field engineer, Ralph Goodwin, worked with the AEE IT team to complete the Windchill installations. Because the AEE team had already deployed Windchill 9.0 during the FY08/09 AEE project, it was expected that the Windchill installation on the new servers would be relatively uneventful. While that was generally true, the installation team did encounter a number of issues that are summarized here:

- Thumbnail images for Office documents (Word, Excel, and PDF) could not be seen or marked up in Windchill. The AEE IT team learned that Microsoft Office worker agents do not work with the 64 bit Windows OS. This issue is not resolved on these systems. The OOTB Windchill solution is to use FTP to connect to a separate 32 bit platform to run the worker agents. Because FTP is not allowed on the classified network, and it shows up as scan vulnerability on the unclassified network, this method is not acceptable. An alternative that was considered, but not implemented, was to set up a separate 32 bit workstation on a private network with the Windchill server to run the worker agents.
- The Windchill method servers, which are a core backbone function for Windchill did not all come up on a system restart. The work-around was to go into the Task Manager and manually restart them.
- There was a problem opening Office 2007 documents which was resolved by upgrading the object adapters.

- There were application compatibility issues with some client machines that were running different builds of Pro/ENGINEER than the AEE servers. The client machines were updated to run Pro/ENGINEER Wildfire 4 M092 to resolve the problem.
- Errors occurred when loading two Windchill lifecycles because the naming scheme was changed to “NSE” when it referred to “NWC” in the load files. The reference was corrected to “NWC” to resolve the problem.
- There were problems exporting library files from the NM production system and loading them onto the unclassified AEE development server. The AEE team determined that this problem was due to the large number of files and instances in the library. This was resolved by exporting/importing small groups of files rather than the entire library.

When the problems were resolved or workarounds were established, the team preformed a basic system validation by performing “picks and clicks” testing of Windchill functionality. This was followed by system memory and performance tuning. Installation details are documented in “Windchill PDMLink 9.0 M070 Installation and Configuration for Sandia National Laboratories.” PTC cautions that this document is applicable only to the AEE development servers and should not be used as a configuration standard.

2.3.2. System Configuration

The AEE team’s intent was to configure the AEE development servers to match the Sandia Windchill 9.0 production servers. Past experience had shown that even slight differences in the Windchill configuration between systems could lead to different results for the same actions. This complicates the development of processes that need to be deployed across multiple systems.

The AEE team used a three step process to configure the servers:

1. Configure to the PRS12005 Issue A specification.
2. Modify the configuration to match the changes that were applied to the older AEE servers when Windchill 9.0 was installed in FY08 (as detailed in “*NWC PDMLink Installation Guide Version 9.0 – Updated for the Sandia-CA Installation*”). This consisted primarily of a number of constraints that needed to be installed.
3. Configure to the Sandia production Windchill 9.0 requirements (those that apply).

The AEE team used an early version of a configuration guide⁶ from Jason Kritter (Org. 2004) as a reference to configure the servers to match the Sandia Windchill 9.0 production systems. Because the production servers are UNIX platforms and the AEE development servers are Windows platforms, much of the content of Jason’s procedure was not applicable. The applicable content included product and library configurations that can be loaded from templates. Prior to loading these templates, the Windchill 9.0 OOTB templates had to be deactivated.

The AEE team discovered that there are differences between the Windchill 9.0 production system configuration and the configuration specified in the PRS12005. The production system follows the convention of using all capital letters with underscores rather than spaces for naming

⁶ SNL PDMLink V9.0_M070 SPDM V1.0 Release, January 2010.

the key fields for all of the Windchill “Roles”. The PRS12005 calls for spaces rather than underscores. The only problem the team observed due to this difference was that a Sandia promotion request lifecycle failed to import into the AEE server. The lifecycle was recreated on the server following Jason’s guide to resolve the problem. Other objects may be affected as well, but no others were identified by the team.

Another difference between the Windchill 9.0 production system configuration and the PRS12005 configuration was that attribute descriptions were more detailed and logical identifiers were omitted for the production system. To resolve the difference, the team removed the logical identifiers and replaced the descriptions. They later found that the ML report generator that was created and used for the product structure pilot relied on the logical identifiers to gather data, and they had to add the logical identifiers back for the specific attributes used to generate the ML report (cage code, description, etc.).

The team also noted that there is an issue with attributes with the same name that appear in different attribute organizers (Attribute Manager and Type Manager), which results in multiple listings of the same attribute name when selecting search criteria. An example is “sigma level” which appears under both “Enterprise Product Model” (EPM) and “WTDocument” objects in the “Pro/ENGINEER” and “Document” attribute organizers. A solution is to use separate names for the attributes, such as “Doc sigma level” and “CAD doc sigma level”, or use a single attribute from either the “Pro/ENGINEER” or the “Document” attribute organizer, but not both. This will eliminate confusion when defining folder and workspace views. Having the same attribute name in both classes will result in two duplicate columns within a folder or workspace view.

The AEE team quickly came to the realization that minor differences in Windchill configurations can lead to different outcomes when using the tool. The team documented their deviations from the Sandia production Windchill configuration in a project document, “Configuration of AEE Servers to SNL/NM Production Settings.”

In summary, the AEE Windchill servers were configured as follows:

- At site level, views were created using the “View Manager” utility.
- The “Design” view was set to be the default at the site, organization, and product levels.
- Four single level “child” views were created under the default “Design” view: Design, Manufacturing, Quality, and Surveillance.
- The “Org Level” preference was set to use the CAD model parameter values “BOM_Description_Asm” and “Part_No” for the WTPart name and number.
- Object initiation rules (OIRs) were created for the following:
 - WT Parts were sent to a “Product Structure” folder and “Auto-numbering” was disabled.
 - Baseline objects were directed to a folder.
 - Auto-numbering for “AEE Document Types” was disabled.

- Attributes were created to support five document soft types. All attributes were of string type except for “MLReport” which is Boolean. The five document soft types included four “described by” documents and one “reference” document:
 - “Described by” soft types:
 - AEE General Document
 - AEE Part Defining Drawing
 - AEE Sandia Spec
 - AEE Support Drawing
 - “Reference” soft type:
 - AEE EA Document

The AEE IT team intended to configure the AEE servers to self-generate an SSL certificate to enable using the HTTPS protocol, which is a corporate requirement. However, they were unable to complete this task because:

1. The team received guidance that self-generated certificates are not allowed at Sandia.
2. The PRS12005 does not allow self-signed certificates.
3. The approved method at Sandia is to use VeriSign, an outside company for the verification certificate. The team was unable to work out a process to accomplish this within the project time constraints and they moved forward using HTTP protocol.

The AEE IT team wanted to use Kerberos user authentication on the Windchill servers so that individual user accounts would not have to be created; however, they found that no one at Sandia has been able to get Kerberos to work with Apache on a Windows server. Apache is the web server for Windchill. Due to time constraints, the team chose to continue to set up individual user accounts.

2.3.3. Classified Development Server Implementation

After all the significant installation and configuration issues were resolved for the unclassified Windchill 9.0 development server, the AEE team proceeded with the deployment of the classified development server. Configurations were copied from the unclassified system and moved to the Sandia Classified Network (SCN) so that they could be installed on the classified server using the Sandia Reliable Automated File Transfer Service (RAFTS). The team saw the same issues they had encountered with the unclassified server installation and were able to resolve them to the same extent.

2.4. Training Server Implementation

The AEE team established a monolithic training server during the FY08/09 project to serve as a platform for Windchill training for Sandia designers and engineers. With PTC support, the team had attempted to save time by cloning the installation from the unclassified three-tier Windchill 9.0 development server rather than performing a full installation from scratch on the training server. This approach was unsuccessful due to the differences between the monolithic and three-tier systems. The cloned application contained references to the separate Oracle server, which did not apply with the monolithic training server. This occurred at the end of the FY08/09 AEE project and there was no time for debugging the installation.

To resolve the training server problems, the FY09/10 AEE team performed a complete reinstallation of Windchill 9.0 on the monolithic training server. The server was then configured using the same settings that had been used for the new three-tier unclassified and classified AEE development servers. The team encountered a few minor problems with the installation and configuration, but those were resolved and the system was put into service for the first of a series of Windchill training classes for Sandia designers.

2.5. Conclusion/Recommendations

The AEE team learned that each NSE site has significant differences in their Windchill 9.0 configurations. The PRS12005 is not specific enough and does not provide examples, and it is therefore open to interpretation. For example, the PRS12005 states the “Role” names with spaces between words in the names. The Sandia production Windchill 9.0 systems are on UNIX platforms, which do not accept spaces in names, thus an underscore is used. To add to the confusion, the defining name of an attribute in Windchill does not accept spaces, and therefore, underscores are required to separate multiple word names. The PRS needs to be more specific and should include examples to fully clarify how to implement configurations in Windchill.

Sandia should adopt a strategy of reporting all serious system problems through the PTC Technical Support Action Manager (TSAM) so that these problems are logged and the information is available to all other NSE sites. This service is paid through the NSE Supplemental Pricing Agreement (SPA) with PTC. An advantage of using this service is that one PTC employee manages all of the critical calls for NSE trouble tickets and will therefore be aware of similar problems that are occurring at the various sites.

3.0 Windchill 9.1 Implementation Pilot

3.1. Summary

Sandia/CA's AEE project team piloted a Windchill 9.1 update on a classified Windchill 9.0 development server (containing CAD and other data) to determine the level of effort required and to identify any potential problems associated with the update. This report is a summary of the work for the benefit of other NSE sites. However, this work is specific to the AEE Windows servers, and therefore, the results may not be directly applicable to other Windchill systems. The software installation and performance tuning were successfully completed with support from Parametric Technology Corporation (PTC) staff in about seven days of effort with only one notable issue: the ProductView thumbnail viewer for Pro/Engineer files did not function within PDMLink. The source of that problem was identified and resolved with assistance from PTC Technical Support.

The AEE evaluation team ran a series of tests with PDMLink 9.1 to validate the functionality of the updated application. Initial tests were based on a Windchill 9.0 validation test plan developed by staff at Sandia/NM⁷. The test plan was completed on a disk-full Windows workstation on Sandia's classified internal network (SCN) and then repeated on a diskless Windows workstation on the same network.

All of the tests were completed successfully using both the disk-full and diskless machines, with the following two exceptions:

- CAD assembly files residing in PDMLink 9.0 on the AEE server prior to the Windchill 9.1 update do not show any information for structure when opened in PDMLink 9.1. For newly created files, and for files imported into a workspace and checked into PDMLink after the update, structure is displayed correctly on their information pages.
- The Pro/Engineer client within PDMLink consistently crashed when a family table generic file was removed from the workspace. This was resolved by updating Pro/Engineer Wildfire 4.0 to version M091 on the application server the AEE team was using. PTC recommends version M090 or higher for compatibility with Windchill 9.1.

The AEE team ran a number of additional tests to validate functions that were not addressed by the Windchill 9.0 validation test plan. The application functioned as expected with the following exceptions:

- Many of the PDMLink "Help" functions did not work as expected. PTC representatives told the team that the help system in Windchill 9.1 is completely new and is web based, which makes many of its functions unusable from Sandia's restricted internal classified network. This limitation will be an issue for other NSE sites as well. The PTC representatives were discussing a solution that would provide a download of the full help system for offline use.
- ProductView did not work within PDMLink. This was presumed to be a compatibility issue, as the ProductView 9.0 client the AEE team was running on their Windows

⁷ SNL PDMLink 9.0 Production Test Plan, June 2009.

workstations was not compatible with PDMLink 9.1. Although the team did not pursue this, installing the ProductView 9.1 client should have resolved the issue.

In conclusion, the Windchill 9.1 update was successfully implemented in this limited pilot. All CAD data and documents residing on the server remained fully accessible after the completion of the update. The AEE team is working with PTC staff to resolve the few problems that were identified through validation testing. This update does require the installation of Pro/Engineer Wildfire M090 or higher, which was not the version used in production throughout the NSE at the time of this evaluation. In addition, an alternate solution for the web based Windchill “Help” system may be required. The AEE team will continue to work with PTC to resolve any additional workspace or client issues that may arise.

3.2. Background

At the April 2009 Model Based Integrated Tools (MBIT) conference at Sandia/CA, representatives from a number of NSE sites expressed concern for the level of effort required to implement PTC’s Windchill 9.1 maintenance build. The AEE project team volunteered to pilot a Windchill 9.1 update on one of its development servers and report the results to all NSE sites.

The AEE team used its existing classified Windchill 9.0 development server for this pilot. This server will soon be replaced by a new AEE development server, which made it available for the update with minimal risk to current users. The classified server currently hosts a modest amount of MCAD data and documents (approximately 3000 files); this data was backed up prior to the update.

The classified AEE development server is a three-tier, Windows Server 2003 system with Windchill, Oracle and Arbortext applications running on separate servers. The PTC applications on the Windchill server prior to the Windchill 9.1 installation were:

- Windchill 9.0 M030
- Windchill PDMLink 9.0
- Windchill ProjectLink 9.0
- ProductView Standard 8.0
- ProductView Client 9.0
- Pro/Engineer Wildfire 4.0, M040

3.3. Windchill 9.1 Implementation

PTC describes Windchill 9.1 as an update, rather than an upgrade. The upgrade manager application is not used to implement this update.

The initial Windchill 9.1 update consisted of the following applications:

- Windchill 9.1, M020
- Windchill PDMLink 9.1, M020

- Windchill ProjectLink 9.1, M020
- ProductView Standard 9.1, M040
- ProductView Client 9.1, M040
- Pro/Engineer Wildfire 4.0, M090

A PTC consultant assisted the AEE team with the Windchill 9.1 update installation. Upon completion of the installation, he ran a series of basic validation tests:

- Checked the installer logs and verified there were no errors.
- Performed application “picks and clicks” to validate basic system functions.
- Verified all page tabs were functional.
- Confirmed connectivity between the client and server, both from a local browser and from a remote system.
- Tested applets, such as lifecycle administrator, principle administrator and file vault administrator.

The only problem revealed by the tests was that the Pro/Engineer file thumbnail image generation did not function within PDMLink. A trouble shooting process was initiated and the problem was resolved with the following three steps:

1. A typo in the user name that the thumbnail application uses to authenticate in Windchill was corrected.
2. Per the suggestion of PTC Technical Support, Pro/Engineer was updated to Wildfire 4, M091 to resolve a security issue within M090.
3. The ProductView 9.1 adapters were installed. These had been missed in the initial installation.

Although the installation, performance tuning and problem resolution occurred over a three week period, the total actual time spent on the process was about seven days.

3.4. Windchill 9.1 Validation

Once the basic Windchill 9.1 operational tests were completed by the PTC consultant, the AEE team conducted functional testing to validate the applications. They used the *SNL PDMLink 9.0 Production Test Plan* as the basis for their initial test process. This plan was developed by a team led by Abe Sego at Sandia/NM for validating their Windchill 9.0 production system.

The test plan was initially conducted from a disk-full Dell model 670 workstation running Windows XP SP2. This workstation resides on the SCN. The data that was used for these tests consisted of B83 files already within PDMLink, test assembly data files that were imported into the workspace and checked into PDMLink, and newly created files as required by the tests.

Because the AEE development server was not fully configured as a Windchill production system, a number of the tests in the *SNL PDMLink 9.0 Production Test Plan* were not applicable.

For example, the system was not configured for “member” roles, and therefore, the tests for this function were omitted. The following are the specific tests from the test plan that were not conducted: 9.1-9.5; 9.10, 9.15, 9.17; 11.2; 12.1-12.22; 17.1.

The remainder of the test plan was completed successfully with the following notes and exceptions:

- Test 2.4. B83 assembly files on the AEE server did not show any information for structure, where used, relationship report, and representations after the Windchill 9.1 update. These CAD assemblies open correctly and appear to contain all of their original components. Modifying the assembly and checking the file into PDMLink restores the structure in the detail page. New files or files migrated after the Windchill 9.1 update display structure correctly on their detail pages.
- Test 2.7. The Pro/Engineer client within PDMLink consistently crashed when a family table generic file was removed from the workspace on both the disk-full and diskless clients. This problem was consistent when tested with both existing and newly created library parts. The team discovered that the only case in which this problem did not occur was when the workspace contained all of the instances of the generic file. In practice, an assembly will typically include only a limited number of instances from a family table, not all of them. The AEE team subsequently found that the problem was resolved by upgrading Pro/Engineer Wildfire 4.0 to M091. Although this version had been installed on the Windchill 9.1 server, the team had been using version M040 from an application server for the tests. PTC requires version M090 or higher for compatibility with PDMLink 9.1.
- Tests 7.1-7.3. The out of the box (OOTB) promotion request was successfully used for this test, as opposed to the Sandia-specific promotion process.

The shortened test plan was repeated on a diskless Dell model 690 workstation running Windows XP SP2. This workstation also resides on the SCN. The team’s objective was to determine if the diskless environment introduced any problems that were not observed with the disk-full client. The results were identical to the tests on the disk-full client, with all tests passing with the exception of tests 2.4 and 2.7, which displayed the same problems that were observed with the disk-full system.

3.5. Additional Windchill 9.1 Testing

The following additional tests were conducted using the disk-full and diskless clients to validate PDMLink functions that were not addressed by the plan and to determine if past problems experienced with the classified AEE Windchill 9.0 system were resolved.

- The OOTB promotion approval reviewer workflow process was run and performed as expected.
- The AEE team successfully opened the B83 Aft Assembly model (existing 9.0 data) which contains over 2000 components and includes cables, inheritance features, and flexible components. The model behaved as expected with no changes in performance over Windchill 9.0.

- The “import to workspace” function was used to successfully migrate two assemblies into a PDMLink workspace:
 - A fifty-five component neutron generator cover assembly which included cables and library parts was migrated using the disk-full client.
 - A B83 firing set case assembly was migrated using the diskless client.
- Multiple file check-in functioned as expected on both the diskless and disk-full clients. Multiple file check-in had been the single biggest issue for users of the AEE classified Windchill 9.0 system. This problem seems to be specific only to this system, as there are no comparable problems with the unclassified AEE Windchill 9.0 system, nor the unclassified Sandia/NM Windchill 9.0 production system. The previously mentioned neutron generator cover assembly was checked-in using the disk-full client and a new six part B83 subassembly was checked in using the diskless client.
- The “export from workspace” function was verified using an internal threaded fastener.
- The team verified the “revise and checkout” function which automatically rolls the file revision and checks out the file.
- The PDMLink “Help” function is not fully operational. PTC representatives told the team that the help system in Windchill 9.1 is completely new and is web based, which is unusable from the SCN. This will be an issue for other NSE sites as well.

The AEE team experienced a problem using ProductView on both the disk-full and diskless workstations, which was presumed to be a mismatch between the application and client. The ProductView 9.1 client had not been installed on the workstations, and the ProductView 9.0 client they were running is not compatible with PDMLink 9.1. Although the team did not pursue this, installing the ProductView 9.1 client should have resolved the issue.

3.6. Conclusion

Compatibility between Windchill 9.1 and the associated applications and clients was the source of some of the problems the AEE team encountered with the implementation of the update. A better understanding of the specific requirements and the current PTC software releases could have prevented these problems.

Through the course of the validation testing, the AEE team had the opportunity to try some of the new features included in Windchill 9.1 and to get a sense of how it performs compared to Windchill 9.0. In general, they were pleased with both the performance and the new features.

The AEE team successfully implemented the Windchill 9.1 update with no significant software problems. Some of the functional problems that were identified through the AEE team’s validation testing were attributed to compatibility issues between the various applications and Windows clients, which emphasize the need to fully understand the requirements for all of the various applications prior to the implementation of an upgrade.

This page intentionally left blank.

4.0 Product Structure Pilot

4.1. Summary

At Sandia National Laboratories, a product structure is defined as the hierarchical representation of all of the data required to define a part, including the material list, support drawings and/or models, specifications, and other relevant documents. The part defining material list (ML) identifies all of this data. This definition is not universal throughout the NSE. The PRIDE Product Structure project goal is to deliver a single set of Product Structure Specifications that can be referenced and aligned to by PRIDE implementation projects across the NSE⁸. To support that effort, the AEE completed a pilot effort to define a product structure for a mechanical product within the Windchill PDMLink 9.0 product lifecycle management (PLM) application. The main thrust of the pilot was the implementation of four use cases that represent the typical lifecycle of a NW component design at Sandia, including the creation of the product, and design revisions and suffixing changes as the design is modified. The AEE team selected the use cases to align to PRIDE objectives and to keep the scope of effort focused to what they felt could be reasonably achieved within the timeframe of the pilot. The team used a simple pen model as the product for the use cases to avoid unnecessary complications. This pen model is being used by the PRIDE Product Structure team and other NSE sites to enable direct comparisons of their product structure work.

The AEE team began their effort by meeting with staff from Sandia/NM, LANL and LLNL to understand the product structure work the staff at those sites had already completed, and to understand where they were headed with future efforts. The AEE team also worked with the Sandia PRIDE Product Structure team to ensure that the work on this pilot would be directly beneficial to the PRIDE effort.

As a foundation for the use case development and implementation, the AEE team worked with a process architect from PTC to conduct a high-level business process analysis. The goals were to define Sandia's current design/product definition processes, propose a future process using the PDMLink 9.0 tool, and identify any significant issues or gaps between those processes and the tool capabilities. The team took a traditional approach by looking first for PDMLink tool solutions that would work without significant changes to Sandia's current processes. A requirement for the pilot was that there would be no customization of PDMLink. The intent was to use PDMLink in its out-of-the-box (OOTB) state, with the expectation that some tool configuration would be required.

In parallel, the AEE team worked with a PTC solution architect to design and implement the four use cases in PDMLink. In summary, a product structure was created for a simple three-part pen cartridge model by using its CAD model structure as the basis for creating a WTParts structure in PDMLink. WTParts are PDMLink objects that represent the elements of a design. Additional WTParts were then created for support drawings, specifications and the bill of materials. This was a hybrid between a model-centric and parts-centric approach to creating the product structure. The use cases also demonstrated how to handle revisions for minor design changes and suffixing for major changes. Because the primary goal of this pilot was to demonstrate an ML-

⁸ *PRIDE Product Structure Specifications: Design Definition*, June 2009, page 5.

based product structure, the PTC solution architect developed a part defining ML report to represent the product structure using the Info*Engine application that is included in the Windchill software suite. This example PDMLink ML report contains most of the information contained in a current Sandia ML report although it is dissimilar in appearance.

