

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



DMDII

+ a UI LABS Collaboration

DIGITIZING AMERICAN
MANUFACTURING

DMDII FINAL PROJECT REPORT

Democratizing the Model-based Domain from Design to Verification: Automatic Generation of Optimized CMM Programs on the DMC

Project Team Lead	Metrosage LLC
Project Designation	15-16-02
UI LABS Contract Number	0220160024
Project Participants	Capvidia NA LLC Naval Air Warfare Center Aircraft Division Lakehurst University of North Carolina Charlotte
DMDII Funding Value	\$ 161,499.00
Project Team Cost Share	\$ 288,889.00
Award Date	12 October 2016
Completion Date	31 December 2017

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

DISTRIBUTION STATEMENT A. Approved for public release; distribution unlimited.

TABLE OF CONTENTS

Page(s)	Section
2-3	I. Executive Summary (1-2 pages)
3-5	II. Project Overview
5	III. KPI's & Metrics
5-8	IV. Technology Outcomes
8	V. Accessing the Technology
8-9	VI. Industry Impact & Potential
9-11	VII. Tech Transition Plan & Commercialization
11	VIII. Workforce Development
11	IX. Conclusions/Recommendations
11-12	X. Lessons Learned
12	XI. Definitions
12	XII. Appendices

I. EXECUTIVE SUMMARY

This proposal addresses a capability gap that hampers effective use of coordinate measuring machines (CMMs) in the assessment of geometric dimensioning and tolerancing (GD&T) callouts. It presents an opportunity to significantly enhance the value of information acquired, while dramatically reducing costs in terms of time and labor. Currently programming of a CMM for specific part measurement can be a tedious process, most commonly requiring manual input of GD&T specifications, manual stipulation of surface sampling strategies and of many other parameters. It is labor-intensive and prone to errors. Moreover, there is generally no *a priori* verification of the adequacy of selected sampling strategies or even of the CMM and sensor system targeted for the measurement tasks.

As GD&T requirements are increasingly incorporated into computer aided design (CAD) applications in the form of *semantic* product and manufacturing information (PMI), direct and automated CMM program generation becomes possible, significantly reducing errors and saving labor. Moreover, in recent years the CMM community has become increasingly aware of the implications of measurement uncertainty and its role in pass/fail assessments risks. Assessment of task-specific measurement uncertainties in coordinate metrology via simulation is now a reality, and provides a basis for *optimizing* measurement strategies to reduce of measurement uncertainties, and for the reporting of the resultant task-specific uncertainty value for each GD&T assessment. **The ability to generate *optimized* CMM programs directly from CAD geometries augmented by semantic PMI represents a dramatically innovation in dimensional quality control. Providing measurement uncertainty data is likewise a vital advance over current practice. Implementation of such capability on the Digital Manufacturing Commons (DMC) will make it accessible to a broad range of users, especially those from smaller manufacturing organizations.**

Developing this capability involved the use of several constituent elements of the open, royalty-free Quality Information Framework Standard (ANSI QIF 2.1) as data linkages among three functional components, each a commercial product offered by a different vendor. The first component, Capvidia's **MBDVidia**, translates native (NX, CATIA, SolidWorks, Creo, etc.) or STEP AP 242 file information into QIF MBD files containing part geometry, inspection features, and pertinent GD&T. The second component, **Pundit CMM**, uses this data, and a user-defined specification (QIF Resources format) of available CMMs and sensors, in Monte Carlo simulations for evaluation of task-specific CMM measurement uncertainties. This process is iterated, exploring different sampling strategies to achieve acceptable measurement uncertainties for all GD&T characteristics and produce a measurement plan in QIF Plans format. The third component, Origin's **CheckMate**, uses the QIF MBD, QIF Resources, and QIF Plans files to generate a detailed CMM program in the widely used Dimensional Measuring Interface Standard (DMIS, ISO 22093:2011) format. The three components, their data linkages, and control software have been bundled in a "black-box" so that start-to-end functionality can be offered with minimal required user interaction. This framework provides a basis for future plug-and-play of newly developed competing components or new versions of these applications, as interchangeable parts of a general solution.

This system of 3 software applications was successfully deployed on the DMC in a limited fashion. It faced two primary roadblocks: (1) the DMC platform was never refined to a fully completed state, which severely limited the User Interface for the tools, and (2) 3rd party PMI translators used by the CAD translation software are unreliable for real industry data, and often, a handful of GD&T requirements were discarded by the workflow. The recommendations for how to address these two gaps are twofold: (1) use a software application platform which allows for the display of 3D data in a graphical UI, and (2)

to use native CAD plugins to translate native CAD & PMI data (these plugins are available on the market and robust).

II. PROJECT REVIEW

Project Scope, Objectives, and Planned Benefits

The objective of this project is to automatically create optimized CMM programs given the input of a CAD model with PMI, and inventory of available measurement equipment. This system was to be deployed on the Digital Manufacturing Commons (DMC), a SaaS platform hosted by DMDII.

Users would be able to connect to the app, upload the model of the part to be measured, upload the available measurement devices. The app would then automatically generate a CMM program – targeted for the user’s CMM software system. This CMM program would often require a minimal amount of manual editing by the user in order to complete.

The general idea is to take an industrial process which is currently 100% manual, and to turn it into a process which is 80% automatic, and 20% manual. Furthermore, as time goes on and the technology is refined, this approximately 20% manual effort would continue to decrease rapidly, until we reach the point where 100% automation is possible.

Technical Approach

The DMC app chained together 3 commercially available software packages: MBDVidia, Pundit CMM, and Origin International’s CheckMate software. These software packages are all 3D based software applications for Microsoft Windows. However, in this setup, the complexity of these tools would be hidden from the end user, and handled purely internally by the DMC app. The app carried out 3 primary tasks: (1) translation of the CAD and PMI into the ANSI Quality Information Framework (QIF) format, (2) high-level optimization of the CMM program, and (3) creation of the CMM program.

ANSI Quality Information Framework

The interoperability of these 3 software systems is enabled by use of the QIF standard data format. This allows for the app chain to communicate CAD, PMI, and measurement plan information from one process to another.

Translation of CAD – MBDVidia

The CAD model with PMI is simultaneously the most crucial input, and the most difficult input to harvest. The CAD model with PMI contains the part geometry and the design requirements which are to be measured. For instance, the PMI contains the feature size tolerances, position tolerances, and other tolerances which are to be measured on the CMM. Capturing this data is fundamental to this metrology process.

It is a difficult accomplish task for two primary reasons: difficulty in translating the PMI, and poor quality of PMI. The CAD models inputted by the user will typically be in non-standard, proprietary formats; e.g., NX/Unigraphics, ProENGINEER/Creo, CATIA, SolidWorks, etc. The structure of these file formats is not publicly documented (as is the case with industry standards), so reading data directly from these data sources is often unreliable. This is particularly the case with PMI data.

The other challenge for a translator is how to deal with poor quality PMI. Often, PMI is added to a CAD model in a “non-semantic”, or non-machine-readable way. PMI can be added to a CAD model in a way in which it can be unambiguously interpreted by a software system – for example, creating a cylindricity tolerance using the proper cylindricity tolerancing tools in the CAD system, and correctly associating the tolerance with the *surface* that it applies to. The alternative is quite common, and causes the PMI to be meaningless to software. For example, the cylindricity tolerance might be created as a general note in the CAD system. Or it might be associated with a model edge (as opposed to a surface), or it might not be associated with any geometry at all! A good analogy for thinking about machine-readable PMI vs. non-machine-readable PMI