The AEE team's effort to develop a proposed future design/product definition workflow within PDMLink 9.0 identified a number of issues and gaps that need further discussion and resolution. These include Sandia business process issues that need to be better defined and specific gaps between the PDMLink tool capability and Sandia business processes. A detailed listing of these issues and gaps can be found in *Section 4.9. Issue and Gap Analysis*. A summary of some of the high-level issues follows:

- There is a significant learning curve for understanding how to use the Windchill PDMLink 9.0 tool to meet the design/product definition requirements at Sandia and the other NSE sites. Windchill generally conforms to commercial CAD design practices, which can differ from those mandated at the NSE sites. As this pilot demonstrated, the tool does not accommodate all of Sandia's business needs in its OOTB configuration.
 - As an example, the Windchill OOTB report generator tools could not produce an ML report with all of the information required by Sandia. These reports show only the parts, not the associated documents that have to be on the Sandia ML report, such as material specifications. To bridge the gap for this pilot, a new report type had to be created using OOTB tools included in the Windchill tool suite.
- The same task can often be performed in different ways within PDMLink and the results can differ. Policies can be set up in PDMLink to control some of these instances, but likely not all of them.
- A general question for the deployment of PDMLink is: Who will be responsible for creating the product structure in the Sandia design/product definition process? The AEE team's assessment is that this process needs to be controlled for consistency in application. That will require the person who is entering data and creating the product structure to have a working knowledge of the PDMLink tool and perhaps Pro/ENGINEER as well.
- There is a need for consistency in the definition of terms by all of the NSE sites and with PTC. Many of the commonly used design/product definition terms have multiple meanings that can lead to confusion for the implementation of business processes in PDMLink. An example is the term "released" which has one meaning within PDMLink and multiple meanings at Sandia. Another example is the term "baseline."
- The guidance provided by the PRS12005 specification for product structure needs to be revised to provide clear guidance for the NSE sites. The AEE team found the guidance for using view versions in PDMLink to be very confusing.
- Resolving gaps between traditional business processes and PDMLink tool capabilities may in some cases require changing the business process. While this may be a logical part of a transformation effort, there may be considerable pressure to resist changing processes that have been in use for many years. Sandia's transition to PDMLink as its

corporate production PLM provides an opportunity to examine current business processes and determine where process transformation is not only desirable, but necessary.

The AEE team's product structure work is documented in a number of presentations and reports in addition to this project report. The team has shared its work with PRIDE team members at SNL/NM, LANL and LLNL through a series of presentations. They also hosted a hands-on workshop for the Sandia System Bill of Materials (SBOM) team who will continue the development of the product structure definition in PDMLink.

Because this pilot used a simple three-part pen cartridge as the model, a logical next step is the use of an actual weapons model to fully expose issues and gaps that will occur with a complex design with hundreds or thousands of parts and related documents. Based on the issues that were identified with the simple model, the AEE team expects that significant additional issues will surface with the weapons model use case.

4.2. Background

At Sandia National Laboratories, a product structure is defined as the hierarchical representation of all of the data required to define a part, including the material list, support drawings and/or models, specifications, and other relevant documents. The NSE PRIDE Product Structure Team is surveying the NSE sites with the intent of developing a common definition of product structure for design definition⁹. The development of a common definition has added importance as NSE sites move forward with the implementation of a new electronic enterprise PLM environment using the Windchill PDMLink application. Product structure is currently instantiated differently at the individual NSE sites, and therefore, the implementation of product structure in PDMLink could vary significantly from site to site. A common definition of product structure and a set of requirements are a necessity for sharing design definition data among the NSE sites.

As an example, LANL initiated their implementation of product structure in PDMLink with a model-centric approach where all elements are treated as WTParts, which is an object that represents physical parts or assemblies in the PDMLink application. A model-centric approach creates a product structure from the CAD model structure using WTParts to represent all of the elements. While this approach may work for LANL, Sandia requires an approach based on the bill of materials (BOM), as the weapons designs that Sandia works with have far more elements than just the parts models. Examples are non-modeled elements such as procedures, specifications, and materials (grease, for example). Sandia also has to address the design revisions which are identified by an issue and/or a suffix change, and legacy designs that have no CAD model. Alternates to the **model-centric** approach are a **part-centric** approach in which WTParts are created and the product structure is built without any CAD-based data, or a **hybrid** approach that uses a combination of the model- and part-centric approaches.

At the inception of the AEE FY09/10 project, the PRIDE team was focused only on the NSE sites' data migration efforts to move MCAD legacy data from existing systems into their new Windchill PDMLink PLM systems. The PRIDE team's next planned step was to develop a

⁹ *PRIDE Product Structure Specifications: Design Definition*, June 2009, page 6.

common definition of product structure within the NSE. The AEE team's intent was to conduct a preliminary product structure pilot in advance of the PRIDE effort to provide a foundation for the PRIDE team to work from as they move forward.

With that understanding, the primary focus for the AEE product structure pilot was to implement a BOM-based product structure for a mechanical product that is representative of the nuclear weapons design and engineering work conducted at Sandia. To minimize complications, a simple pen model that was developed by Y12 for the PRIDE Product Structure team was chosen as the MCAD model for this pilot effort¹⁰. By using the same pen model, direct comparisons of the product structure work at the NSE sites is much easier than if each site uses their own unique model.

A primary tenet for this pilot was to develop a product structure definition using PDMLink in its OOTB state, with no customization of the base code. Changes to configuration, which will not be impacted by PDMLink product upgrades, were considered allowable. The intent was to leverage the OOTB product to meet business needs, and to identify gaps where the product cannot accommodate current business practices. In those cases, the business model itself should be examined to determine if it should be changed to meet the capabilities of the PDMLink tool, or if an alternate solution is required.

4.3. Alignment with PRIDE Product Structure Effort

The goal of the PRIDE Product Structure project is to deliver a single set of product structure specifications that can be referenced and aligned to by PRIDE implementation projects across the NSE¹¹. To ensure that the work performed on this AEE product structure pilot was fully in line and directly applicable to the larger PRIDE product structure effort, the AEE team initiated this pilot with several meetings with the PRIDE Product Structure team. The PRIDE team presented the work they had completed to date and provided guidance for the AEE team's project focus.

Based on the PRIDE team's input, the AEE team determined that the demonstration of a BOM-based product structure definition with several typical design definition use cases was a priority. A specific request from Tim Meeks, the PRIDE Product Structure Chair, was to look at an ML and determine how to populate it in PDMLink. All of the data identified in the fields of the current ML have to reside somewhere in PDMLink. He also suggested that the format of the ML can be changed if it does not meet current needs, as it has been in its current form for a long time.

AEE team members and the AEE's PTC process architect met with PRIDE team members to discuss possible use cases to implement in the pilot project. In addition, the PRIDE team provided the AEE team with their high level use case model that they developed for their prototyping activities (Figure 2). This diagram is specific to mechanical parts only to limit the scope of effort. Electrical parts are not specifically addressed and it was thought that they can be handled as product support documents. The AEE team used this diagram as the basis for

¹⁰ *PRIDE Product Structure Specifications: Design Definition, Version 2.0*, June 2009, Appendix C.

¹¹ *PRIDE Product Structure Specifications: Design Definition, Version 2.0*, June 2009, page 5.

developing their use cases. Their intent was to address as many of the specific activities from the process for defining a part as was reasonably possible within the AEE pilot's scope and timeframe.

After considerable discussion and deliberation, the AEE team decided to move forward with the product structure definition using the PRIDE Product Structure pen model with four use cases that represent a product lifecycle:

- A new product introduction.
- A simple change requiring a part revision (no form fit or function change).
- A complex change to a component (a change to form, fit or function) requiring a suffix change to the component and the next higher level assembly.
- A second complex change to the component requiring a suffix change and a tabulated part defining drawing.

Figure 2. PRIDE Product Structure "Define a Part" use case diagram¹².

¹² PRIDE Product Structure Specifications Development General Briefing, July 2009, slide 20.

These four use cases could be considered a single use case with four separate stages, but for the purposes of this pilot project, they were treated as four individual use cases. These use cases provided the opportunity to address many of the activities from the PRIDE use case diagram. The following is a list of specific design/product definition activities the AEE team intended to exercise with the use cases and an explanation of the planned approach:

- Create a PDMLink product-part support document:
 - This will be represented by a content document file (Pro/Engineer drawing file, other CAD drawing file, or an image file) associated to its related WTPart. Associated content will be extracted in a report to appear on the ML.
- Create a PDMLink non-CAD product-part (includes special design cases, commercial product-parts, and the public domain library):
 - These will all be represented by WTParts populated with defined attribute data. They will appear on the ML.
- Create a PDMLink material:
 - Material will be represented with a document associated to its relevant WTPart. Associated content will be extracted in a report to appear on the ML.
- Create a PDMLink product-part structure:
 - Product Structure will be represented by a hybrid of model-centric and part-centric practices using WTParts. The “Design” view will contain all data related to development and release. Supplemental data outside of the design realm will be represented by other views (Manufacturing, Quality, and Surveillance).
- Derive a product-part defining ML drawing viewable:
 - This will be derived in the form of a report generated from the product structure and the product structure content (associated documents and CAD documents).
- Create a PDMLink product-part defining drawing:
 - The derived document (viewable) for each released configuration will be associated to a base numbered WTPart End Item. Each released product-part structure will be represented by a saved configuration and a CAD document baseline object. Individual product structures (copied from the base “-00” suffix) will represent suffix changes.

4.4. Product Structure Work at Other NSE Sites

As a starting point for the AEE product structure pilot, team members met with staff at Sandia/NM and Los Alamos National Lab (LANL) to discuss and learn from their product structure work. The team also wanted to establish lines of communication with these staff to ensure that the AEE product structure work would be fully communicated to those sites.

The team began with a review of a PowerPoint presentation that Gene Kilmer at LANL had created to show the product structure development work he had completed in PDMLink using a simple pen model as the subject. Although the presentation was extensive, it was short on explanatory text and the AEE team had difficulty understanding all of the content that was presented. These slides gave the AEE team an idea of where they wanted to go, but did not provide the detail of how to implement this approach. The approach was also very generic in that

it applied to a typical commercial product with commercial processes. It did not demonstrate NSE practices such as suffixing.

The AEE team then visited Gene at LANL to understand how he had implemented the product structure and to determine if his approach would work for the AEE pilot. Gene's approach was model-centric, with all elements designated as WTParts in PDMLink. The other notable aspect of Gene's approach was his use of multiple view versions and instantiations in PDMLink. Design data was broken into different view versions. When a new view version was created, the associated content of a WTPart would be copied to the new view version (documents and CAD drawing documents), but not the associated CAD model. New versions were implemented as revisions to the original WTPart design view. Every time the original WTPart was revised, new supporting view versions had to be created as well. This approach requires considerable effort to manage all of the various view versions.

While at LANL, AEE team members also talked with Chris Scully, LANL's PRIDE MCAD representative, about his perspective for where LANL was going with product structure. They had already initiated a product structure implementation in their Windchill 8.0 system and were looking to make improvements using Windchill 9.0. Chris said they were considering using configuration names to handle suffixing to reduce the duplicated product structures they were seeing with their Windchill PDMLink 8.0 practices.

AEE team members also met with Rick Chavez-Hatton at Sandia/NM to discuss his PDMLink product structure work. Rick was early in the learning process, but he had created WTParts to represent satellite components on the Sandia PLM sandbox server. In his approach, Rick was associating documents to WTParts. A WTParts revision was not synchronized with associated CAD documents and no associations were made. Rick was using end item configurations to historically preserve CAD structure.

The AEE team came back with the idea that using view versions would be a good starting point for the AEE pilot and would be compliant with PRS12005 as they understood it. This approach would also be consistent with the direction LANL was going.

4.5. Initial Product Structure Development Work

In preparation for their product structure work, the AEE team looked at the course offerings in PTC's Precision Learning resources and the primary team members completed two courses to get a basic understanding of the tool: "Windchill PDMLink 9.0 Product Structure Management" and "Windchill PDMLink 9.0 Data Management." Although the courses provided some helpful information for how to do things in PDMLink, they did not provide an explanation of the "why" and "what for" aspects of the tool. This training was intended only as basic information. The team clearly needed assistance from PTC consultants at this point, which was the original intent when the AEE product structure pilot project plan was developed in concert with the PTC business development manager and the PTC NSE project manager. The essential learning was intended to come from PTC consultants who would provide hands-on guidance for the team as they worked together on the development of the product structure definition.

Based on project planning discussions, the AEE team had expected that PTC would assign Rusty Abernathy to the product structure project. Rusty is a PTC solution architect who has working knowledge of NSE business processes and experience with the PDMLink tool and the team felt that his expertise would be highly beneficial to their development work. Unfortunately, when the AEE team initiated its work, Rusty was committed to several other projects, including the very high priority Sandia MCAD data migration project. Due to his limited availability and time constraints, the AEE team had to move forward with limited guidance from PTC at this point. Based on information provided by PTC, and the information they had gathered by visiting with staff at other NSE sites, the team decided its first step would be to investigate if a product structure could be defined using view versions in PDMLink.

4.5.1. Using View Versions in PDMLink

The AEE team considered the NSE Product Realization Standard (PRS) 12005 a reference for its product structure work plan. The PRS specifies the configuration settings that should be used for PDMLink to maintain tool uniformity across NSE sites. Included in the as yet unreleased updated PRS 12005 Version B is a set of PDMLink view versions. It is implied that these various view versions (CAD, manufacturing, quality, etc.) should be used for the development of product structure. The PRS is very specific in defining a view version structure where the design view is a parent to all other views. Each view is named for a category of design data which implied that all views were essential to a successful product structure. Based on this information and a number of conversations with PTC's Rusty Abernathy, the AEE team presumed that it would need to implement a product structure definition using view versions. The team moved forward to understand how to create, edit, and use view versions in PDMLink. Objectives included:

- Learn how data is managed in view versions.
- Understand revisions, iterations and releases: what is their impact on the stored data?
- Determine how configurations preserve or interact with view versions. The team believed that view versions were captured by configuration.

The team spent a considerable amount of time trying to implement LANL's approach to product structure using view versions. The following assumptions were made:

- One must address the entire product life cycle with all of the various views (CADs, manufacturing, quality, etc.), not just a design or a manufacturing view.
- View versions are meant as a storage area to collect data to describe product structure.
- For the design agencies (DAs) the EBOM view is where they create WTParts and views are configurations.
 - Note: In this context, EBOM is a view version within PDMLink.

The AEE team used the three-part cartridge assembly from the simple PRIDE Product Structure pen model and represented the model as two fabricated parts and one non-model part (ink). They created mock PDMLink documents to represent manufacturing, inspection and material list documents. Their intent was to populate the WTParts (the placeholders in PDMLink) with the pen model parts using view versions as storage locations. However, when they created configurations of these view versions, only the design view was captured, not the various

multiple views, which left them wondering how they could preserve all of the configurations using multiple view versions. The team also observed that the maintenance of multiple view versions, namely creating views, linking and unlinking files, and releasing the views, seemed labor intensive and prone to human error.

The team thought that by creating named configurations, they could capture suffixing and revisions. Their plan was to save configurations with a naming convention. For example:

- The cartridge structure was represented with a WTPart structure.
- Four view versions were made (CAD Structure, EBOM, MBOM, and Quality).
- The associated documents were placed on those versions:
 - The CAD structure view held all associated CAD docs.
 - The EBOM view held documents that one would expect to see on the ML.
 - The MBOM view held all manufacturing documents.
 - The Quality view held all inspection docs.

A configuration was created from the CAD structure view with the expectation that it would capture all of the view versions and the data associated with that configuration. The naming convention used was 100010-00-A (base#-suffix-revision). The team found that the configuration could only be captured as single view. Because the data was placed only in the appropriate view versions, additional configurations would be required to capture the data in the other view versions. Baselines and End Item configuration only apply to a single view. Multiple view versions would require multiple baselines and configurations. It was also not possible to view all of the product structure links in one place using configuration alone.

The use of view versions spawned concerns related to the release of data. For example, because a view version is an iteration of a WTPart, with each new view version one would create another WTPart that has to be released and revised. All of these view versions would have to be managed and each would need the support of a separate configuration that would also have to be managed.

To illustrate, what appears to be a single WTPart with four views is represented in PDMLink by four WTParts:

- 100010-00 A.0 (CAD Structure)
- 100010-00 A.A.0 (EBOM)
- 100010-00 A.A.A.0 (MBOM)
- 100010-00 A.A.A.A.0 (Quality)

Using view versions in this manner the simple cartridge assembly and its three component parts would be represented by sixteen view versions and four configurations.

At this point, the team looked to PTC to answer fundamental questions about the use of view versions and the proposed use of the view version structure that is described in PRS12005. The team understood that the configuration is a template for instantiating, and that the EBOM view would be used for that purpose. They did not understand why configurations of the other views

are required other than to capture the data specific to that state of development. They also did not understand if or when to release a WTPart.

While waiting for guidance, the team tried to represent Sandia's suffixing practices with copies of an initial product structure. For example a "-00" product structure would be created with all its view versions and a copy would be used to create the "-01" version. The result was the same as with configurations in that only one view version could be copied and it contained only partial data.

PTC provided the AEE team with access to an experienced PDMLink consultant, Walt Morrow, to clarify the use of view versions and to answer some of the team's questions. The guidance they received was:

- View versions are NOT intended as storage areas, they are meant to generate BOMs with a specific content.
- All data pertaining to a part is stored at the default view.
- Use multiple views only when necessary to reflect different structure or content.

As a follow-up to the guidance, Rusty Abernathy of PTC advised the team that the advice provided by Walt was based on an enterprise within a single company, not on the NSE model with its multiple sites. This approach may not work for a federated system.

At this point, PTC was able to provide two consultants to work with the AEE team: a process architect, Lisa Monk, and a solution architect, Jay Hunt. These consultants would work with the team to define a BOM-based product structure through the development and deployment of the four use cases in PDMLink. One of the first activities on the part of the consultants was to provide the team with a general briefing on the use of views in PDMLink with their "*AEE Product Structure Capability Aid*" presentation. While this presentation did not address all of the specific questions team members had, it did provide a basis for a common understanding of the use of views from PTC's perspective. One of the clarifications for the team was the fact that once a new view is created, it is no longer linked to the original view, so any changes to one view or the other are not propagated between views.

The PTC team also provided some clarification for the AEE team to help resolve confusion with the references to PDMLink views in the PRS12005. Rusty Abernathy provided the following guidance:

- The PRS was not intended to tell one how to do things; its purpose is to provide the requirements to facilitate the exchange of data (import/export) between sites.
- There are too many views specified in the PRS12005.
- "Views" are filters; one would create a master view and then the other views (MBOM, Quality, Surveillance, etc.) are specific to those needs. For example, one could remove unneeded data and add more data to the MBOM as needed without affecting the master view.

With a better understanding of how view versions function in PDMLink, and with the two PTC consultants fully engaged, the AEE team was able to begin to define a BOM-based product structure definition for the four use cases. Before the use case solutions could be developed, the AEE team had to define how the PDMLink development work would be conducted.

4.5.2. VMWare Image for Development Work

Although the AEE team wanted to have the PTC solution architect, Jay Hunt, perform all work directly on the AEE unclassified development server, Jay convinced the team that this would be an impractical approach. First, it would limit his development time because he was not allowed remote access to the server. Second, the differences between the AEE server and Jay's development laptop could create problems that would hinder the transfer of work between systems as the development work progressed.

The team had already experienced this problem as they had attempted to implement the LANL product structure approach using view versions. Specifically, when a new view version was created on the AEE development system, there was no reference to the CAD documents (model and drawing). This was different from Gene Kilmer's LANL development system, where these links were maintained in the new view version. Through conversation, the AEE team discovered that LANL staff was seeing differences between their development and production systems, and the results from those two systems were different from those on the AEE system. These inconsistencies were a source of considerable frustration for the AEE team.

When discussing these issues with PTC consultants, the AEE server was providing different results than the consultants were getting on their own systems. To address this issue, the PTC consultants proposed that they create a VMWare virtual image for the AEE team to use for testing and validating use cases. In theory, it would provide a uniform PDMLink configuration that would ensure that the team and the consultants all have the same working environment. It would also allow Jay Hunt to perform development work without a proscribed schedule and it would allow him to troubleshoot problems without having to be onsite to work directly on the AEE development server.

The AEE team did have some concerns for the VMWare approach. Their past experience with PDMLink problems and their lack of expertise to resolve them raised concerns that a VMWare image might not leave them with a fully operational PDMLink development system once this short term pilot concluded. However, the PTC team convinced the AEE team that it would be far more expedient to use the VMWare solution. VMWare is commonly used as a testing environment for parallel development, common configurations for troubleshooting, and as a platform for trying alternative configurations that can be reset to the original configuration. The AEE team ultimately agreed and this approach was initiated.

The plan for moving forward was to have PTC create a PDMLink 9.0 virtual image and configure it to PRS12005. The AEE team would install a VMWare player on the AEE unclassified development server and then load the VMWare image to use as the product structure development environment. The PTC consultants would also use this image for development and testing, and in theory any problems that were encountered would be easy to identify, as they

would be in the PDMLink application and not due to any differences between the AEE and PTC systems.

Unfortunately, the VMWare turned out to be a more complicated effort than was expected. Something in the VMWare configuration prevented the use of PRO/Engineer from outside of the VMWare image on the AEE server. As a result, the pen model could not be loaded into the VMWare image on the server. This problem was specific to the VMWare image that PTC provided. The AEE team and the PTC consultants spent considerable time trying to resolve the issues, but did not find a workable solution. The AEE team ultimately felt that they had to move forward without the VMWare image on the AEE server. The approach that was ultimately employed was for the PTC solution architect to develop the use cases on his VMWare image and then work with the AEE team to manually transfer files and configuration settings directly into PDMLink on the AEE development server, rather than with a VMWare image. This enabled the team to move forward with the use case development, and it provided the AEE team with confidence that they would have a functioning PDMLink development server at the conclusion of the pilot project. However, this approach did require more time and it also provided the opportunity to introduce errors through the manual transfer of configurations.

Although Jay Hunt configured his VMWare image to look like the AEE development system, there were differences. Jay's system did not conform to the PRS12005 and while the AEE development system was originally configured to the PRS12005, it was later changed to mirror the NM production system. The NM server configuration lacked "Logical Identifiers" for its attributes which prevented the PDMLink report generator from collecting attribute values from product structure document links. Another difference was in the way PDMLink "Roles" were named. The AEE server, per the PRS, included spaces in the name of the role where the NM server used the underscore character in place of a space. It is unclear if these differences will affect the product structure.

4.5.3. Additional PDMLink Information

The AEE team began this project with a limited understanding of PDMLink and how a BOM-based product structure could be defined with the tool. The following information is a summary of some of the details of the product structure implementation that were not addressed in other sections of this report.

A product structure in the PDMLink application is a collection of objects called WTParts. Each WTPart has the option of being owned by a specific CAD model and linked to any associated drawings and documents. The collection of WTParts is arranged into structures to represent an assembly of components with the highest level WTPart (the top level assembly) deemed an "End Item". Reports are applied to the End Item's structure to generate bills of materials and PDMLink "Baseline" objects capture an accurate configuration of the component revisions used in the assembly at significant milestones. As the product life matures in the form of revisions to the models, drawings, and documents, the WTParts are revised to maintain new links to the revised data.

For the use cases, the team set all CAD models and drawings to the "released" state before the creation of the product structure. Their intent was to demonstrate that WTParts could be initially

synchronized with the revisions of the drawings and models.¹³ However, maintaining synchronization is not practical. Any change to an associated document would drive a revision to the WTPart if it is released. One of the team's recommendations was to change the revisioning scheme for the WTPart to numbers rather than letters so that it differs from the letter scheme used for CAD documents, which is defined as the CAD model and drawings or EPM documents in PDMLink. This could reduce confusion, but it might also cause problems when exchanging data between NSE sites, so it needs to be looked into further.

WTParts created after the Pro/ENGINEER model is released will require administrator privileges to iterate the released file. As a good practice, the model's comment field should note that the iteration was required to create the product structure.

For this pilot, the team used the BOM description parameter as the WTPart name and the WTPart number was taken from the part number parameter (PART_NO). This naming convention was chosen to make the use case demonstrations more understandable. The advantage is that there is a clear correlation between the WTPart and the ML report. The implications, if any, for applying this scheme to a production system are unknown.

PDMLink uses "Soft Type" objects to define different types of documents that require different attribute values. PRS12005 specifies attributes for engineering authorizations (EAs), support drawings and part defining drawings. The AEE team did not want to alter the "Soft Type" objects that are defined by the PRS12005, so they created new "Soft Type" objects and used an "AEE" prefix to identify and isolate them. The following soft type documents were created for this pilot:

- AEE General Document
- AEE Part Defining Drawing
- AEE Sandia Specification
- AEE Support Drawing
- AEE EA Document

All work was conducted within a single product container in PDMLink and additional configurations such as object initialization rules (OIRs) and preferences were set at the product container level. OIRs were written to disable the auto numbering of documents and WTParts and to set folder locations.

4.6. Sandia Design Definition Business Process Analysis

As the AEE team and PTC solution architect were working the development of the VMWare image, the PTC process architect, Lisa Monk, began a parallel effort to analyze Sandia's design/product definition business processes. Before the PTC consultants could propose a PDMLink product structure solution, they first had to understand Sandia's current workflow and business processes. The plan was to have Lisa work with Sandia subject matter experts (SMEs) to conduct a "big picture" analysis of the current Sandia design/product definition business

¹³ Kiba, Grant and Pomplun, Alan, *AEE Product Structure Pilot Use Case Demonstration*, December 2009.

process, propose a future state process using PDMLink 9.0, and identify gaps where Sandia processes cannot be directly implemented in PDMLink in its OOTB configuration.

4.6.1. Current Design/Product Definition Business Process

Because this was a limited term project, the AEE team had no expectation that they could fully address all aspects of the design/product definition workflow. Their goal was to select and implement a small number of select use cases that would address typical steps in the workflow to serve as a starting point for work that can be built upon by the PRIDE Product Structure team. Lisa began her work by interviewing AEE team members and subject matter experts (SMEs) at the CA and NM sites to get an understanding of the design/product definition workflow. These included interviews with members of the PRIDE Product Structure team.

The AEE team asked Lisa to focus on the process steps between the initiation of a design concept by an engineer and the “release” of the part into the Image Management System (IMS). Although Sandia’s design/product definition business processes are fairly well documented, Lisa found that in general, not all designers/engineers are fully aware of all requirements, nor are all requirements being strictly adhered to. Although this was not an impediment for this pilot, it should be noted for future product structure efforts, as it is hard to define a new process if the current process and its implementation are not fully understood. Using the data she gathered, Lisa developed a set of draft process diagrams for the current design/product definition workflow. She then worked with the AEE team and the subject matter experts through several iterations to arrive at a “final” set of high level current process diagrams (refer to Appendix B).

4.6.2. Future Design/Product Definition Business Process

Once she better understood the requirements of Sandia’s current design/product definition processes, Lisa began developing a high level proposed future process. She worked with AEE team members and other Sandia SMEs to understand what happens at each step of the current process so that she could determine how to implement those processes within PDMLink.

The AEE team asked Lisa to focus her efforts for the future state process on those areas that will have significant changes from the current process. They also asked that she keep the level of detail high enough that it applies to both the NM and CA sites. Differences in processes at the two sites would be seen in the next level of detail if her diagrams were developed further. The development and refinement of these future state business process diagrams is an ongoing effort and the diagrams presented in this report should be considered works in progress (refer to Appendix C).

4.7. Product Structure Use Case Implementation

As previously described, the AEE team had decided to implement four use cases that represent a product lifecycle as the starting point for the product structure definition. Their approach was to use the simple three-part PRIDE Product Structure pen model and work stepwise through the four product lifecycle stages one step at a time, documenting the process as they progressed. The team took a conventional approach with the use cases, choosing to adapt solutions, where possible, to match current Sandia business practices and some newly recommended NSE practices, rather than modifying those practices to meet the capabilities of the PDMLink tool.

Equally important considerations were data integrity, ease of use, flexibility of implementation with regard to roles and sequences of events, and future incorporation into workflows and change processes.

The particular use case scenario chosen for the pilot represents the majority of what typically occurs in the lifecycle of a CAD model and drawing at Sandia. First, an assembly of CAD models and drawings nearing the end of development are released into a documentation system. Second, a minor change to a drawing requires a revision from “A” to “B”. Third, a significant revision to a component requires a suffix change from “-00” to “-01”. Then finally, a change in components drives an additional suffix change from “-01” to “-02”, requiring a tabulated form of the bill of materials.

The team felt that this use case scenario applied to simple CAD models and could support many different approaches of product structure representations in the PDMLink software. Additionally, the simple models could be removed from the product structure to represent an application of legacy data for which no CAD models exist.

As was expected, the first use case took the longest to develop and demonstrate, as the AEE team and the PTC solution architect had to resolve a number of issues, including:

- The team’s established perceptions and misunderstanding that view versions were necessary for the creation of product structure, based on what they learned from LANL staff and the implications in PRS12005.
- PDMLink configuration settings on the AEE server and on the PTC solution architect’s VMWare image.
- Establishing common terminology: The team quickly discovered that many common terms had different meanings for the PTC consultants compared to AEE team members. Examples are the terms “released” and “part.”
- Defining the Sandia business processes and requirements that the solution architect had to work within.

Although the team had originally intended to investigate multiple potential solutions for each use case and determine the pros and cons of each solution, time constraints drove the team to focus its efforts on finding and demonstrating a single solution for each of the four use cases. Any remaining project time would be spent addressing additional high priority use cases.

4.7.1. Use Case Definition

As the team began implementing the use case solutions, the PTC process and solution architects compiled a process document in parallel to detail all of the PDMLink actions that were required for the use case solutions¹⁴. The use case definitions and high-level solution process diagrams from that document are included in this section. The document also contains highly detailed step-by-step definitions of each use case solution, including assumptions, pre-conditions, the normal sequence of steps, post-conditions, and any alternate sequences.

¹⁴ *Detailed Use Case Document – AEE Project*, February 2010.

The system level use case diagram (Figure 3) shows the release information for the four use cases, including the suffix and revision changes to the components and the assembly. The details for each use case are as follows:

1. A new product introduction: the release is suffix 00, revision A (00.A)
2. A simple change to the “Well” component requires a revision to that component and the higher level assembly with no suffix change: the release is suffix 00, revision B (00.B)
 - a. This use case has no change in form, fit or function (FFF); it is a simple change such as the addition of a note, a change to a part tolerance, or the addition of a specification.
 - b. No WTParts are created.
 - c. This is a CAD change to a model or drawing, not to documents.
3. A significant (FFF) change to the “Well” component requires a suffix change and a revision to both the component and the assembly: the release is suffix 01, revision C (01.C).
 - a. A dimension change on the “Well” part geometry requires the suffix change.
4. The “Ink” component is replaced (red ink is replaced by black ink), requiring a suffix change and a revision to both the component and the assembly: the release is suffix 02, revision D (02.D).
 - a. Both this configuration (02.D) and the previous configuration (01.C) are combined into a single ML report in this use case step to represent a tabulated part defining drawing.

	Use Case 1:		Use Case 2:			Use Case 3:			Use Case 4:		
	New product		Note change to "Well"			Geometry change to "Well"			"Ink" component is replaced		
	Release		Release		Action	Release		Action	Release		Action
	Suffix	Revision	Suffix	Revision		Suffix	Revision		Suffix	Revision	
Assembly	00	A	00	B	Revised	01	C	Revised	02	D	Revised
Well	00	A	00	B	Revised	01	C	Revised	01	C	
Tip	00	A	00	A		00	A		00	A	
Ink	00		00			00			00		Changed part

Figure 3. System level use case diagram.

The following diagrams represent the process followed for each of the four use cases¹⁵.

Figure 4. Use Case 1 process diagram.

Figure 5. Use Case 2 process diagram.

¹⁵ Detailed Use Case Document – AEE Project, February 2010.

Figure 6. Use Case 3 process diagram.

Figure 7. Use Case 4 process diagram.

4.7.2. Use Case Development Details

The implementation of the use cases was a team effort between the PTC solution architect, Jay Hunt, and the AEE Product Structure team, led by Grant Kiba. Because of the previously described problems with using VMWare on the AEE development server, the development approach that was used was parallel development on two machines:

- Jay performed his development work in VMWare on his laptop.

- The AEE team worked directly in PDMLink on the unclassified AEE development server. Jay's configurations were manually input on the AEE server throughout the development process.

Jay presented the following guidance for how/when to create product structure in PDMLink¹⁶:

- How:
 - Manually.
 - Automatically via the user interface.
 - Programmatically.
- When:
 - Prior to the existence of CAD data.
 - After the CAD data is created.
 - In parallel to the creation of CAD data.

The team elected to automatically create product structure from existing CAD models using existing user interface options and made the assumption for this effort that no product structure currently existed in the PDMLink System. The PDMLink software allows product structures to be created with or without CAD models. When models exist, the tool can leverage the model structure to quickly create a product structure of WTParts for a significant portion of the product. However, documents and other BOM items without CAD models still need to be added. This is done by manually creating WTParts and associating these documents and BOM items to the WTPart. This was the technique that was used for the use cases in this pilot. The team also intended to examine the use of PDMLink's import function to create WTParts from information listed in an Excel spreadsheet. Because of limited time and the absence of the Microsoft Office Worker on the AEE servers, the team was unable to complete this task.

The process used to create the product structure can be summarized as follows:

1. A CAD model is checked into PDMLink to create WTParts.
2. The WTPart associated with the assembly level contains the product structure.
3. The structure is edited to create and associate non-modeled WTParts.
4. Documents are associated to the assembly WTPart.
5. The ML report is generated.
6. The part defining drawing is created.
7. A baseline is created to capture the product structure and WTPart contents.

Use Case 1 shows how this process is implemented for the simple pen model. Use Cases 2-4 demonstrate solutions for revisions and suffixing in PDMLink.

Only the "Design" view version was used for this pilot. The AEE team had created the view names and their structure as part of the PDMLink configuration on the development server by following the revision B draft of the PRS12005 (design, manufacturing, quality, and surveillance views). The team assigned the design view as the default for WTParts and this is the only view that was used for the use case demonstrations.

¹⁶ *Product Structure Pilot: Close Out Documentation*, November 2009.

4.7.3. Use Case Deployment

While the aforementioned “*Detailed Use Case Document – AEE Project*” use case documentation contains considerable information, the AEE team felt that a better method for demonstrating the use case solutions was to develop a PowerPoint presentation using screen captures from the PDMLink 9.0 application¹⁷. The presentation slides include the PDMLink screens that have to be accessed and annotations that walk the user step-by-step through the implementation process. The presentation is lengthy and is, therefore, not included in full in this project report. A brief summary of the basic steps that were followed for the use case implementation and examples of supporting graphics from the presentation follow. The team’s efforts to develop the ML report are detailed in the following section.

¹⁷ *AEE Product Structure Use Case Demonstration*, December 2009.

4.7.3.1. Use Case 1 Summary of Steps

1. Create WTParts and the WTPart Structure from CAD models for the 100010-00 assembly and the 100011-00 Well part.
2. Create a non-modeled WTPart to represent Ink, a purchased part (Figure 8).
3. Associate documents (all three WTParts).
4. Generate the ML report and create the Part Defining Drawing document.
5. Associate the Part Defining Drawing.
6. Create a Baseline to historically capture the product structure and WTPart contents.

Create the WTPart for Non-Modeled Data

The “Edit Structure” action is used to create and associate the non-modeled WTPart representing the Red Ink in one step.

The part data is entered for the new WTPart

Attribute	Value
Number	MC001-00
Name	INK, RED
View	Design
Assembly Mode	Component
Default Trace Code	Lot/Serial Number
Default Unit	each
Life Cycle Template	(Generated)
Location	Autoselect Folder (Product Structure Evaluation Product)
Source	Buy
Team Template	(Generated)
BOM Description Assembly	INK, RED
Cage Code	74925
On ML Report	Undefined
Part Classification	UNCLASSIFIED

Figure 8. Presentation graphic for Use Case 1, step 2.

4.7.3.2. Use Case 2 Summary of Steps

1. Create revision B CAD model, drawing, and WTPart (100011-00).
2. Revise the CAD model and drawing.
3. Revise the associated Support Drawing documents (Figure 9).
4. Generate the ML report.
5. Revise the associated Part Defining Drawing document.

Revise the Support Drawing

Select the Support Drawing's link to go to its detail page.

Select New Revision to create revision B of the Support Drawing.

Select "Check Out and Edit" and attach the new revision B PDF file to the document. Press "Check In" to finish.

Release the Support Drawing document

The 'Edit' dialog box shows the following fields:

- Primary Content Source: Local File
- Local File Path: ☐ Keep existing primary file: AY100011-000-UNC A WELL.pdf
- Local File Name:
- Local File Description:

Attributes (5 total objects):

Name	Value
Name	WELL
Description	WELL
Cage Code	14214
On ML Report	☐ Undefined ☒ Yes ☐ No
Part Classification	UNCLASSIFIED

* Indicates required fields.

Buttons: Next, Save, Check In, Cancel

Figure 9. Presentation graphic for Use Case 2, step 3.

4.7.3.3. Use Case 3 Summary of Steps

1. Use “Save As” to create the -01 suffix for CAD models, drawings, and WTParts.
2. Set the correct revision letter.
3. Revise 100010-01 and 100011-01 CAD models and drawings (Figure 10).
4. Remove -00 and add -01 document links from both WTParts.
5. Generate the ML Report.
6. Associate the Part Defining Drawing document.

Revise 100011-01 Model and Drawing

Open the **100011-01-UNC.PRT** model. Update its parameters, revision description and change model geometry.

DESIGNED	String	G. PILE	✓
MATERIAL	String	7341723-00	✓
PART_NO	String	100011-01	✓
CONTROL_NO	String	AY100011-001	✓
TABULATION	Yes No	NO	✓
STATUS_DATE	String	11-23-09	✓
ISSUE	String	C	✓
CHECKED	String	J. KIRK	✓
APPROVED	Status	D. VADER	✓

Open the part's drawing (100011-001.DRW)

- Update the sheet and regenerate draft
- Create the PDF image. Save the file locally for the Support Drawing. Use name **“AY100011-001 C WELL”**
- **Check In** the model and drawing.

Figure 10. Presentation graphic for Use Case 3, step 3.

4.7.3.4. Use Case 4 Summary of Steps

1. Use “Save As” to create the -02 suffix of the assembly, drawing, and WTPart.
2. Set the correct revision letter in a workspace.
3. Revise WTPart structure to include the black ink and remove the red.
4. Remove -01 document links from the -02 WTPart.
5. Create the 100010 Base Number End Item and set revision to C.
6. Revise the 100010-01 suffix to revision C (remove -01 ML Report).
7. Generate the ML Report -01 and -02 (Figure 11).
8. Associate the Part Defining Drawing document to the Base Number End Item.

Create an ML Report for 100010

Search for (number='100010')

Number	Name	Revision	Iteration	Status	End Item	Folder	LifeCycle	View
100010	CARTRIDGE ASSEMBLY	A	0	c/o	true	Product Structure	INWORK	Design
100010	CARTRIDGE ASSEMBLY	A	0	chk	true	Checked Out	INWORK	Design

ML Part Base Info

Number	Name	View	Description	Classification
100010	CARTRIDGE ASSEMBLY	Design	CARTRIDGE ASSEMBLY	UNCLASSIFIED

EA-Type Documents

Cage Code	Number	Description	Classificat	EA Type	ML Issue	ML Suffix
20000012SL	Use Case 3 revision and suffix change	UNCLASSIFIED	ACO			01

Notes for ML Base Part: CARTRIDGE ASSEMBLY

Note Name	ML NOTE2	Note Number	100010-00 NOTE2	Version	A.2
Location:	Notes	State:	WIP		
Note:					
1:	DRMS: AS REQUIRED PER ASSEMBLY				
2:	THESE ITEMS HAVE A LIMITED				
3:					
4:					
5:					

Related Documents for ML Base Part: CARTRIDGE ASSEMBLY

Cage Code	Number	Description	Classification
14214	AT100010-001	CARTRIDGE ASSEMBLY	UNCLASSIFIED
14213	SAFETY SPECIFICATION	SAFETY SPECIFICATION	UNCLASSIFIED
14214	9910100	MARKING, GENERAL METHODS (from description)	UNCLASSIFIED
14214	9900000	GENERAL REQUIREMENTS	UNCLASSIFIED

Summary View

QTY	UOM	Cage Code	Related Par	Description	Classificat	Item Number	Line Number
1	ea	14214	100010-01	WELL 08	UNCLASSIFIED	2	2,530
1	ea	74625	MC001-00	INK RED	UNCLASSIFIED	3	2,520
1	ea	14214	100012-00	TP 08	UNCLASSIFIED	1	2,500

Create the ML Report on the 100010 Base number WTPart.

ML Part Base Info

Number	Name	View	Description	Classification
100010	CARTRIDGE ASSEMBLY	Design	CARTRIDGE ASSEMBLY	UNCLASSIFIED

EA-Type Documents

Cage Code	Number	Description	Classificat	EA Type	ML Issue	ML Suffix
20000013SL	Use Case 4 Suffix change	UNCLASSIFIED	ACO			02

Notes for ML Base Part: CARTRIDGE ASSEMBLY

Note Name	ML NOTE2	Note Number	100010-00 NOTE2	Version	A.2
Location:	Notes	State:	WIP		
Note:					
1:	DRMS: AS REQUIRED PER ASSEMBLY				
2:	THESE ITEMS HAVE A LIMITED				
3:					
4:					
5:					

Related Documents for ML Base Part: CARTRIDGE ASSEMBLY

Cage Code	Number	Description	Classification
14213	SAFETY SPECIFICATION	SAFETY SPECIFICATION	UNCLASSIFIED
14214	AT100010-002	CARTRIDGE ASSEMBLY	UNCLASSIFIED
14214	9910100	MARKING, GENERAL METHODS (from description)	UNCLASSIFIED
14214	9900000	GENERAL REQUIREMENTS	UNCLASSIFIED

Summary View

QTY	UOM	Cage Code	Related Par	Description	Classificat	Item Number	Line Number
1	ea	14214	100011-01	WELL 08	UNCLASSIFIED	2	2,530
1	ea	14214	100012-00	TP 08	UNCLASSIFIED	1	2,500
1	ea	74624	MC002-00	INK BLACK	UNCLASSIFIED	3	2,530

Associate the Part Defining Drawing to the 100010 Base Number WTPart.

Check In and Release the completed WTPart.

Product Structure

Number	Actions	Version	Name	On ML Report	State	Quantity
100010	Actions	C.0 (Design)	Cartridge Assembly	Yes	In Work	
100010	Actions	C.0	CARTRIDGE ASSEMBLY (ML Report)	No	Released	
100010-01	Actions	C.1 (Design)	CARTRIDGE ASSEMBLY	Yes	Released	1 each
100010-02	Actions	C.1 (Design)	CARTRIDGE ASSEMBLY	Yes	Released	1 each

Figure 11. Presentation graphic for Use Case 4, steps 7 and 8.

4.7.4. Suffix Changes

While other methods for creating and managing suffix changes in a product structure were considered, the AEE team's approach was to follow current and common CAD practices for this pilot. In Sandia CAD practices, a suffix change is handled by making a copy of the model and drawing, renaming them to the new suffix, and making the necessary changes. The same approach was used in PDMLink; the model and drawing were copied and renamed to the new suffix. Two other approaches were also considered, but not implemented:

1. Using PDMLink "End Item" configurations to represent suffix changes.
 - a. This approach would require a change in file naming practices to move the suffix from the file name to the end item configuration name and use the file base name throughout product the lifecycle.
2. Using PRO/Engineer "Family Tables" in assemblies.
 - a. It is not a common practice at Sandia to use Family tables for suffixed assemblies.
 - b. The handling of family table documents in PDMLink is complex.

4.7.5. Additional Use Cases

The AEE team completed several additional use case exercises to continue to investigate the capabilities and identify gaps between Windchill capabilities and Sandia business practices.

4.7.5.1. New View Version Use Case

The use of new view versions was successfully applied to the cartridge assembly to demonstrate how these views can be used to associate data after a design is released. Once a design issue is released, additional documents cannot be associated to that design without a revision to the WTPart. By creating a new view version in PDMLink, additional data that does not affect the design can be associated to the new view. The new view versions are appropriately named for the data links they contain and are a "single level" child of the "Design" view. As previously described in Section 2.3.2 *System Configuration*, the PDMLink view structure had already been created at the site level when the system was configured.

Figure 12. A “Manufacturing” view was created to support manufacturing data links for the 900011-00 revision “A” Well part.

For this exercise, the default “Design” view for the cartridge assembly, which contains links to all design related documents and data, was released, baselined, and maintained under configuration control. The use case was implemented with the following steps:

- A “Manufacturing” view was created to support manufacturing data links for the 900011-00 rev “A” Well part (Figure 12).
- The new “Manufacturing” view version is represented by a new WTPart. For ease of accessibility, the WTPart was moved to its own separate folder.
- Data was associated with the new view. In this use case, manufacturing CAD documents were used.

4.7.5.2. Serialization Use Case

Serialization was successfully applied to the cartridge assembly to provide a means of traceability for individual assemblies and components as they were fabricated and assembled. The serialization use case was implemented with the following steps:

- The WTPart trace code attribute was set to “Serial Number” by using the “Edit Common Attributes” action (Figure 13).
- A configuration was created for each component and assembly by using the “New Configuration” action.
- Part instances were created and assigned individual serial numbers using the “New Part Instance” action in the Configuration.
- Serialized component instances were allocated to assembly instances using the “Allocate Existing” action within a specific assembly instance.

- Incorporate (assign effective date) the instances in the assembly part instance.

Figure 13. The WTPart trace code attribute was set to “Serial Number” by using the “Edit Common Attributes” action.

4.7.5.3. Library Parts Use Case

This use case examined how family table library parts are handled in the product structure. First, WTParts were created within the library for all of the instances of a family table library part: a fastener (Figure 14). Second, a product structure was created for an assembly that used one of those instances. The result showed that the fastener was correctly represented by the WTPart defined in the library.

Folder Contents (105 of 105 total objects)					Current View: All			
File Edit								
Cut Copy Paste Delete Move Check out to workspace Add to Workspace					New Folder New Document New Multiple Documents New Document from Template New Part New Multiple Parts New Link New Problem Report			
		Number	Name		Actions	Version	State	Last Modified
☐		VL-NASM16995.PRT	-O-		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-1.PRT	SCR.SCH_086-56UNC-3A_18...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-1S316.PRT	SCR.SCH_086-56UNC-3A_18...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-2.PRT	SCR.SCH_086-56UNC-3A_25...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-2B.PRT	SCR.SCH_086-56UNC-3A_25...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-3.PRT	SCR.SCH_086-56UNC-3A_37...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-3S316.PRT	SCR.SCH_086-56UNC-3A_37...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-4.PRT	SCR.SCH_086-56UNC-3A_50...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-4S316.PRT	SCR.SCH_086-56UNC-3A_50...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-9.PRT	SCR.SCH_112-40UNC-3A_25...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-9B.PRT	SCR.SCH_112-40UNC-3A_25...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-9S316.PRT	SCR.SCH_112-40UNC-3A_25...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-10.PRT	SCR.SCH_112-40UNC-3A_37...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-10B.PRT	SCR.SCH_112-40UNC-3A_37...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-10S316.PRT	SCR.SCH_112-40UNC-3A_37...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-11.PRT	SCR.SCH_112-40UNC-3A_50...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-11B.PRT	SCR.SCH_112-40UNC-3A_50...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-11S316.PRT	SCR.SCH_112-40UNC-3A_50...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-12.PRT	SCR.SCH_112-40UNC-3A_62...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-13.PRT	SCR.SCH_112-40UNC-3A_75...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-16.PRT	SCR.SCH_138-32UNC-3A_25...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-17.PRT	SCR.SCH_138-32UNC-3A_37...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:
☐		VL-MS16995-17B.PRT	SCR.SCH_138-32UNC-3A_37...		Actions ▼	A.0 (Design)	Released	2009-12-03 1:

Figure 14. WTParts represent the instances of a family table library part.

4.7.5.4. Tabulation Use Case

The AEE team’s examination of if and how tabulated part drawings can be implemented into a product structure revealed that there are different reasons for employing tabulation as a documentation practice. These reasons are supported by CAD software practices that have become well established in the NSE design community. Implementing a successful product structure will require a good understanding of the resulting database structures and the possible alternatives to current practices. The following three tabulation use cases are presented as examples of how tabulation can be integrated into the product structure.

Tabulation Case 1

A tabulated CAD model and drawing is sometimes used to document multiple suffix part versions. For example, the CAD model “1E2345-00” has a 0.500” diameter hole. Its drawing AY1E2345 shows the model in its drawing views. Two new suffixes, “-01” and “-02”, require the same hole but at different diameters (0.125” and 0.250”). At this point, the designer has a choice:

- The designer can copy the “-00” model and drawing and edit them to create two new sets of models and drawings to support the new “-01” and “-02” configurations; this results in no tabulated drawing.
- The designer can use a family table to create instances representing the “-01” and “-02” versions within the original “1E2345-00” model. The original drawing (“AY1E2345”) would then include a table or tabulation that lists the three suffixes and their unique hole diameter dimension values. This is a tabulated drawing that lists multiple suffixes.

Tabulation Case 2

A tabulated CAD model and drawing are sometimes used to document different part numbers with similar geometry. For example, the CAD model geometry for part “A00001-00” is similar to part “A00002-00” in shape, but not in size. The parts have two different feature lengths as their only difference. The designer’s choices are:

- The designer can copy the “A00001-00” model and drawing and edit them to independently represent the “A00002-00 part”; this results in no tabulated drawing.
- The designer can use a family table to create instances representing both parts within the original “A00001-00” model. The “A00001-00” drawing would then include a table or tabulation listing each part with its unique part numbers and dimension(s). This is a tabulated drawing that lists multiple part numbers.

Tabulation Case 3

A tabulated CAD model and drawing are sometimes used to document different suffixed assemblies. For example, a CAD assembly “1E0000-00” requires a suffix change by a “form, fit or function” modification to one of its components. In creating the new assembly (“1E0000-01”) the designer has the following choices:

- The designer can make a copy of the original assembly (“1E0000-00”) and drawing (“1E0000-000”) and edit them to create the updated “1E0000-01” assembly. The model would be named “1E0000-01” and the drawing would be named “1E0000-001” (incrementing its control number). No tabulated drawing is created.
- The drawing would **NOT** be copied and the original drawing could be modified to support the new model suffix. The updated “1E0000-01” assembly could be added to the original “1E0000-000” drawing. The drawing would contain both assemblies (multiple models in a drawing) and the “-01” assembly may appear on additional drawing sheets or in specifically labeled drawing views.
- The designer can use a family table to create assembly instances representing both assembly suffixes within the original “1E0000-00” model. The “1E0000-000” drawing would then include a table or tabulation listing both suffixes defined in the single drawing. This is a tabulated assembly drawing.

Conclusions

The AEE team drew the following conclusions from the tabulation use cases:

- The AEE team addressed three known tabulation use cases; there may be more that need to be considered.

- All three tabulation use cases can be incorporated into a product structure with no apparent difficulties.
- There is a problem with a manner in which family tables and tabulated drawings are created that affects configuration management in PDMLink. When a family table is employed, it is important that the generic **NOT** be used to represent any specific part or suffix number. Each part or suffix number must be represented by its own instance. Additionally, the generic should not be used in a drawing; drawings should reference only specific instances. When a generic is used to represent a specific part or suffix number, the revision or issue of that part or suffix number cannot be controlled independently from the issue of the generic file. The generic file must be revised each time that any of its instances are revised. The part or suffix number assigned to the generic will be subject to unnecessary and undocumented issue changes. If, however, each part or suffix number in a family table is assigned to a specific instance, the revision of the generic file becomes irrelevant and each instance maintains its own separate revision or issue letter. Existing models and drawings that currently use the generic as a specific part number can be corrected in PDMLink.

4.8. Development of the ML Report

Sandia defines product structure as all of the data required to build a part, including the CAD data, support drawings, specifications, and other relevant information. The part defining ML identifies all of this data. For this pilot, the AEE team wanted to derive the ML from the product structure and have the capability to produce an ML report comparable to the ML report currently in use.

4.8.1. ML Requirements

The PTC solution architect, Jay Hunt, worked with the AEE team and SMEs to understand the requirements for the ML report, which differ significantly from typical industry requirements. For example, the PDMLink OOTB ML report does not include any documents within the product structure that are normally on the Sandia ML report, nor does it include the part defining drawing document, which for Sandia is the ML report itself. The team derived the following set of rules for the development of the ML:

- Every "deliverable item" will have an ML report.
- The ML report includes only objects on the product structure.
- The ML report includes only objects in the top and first sub-level of the product structure, not all elements at all levels.

The AEE team provided the PTC solution architect with the following ML report requirements: Engineering Authorization (EA) Documents must report:

- Name (CER, ACO, FCO)
- Number
- Revision

- Description
- Date
- Issue of the ML
- Suffix of the ML

WTParts, Support Drawings, and Sandia Specs report the following:

- Quantity (use “NA” for quantity on support drawings)
- Cage Code
- Part number
- Item description
- Item number
- Line number

Sandia General Documents and Part Defining Drawings should report:

- Name
- Item description

4.8.2. ML Report Generation

This AEE team intended to exercise the use cases with PDMLink in its OOTB configuration and would not use customization of the application’s base code to implement a solution. Although the Windchill software suite includes a number of report generating tools, Jay Hunt determined that none of those tools could meet all of the Sandia ML report requirements. He provided the following assessment of the available tools¹⁸:

- Query Builder:
 - Pros:
 - User interface: Good for a simple object report.
 - Can report on related objects (to an extent).
 - Quick and dirty, available to all.
 - Cons:
 - Requires knowledge of the Windchill object model.
 - Limited linking capability.
 - Limited user interface displaying.
 - Not able to conduct BOM processing with linking of additional data.

4.8.2.1. Cognos Windchill Business Reporting:

- Pros:
 - Standard third party technology.
 - Robust and professional reporting.

¹⁸ *Product Structure Pilot: Close Out Documentation*, November 2009.

- Cons:
 - Additional Software used.
 - Primarily for data display
 - Limited data processing.
 - Requires data group be passed in.
 - Has a learning curve.

4.8.2.2. Info*Engine/JSP

- Pros:
 - Standard Windchill / PTC technology.
 - Sufficient documentation is available.
 - Components are flexible and re-usable.
 - Info*Engine tasks are the processing power for many Windchill functions.
- Cons:
 - There is a learning curve.
 - Data / configuration to maintain.

Jay recommended using Info*Engine/JSP (I*E/JSP) as the solution for generating the ML report. He had consulted with his PTC peers and the general consensus was that I*E/JSP was the best solution based upon the ease of use, the available documentation, and its flexible and scalable architecture. To meet all of the AEE team's ML report requirements, some configuration and development work would be required to configure a new report using I*E. This report relied on existing data linkages in PDMLink to pull all of the required data into a single ML report. All of the work was done using OOTB applications and there was no modification of PDMLink base code (customization). For future PDMLink upgrades, it is probable, although not guaranteed, that this report will not be affected and should not have to be recreated.

Jay provided the following explanation for how the data would be gathered for display in the I*E generated ML report¹⁹:

*Data from multiple sources are included to produce this report. Step one is to locate a "Base Part" for which this report is to be run. Step two is to walk the product structure tree of the selected base part and gather the needed data. The interrogation of the existing product structure utilized an OOTB Info*Engine code snippet (webject) called "Query-Tree". Step three is the display of the needed data. This is accomplished with an available OOTB Info*Engine code snippet called "Display-Table".*

A fundamental issue for producing the ML report was to determine which objects should be included in the report. Time constraints prevented an in-depth analysis of how business rules could be applied in PDMLink to define the ML report content. The AEE team decided that a simple and effective solution was to use a flag attribute on objects to include or exclude them from the ML report. The following are details for the use of the flag attribute:

- The "On ML Report" attribute is a yes/no flag that was added for all WTParts and WTDocuments.

¹⁹ *Product Structure Pilot: Close Out Documentation*, November 2009.

- CAD data does not have the ML report flag.
- EPM documents (CAD data) are always excluded from the ML report.
- The “On ML Report” attribute is specific to Sandia. Other sites can create their own attribute as needed.
- The “On ML Report” attribute is in addition to the PRS12005 attributes.
- A WTDocument’s “On ML Report” attribute setting is constant for all product structures.
- To avoid iterating a WTPart, the “On ML Report” attribute must be set when the WTPart is created. This will keep the iteration as “X.0”. If the “On ML Report” attribute is added later, the WTPart will iterate to “X.1.”

The team went through a series of iterations in the development of the ML report. Through this process, the team decided that two separate reports would be beneficial:

- The first report was named “Sandia ML Report” and it contains all of the information required in the current Sandia ML report (Figure 15). While the content is the same, this report’s format differs from the Sandia ML report.
- A second report named “Product Structure Interrogation Report” was also generated. It retains all of the information included in the “Sandia ML Report” and it has additional information that is helpful for the team’s ongoing product structure development work. Specifically, it reports the configurations for revision tracking, which the “Sandia ML Report” does not.

Sandia ML Report

ML Part Base Info

Number	Name	View	Description	Classification
100010-00	CARTRIDGE ASSEMBLY	Design	CARTRIDGE ASSEMBLY	UNCLASSIFIED

Executed by UID: gwilba
Executed On: 2009-11-25 09:54:08
Executed from IP: 134.252.16.195

EA-Type Documents

Cage Code	Number	Description	Classificat	EA Type	ML Issue	ML Suffix
	20090001SL	Initial Release (Use case #1)	UNCLASSIFIED	CER	A	00

Notes for ML Base Part: CARTRIDGE ASSEMBLY

Note Name: ML NOTE1 Location: Notes Note: 1. SHIMS, AS REQUIRED PER ASSEMBLY 2. THESE ITEMS HAVE A LIMITED..... 3. 4. 5.	Note Number: 10010-00-NOTE1 State: WIP	Version: A.3
Note Name: ML NOTE2 Location: Notes Note: 6. ADD PART XXX...	Note Number: 100010-00-NOTE2 State: WIP	Version: A.2

Related Documents for ML Base Part: CARTRIDGE ASSEMBLY

Cage Code	Number	Description	Classification
14214	AY100010-000	CARITDGE ASSEMBLY	UNCLASSIFIED
14214	9900000	GENERAL REQUIREMENTS	UNCLASSIFIED
14214	9919100	MARKING, GENERAL METHODS (from description)	UNCLASSIFIED
14213	SAFETY SPECIFICATION	SAFETY SPECIFICATION	UNCLASSIFIED

Summary View

QTY	UOM	Cage Code	Related Par	Description	Classificat	Item Number	Line Number
1	ea	14214	100012-00	TIP 08	UNCLASSIFIED	1	2,500
1	ea	14214	100011-00	WELL 08	UNCLASSIFIED	2	2,510
1	ea	74925	MC001-00	INK, RED	UNCLASSIFIED	3	2,520

Figure 15. The Sandia ML report generated in PDMLink.

In Use Case 2, a new ML was generated to capture a simple change to a component. To reflect the changes in the product structure on the new ML, the product structure and WTPart must be modified (remove old version and add new version), which results in the iteration of the WTPart and the top level assembly. In addition, by putting the ML report in the product structure (to track it), any time the ML changes, so does the product structure, which is a circular reference. Solutions that were considered included:

- Attaching the ML report document to the top level assembly using a references link. The latest version of the ML report would always be referenced.
- Place the ML in a separate folder rather than on the product structure.
- Do not include the versions and iterations on the ML since this will be the only change in this scenario. This only applies to WTParts in the work in process (WIP) state, not once a WTPart is released. This is the solution that was selected.

For Use Case 4, the suffix change that is required with the replacement of the “Ink” component provided the AEE team with the opportunity to demonstrate tabulation for the ML report. The ML report includes both the original and suffixed configurations as separate MLs in the same report (Figure 16). While the team proved they could extract all of the data for both reports, they did not format the report to replicate the appearance of the current Sandia tabulated ML report.

ML Part Base Info

Number

Name

View

Description

Classification

100010-01

CARTRIDGE ASSEMBLY

Design

CARTRIDGE ASSEMBLY

UNCLASSIFIED

EA-Type Documents

Cage Code

Number

Description

Classification

EA Type

ML Issue

ML Suffix

20090012SL

Use Case 3 revision and suffix change

UNCLASSIFIED

ACO

01

Notes for ML Base Part: CARTRIDGE ASSEMBLY

Note Name: ML NOTE2

Location: Notes

Note: 6. ADD PART XXX...

Note Number: 100010.00.NOTE2

State: WIP

Version: A.2

Note Name: ML NOTE1

Location: Notes

Note:
1. SHIMS, AS REQUIRED PER ASSEMBLY
2. THESE ITEMS HAVE A LIMITED...
3.
4.
5.

Note Number: 10010.00.NOTE1

State: WIP

Version: A.3

Related Documents for ML Base Part: CARTRIDGE ASSEMBLY

Cage Code

Number

Description

Classification

14214

AY100010.001

CARTRIDGE ASSEMBLY

UNCLASSIFIED

14213

SAFETY SPECIFICATION

SAFETY SPECIFICATION

UNCLASSIFIED

14214

9919.00

MARKING, GENERAL METHODS (from description)

UNCLASSIFIED

14214

9900000

GENERAL REQUIREMENTS

UNCLASSIFIED

Summary View

QTY

UOM

Cage Code

Related Par

Description

Classification

Item Number

Line Number

1

ea

14214

100011-01

WELL 08

UNCLASSIFIED

2

2,510

1

ea

74925

MC0011-00

INK, PFD

UNCLASSIFIED

3

2,520

1

ea

14214

100012-00

TIP 08

UNCLASSIFIED

1

2,500

ML Part Base Info

Number

Name

View

Description

Classification

100010-02

CARTRIDGE ASSEMBLY

Design

CARTRIDGE ASSEMBLY

UNCLASSIFIED

EA-Type Documents

Cage Code

Number

Description

Classification

EA Type

ML Issue

ML Suffix

20090013SL

Use Case 4 Suffix change

UNCLASSIFIED

ACO

02

Notes for ML Base Part: CARTRIDGE ASSEMBLY

Note Name: ML NOTE2

Location: Notes

Note: 6. ADD PART XXX...

Note Number: 100010.00.NOTE2

State: WIP

Version: A.2

Note Name: ML NOTE1

Location: Notes

Note:
1. SHIMS, AS REQUIRED PER ASSEMBLY
2. THESE ITEMS HAVE A LIMITED...
3.
4.
5.

Note Number: 10010.00.NOTE1

State: WIP

Version: A.3

Related Documents for ML Base Part: CARTRIDGE ASSEMBLY

Cage Code

Number

Description

Classification

14213

SAFETY SPECIFICATION

SAFETY SPECIFICATION

UNCLASSIFIED

14214

AY100010.002

CARTRIDGE ASSEMBLY

UNCLASSIFIED

14214

9919.00

MARKING, GENERAL METHODS (from description)

UNCLASSIFIED

14214

9900000

GENERAL REQUIREMENTS

UNCLASSIFIED

Summary View

QTY

UOM

Cage Code

Related Par

Description

Classification

Item Number

Line Number

1

ea

14214

100011-01

WELL 08

UNCLASSIFIED

2

2,510

1

ea

14214

100012-00

TIP 08

UNCLASSIFIED

1

2,500

1

ea

76524

MC002-00

INK, BLACK

UNCLASSIFIED

3

2,530

Figure 16. The ML report with tabulation.

Further details for Jay Hunt’s development of the PDMLink ML report can be found in his “*Product Structure Pilot: Close Out Documentation*” report. Additional information and comments for the ML report generation and use are summarized here:

- Ideally, a user should have the ability to generate the ML report in PDMLink by selecting an action from a pull-down menu. Due to time constraints, a hyperlink to the ML report generator was used for this pilot. This is a gap that should be addressed.
- The ML report includes the ability to do a wildcard search (xxxx*).
- If an assembly is checked out, it will be listed twice on the ML report “Search” view:
 - One is the checked out assembly.
 - The other is the assembly the PDMLink system sees.
- Users can run the ML report on a WTPart whether it is checked out or checked in; however, here is where the ML being in the product structure causes a circular reference. The WTPart has to be checked out to add the ML to the product structure. After the ML is added, a baseline is created. When the WTPart is checked in, the WTPart iterates and this may cause confusion. Although the ML and baseline will appear with the new iteration number in the product structure, when one opens the ML report, it will be identified on the report as the previous iteration. Note that the “Iteration” column in the product structure refers to the WTPart, not the CAD part. This should not be an issue, since we are eliminating the “Iteration” column from the new version of the “Sandia ML Report” and it will only remain in the new “Product Structure Interrogation Report.”
- For this pilot, the ML report flag was attached to WTParts. PTC recommends that a better practice would be to attach the flag to the link between the WTPart and the assembly. This would allow a document to appear on the ML in one product structure and not appear in another. However; it would also mean that the link would need to be manually

set. A better idea would be to have the documents type have a default flag setting but let the flag on the link override the document's flag when needed.

- Comments for “Notes” on the ML report:
 - Users can search for “Notes” on the ML by using the “Search” user interface.
 - Notes could be categorized and made available through a pull-down menu. Jay Hunt indicated that an instance based attribute (IBA) could be added to accomplish this task.
- The EAs, Support Drawings, and Sandia Specs all appear in the ML Report. The ML Report and general document types in our use case are never listed on the ML. The team considers it a good option to have the ability to include all documents on the ML Report.

4.9. Issue and Gap Analysis

As the AEE team and the PTC consultants worked through the development of the future business process diagrams and the use case solutions, they identified a number of process issues that require clarification and/or resolution, and they identified gaps between current Sandia business processes and PDMLink capabilities. The gaps are instances where there may be no directly applicable solution using PDMLink which may require either a change to the Sandia business process, or the investigation of an alternate tool solution. These gaps will need to be addressed by the PRIDE team as they move forward with their product structure definition.

The AEE team had hoped to address some of these identified gaps to determine whether the business process could be changed to meet the tool capabilities, or if an alternate tool solution would be required. However, time constraints prevented further investigation in this area. The team did note that many of the current processes have been in place for a long time and were developed for a drawing-based system so there may be potential for some process changes.

The PTC process architect, Lisa Monk, documented her findings in an issues analysis presentation for the AEE team²⁰. Her findings are included in the following section, along with additional issues and gaps that the AEE team identified through their use case implementation.

4.9.1. Design Related Process Issues/Gaps:

1. Inconsistency in the process for determining if a design change requires a new suffix or only a revision change.
 - a. Cause: The project engineer makes the determination based on technical business practices (TBPs).
 - b. Comments: Rolling the suffix adds work within Pro/ENGINEER, so project engineers are not always following the proscribed process. Leaving the suffixing decision to the discretion of the project engineer leads to inconsistencies.
 - c. Options/Solutions: Reeducate project engineers on the TBPs. Processes need to be followed to ensure consistency.
2. Engineers are not always informed when changes are made to the drawings for which they have ownership.

²⁰ AEE Product Structure, December 2009.

- a. Cause: Once Engineering does a drawing transfer engineering release (DTER) to transfer a design to a production agency (PA), the project engineer at the design agency (DA) is still responsible for any changes to the drawing. Changes made at the PA are not always formally communicated back to the DA.
 - b. Comments: The way changes are communicated is not a documented process.
 - c. Options/Solutions: Educate the Engineering staff on design maintenance and ownership requirements. Consider an automated notification process for design changes.
3. Will the files that create the AY support drawing be processed in PDMLink through the CAD Release Process?
 - a. Cause: How will the AY support drawing be released in PDMLink?
 - b. Comments: This is a discussion point that needs to be resolved.
 - c. Options/Solutions: Need to define a transition from the current process to a new process.
4. To create a new revision of a CAD model that is already linked to a WTPart in PDMLink, the WTPart has to be revised as well.
 - a. Cause: The AEE team expected that a revision to an EPM Document (CAD document) in PDMLink would cause the associated WTPart to be revised to keep the revisions in sync. That is not the case.
 - b. Comments: The WTPart is a separate object in PDMLink and it does not revise if the EPM Document is revised. This is the OOTB functionality of PDMLink.
 - c. Options/Solutions: If desired/required, revise the EPM and WTPart at the same time.

4.9.2. Product Structure and Design Release Issues/Gaps:

1. When should the product definition be defined (the association of documents, etc)?
 - a. Cause: Multiple departments/people are involved in the complete/full product definition and the data is housed in multiple systems.
 - b. Comments: At what stage of the "Released" lifecycle do we build product structure and who builds it? This is a discussion point.
 - c. Options/Solutions: The AEE team discussed the possibility of building the product structure during the CAD object review and release process.
2. The definition of the term "Released".
 - a. Cause: Lifecycle states should be evaluated to determine if there is a need to expand the "Released" state into multiple levels. PDMLink's "Released" lifecycle state does not match Sandia's definition of the "Released" lifecycle state.
 - b. Comments: This would affect the PRS12005 which specifies the lifecycle states.
 - c. Options/Solutions: Consider adding more lifecycle states to accommodate the different levels of the "Released" state.
3. Creating the product structure from the released model files creates an iteration change of the released CAD model.
 - a. Cause: When it is time to create the product structure, most of the CAD documents will be marked as "Released". Creating a product structure with

released CAD documents will create a WTPart that is released (undesired) due to checking the item out to “Auto Associate” in PDMLink.

- b. Comments:
- c. Options/Solutions: Possible solutions include:
 - i. WTParts and the product structure are created at CAD document release by the workflow/release process.
 - ii. Create the product structure before releasing the top level assembly (not practical).
 - iii. Administrative privileges can be used to create the product structure from a revised assembly which would cause iteration to the CAD assembly.

4.9.3. Documents Issues/Gaps:

1. There are a few documents that should always be attached to the product structure (e.g., documents 9900000 and 9919100).
 - a. Cause: There are two specifications that must be called out on every part.
 - b. Comments: Designers know these two specifications must be added to all designs.
 - c. Options/Solutions: Could these two specifications be automatically added to the product structure? Alternately, could they be listed as notes on the ML report only?

4.9.4. Material List Report Issues/Gaps:

1. The capture and display of “Notes” on the ML report (as a result of a change).
 - a. Cause: PDMLink and the EA system do not exchange data.
 - b. Comments: The EA change process and note capture were out of scope for this project. SMEs would like the notes displayed at the top of the new ML report just as they are on the current ML report.
 - c. Options/Solutions: For the use cases, the ML report was reformatted to place the notes at the requested location.
2. How is change history captured on the ML report?
 - a. Cause: The current ML report is under configuration control outside of PDMLink (in the Sandia EBOM system).
Note: In this section, EBOM refers to the commercial product data management application used by Sandia and other NSE sites to manage MLs, EAs and other information that define a product.
 - b. Comments: Should the ML be under configuration control in PDMLink? This is a discussion point.
3. The ML report cannot always utilize part numbers in a higher chain.
 - a. Cause: There are inconsistent timing requirements for when part numbers are submitted into the Sandia EBOM system (no formal NSE process).
 - b. Comments: Only part numbers that have been entered into the Sandia EBOM system can be recorded on the ML. Sandia/CA enters metadata into EBOM when a drawing number is pulled. Sandia/NM does not enter data into EBOM until the drawing release, so if Sandia/CA is creating an ML with Sandia/NM parts that are not yet released those parts cannot be called out on the ML. In this case, Sandia/CA cannot release the ML because the Sandia/NM data is missing.

- c. Options/Solutions: To be determined.
- 4. Should the ML report be associated to the top level of the product structure?
 - a. Cause: Configuration management of documents currently happens outside of the PDMLink system.
 - b. Comments: This was documented in the use cases where the team decided to link the ML Report to the top level of the product structure.
 - c. Options/Solutions: To be determined.
- 5. Accessing the ML report.
 - a. Cause: Ideally, the ML report should be available from a pull-down menu within PDMLink, but this is not available in the OOTB configuration and would require a customization.
 - b. Comments: The AEE team used a URL to access the ML report search page as a solution for the use case demonstration.
 - c. Options/Solutions: Add an icon to directly start the ML report or add a shortcut directly to the "Properties" page.
- 6. Revisions to the ML when it is included in the product structure.
 - a. Cause: Once an ML report has been created, attaching the ML report to the product structure will iterate the objects to which the ML report is being attached. The attached ML report is always out-of-sync with the product structure.
 - b. Comments: Is this really a problem?
 - c. Options/Solutions: A solution is to not include the versions and iterations on the ML report.
- 7. "Notes" formatting.
 - a. Cause: PDMLink displays Notes without carriage returns, as a straight string field.
 - b. Comments: Notes were initially displayed in the ML report without line breaks. A "fix" for the use cases was the manual insertion of html line breaks to separate the notes in the ML report.
 - c. Options/Solutions: Sandia would like standard formatting applied to the Notes in PDMLink.
- 8. The "On ML Report" attribute affects a document's visibility on all product structures.
 - a. Cause: There could be a situation where the document should be visible on Product Structure 1 and not visible on Product 2. The "On ML Report" attribute does not allow this option.
 - b. Comments: Under the current approach this is a gap.
 - c. Options/Solutions: Apply the attribute to the link, not on the WTPart.
- 9. Determining if a supporting document should be part of the ML report by using an attribute.
 - a. Cause: There are no business rules defining whether an item will ALWAYS be part of or NEVER be part of an ML Report.
 - b. Comments: Adding an "On ML Report" attribute is currently not part of the PRS12005. Each NSE site would need business rules for creating their own "On ML Report" attribute.

- c. Options/Solutions: A report can be modified to look at a properties file to pick up the value of the attribute prior to creating the report.

4.9.5. Practices and Processes Issues/Gaps:

1. Naming procedures for baseline and configuration objects need to be addressed; baselines are not currently utilized in the NSE.
 - a. Cause: Currently there is no naming convention for electronic baselines.
 - b. Comments: Date/time is not the preferred naming convention for baselines.
 - c. Options/Solutions: Sandia and the NSE need to define if/how they will use baselines.
2. Part numbers are inconsistent across the NSE for the same drawing. The base number is consistent but extension numbers can vary.
 - a. Cause: Inconsistent requirements allow each production agency to use their own software. Each could have a different extension for part numbers beyond the base number.
 - b. Comments: This can be a problem for electronically sharing data between NSE sites.
 - c. Options/Solutions: This issue should be discussed among the NSE sites to determine if a common business practice can be employed.

4.9.6. Additional Issues/Gaps:

The AEE team identified the following specific PDMLink gaps and desired functionality through their use case implementation:

1. Running a search in PDMLink for a part that doesn't exist results in an error screen.
 - a. Options/Solutions: PDMLink should be corrected to display a message that says the part can't be found rather than the error screen.
2. Desired PDMLink functionality:
 - a. Add categories for Notes for the ML report and make them available via a pull-down menu.
 - b. Write the ML report directly to a PDF file with embedded text so that the data can be read from the file.

4.10. Conclusions and Recommendations

The need to define a common product structure definition for the NSE in the Windchill PDMLink tool has been the subject of discussion for some time. The AEE team's successful demonstration of a BOM-based product structure is intended as a starting point for follow-on work by other Sandia development teams. The AEE team demonstrated that a BOM-based product structure can be defined for a simple assembly in PDMLink 9.0 while adhering to current Sandia design definition business processes. No customization of the OOTB PDMLink tool was required, although configuration and development work with OOTB software tools was required to create an ML report that would include all of the required data.

The product structure solution employed by the AEE team was neither model-centric nor fully part-centric; it was a hybrid of the two. The initial product structure was created using the CAD

model and then documents and non-modeled represented parts were added to this structure. This approach will accommodate legacy data that are represented only by drawing images, as well as materials and documents with no CAD models that must be included in the ML.

The AEE team does not suggest that the solutions they implemented for the four use cases are the only solutions or necessarily the ideal solutions. Time constraints limited the team to the pursuit of a single solution for each use case, rather than trying and comparing multiple potential solutions as had been the team's original intent. AEE team members invested considerable time and energy to successfully demonstrate these use case solutions. Some of the barriers the team encountered include:

- A big learning curve to understand how PDMLink can be used to define a product structure.
- Resolving PDMLink 9.0 configuration issues on the AEE development servers.
- Communicating Sandia business processes and project requirements to the PTC process and solution architects.
- Confusion over the use of "view versions" in PDMLink as a possible solution for product structure, as implied in PRS12005.
- Difficulties deploying the common development environment VMWare image on the AEE project server.

The AEE team continues to learn new things every time they use PDMLink. They have noted that deploying the tool requires many little, but significant steps.

4.10.1. Recommendations

The AEE team provides the following recommendations for the PRIDE and SBOM teams as they continue product structure development:

- Representatives from the NSE design agencies should get together to compare notes on using Windchill 9.0:
 - Are they all moving in the same direction, finding the same issues, reaching the same conclusions?
 - What are the other NSE sites doing with baselines? Do they baseline every revision or only significant revisions?
 - How are the other sites implementing design suffixes?
 - Sandia does multiple versions of tabulations. What are other NSE sites doing and how do they define them?
- Use a complex weapons design to exercise use cases to better reflect real world application of the PDMLink tool; this will identify more gaps/issues.
- Consider the impact of having multiple ways of doing the same task in PDMLink and how to address the problem or limit the choices where possible to ensure consistency in tool use. Setting up policies in PDMLink is one way to ensure that designers/engineers always follow the same procedure; however, policies will not likely address all of these issues.

- Determine who will create the product structure: the product engineer, the designer, an administrative clerk, or others?
 - Managing product structure in PDMLink is complex and requires a working knowledge of both the tool and business practices. This differs from Sandia's current EBOM system, which can be administered by staff without that expertise.
 - Designers may only care about the Pro/Engineer structure, not the PDMLink product structure.
- By associating data within PDMLink, the classification of that data might be changed. Classification experts should be included in PDMLink development work to ensure that this issue is addressed.

4.10.2. Topics for Further Discussion and Definition

Throughout the course of the AEE team's product structure work and their discussions with staff from other NSE sites, a number of issues that could not be addressed in this pilot were noted. Those issues are presented here with the recommendation that they be addressed by the NSE PRIDE Product Structure Team:

- View versions work differently in PDMLink 9.0 compare to PDMLink 8.0. Will there also be significant changes to PDMLink 10 when it is released and with what impact?
- There is a need for consistency in the definition of terms within the NSE sites and PTC to eliminate confusion. Examples are:
 - "Configuration" is a broad term that seems to be used in many ways and in different contexts, which creates confusion.
 - "Released" is a lifecycle state in PDMLink. At Sandia it is a term that has multiple meanings in common usage.
 - "Tabulation" is another term that has different meanings in common usage at Sandia. Some of this confusion may stem from the fact that there are multiple techniques for doing tabulation within Pro/ENGINEER.
- Applying taxonomy (smart naming conventions to enable searches) in PDMLink.
 - A designer gave an example of a spacer for a #4 hex screw, which might be duplicated in the PLM system multiple times (perhaps 10-20) because designers did not know it already existed in the system due to differences in the naming of the part.
- The PRS12005 should have a definition for product structure so that all NSE sites have guidance.
- Consideration for additional PDMLink lifecycle states to better address the various states of a product release rather than using a single "Released" lifecycle state.
- Additional business processes that were out of scope for this project need to be addressed, such as Engineering Authorizations.

4.10.3. Suggested Additional Use Cases

The AEE team had planned to continue their work with the following additional use cases, but they did not have time to pursue them. The team suggests that the PRIDE product structure team consider these as logical next steps to the work completed in this pilot:

1. Create a non-modeled version for Use Cases 1-4. Investigate the use of Excel or other automated ways to create product structure for legacy data.
2. Incorporate the following requirements into Use Case 1:
 - a. Weldments, castings and forging drawings
 - b. Multiple fabricated parts
 - c. Library and commercial parts
 - d. Tabulated fabricated parts
 - e. Access policies for documents and WTParts
 - f. Work flows and processes for revising and releasing WTParts
 - g. The use of a non-lettered versioning scheme for WTParts
 - h. The use of a common drawing to support multiple suffix models

4.10.4. Project Documentation

The AEE team worked with SME's from Sandia/NM, LANL and LLNL throughout the pilot to learn from their work, to get their input for defining the AEE pilot effort, and to share information as the team's work progressed. The AEE team delivered several presentations and hosted a hands-on workshop to share their results with that same group. In addition to this project report, the following documents detail the work completed in this pilot project:

- The AEE team's presentation of a step-by-step walk-through of the four use cases using PDMLink screen captures.
- A PTC detailed report for the execution of the four use cases: "Detailed Use Case Document – AEE Project".
- A PTC close-out report for the use case implementation: "Product Structure Pilot: Close Out Documentation".
- A PTC summary presentation of the current design definition business process and the proposed future state; the use case process diagrams; and the issue analysis and gap summaries: "Where are we at with Product Structure?" (December 2, 2009).
- A PTC presentation for understanding view versions in PDMLink: "AEE Product Structure – Capability Aid", Oct. 27, 2009, Jay Hunt and Lisa Boyer, PTC.
- The AEE team also compiled several unpublished project documents to detail the AEE development server configuration: "AEE Soft Documents", "Configuration of AEE Servers to SNL/NM Production Settings", and "AEE Product Structure Settings".

This page intentionally left blank.

5.0 Mentor Graphics ECAD Library Administration Tool Pilot

5.1. Summary

The AEE team piloted the implementation of Mentor Graphics Data Management System (DMS) software as a solution for resolving NSE multi-site shared ECAD library administration problems. This was a partnership effort between Sandia National Laboratories/California (SNL/CA), Sandia National Laboratories/New Mexico (SNL/NM) and Honeywell FM&T (KCP).

The AEE team evaluated only the DMS library administration tool in this pilot. There are additional data management modules available for DMS that might be worthy of evaluation in the future. DMS is Oracle-based mid-ware, and with additional modules, it can provide project lifecycle maintenance capabilities, including a link between the ECAD engineering community and a PDM system (such as PDMLink).

ECAD library administration is currently a manual process which cannot be performed in “real-time” because of the inherent complexities of multi-site administration. DMS affords solutions in the form of tightly controlled and configurable ECAD library administration and data management tools. It provides automated solutions for many of the time-consuming manual tasks currently required, and ensures real-time library updates from multiple sources without the threat of conflicting data. Implementation will provide real-time access to library updates by designers and librarians at all three sites, it will make all sites equal participants in the process, and it will significantly reduce the burden of ECAD library administration.

The AEE team’s objective for the ECAD tool pilot was to determine the value added to the design agency (DA) and production agency (PA) ECAD library administration processes through the use of the DMS tool. The team’s evaluation was limited to the library administration tools included in DMS and included exercising and documenting the following capabilities:

- Incorporation of real-time library updates from multiple sites to a DMS library specification.
- Management of a multi-site library to avoid conflicts and duplication of effort by librarians at different sites.
- Tracking of existing part modifications via a formal check-in/check-out and versioning process.
- Application of permissions and constraints to restrict library editing and need-to-know (NTK) access to appropriate individuals.
- Export and synchronization of production libraries outside of DMS.
- Instantiation of parts into a schematic directly from DMS.
- Ability to access DMS component data from within a DxDesigner schematic.
- Export of reports and spreadsheets from DMS queries.

The AEE team used an SNL/NM Windows virtual server on the unclassified Sandia Restricted Network (SRN) as the DMS server for this project. DMS installation was performed at SNL/NM in collaboration with a Mentor Graphics applications engineer (AE) via telephone. Minimal adjustments to the OOTB configuration were required to ensure error-free functionality in the NSE environment. Librarian and user accounts were created with appropriate access permissions assigned to each type. Because KCP is not allowed to have direct write access to a Sandia server, a crypto-card account was set up for the KCP librarian and an SNL/CA computer was identified for KCP remote login.

The current version of the NSE library was uploaded into a new DMS library specification. Release status assignments for “approved”, “development” and “obsolete” parts were applied and validated. Datasheets were successfully attached to components in DMS. Component and part element check-in, check-out and versioning processes were exercised and metrics were obtained to determine durations for the various processes relative to the different sites. Reports of various types were extracted and printed from within DMS.

DMS allowed real-time updates from multiple sites. Full-up and filtered production libraries were extracted from DMS. Instantiation of parts into DxDesigner schematics was successfully accomplished.

On the whole, DMS worked as advertised. Several areas for DMS improvement were identified and documented. The biggest concern was not a DMS tool problem but a networking “bottleneck”. Sandia/CA was seeing process durations 10 times those of equivalent exercises that were run at Sandia/NM. Because KCP will be running processes through a remote login to an SNL/CA computer, it is logical that they will see similar delays. Investigation into this SNL/CA bottleneck is under way.

Under the current non-DMS process, library updates have to be performed by overwriting the entire master library with a new version. KCP is not allowed direct write access to Sandia servers, so they cannot overwrite the master library. To date, SNL/CA has been updating this library for KCP. With DMS, KCP can upload their own parts in real time via remote login to a Sandia computer. One drawback to the KCP remote login process is that it ties up an SNL/CA Library Manager license. Even so, the annual savings from this and other DMS-simplified library administration processes approximate the annual maintenance cost of the DMS licenses purchased while at the same time decreasing opportunities for human error.

Testing revealed that there is no real advantage to designer instantiation of parts into a schematic from DMS. Instantiation from the conventional library provides the same capability without having to log into a special environment. If additional parametric and support information is added to the DMS component information, the value of DMS instantiation becomes more tangible.

5.2. Mentor Graphics Data Management System

DMS provides tightly controlled and configurable ECAD library administration. It affords automated solutions for many of the time-consuming manual tasks currently required, with the capability for real-time library updates from multiple sources without the threat of conflicting

data. DMS provides a single point where all librarians, designers and engineering customers can access the most current library content. Information is passed from the DMS Data Store to librarians and the ECAD engineering community through DataFusion.

Implementation of DMS library administration tools will provide:

- Real-time access to library updates by designers and librarians at all three sites (with capability for expansion if there is interest from other NSE sites).
- Equal participation of all subscriber sites in the library administration and maintenance process
- A significant reduction in time and resources required by the current manual processes of ECAD library administration.

DMS stores one or more library specifications created by uploading a populated library into a DMS data model. The upload procedure includes the creation of a component information “container” for each part number. Additional data not found in the library itself can be added to this component container. An example would be the addition of datasheets and other pertinent support documentation: this is information that one would not be able to access from within a conventional library. The capability for uploading enterprise Component Information System (eCIS) parametric information into the component store is an attractive possibility. The eCIS tool is a component database system used to manage commercial and NSE part data. It provides searchable parametric data on component characteristics. eCIS is maintained by KCP and is accessible by the Sandia and KCP engineering communities.

DMS itself is not a library; it is a library data store. Therefore, instantiation of parts into a schematic still requires that a design point to a conventional central library. This library is referred to as a *production library* and is periodically synchronized with current DMS information via an update wizard. The content of a production library is determined by a set of named production library filters defined in DMS.

Each librarian works in a local “sandbox”. A sandbox can be an empty library created from DMS or it can be an existing populated library attached to DMS by the librarian. The sandbox provides a familiar library environment in which new or modified parts can be worked. Parts do not have to be built inside the DMS environment: they can be built in a conventional external library and imported into a sandbox library for check-in to DMS. This permits library work to continue in the event of a network or hardware outage.

When checked out from the library specification into the sandbox, parts and/or part elements are locked, preventing inadvertent overwriting of data by other librarians. Checking out a part does not check out the sub-elements of that part. These must be explicitly checked out if they are to be modified.

New or revised parts are checked into the DMS library specification in real time. A search filter lists all the new parts in DMS that are not in the production library, and these search results are then ported directly into the production library list. The master production library is then incrementally updated via the update wizard. At this point, the new or revised parts are available

to the design community for instantiation. The production library is also available for distribution to classified networks or to local drives, if that is the site-specific practice.

DMS provides a web-based search interface through which designers and librarians can search for either specific part numbers or to do parametric searches to provide a list of acceptable parts to choose from. Because DMS has the potential for more information than a standard library, more thorough component searches can be performed.

Figure 17. Typical DMS desktop configuration.

A DxDesigner schematic interface into DMS allows DMS search results to be placed into the schematic directly (instantiation). DxDesigner is the schematic capture tool used by SNL/CA, SNL/NM and KCP for new designs.

All areas of DMS are capable of filtered searches. Query results can be extracted in part or as a whole to a .csv spreadsheet file format.

The DMS library administration capabilities being evaluated utilize two DMS modules:

1. **DMS Librarian:** This is a DMS add-on to the conventional Library Manager software currently used by librarians for part creation and modification. The editing environment is exactly the same as the standard Library Manager. The only difference is that once a library is “attached” to DMS as a sandbox it can only be opened from within DMS Librarian. Since the sandbox has an association with DMS contents, the part and element status can be seen in the DMS Librarian navigation tree. Parts and elements can be checked out into the sandbox, edited, and checked back into DMS via this navigation tree.
2. **DMS Desktop:** This application is accessed on the Web via a URL to the DMS server. A login and password are required to enter the dialog. Overall DMS administration and configuration are handled within this tool. It can also be used to view the DMS contents. The desktop is also used as a conduit for designer component searches and instantiation of parts into schematics. Figure 17 represents a typical DMS Desktop configuration.

5.3. DMS Installation and Configuration

SNL/CA, SNL/NM, KCP and LANL utilize Mentor Graphics Expedition as their standard printed wiring board (PWB) design tool. Expedition is available with either a Design Capture (DC) or DxDesigner (DxD) schematic capture front end. Both flows are currently available to designers, and parallel DC and DxD libraries are maintained. For purposes of evaluation, the DxD/Expedition flow was chosen as the flow of choice. DC is considered a point tool and will not be enhanced by Mentor. The NSE sites are adopting DxDesigner for all new designs.

The specific software versions evaluated were:

DMS2007.7 Update 1	(DxDesigner flow)
EE2007.7 Update 6	(DxDesigner/Expedition flow)

5.3.1. Installation

Cost austerity dictated that the team streamline the license types and quantities to determine the minimum required to perform the evaluation. The types of licenses purchased dictate how much of DMS Librarian and DMS Desktop are accessible and which functions may be in simultaneous use at the various sites. The licenses purchased for this project were:

DMS Base Server DataFusion	(1 base application required)
DMS Library Solution EE	(1 floating)
DMS Libr Mgr Add-on	(2 floating, one of which is supplied by Library Solution EE, requires an existing Library Manager license)
DMS Designer Pro	(3 floating, required for instantiation from DMS)
DMS LFM EE Add-on	(1 floating, supplied by Library Solution EE)

[illegible]

A Windows Server 2003 virtual machine on the unclassified Sandia Restricted Network (SRN) at SNL/NM was used as the DMS license server. Oracle 10g, Apache Tomcat (with http daemon and servlet container), Java Runtime and the DataFusion software were loaded on the server. The DMS licenses were also installed on the server. Mentor Graphics applications engineers provided support for this portion of the installation via telephone.

Figure 18. NSE DMS landscape.

Any machine to be used for librarian functions required some additional configuration. This involved several phone calls to Mentor Graphics and some trial and error before a successful link to DMS was achieved. The actions required consisted of:

- Installation of the Oracle 10g client software.
- Installation of DataFusion, DMS, DMS Librarian and DMS Librarian Flow Manager.
- Assignment of new and revised system environmental variables.
- Installation of the latest cygwin software (a UNIX-like environment and command line interface for Windows).
- Setup of Oracle server recognition (via creation or edit of the tnsnames.ora file).
- Software recognition of the DMS server alias.
- Verification of key internal settings within DMS Librarian.

KCP participants are not allowed direct write access to Sandia servers, and therefore, must use Remote Desktop to log into a Sandia computer for DMS librarian operations. This required the use of a crypto card and the establishment of a user account on the SRN. SNL/CA provided a librarian-configured machine for this purpose. KCP designer interactions are carried out through a Web interface and are read-only, so there is no need for KCP designers to use Remote Desktop for DMS access. Figure 18 shows the relationship between the installed items and their place in the proposed NSE configuration.

5.3.2. Setting User and Group Account Objects (Access Controls)

DMS account objects are entities created in the Administration area of DMS Desktop that tie appropriate library access controls to specific DMS user roles. Only members of the DMS_SUPERUSER group are authorized to create or edit DMS account objects.

Account objects can be either User or Group entities. User accounts are normally created with “empty” permissions, and are then added to group accounts as “members”. Once assigned to a group, the user account inherits the permissions of that group.

Predefined group accounts are available in the OOTB DMS install. Additional group accounts with varying permissions can also be created.

User accounts were created for each of the three site librarians and assigned to these predefined group account objects:

- DMS_SUPERUSER
- DMS_LIBRARIAN
- DMS_ADMINISTRATOR

An additional generic designer account was created for each site and assigned membership to the pre-defined DMS_DESIGNER account object.

5.3.3. Creating a Library Specification

A library specification is a volume within DMS that contains all the component data of a given library. DMS can handle multiple libraries through multiple library specifications. DMS library

specifications are independent and isolated from each other. Parts present in multiple library specifications can have different parametric data or access controls in each specification. An empty library specification called “nwc” was created through the DMS Desktop. This specification would house the latest NSE library. Within the “nwc” specification is the actual database that would store the library data. This evaluation database was called “nwclib”.

5.3.4. Library Upload into DMS

Before populating DMS with the NSE library contents, a component structure needed to be defined. This structure provides the framework and mapping for uploaded library elements and additional information.

The component structure can be as flat or as deep as one requires. The primary advantage to a deep, detailed structure is that it reduces search times by constraining the search to specific component types. The drawback is that more thought has to be placed in determining the appropriate part information, particularly if a part is equally suited to multiple component categories. The negotiated structure depth should be a balance between value-added search time reductions and intuitive partitioning.

The team originally considered uploading data from eCIS as well as the NSE library contents, but realized that the time and resources required to map the breadth and depth of the eCIS parametric information would have taken up all available evaluation resources. Therefore, the component structure chosen was identical to the existing NSE library partitioning.

The library was imported into DMS using the “Add to DMS” command within DMS Librarian. This command turns the populated library into a sandbox associated to the DMS library specification. The contents of the library were then uploaded into the “nwc” library specification. After all parts and part elements were imported, DMS created a component identity (“container”) for each part. This is the area where additional part information (datasheets, graphs, etc.) can be loaded.

The library (10,000+ parts) was imported from a client machine at SNL/CA to the DMS server at SNL/NM. There were 79 errors and hundreds of text truncation warnings. All but two of these errors and warnings were addressed by changing the OOTB DMS settings to accept part numbers with spaces and to change the value length limit of cell and part descriptions from the default 64 characters to 170 and 200, respectively. The two remaining errors were for part numbers with a “:” (colon), which the “Add to DMS” function cannot handle. A manual workaround to import these two parts was identified and implemented for a 100% success rate.

5.3.5. Setting Element Release Status

DMS provides several release status settings that are applied to parts and the elements that make up those parts. Release status can be used as a filter to allow designers to see just those parts that the librarians allow them to see. The status can also be used as a filter when selecting parts for populating a production library.

The available status settings are:

U – Under Construction	D – In Development	A – Approved/Released
R – Restricted	X – Obsolete	

In addition to these letter codes, elements at different status levels have different colored text.

Status can be changed on individual elements or on many elements at a time. The latter is accomplished via a DMS query and the application of bulk modification to the query results. Bulk modification requires some initial tweaking to the “Status” characteristic, as it is not a bulk-modifiable entity in the OOTB DMS configuration.

Since the entire evaluation library was loaded into DMS from a sandbox, all components and their sub-elements were automatically assigned the DMS default “D” (Development) release status. To change a component status to “A” (Approved), one must first change the status of all sub-elements that make up that component to “A”.

The team’s first thought was to change everything to “A” status except for the parts in the “Obsolete” partition, which would be assigned an “X” (Obsolete) designation. After learning that “A” objects have to be checked out and changed to “D” status before they can be edited by the librarian, the decision was made to leave the parts, cells and symbols at “D” status. The smaller objects that are not often edited (padstacks, pads, holes and custom drill symbols) were changed to “A” status. The parts in the “Obsolete” partition were changed to “X” status as originally planned.

New elements being added into DMS will retain this same philosophy. The team may revisit this if they see value added in the more restricted release status setup as more work is done in the DMS environment.

5.3.6. Uploading Additional Information to a DMS Component

Datasheets and other support documentation can be uploaded into DMS and associated to DMS components. There are bulk loaders available for this process, but the resources required to collate the data and set up the mapping files for automated attachment were beyond those available for this evaluation. Therefore, manual uploading of a small sampling of datasheets was tested and documented. PDF is the default file type supported, but other file types can be attached as well.

Support documentation is attached to a component by first creating a documentation object and then attaching the desired information to that object. The component “container” is then modified to associate the documentation object to the part number. Multiple documentation objects can be associated to a single component. Conversely, multiple components can point to the same documentation object.

Documentation is accessible from the DMS component’s search results panel by hitting the “Datasheet...” button. If there are multiple attachments to a drawing object, a list is presented for down-select. If multiple drawing objects are attached to the component, each is available as a separate “Datasheet...” button.

5.3.7. Creating a Production Library in DMS

Designers must point to a conventional central library for parts instantiation. DMS can export a production library for this purpose. A production library is essentially a conventional central library created by DMS using a filtered component list to restrict parts inclusion. For example, a production library might be created from a list of all the parts in DMS except those in the “Obsolete” partition.

Part numbers are added to a full production library list by performing a search on components in a library specification with a “D” or “A” release status and “sending” the search results to the production library list. The “X” status obsolete parts are therefore filtered out of the search, and will not become part of the list, although the part entries still exist within DMS itself. As new parts are created, they can be easily identified via a NULL search, which will list all parts not already in a production library list. The search results can then be ported to the production library list.

Multiple production libraries can be created, each with differing content depending on the need. For example, one production library may contain all of the commercial and NSE parts in a DMS library specification, while another is filtered to contain only qualified NSE parts. The content of a production library is determined by named production library filter lists that are defined in DMS.

A production library begins life as an extraction from DMS using the appropriate production library filter list. Once created, the librarian synchronizes the production library with the latest DMS content via an incremental update wizard.

A synchronized production library contains no “hard” links to DMS, and can be used by any designer, not just designers with DMS access. The production library can also be copied to other locations. This allows the latest library to be copied to a classified network or to local drives, if that is the site-specific practice.

An empty production library filter list called “nwcdxd” was created in DMS Desktop. A query on all parts with “A” or “D” release status resulted in the full parts list for our nwcdxd production library. A right-click “Send To...” function added these search results to the production library filter list. This worked fairly quickly; 10,000+ results were added to the filter in about a minute.

The update wizard was launched in “Create New Central Library” mode to generate a fresh conventional central library extracted from the “nwcdxd” production library list. The process ended with many errors because the OOTB configuration had a part limitation of 10,000 parts. Changing the preferences to 15,000 parts allowed the process to run to completion.

There were 52 errors caused by part numbers with spaces or colons, which the wizard cannot recognize. These parts were imported into the new library manually. These parts were also removed from the “nwcdxd” production library list because the errors they generated slowed down the synchronization process. They can be reinstated to the list if/when Mentor rectifies this shortcoming.

5.4. Testing

A checklist-style test plan and associated support document were developed early on and modified as capabilities and interfaces became clearer.

5.4.1. Check-in/Check-out of Parts and Part Elements

SNL/CA had been checking all of their new and updated parts into DMS for a couple of months without any problems at the time of this report. All check-in and check-out operations are performed from a sandbox in DMS Librarian. Parts that are not in a sandbox cannot be checked out.

Visual markers in the navigation tree indicate part status:

- **Locked Parts/Elements:** The entry has a yellow lock icon attached. These are parts or elements that are checked out for edit by another librarian and cannot be checked in by anyone but that librarian.
- **Hijacked Parts/Elements:** The entry has a blue check covered by a red “X”. “Hijacked” entities (the official DMS terminology) are parts/elements that have existing DMS entries but have been brought into the sandbox from a source other than DMS, usually through Library Services. They are recognized as “locally modified”. Because of their unknown pedigree, the librarian has discretion as to whether a “hijacked” part overwrites an existing DMS entry on check-in.
- **Checked-in Parts/Elements:** The entry has no visual markers. These are parts/elements in the sandbox that already exist in DMS.
- **New Parts/Elements:** The entry is followed by a blue asterisk. These are parts/elements that are not in DMS.
- **Checked-out Parts/Elements:** The entry has a blue check. These are parts/elements that are checked out by the librarian for editing. This entry will show up as “Locked” to all other librarians.

“Check-in”, “Check-out” and “Add to DMS” actions are initiated by right-clicking on the entry and selecting the action. Only the appropriate actions for the status type are available.

For hijacked parts, the librarian is asked if they want to make the locally modified version the checked-out part. If answered “No”, the DMS version is checked out and it overwrites the local version. If answered “Yes”, the local version is designated as checked out and will overwrite the DMS version upon check-in.

Figure 19 shows the various visual status cues of parts/elements. The KTJ101B106M55BFT00 part hierarchy has been expanded to illustrate how DMS provides sub-element status information.

Figure 19. Typical status visual cues.

- The navigation tree does not allow multiple selections, so sandbox parts can only be processed one at a time.
- An entire external (non-sandbox) library *can* be processed using the “Add to DMS” command, but this is an overwrite function that will totally replace any existing DMS entries with the external library versions. This would also reset the element version to zero.
- Parts and sub-elements can be checked in or checked out from a sandbox either as individual “pieces” or as a hierarchy.
 - Individual check-in requires that the lower level part elements (padstacks, cells and symbols) be checked in before the parts that utilize them can be checked in. Checking out an individual part does *not* check out the sub-elements of that part. These must be explicitly checked out if they are to be modified. The example in Figure 20 shows that the librarian has explicitly checked out the cell used by the KTJ part. That cell is now locked from modification by any other librarians. The symbols have not been checked out and are therefore eligible for check-out and modification by any librarian. Normally, sub-elements are not revised, so this is not a problem.
 - Check-in and check-out of a hierarchy will process a part and its sub-elements in a single operation. A table showing the status of the part sub-elements allows the librarian to select which elements of the hierarchy are to be processed (Figure 20).

Select objects for Checkout Hierarchy					
	Name	Partition	Type	Status	Released
☒	egd_hierarchy_test_part	Clks-Crystals	Part	Checked In	No
☒	egd_hierarchy_test_cell1	PTH_IC	Cell	Checked In	No
☐	Thru Rd 59 PTH 36th	---	Padstack	Checked In	Yes
☐	Thru Sq 59 PTH 36th	---	Padstack	Checked In	Yes
☒	egd_hierarchy_test_cell2	PTH_IC	Cell	Checked In	No
☐	Thru Rd 51 PTH 34th	---	Padstack	Checked In	Yes
☐	Thru Sq 51 PTH 34th	---	Padstack	Checked In	Yes
☒	egd_hierarchy_test_symbol	IC_Digital	Symbol	Checked In	No

Figure 20. The Checkout Hierarchy Selection table.

- When two or more librarians are in DMS Librarian, status changes resulting from one librarian's actions will not be visible on the other librarians' screens until a refresh is performed. DMS does recognize any status changes regardless of the screen depiction, so proper lockouts are in place to prevent a librarian from checking out a part that is under modification at another site.

5.4.2. Creating New Library Parts

There are four ways to generate new parts in the DMS environment, and each has its own set of advantages and drawbacks:

1. Perform all of the work in DMS Librarian within a "full" library sandbox. This requires that a full production library be made into a sandbox, which requires a significant initial time investment for processing. From that point, it can be synchronized with DMS as a production library. Standalone production libraries could be created from this sandbox by making a copy and then deleting the copy's .sbx and .dat files. These are the files that tie the library to DMS as a sandbox.
 - Pros: All parts/elements are available for immediate check-out. There are no hijacked elements to worry about.
 - Cons: There is a significant time lag each time a DMS Librarian opens the sandbox because the application searches through the sandbox to update the visual part/element status in the navigation tree. Since there are thousands of parts in the library, this process could take a while.
2. Perform all of the work in an empty DMS sandbox using the "Add to Sandbox" command to search and retrieve appropriate DMS parts into the sandbox for editing.
 - Pros: The initial sandbox update is relatively quick. The sandbox retains the visual status cues of the existing DMS elements.
 - Cons: Does not provide a "full" library for immediate search/edit.

3. Perform all of the work external to DMS in a transfer library and then update DMS with the total contents of the transfer library. The library is attached to DMS as a sandbox and its parts/elements are checked in via the “Add to DMS” command.
 - Pros: The initial sandbox update is relatively quick.
 - Cons: The "Add to DMS" action overwrites any existing DMS part/element with the transfer library's version. It will not "rev" the element like a "check-out/check-in" does; it will be listed as the first and only revision. This option does not take advantage of the DMS part status visual cues because there is no real attachment to DMS.
4. Use Library Services to bring new parts/elements into an empty DMS sandbox for manual addition into DMS through the DMS Librarian navigation tree.
 - Pros: The initial sandbox update is relatively quick. Elements from the transfer library that already exist in DMS are “hijacked” parts which allows some flexibility for the librarian to determine whether to overwrite legacy DMS entries with local updates.
 - Cons: Because of the hijacked element status, the librarian must keep track of any element modifications that are done in the transfer library to insure that the local version of that element overwrites the DMS version.

As stated earlier, when *existing* elements are checked out in DMS, a “lock” is automatically placed on that element so that only the editing librarian has modification access to it. When the element is checked back into DMS, the lock is released. This is how DMS manages conflicts and prevents duplication of effort.

However, none of the processes as described above will lock a *new* part/element because nothing exists in DMS until that part/element is checked in. Even the DMS Parts Request module (not purchased for this evaluation) would cover only the part level lockout, not the symbol and cell element locks required for full conflict management.

The team concluded that the only way to force a lock on a new element is to create a component entry in the sandbox, create/assign a symbol and cell to that entry, check in the part hierarchy and immediately check it out for edit. This puts a lock on that part and its elements, protecting them from edits by other librarians.

5.4.3. Part and Element Searches

All sections of DMS were searchable. Wild cards were accepted, and there were several filters that could be configured to tune the search. The filtering capabilities changed to match the type of information one was searching for.

5.4.4. Updating an External Library from a DMS Production Library List

The Update Production Library Cache wizard was run in “Create New Central Library” mode to create a fresh production library from the “nwcxdx” production library filter list. The wizard has since been run many times in “Incremental” mode to keep this library in sync with DMS. SNL/CA has been using the resulting library for production work on the SRN and on a copy of that library placed on the SCN and has seen no problems. Typical DMS incremental updates

average between 2 and 12 minutes depending on the extent of information to be updated. Update duration and any errors or warnings are captured in a log file.

5.4.5. Designer Testing

A sample design was populated using instantiation from DMS and there were no errors or warnings during packaging. Instantiation from DMS is performed by creating an IPC-DMS connection. This connection links the searchable information in DMS to the actual synced-up central library mapped by the design. The IPC-DMS link is made via setup of an environmental variable and by launching an executable file that adds a DMS pick to the DxDesigner menu bar. The menu bar can be customized so that the executable can be easily initiated from within DxDesigner.

DMS has a toolbox that can be configured to tell DMS what properties to pass to the schematic design. Once the toolbox is configured, all of the proper values are passed through to DxDesigner for packaging.

Currently, DMS has the exact same parametric information as that of the central library. Because part instantiation is done through a conventional central library, DMS would provide no added advantage to designers for identifying available parts. The fact that datasheets can be attached to components in DMS does provide some advantage, but the current DMS store has only a handful of datasheets in place; it would require extensive resources to adequately populate the DMS datasheets store.

If eCIS information is ported to DMS, the advantage increases. At that point, DMS would contain much more information than the conventional central library. The team would then envision a parts search to be initiated through DMS to determine if an acceptable part is available in the NSE library. Instantiation could then be performed from the search results.

Designers can select a DMS-instantiated symbol within their DxDesigner schematic and view the associated DMS component data via a “View in DMS” pull-down menu function. This provides the designer with the typical DMS Desktop configuration illustrated in Figure 17.

5.4.6. Data Export

Several reports were successfully generated from DMS. One can export any search result (in all or in part) using the “Export to CSV” or “Export Selection to CSV” options. This exported data does not include column headings, however.

5.4.7. Printing

DMS prints search results or individual component text information in report form via the “File>Print” pull-down menu. Various subsets of the component information, including graphics of a cell or symbol, can be printed by placing the cursor over the appropriate window and using the “Print” command from the right-click pull-down menu.

5.5. Library Administration Process

When SNL/NM, SNL/CA and KCP moved from VeriBest to Mentor Graphics Expedition as their production ECAD design software, the decision was made that all three sites would utilize a single master library. This would ensure standardization of library contents, remove duplication of effort, and provide a set of librarians that could smooth out peaks and valleys of workload at any one site.

The master central library exists as a compressed file located on an SNL/NM server. Upon notification of a library update, each site's librarian downloads a copy of this server library for local use at their site. DMS was evaluated to see how its usage might enhance and streamline our current library administration processes and to determine its ability to provide a more collaborative environment for multi-site library management.

5.5.1. Current Library Administration Process

Current library administration is a 100% manual process and cannot be performed in real-time because of the inherent complexities of multi-site administration. The addition of new parts to the library involves each site adding their new parts to a local copy of the most recent master library, performing the many manual steps required to make the libraries fully useable and searchable by the customer base, and then overwriting the existing master library with a totally new library.

The library update process as a whole must be carefully choreographed to insure that the sites are not overwriting each other's information or introducing conflicts. For example, there is a window of opportunity once every other week when SNL/CA can perform a master library overwrite. But during that 2-week period, the SNL/CA librarian has to update a local SNL/CA library with the new parts so that SNL/CA designers can access them and then repeat the process for the master library update. This is a duplication of effort. Additionally, KCP is not allowed write access to the NM library server, and must "ship" new parts to SNL/CA for inclusion.

Figure 21. Comparison of the current library administration process and the new DMS library management process.

The current library administration process consists of the following steps, which are illustrated in Figure 21:

1. The Site Librarian receives a request for a new library part.
2. The Site Librarian opens a pre-defined template copy of an empty library. This empty library has the same part, cell and padstack partitions as the “master” library on the SNL/NM server; however, it has no content.
3. The Site Librarian creates the new part(s) in the Library Manager. Normally, parts or part elements that are similar to the part being requested are brought into the empty library via Library Services and then renamed and edited to become the new part. The parts are tested in a new job to verify that all aspects are present and addressed. This verification includes running Packager and generating a BOM. The verification is considered a success after the placement of the part into a physical PWB test layout without errors. However, special cases where new processes are being tried may require the generation of Gerber artwork and drill files to verify the correctness of the final output.
4. If the SNL/NM Site Librarian is creating the parts, skip to Step 5.
 - a. If the SNL/CA Site Librarian is creating the parts, the librarian must negotiate with SNL/NM to determine an appropriate date and time for a library update. This is because the update will overwrite the entire existing “master” library. If this is not coordinated, new SNL/NM parts could be lost in the overwrite. If additional new parts are requested in the lag time before the scheduled update time, these new parts are created. A local SNL/CA library is then updated with the new parts so that SNL/CA designers have access to the parts in a timely fashion; designers normally cannot wait up to 2 weeks for the negotiated periodic update.
 - b. If a non-Sandia Site Librarian (KCP, LANL, or other agency) is creating the parts, the librarian must forward a library containing the completed parts to the SNL/CA Site Librarian, who will include the parts in their update. This is because non-Sandia sites do not have write access to the SNL/NM server, so they cannot overwrite the master library.
5. Just prior to the update, the Site Librarian downloads a copy of the latest “master” library from the NM server.
6. The Site Librarian uses Library Services to copy the new parts from their transfer library into the local copy of the “master”.
7. The Site Librarian deletes the index file and rebuilds the .lmc file, inspecting for any warnings or errors in the log files.
8. The Site Librarian sets the default printer to Adobe Acrobat and runs Report Writer to create the library part, cell, and symbol and padstack tables. The tables are then printed to PDF files that are added to the library folder. These PDFs are how designers check to see what parts are available in the library.
9. The Site Librarian deletes the “vbreport” folder generated by Report Writer (this can be a large file, on the order of 34MB). Log files are also deleted to streamline the library folder size.
10. The Site Librarian creates a compressed ZIP file of the updated library.

11. The Site Librarian goes to the SNL/NM server, moves the existing (old) library ZIP file into an archive directory and then copies the updated ZIP file onto the server. From SNL/CA, this can take upwards of 10 minutes due to network interactions. A copy of the new ZIP file is also placed in a shared folder in Sandia's Dropzone data repository so that non-Sandia customers can access it. Downloading the ZIP file from the Dropzone folder can take upwards of 10 minutes for these customers.
12. The Site Librarian sends an email to all sites to notify them of a library update.

5.5.2. New Library Administration Process Utilizing DMS

As a baseline, DMS provides the designer community access to new parts through the update of a production library. Typically, a site librarian would update their site's master library from DMS using the update wizard. All designers at that site can point to this library. Designers still have the flexibility to copy the site production library to their local drive if network issues are a concern. Designers can run the update wizard at any time on a local library to sync it to DMS. Only librarians can run the update wizard on a site's master library.

The proposed DMS process is as follows:

1. The Site Librarian checks the new or modified parts into DMS.
2. The Site Librarian updates the DMS production library filter list to include the new parts.
3. The Site Librarian synchronizes their site's master production central library with DMS using the Update Production Library Cache wizard.
4. The Site Librarian inspects the update log and addresses any errors or warnings.
5. The Site Librarian sends an email to the appropriate library requestors to notify them of the library update.

5.6. Evaluation Results and Recommendations

The following section summarizes the AEE team's DMS evaluation and provides recommendations for product improvement.

5.6.1. DMS Library Administration Evaluation

DMS allows the sites to perform updates on their own individual timetables, requiring no inter-site negotiation. The ability to update the library in real time without worrying about overwriting another site's library reduces the amount of administration needed to keep track of new or modified parts. Currently, the SNL/CA librarian has to retain stockpiles of updated parts until a window of opportunity opens and then has to spend an entire day coordinating and performing the actual upload of the new parts into the "master" library.

Under the current non-DMS process, KCP is not allowed direct write access to Sandia servers, so they cannot update the master library. SNL/CA must update this library for them. Using the DMS process, KCP can remote log into an identified Sandia/CA computer, and upload parts into DMS. They can then run the production library update wizard to update the production library.

The savings from this and other DMS-simplified library administration processes approximate the maintenance cost of the DMS licenses purchased while decreasing opportunities for human error.

5.6.2. DMS Designer Evaluation

Because DMS contains little more than what is already in the conventional central library, there is no real advantage to designer instantiation from DMS. This scenario may change as DMS is populated with additional component support information.

5.6.3. Comparison to Other Applications

There are no comparable applications that provide the administration tools afforded by DMS. PTC's ECAD Workgroup Manager attempts to address the creation of production libraries through the use of WTParts, but it does not have all of the capabilities of DMS. Also, the ECAD Workgroup Manager is currently incompatible with any version of Mentor Graphics Expedition above 2005.3.

5.6.4. Recommendations for DMS Improvement

The AEE team makes the following recommendations to Mentor Graphics to improve DMS:

- **Publish a “Quick-Start” guide for first-time installers.** There are no “quick-start” write-ups that gather basic Oracle, DataFusion, Apache Tomcat and DMS installation and setup processes into a single document. Individual setup operations are spread across several multi-hundred page manuals. The team had to construct such a document from phone calls to the Mentor AE and from trial and error. None of the processes defined were site-specific, so there was no reason that an acceptable generic quick-start guide could not be written.
- **Application login interfaces should be more consistent.** Most DMS logins require the user to enter an IOR file path. Example:
`http://cadvm02.srn.sandia.gov:8090/DMSBrowser/DataFusion.ior`
- For some logins, the IOR path fills in automatically, while for others the user has to type or paste the path into the text box. Also, the language setting doesn't remain static in all logins. Sometimes it is set to “e” (for English), and sometimes it resets to the default “g” (for German).
- **Make “area-based” functionality changes more visible to the user.** Available options change as one moves the cursor into different areas of a DMS dialog. This functionality change is sometimes only recognizable by the addition of a tiny blue icon associated to the cursor. This particular functionality change did not seem to be documented anywhere, and was only explained via a teleconference with the Mentor AE.
- **Action options on search results should be available without having to first select the line entry.** Right-mouse-click functions (Add, Modify, View, etc.) do not become active until one selects an object; this adds an additional, non-intuitive step to the process and differs from how most applications outside of DMS function. This type of selection

makes sense only if one wants to select multiple line entries and perform a function on the group.

- **Library upload/component synchronization speed needs to increase significantly.** A full library upload using the “Add to DMS” command took 46 hours at SNL/CA. There really isn’t anything one can do on the computer while this is processing, since the upload ties up all the machine’s resources. Granted, there appears to be some sort of networking bottleneck at SNL/CA, but the SNL/NM upload took upwards of 6 hours. Note that this type of “full” upload is really only performed once, when the entire library is uploaded into DMS. The normal upload is incremental in nature and takes significantly less time, normally between three and thirty minutes. However, there are occasions when bulk uploading of datasheets will push this upload time outward to an hour or two. This is a long time to tie up a production machine.
- **DMS constraints should not be tighter than those of the applications that interface with it.** For example, DMS can be configured to upload part numbers with spaces. However, the Update Production Library Cache Wizard cannot process part numbers with spaces. What goes in should be able to come out.
- **DMS should lock new parts and elements as soon as they are named.** Testing showed that DMS does a good job locking *existing* elements but that the only way to lock a *new* element is to create it, check it into DMS, and then check it out. If the part is a 1000-pin component, there will be a long lag time before the part definition is sufficiently defined to pass the minimum definition DMS requires for check-in eligibility. During that lag time, there is no lock in place and other librarians could try and create the same element, not realizing they are duplicating another librarian’s efforts.
- **DMS should not rely on designer “tweaks” for workarounds to internal problems.** Per the previous example, the Update Production Library Cache Wizard should accept part numbers with spaces and/or colons. Because it does not, librarians have no alternative than to upload these “unrecognized” parts into the production central library manually. Mentor’s suggestion to change these parts to remove the offending characters is not acceptable, as these are all valid vendor procurement part numbers and cannot be changed without downstream consequences.
- **DMS should not tie up non-DMS licenses in order to function.** DMS requires a Library Manager license in order to function. Library Manager is an application purchased years ago as an Expedition license, *not* as a DMS license. Since KCP logs into an SNL/CA client machine to perform DMS Librarian tasks, the only SNL/CA Library Manager license is pulled, locking out the SNL/CA librarian from doing any Library Manager work.
- **Dialog changes in DMS should be more visible to users.** DMS search and modification panel changes are very subtle. It is often difficult to tell where one is in the DMS environment because the various panels are almost identical. One must pay close attention to the tabs and panel headers to make sure they are performing a function in the intended area. A text or background color change to indicate when a panel changes is desirable.

- **New DMS panels should not be constrained by the size of legacy panels.** New panels are often squeezed to fit and sometimes overlap and occlude other panels. Important areas of the new panels are often hidden until they are scrolled into view. The team recommends that the legacy panels be squeezed instead.
- **Allow DMS check-in to process multiple parts at a time.** Currently, parts/elements can only be checked in one at a time. There is no capability for selecting multiple elements in the DMS Librarian navigation tree.
- **Allow instantiation directly from DMS without requiring an external production library.** The team was surprised to find that an external library was a requirement for DMS-based instantiation. At first, this seemed to defeat the purpose of DMS. If all the component data and the production library filter list information are stored within DMS, why could parts not be instantiated from DMS without an external library? The team then realized that they have to generate an extracted conventional library to update classified systems anyway, so the actual impact was minimal.
- **Multiple attachments within a single documentation object should be more intuitively accessible.** When a documentation object contains multiple attachments, a designer clicking on the datasheet's tab only sees a list of the document object title and the attachment index number as shown in Figure 22. There should be some way to display a column of attachment titles at this level so designers do not have to "dig into" the documentation object for more info. Having separate drawing objects for each datasheet is not a solution because a single component would be listed multiple times in a search (once for each attached drawing object).

Figure 22. The datasheet selection tab.

- Have the option to group items in the DMS Librarian navigator tree by status (checked-in, checked-out, new, hijacked, locked), not just alphabetically. The tree can become quite long and one often has to scroll to perform a function. Every time an operation is performed, the screen has to do a visual refresh to capture the changed part status and it snaps back to the top, so the user is constantly scrolling to get to the next part for processing. The team recommends that a grouping option to display all parts and sub-elements by status be provided.

- Stop the DMS Librarian navigator tree from snapping back to the top after a post-operation refresh. Again, this snap-to-top action requires the designer to constantly scroll to get to the next part.
- Provide more intuitive identification of internal characteristic names. Several of the OOTB characteristics settings had to be changed, but it was difficult to locate the characteristics because of their obscure names. For example, the cell description characteristic is called 130descr, and was only identifiable by eliminating the obvious and through trial and error with what remained. At the very least, there should be a searchable column for the intuitive DxD and Expedition property names associated to the internal characteristic callout.
- When exporting data to a .csv file, include the headers. Without a header, it is difficult to determine what the data actually represents.

5.6.5. *Lessons Learned*

The AEE team documented the following lessons learned as they performed the DMS evaluation:

- **Don't add library parts to DMS using the "Add to DMS" command if one can avoid it.** It is time consuming. If there are bulk uploads, do them from an SNL/NM machine, preferable over a weekend.
- **Unless all parts/elements are new, check-in "hijacked" elements into DMS instead of using the "Add to DMS" command.** There is more librarian control and flexibility over modifications to existing items, and versioning is retained. Many times legacy parts are updated to reflect new requirements. The "hijacked" parts allow for the librarian to choose whether they want to overwrite the DMS entity with the modified sandbox version.
- **DMS administrators should have a Mentor AE's cell phone number handy for installation and for nuances of standard operations.** Instructions are often hidden in one of many multi-hundred-page manuals. There also appear to be a lot of undocumented features in DMS that can be useful.
- **DataFusion stops occasionally, usually after SNL/NM's scheduled Thursday night Oracle maintenance.** This requires a server administrator (2-factor identification) to log into the DMS server and restart the DataFusion service. There is one team member at SNL/CA and one at SNL/NM with this capability. KCP cannot be a server administrator on a Sandia machine.
- **Document any part numbers containing special characters that the Update Production Library Cache Wizard cannot process.** These parts will have to be added to the production central library manually. A spreadsheet documenting these parts can save librarians some time researching error messages. A service request has been submitted to Mentor to see if this can be fixed.
- **Get used to the "Specify/Send To" functionality within DMS Desktop.** This is a right-click methodology is used in conjunction with a "specify" icon to move data from a

search result to another area of the dialog. “Specify/Send To” is used consistently in many of the DMS Desktop environments.

- **Older versions of an entity are listed in DMS, but are not easily retrievable.** Since most part/element revisions are made to incorporate added capability or to correct defects, there probably won’t be much need to do a rollback to a previous version.

5.6.6. Next Steps

The 2009 AEE evaluation was confined to library management, but there are additional data management modules available for DMS that might be worthy of evaluation in the future. DMS is Oracle-based mid-ware, and with additional modules, it can provide project lifecycle maintenance capabilities, including a link between the ECAD engineering community and a PDM system (such as PDMLink).

Although not targeted for this FY effort, DMS can be expanded via additional modules to interface with PDMLink via a PLM2PLM Adaptor. Mentor’s latest incarnation of this adaptor is supposedly much less complex than its predecessors, and relies on XML files to make the required connections. Mentor consulting would only be required for setting up specific NSE customized interfaces. This interface can allow mapping and synchronization of component information, release of finished designs and BOMs, and check-in/check-out from PDMLink through DMS for mid-life changes. DMS can also provide configuration management and access control for the many ECAD manufacturing files that are currently just “zipped up” with the design and checked into PDMLink as a single object.

The DMS library part request tools should be evaluated and compared to the current eCIS request form. Also, the feasibility of an eCIS-DMS part request interface should be investigated.

Direct import of component data from other databases, including eCIS, should be exercised. DMS accepts input via an ASCII loader or XML.

Setup of component type-specific search criteria can be explored. Different types of components use different types of properties. DMS searches and search results can be set up to display only those properties specific to a component type. For example, a capacitor search would not need resistance or power rating as search criteria, so these properties would not be part of the search parameters for capacitors. Component type-specific property tables had been developed pre-DMS as librarian aids and need to be reviewed to ensure they are complete and current. Once the tables have been verified, the DMS component search configurations should be tailored to match them.

6.0 MCAD Data Migration Project

6.1. Summary

Sandia is transitioning to a new electronic enterprise product lifecycle management (PLM) environment using the Windchill PDMLink 9.0 application. As part of this transition, legacy mechanical computer-aided design (MCAD) data must be moved from Sandia's existing PLM systems into new Windchill 9.0 systems in both classified and unclassified environments. This effort is managed by Org. 2996, MCAD Design for Defense Programs & Missile Defense. The AEE project contributed to this effort by providing funding to support the Data Migration team's ongoing efforts to refine and deploy a previously developed data migration plan. The AEE funding was used only for the PTC consultant costs; Sandia project staff and other costs were funded from a different source. The funding period was approximately two months in duration.

Project activities included preliminary testing and evaluation for moving legacy MCAD data from the unclassified MatrixOne PLM system into a Windchill PDMLink 9.0 development system, followed by a full migration of the legacy data. This process required the use of an intermediary application, Pro/INTRALINK, to transition the data between systems. The team also completed testing and a full migration of legacy MCAD data from the unclassified Windchill 8.0 production system into the Windchill 9.0 development system, again with the use of the intermediary system.

A number of significant problems were encountered with the data extractions from both legacy systems. There was a higher than expected number of errors in the extracted configurations, primarily due to external references to other objects that could not be found. The software scripts that were used to programmatically extract data from the legacy systems were modified to rectify some of these problems, but could not address them all. At the conclusion of the AEE-funded effort, the data migration team was at a decision point: Do they continue to try to refine the data extraction scripts to reduce the number of errors and run additional full data extractions, or do they move forward and manually address each configuration error individually? Time constraints led to the decision to move forward and begin data migration with a plan to manually migrate the data that was not included in the programmatic migration.

This report provides a summary of the AEE funded data migration work only. A report documenting the complete MCAD data migration effort will be published by Org. 2996 at the conclusion of the project.

6.2. Objective and Scope

The objective of this project was to support the migration of legacy MCAD data from Sandia's unclassified MatrixOne and Windchill PDMLink 8.0 PLM systems into a new Windchill PDMLink 9.0 system. The scope of this effort was limited to funding the continued work of two PTC Windchill consultants to support the data migration team's implementation of the data migration plan. This work was underway, but the team was facing a stoppage in the consultants' work due to a gap in project funding. The AEE funding enabled the data migration team to bridge this gap and continue the consultants' work uninterrupted.

6.3. Introduction

Sandia currently manages its MCAD data for nuclear weapons engineering and defense systems and assessments work (DS&A) with two product lifecycle management systems (PLMs): MatrixOne and Windchill PDMLink 8.0. MatrixOne has been in place as a data repository since 2001, and has been used by MCAD design staff at both Sandia sites to manage Pro/ENGINEER CAD files. Engineering customers also use MatrixOne through the web based interface known as Design Definition Manager (DDM) to manage non-MCAD objects and documents. This latter user base makes up the majority of MatrixOne users.

Windchill PDMLink was selected in 2007 to replace MatrixOne on the basis of its enhanced capability to manage all forms of product data. Windchill PDMLink, unlike MatrixOne, is fundamentally based upon industry configuration management principles with workflows and processes which will support SNL's transition to an integrated product lifecycle environment for all product related data. A Windchill PDMLink 8.0 production system was installed in April, 2008, which began the user transition from Matrix to the new PLM solution. The Windchill 8.0 environment was used primarily for design process prove-in (workflows, team roles, access controls) and management of Pro/E MCAD files that had not been managed in MatrixOne. Other functional areas such as non-CAD documents, change management, and product structure were intended to be rolled out in later production phases.

Sandia's intent was to migrate relevant legacy data from these two PLM systems into a new Windchill PDMLink 9.0 system and use this new system to realize the following benefits:

- Consolidation of data into a single environment.
- PLM tool integration with PRO/ENGINEER, the NSE CAD tool of choice.
- Establishment of a true enterprise PLM tool.
- Support for formal change processes based upon CMII fundamentals; this can potentially replace the Sandia EAWEB system.
- Supports product structure with all product related data; this can potentially replace the Sandia EBOM system.
- A single source, trusted repository for design definition; a potential Image Management System (IMS) replacement.
- NSE sharing of design definition is more transparent.

To facilitate this data consolidation effort, Sandia contracted Parametric Technology Corporation (PTC) to work with the Sandia data migration team to develop and implement a data migration plan. This plan presents a step-wise approach to test and validate the data extraction and data loading processes on development systems prior to a full migration of legacy MCAD data. The data migration effort is not a simple exercise, as the data contained in Sandia's PLM systems consists of tens of thousands of data files that have been generated over many years. Although Sandia employs detailed processes for maintaining these PLMs, data inconsistencies coupled with the complex relationships and dependencies between CAD files will complicate the data migration process. The data migration plan presents a step-wise approach to test and validate the

data extraction and data loading processes on development systems prior to a full migration of legacy MCAD data.

This project continues work that was already completed by the Sandia MCAD data migration team, which includes the development of a high-level data migration plan and first iteration scripts for extracting data from Sandia's unclassified MatrixOne and Windchill PDMLink 8 PLMs. The Data Migration team had already completed most of the required planning and infrastructure development work. This project focused on the implementation phase of the data migration process, which included rehearsals for data extraction, loading the data into an interim system, validating the loaded data, refining the data extraction scripts as needed, and loading the data into the Windchill 9.0 development system. These rehearsals were intended to test and refine the process in preparation for a full-scale data migration of all of Sandia's legacy MCAD data, which is out of the scope of this project.

6.4. Data Migration

Although data from the Windchill 8.0 system can in theory be moved into the Windchill 9.0 system through an application upgrade, a previous PRIDE funded effort to validate this process determined that there were insurmountable obstacles and an alternative data transfer method was required. There is also no direct path for migrating data from the MatrixOne system into PDMLink 9.0. The solution proposed by the PTC consultants was to use an interim application called Pro/INTRALINK to transition data from both legacy systems into the new PDMLink 9.0 development system.

For this process, data from each system is first extracted using custom JAVA extraction scripts and then loaded into the Pro/INTRALINK system where it is validated to ensure no errors or loss of data. If significant errors or missing data files are identified, the source of the problems are identified, the data extraction scripts are modified, and another data extraction is performed. Once data is validated to a level deemed acceptable, the data is loaded from Pro/INTRALINK into the PDMLink 9.0 PLM system using a Windchill migratory tool that takes the Pro/INTRALINK data and maps data types, attributes, states, versions and other characteristics to the PDMLink 9.0 equivalents.

Per the data migration plan developed by Sandia and PTC²¹, a series of rehearsals were designed to test and refine the custom JAVA scripts that are used to extract data from the legacy PLM systems. The process for the data extraction and validation used for each of the two legacy PLMs was as follows:

- Develop a data extraction script.
- Perform a data extraction rehearsal.
- Validate extracted data using PTC's ModelCHECK application. ModelCHECK provided useful information regarding model quality and data integrity, but results were not used to refine extraction scripts. ModelCHECK was only run on the first and initial extraction from MatrixOne. Subsequent data extractions did not have a ModelCHECK component.

²¹ *Data Migration Plan – Sandia National Labs (draft)*, May 2009.

- Load data into the interim Pro/INTRALINK system.
- Process errors and re-run all configurations that had conflicts on a single CAD worker machine (rather than on multiple machines).
- Validate data in the Pro/INTRALINK environment using Pro/ENGINEER Wildfire 4.
- Identify the source of errors or missing data files.
- Modify the data extraction script as needed. Most errors were due to data integrity problems such as external references, missing family table instances or same named instances with different generics, or regeneration failures.
- Perform another data extraction rehearsal.

The data migration development work was conducted on two development servers, “Dev1” and “Dev2”. These servers were set up as clones of the Windchill 9.0 production environment to serve as test environments into which the data from the interim Pro/INTRALINK system will be migrated.

6.4.1. MatrixOne Data Extraction: Rehearsal 1

The first MatrixOne data extraction rehearsal was limited to extracting data from the unclassified MatrixOne system and loading it into Pro/INTRALINK only. This rehearsal was used to confirm the fidelity of the extracted configurations and the connectivity between the iLink data loader, Pro/INTRALINK, and the CAD worker machines that were used to verify the loaded data.

The MatrixOne data extraction was performed using the first iteration custom data extraction scripts developed by Sandia staff. The data extraction process creates list files for each extracted configuration which are used by the Pro/INTRALINK data loader and the ModelCHECK verification tool. These list files had to be processed to modify each entry to achieve compatibility with both tools, as requirements were different for each.

The extracted data included all of the latest released configurations of drawings, assemblies, parts, and all subsequent in-progress work. Once the configurations were extracted from MatrixOne, ModelCHECK was used to identify any data integrity issues. Nineteen individual CAD worker machines were used to complete the verification process.

Before loading the extracted data into Pro/INTRALINK, the ParaCAD Lite library set, as defined by SNL/CA, was loaded into the Pro/INTRALINK database. This library had to be loaded into the database before the configurations so that ParaCAD library parts in assemblies that were extracted from MatrixOne would be replaced in Pro/INTRALINK. The Pro/INTRALINK loading infrastructure was tested and proven with the loading of the ParaCAD library, which consisted of approximately 11,000 files.

A preliminary test load into Pro/INTRALINK was conducted using a small configuration comprised of about 20 parts. This configuration was successfully loaded and used to validate version and state mapping. A second test load using a larger assembly comprised of approximately 800 parts followed. However, data integrity errors prevented the configuration

from being loaded. An investigation and analysis by the PTC consultants narrowed the causes to a configuration setting in the Pro/INTRALINK data loader and a data integrity issue.

The team made the decision to attempt to load all of the MatrixOne data into Pro/INTRALINK. The loading process took approximately five days to complete. Of the 155,250 extracted configurations, approximately 69% of the data loaded successfully, based upon validation of 555 unique configurations. The PTC consultants ran error code interrogation on the failed configurations and the configurations with “conflict 160” errors, errors due to duplicate parts in Pro/INTRALINK or parts that were locked by another CAD worker, were rerun using a single CAD worker machine.

With the data load completed, nine Sandia staff performed a validation of the data per a plan that was developed by SNL/CA staff²². These validators loaded Pro/INTRALINK on their client machines, configured it to work with Pro/ENGINEER Wildfire 4, and followed a proscribed process to validate the data. Between the nine validators, approximately 550 randomly selected configurations were searched for in Pro/INTRALINK and opened in Pro/ENGINEER.

Approximately 30% of the data was either not found in the Pro/INTRALINK database or had regeneration problems. Based upon these preliminary results the team decided to modify the MatrixOne extraction scripts to exclude version mapping. They had determined that a common version scheme, “LGY”, that was assigned to all objects with non-standard version schemes had unintended consequences which adversely affected some of the configurations.

Using the modified MatrixOne extraction scripts, the team extracted the entire MatrixOne data set again in preparation for a second Pro/INTRALINK load. Pro/INTRALINK was reconfigured so that only the ParaCAD Lite and drawing formats were present in preparation for the data load. This step was necessary to ensure a “clean” environment into which the data would be loaded. This same step was performed prior to the first attempt to load data as well; just a necessary preparatory step for loading the data for the first time. In subsequent loads this was not done as the team continued to add data instead of starting from a clean slate.

6.4.2. MatrixOne Data Extraction: Rehearsal 2

The second MatrixOne data extraction activity culminated in a full data migration “rehearsal”, including data extraction from MatrixOne, data loading into Pro/INTRALINK, and data migration to one of the PDMLink development servers. Using revised extraction scripts, all MatrixOne data was extracted and loaded into Pro/INTRALINK. This load became the Pro/INTRALINK baseline to which additional MatrixOne extractions and WC8 production data were added.

A second validation activity (Validation 2) was completed on the new Pro/INTRALINK data load. The team took a conservative approach to complete this validation. Those configurations that were identified during the first validation as either not existing in the Pro/INTRALINK environment or not regenerating properly were targeted first. These approximately 160 configurations were divided among the validation team to address first. Each validator was also

²² *Data Migration Validation Plan Version 1*, June 2009.

tasked to look for 10 additional configurations that were part of a new, random sample of data from the second MatrixOne extraction. The team found that approximately 87% of the “problem” configurations from the first validation were still missing, or had regeneration problems during Validation 2. This led the team to a key conclusion: Data integrity issues, primarily the existence of external references, were affecting the load of approximately 30% of the MatrixOne data into Pro/INTRALINK. Despite the high error rate, the team decided to move forward with the migration of the data from Pro/INTRALINK to one of the Windchill 9.0 development systems. Additional strategies/contingencies would be needed to help quantify these results and reduce the error rate.

6.4.3. Windchill 8.0 Data Extraction

In parallel with the MatrixOne extraction and loading activities, the PTC consultants were developing data extraction scripts that could be used to migrate data from the unclassified Windchill 8.0 production system to the Windchill 9.0 system. This process would use the same interim Pro/INTRALINK infrastructure that had been established for the MatrixOne data migration.

In earlier preparation for the production data extraction from the Windchill 8.0 system, the data migration team had determined that a programmatic extraction using software scripts was preferred over a manual extraction. A programmatic approach would include the iteration history for data and it would have less impact on designer resources.

Prior to the start of this project phase, the PTC consultants had nearly completed the data extraction scripts, which they proceeded to finish as this effort moved forward. The scripts were tested on a copy of a set of Windchill 8.0 data to ensure that no production data was adversely impacted during script testing and validation. Multiple iterations of the extraction scripts were required to overcome unforeseen obstacles which included: PDMLink file renames; external references; references to library part instances that were different than what was contained in Pro/INTRALINK, and family tables. This was PTC’s first effort to use custom scripts to extract data from Windchill 8.0 and the process was more complicated than expected.

Once a “final” script was tested and verified, it was used to extract all data from the Windchill 8.0 production environment. List files were created for the configurations and scrubbed for iLink loader compatibility. The configurations were then loaded into Pro/INTRALINK using the iLink loader tool. The team observed some of the same challenges with the Windchill 8.0 data that they had identified during the MatrixOne load. Specifically, external references prevented some of the Windchill 8.0 data from loading into Pro/INTRALINK. As a result, a manual migration activity would be required to load these data as a separate post-migration activity.

The team did not validate data that was loaded from Windchill 8.0 into Pro/INTRALINK. They chose to validate the data only after it was migrated to the Windchill 9.0 system. However, error logs created during the data extraction and data load were useful in helping the team determine what percentage of configurations had been extracted from Windchill 8.0 and loaded successfully into Pro/INTRALINK, and which configurations had loading errors due to external reference issues.

6.5. Conclusion

The data migration team made considerable progress during the two months of AEE project funding support for the PTC consultants. Funding to continue the work will come from other sources. The data migration effort will be completed for the two unclassified legacy PLM systems and the lessons learned will be applied to migrating data from the classified legacy MatrixOne and Windchill 8.0 systems to a classified Windchill 9.0 production system.

The data migration effort required considerably longer time than was expected due to an unexpectedly high percentage of errors in the extracted configurations. Although several iterations of extraction scripts were developed to reduce the number of errors, a significant percentage of configurations continued to display problems. The data migration team reached a decision point and chose to proceed with the full data migration and try to resolve the errors at a later time. That resolution might require the manual loading of every configuration with an error into PDMLink, which would require a great deal of labor.

This page intentionally left blank.

7.0 Modernization of the Sandia/CA CAD Infrastructure for Design and Engineering Applications.

7.1. Summary

The Sandia/CA CAD IT Services team, Org. 8945, manages a combined forty-eight CAD infrastructure servers on the classified and unclassified networks to support the Sandia/CA site's design and engineering activities. Eight of these servers directly support PTC software applications. These aging servers, ranging from three to nine years old, are all out of service warranty and represent a significant risk to the site's NW design and engineering capabilities. AEE provided the funding to replace the four most critical of these systems and to install, test and validate PTC applications on the new systems.

7.2. Implementation

The CAD IT Services staff identified the following four servers as their highest priority for replacement:

1. A Windchill 9.0 cache server on the classified (SCN) network.
2. A Windchill 9.0 cache server on the unclassified (SRN) network.
3. An MCAD applications server on the unclassified (SRN) network, "RAN Apps", which hosts all PTC applications and other related applications for production.
4. A Virtualized FLEXlm license server which provides PTC application licenses and other related licenses to site users.

The cache servers were considered critical due to the implementation of the Sandia/NM based Windchill 9.0 production systems that will be utilized by the Sandia/CA site. These cache servers provide local buffers for files that are checked out of the remote production servers. Utilizing cache servers provides several benefits to Sandia/CA users:

- Network traffic between the sites is reduced; when a file is checked out of the production Windchill system, it is stored locally in the cache server.
- If the network or the production Windchill system goes down, Sandia/CA users can continue working in Windchill with the locally stored files.
- The cache server updates the production system on a nightly basis to update files on a daily basis.

The CAD IT Services team manages all of the previously mentioned servers and they performed this new server implementation using AEE funding. In keeping with their policy of standardization where possible, they purchased Dell R610 servers, the current Sandia/CA standard server. These servers met the requirements for the applications and were therefore the ideal choice. By standardizing, the CAD IT Services team can minimize the ongoing maintenance and support costs for these servers.

7.3. Server Virtualization

The CAD IT Services team has limited budget and staff, and one of their goals is to minimize the hardware required to support design and engineering applications at the Sandia/CA site. At a

recent Model Based Integration Tools (MBIT) conference, PTC announced its support for the virtualization of its software applications. This provided the CADs IT Services team with the opportunity to analyze the potential to implement virtualization as a means to consolidate their current forty-eight-four CAD infrastructure servers to a smaller number.

Virtualizing a server platform provides the ability to run multiple virtual servers on a single piece of hardware, rather than requiring separate hardware for each application. This is highly advantageous with Windows platforms, which cannot serve multiple licenses that utilize the same port. This problem is resolved by creating multiple separate virtual servers on the same hardware to serve each license. Virtualization cannot work for applications that have high processing requirements, but can be very effective for those applications that require intermittent or brief data throughput.

The CAD IT Services team used the AEE project as an opportunity to investigate the virtualization of the four newly purchased servers as the basis for their future modernization of all of their server hardware. Before implementing them, they wanted to understand how virtualization can be implemented, and whether it made sense to virtualize these new servers. To avoid duplication of effort, they did not want to implement them as standard servers at this time, and six months from now, reconfigure them as virtual servers.

The team concluded that most of their current servers are virtualization candidates, but the time required to perform the setup, testing and validation required for implementation was beyond the scope of this limited pilot effort, with the exception of the FLEXlm license server. Because there was a time-critical need to get the two Windchill cache servers operational, the decision was made to implement them as standard servers. Ultimately, all CAD infrastructure servers will be virtualized. The knowledge gained by researching the virtualization option in this pilot will be the foundation for the future virtualization effort. This was an important step, as funding for all of these systems is limited and server virtualization is a necessity.

7.4. Conclusion

The limited CAD Infrastructure modernization was near completion at the time of this report. Four new Windows servers were installed and were fully operational, but not all applications were fully implemented. The remaining tasks are as follows:

- “RAN Apps” MCAD Application Server: All applications are functional and the server is in production use.
- Flex LM Virtualized License Server: This virtual server is currently undergoing testing prior to being put into production use.
- Windchill 9.0 Cache Servers (unclassified and classified): These two servers require coordination with the Sandia/NM PDMLink team before they can be put into production.

It is expected that all systems will be fully implemented in the coming months. However, that timetable could be accelerated if required. If one of the older systems fails, the new replacement system can be put online quickly to replace it. Otherwise, the new systems will be completed within the availability of the CAD IT Services staff.

The virtualization effort conducted in this pilot will be the foundation for CAD IT Services team's eventual virtualization of the majority of their servers. Virtualization will allow them to cost effectively modernize their infrastructure, as they can't afford to purchase, install and maintain the 48 servers currently required to support the Sandia/CA site's design and engineering activities. This in turn will provide valuable information and experience for the virtualization of the site's enterprise application servers, which will impact approximately 500 users.

The new Windchill 9.0 cache servers present an additional capability that could be beneficial to Sandia. PTC has upgraded its Windchill cache server software to what it now calls file server software. In addition to supporting the cache server application, it provides the ability to fully replicate the host server. The Sandia/CA cache servers could in effect become complete replicas of the Sandia/NM production systems, which would provide full disaster recovery/business continuity (DR/BC) capability in the event of a catastrophic failure. The CAD IT Services team will work with the Windchill production team to determine if this is a capability to implement.

This page intentionally left blank.

8.0 References

Dencklau, Jeremy and Sego, Abraham, *Migration and Matrix Divestment Options – Decision Paper #8*, March 2009, Sandia National Laboratories, Albuquerque, NM.

Goodwin, Ralph, *Windchill PDMLink 9.0 M070 Installation and Configuration for Sandia National Laboratories*, September 2009, Parametric Technology Corporation, Needham, MA.

Hunt, D. Jay, *and Product Structure Pilot: Close Out Documentation*, November 2009, Parametric Technology Corporation, Needham, MA.

Hutcheson, Will, *AEE Product Structure Capability Aid*, October 2009, Parametric Technology Corporation, Needham, MA.

Kiba, Grant and Pomplun, Alan, *AEE Product Structure Pilot Use Case Demonstration* (unpublished internal presentation), December 2009, Sandia National Laboratories, Livermore, CA.

Kiba, Grant, *Configuration of AEE Servers to SNL/NM Production Settings* (unpublished project documentation), September 2009, Sandia National Laboratories, Livermore, CA.

Kritter, Jason, *SNL PDMLink V9.0_M070 SPDM V1.0 Release*, January 2010, Sandia National Laboratories, Albuquerque, NM.

Meeks, Timothy, et al., *PRIDE Product Structure Specifications: Design Definition*, Official Use Only (OUO), June 2009, Sandia National Laboratories, Albuquerque, NM.

Meeks, Timothy, *PRIDE Product Structure Specifications Development General Briefing*, July 2009, Sandia National Laboratories, Albuquerque, NM.

Monk, Lisa, *Sandia Future Process Description*, February, 2010, Parametric Technology Corporation, Needham, MA.

Monk, Lisa, *Where Are We At With Product Structure?* December, 2009, Parametric Technology Corporation, Needham, MA.

Monk, Lisa and Hunt, D. Jay, *Detailed Use Case Document – AEE Project*, February 2010, Parametric Technology Corporation, Needham, MA.

NSE Model Based Integrated Tools (MBIT) Committee and PRIDE MCAD Project, *NSE Product Realization Standard – PDMLink Standard Installation and Configurations*, PRS12005 Issue B (Draft), September 2009.

NWC Model Based Integrated Tools (MBIT) Committee and PRIDE MCAD Project, *NWC Product Realization Standard – PDMLink Standard Installation and Configurations*, PRS12005 Issue A, November 2007.

Otto, Catherine and Sego, Abraham, *Data Migration Validation Plan Version 1*, March 2010, Sandia National Laboratories, Livermore, CA.

Pomplun, Alan R. et al., *Advanced Engineering Environment Collaboration Project*, SAND2008-7885, January 2009, Sandia National Laboratories, Livermore, CA.

Pomplun, Alan R., *Advanced Engineering Environment Project Windchill 9.1 Implementation Pilot Summary Report*, SAND2009-4650P, July 2009, Sandia National Laboratories, Livermore, CA.

Sego, Abraham, *PDMLink Data Validation (WFS1055624) – Decision Paper 22*, March 2010, Sandia National Laboratories, Albuquerque, NM.

Sego, Abraham, *Migration Options and Configurations – Decision Paper #12*, June 2009, Sandia National Laboratories, Albuquerque, NM.

Sego, Abraham, *SNL PDMLink 9.0 Production Test Plan Version 1*, June 2009, Sandia National Laboratories, Albuquerque, NM.

Wood, Pete, *NWC PDMLink Installation Guide Version 9.0 – Updated for the Sandia-CA Installation*, November 2008, Parametric Technology Corporation, Needham, MA.

Zyblut, J. P. and Formales, Marc-Wayne, *Data Migration Plan – Sandia National Labs* (draft), May 2009, Parametric Technology Corporation, Needham, MA.

Appendix A: AEE Server Specifications

A.1 Three-tier Development Servers

This hardware and software details for the AEE development servers are provided here:

- Oracle and Arbortext Servers
 - Hardware
 - Dell PowerEdge R610
 - 1-RU Chassis for up to six 2.5-inch hard drives.
 - Single processor: Intel® Xeon® X5550, 2.66Ghz, 8M cache, 6.40 GT/s QPI, turbo, (Quad Core with HyperThreading).
 - Memory: 6GB memory (3x2GB), 1333MHz dual ranked UDIMMs for 1 processor, optimized.
 - Hard drive configuration: RAID 1/RAID 5 for PERC 6/i controller.
 - Hard Drives:
 - (2) 73GB 10K RPM serial-attach SCSI 2.5" hot plug hard drive; the OS is installed here, configured as a RAID 1 (mirrored).
 - (4) 146GB 10K RPM serial-attach SCSI 2.5" hot plug hard drive (RAID 5 = 3x146GB striped with the 4th acting as a parity drive; formatted capacity is 408GB).
 - Software
 - Oracle Server
 - Windows Server 2003 Enterprise, 64 bit.
 - Oracle 10.G.
 - Microsoft Office 2007
 - Arbortext Server: This server was not configured.
- Windchill Server
 - Hardware
 - Dell PowerEdge R710
 - 2-RU Chassis for up to six 3.5-inch hard drives.
 - Processor: Intel® Xeon® X5550, 2.66Ghz, 8M cache, 6.40 GT/s QPI, turbo, (Quad Core with HyperThreading).
 - Additional processor: Intel® Xeon® X5550, 2.66Ghz, 8M cache, 6.40 GT/s QPI, turbo, (Quad Core with HyperThreading).
 - Memory: 36GB memory (18x2GB), 800MHz dual ranked RDIMMs for 2 processors, optimized.
 - Hard drive configuration: RAID 1/RAID 5 for PERC 6/i controller.
 - Hard Drives:
 - (2) 146GB 15K RPM serial-attach SCSI 3.5" hot plug hard drive; the OS is installed here, configured as a RAID 1 (mirrored).
 - (4) 450GB 15K RPM serial-attach SCSI 3.5" hot plug hard drive (RAID 5 = 3x450GB striped with the 4th acting as a parity drive; formatted capacity is 1.22TB).
 - Software
 - Windows Server 2003 Enterprise, 64 bit.

- Windchill 9.0 M070.
- Pro/ENGINEER Wildfire 4 M092.
- Microsoft Office 2007
- Oracle Client 10.G.
- Adobe Acrobat Professional 9.0.
- ProductView Standard 9.0.

A.3 Monolithic Training Server

The hardware and software details for the training server are provided here:

- Hardware:
 - Dell PowerEdge 2950.
 - Two dual core Intel Xeon processors, 2.66 GHz, 800 MHz FSB.
 - 4GB DDR2 400MHz.
 - RAID 5 controller.
 - Hard drive configuration: Four 73GB 15K rpm SAS (Serial Attached SCSI) drives.
- Software:
 - Pro/ENGINEER Wildfire 4 M092.
 - Windchill 9.0 M070.
 - Windows Server 2003 OS (standard version).
 - Acrobat Distiller add-in.
 - ProductView Standard 9.0.
 - Oracle 10.g.

Appendix B: Sandia Design Definition Current Workflow Process Diagrams

The high level process diagrams that Lisa Monk (PTC) developed for Sandia's current design definition workflow are included here.

Figure 23. Current design definition process overview.

Figure 24. Current process Step 1: Design effort.

Figure 25. Current process Step 2: Final design.

Figure 26. Current process Step 2: Mechanical design.

Figure 27. Current process Step 2: Electrical design.

Step 3 (Activity): Supporting Documentation –
Can run parallel with Design

Last Updated Date: November 22, 2009

Figure 28. Current process Step 3: Supporting documentation.

Figure 29. Current process Step 4: Compare Information in the Sandia EBOM system.

Figure 30. Current process Step 4: Material list.

Figure 31. Current process step 5: Store/release design.

This page left intentionally blank.

Appendix C: Future Design Definition Workflow Process in Windchill PDMLink 9.0.

The high level process diagrams that Lisa Monk (PTC) developed for a proposed future design definition workflow within the Windchill PDMLink 9.0 application are included here.

Figure 32. Future process overview.

Figure 33. Future process Step 1: Final design (same as current process).

Figure 34. Future process Step 2: Mechanical design overview.

Figure 35. Future process Step 2: Mechanical design for legacy designs with an existing CAD model.

Figure 36. Future process Step 2: Mechanical design for legacy designs with no existing CAD model.

Figure 37. Future process Step 3: Part structure.

Step 4 (Activity): WTParts

Last Updated Date: November 30, 2009

Designer

Figure 38. Future process Step 4: WTParts.

Figure 39. Future process Step 5: Supporting documentation (same as current process).

Figure 40. Future process Step 5: Material list (same as current process).

Figure 41. Future process Step 6: Finalize design.

Distribution

Honeywell FM&T
Attn: K. Baughn (electronic copy)
kbaughn@kcp.com

Honeywell FM&T
Attn: J. Galichia (electronic copy)
jgalichia@kcp.com

Honeywell FM&T
Attn: B. Grantham (electronic copy)
bgrantham@kcp.com

Honeywell FM&T
Attn: D. Lewis (electronic copy)
dglewis@kcp.com
Parametric Technology Corporation
Attn: R. Abernathy (electronic copy)
rabernathy@ptc.com

Parametric Technology Corporation
Attn: C. Cash (electronic copy)
ccash@ptc.com

Parametric Technology Corporation
Attn: R. Goodwin (electronic copy)
rgoodwin@ptc.com

Parametric Technology Corporation
Attn: F. Fryer (electronic copy)
ffryer@ptc.commailto:

Parametric Technology Corporation
Attn: D. J. Hunt (electronic copy)
jhunt@ptc.com

Parametric Technology Corporation
Attn: W. Hutcheson (electronic copy)
whutcheson@ptc.com

Parametric Technology Corporation
Attn: B. Johnson (electronic copy)
brjohnson@ptc.com

Parametric Technology Corporation
Attn: L. Monk (electronic copy)
lbarnhart@ptc.com

Parametric Technology Corporation
Attn: C. Stelse (electronic copy)
cstelse@ptc.com

Los Alamos National Laboratory
Attn: E. N. Kilmer (electronic copy)
maggot@lanl.gov

Los Alamos National Laboratory
Attn: C. J. Scully (electronic copy)
chscully@lanl.gov

Lawrence Livermore National Laboratory
Attn: B. T. Grover (electronic copy)
grover5@llnl.gov

Lawrence Livermore National Laboratory
Attn: R. E. Santos (electronic copy)
Rey-santos@llnl.gov

MS0340	M. E. Gonzales (electronic copy)
MS0374	H. P. Walther (electronic copy)
MS0464	R. J. Chavez-Hatton (electronic copy)
MS0464	J. R. Martinez (electronic copy)
MS0576	A. A. E. Barnes (electronic copy)
MS0622	A. B. Clark (electronic copy)
MS0622	S. J. Pinnick (electronic copy)
MS0624	T. Cox (electronic copy)
MS0624	B. M. Evanetich (electronic copy)
MS0624	J. M. Kritter (electronic copy)
MS0624	T. M. Meeks (electronic copy)
MS0624	R. J. Sikorski (electronic copy)
MS0624	J. R. K. Smith (electronic copy)
MS0625	R. D. Graham (electronic copy)
MS0626	T. D. Hernandez (electronic copy)
MS0628	L. S., Ashford (electronic copy)
MS0628	M. F. Current (electronic copy)
MS0648	T. A. Burrows (electronic copy)
MS0648	A. L. Sego (electronic copy)
MS0648	P. L. Turner (electronic copy)

	MS0640	R. N. Harris (electronic copy)
	MS0806	L. G. Martinez (electronic copy)
	MS1138	A. E. Starcher (electronic copy)
	MS9039	E. M. Allen (electronic copy)
	MS9039	E. G. Dutra (electronic copy)
	MS9039	D. J. Herron (electronic copy)
5	MS9039	G. W. Kiba, 8243-1
15	MS9039	J. A. Lamph, 8243
	MS9039	A. A. Machado (electronic copy)
	MS9039	C. Otto (electronic copy)
	MS9039	M. L. Tootle (electronic copy)
	MS9039	M. Wong (electronic copy)
	MS9040	T. D. Bersie (electronic copy)
	MS9040	L. A. Brown (electronic copy)
	MS9040	R. J. Dankiewicz (electronic copy)
	MS9040	L. R. Navarro (electronic copy)
	MS9151	J. E. Costa (electronic copy)
	MS9151	L. M. Napolitano (electronic copy)
1	MS9151	C. T. Oien, 8940
10	MS9152	A. R. Pomplun, 8953
	MS9153	M. F. Hardwick (electronic copy)
	MS9154	M. K. Lau (electronic copy)
	MS9154	R. A. Van Cleave (electronic copy)
1	MS9155	R. L. Clay, 8953
1	MS9957	S. C. Carpenter, 8945
	0899	Technical Library, 9536 (electronic copy)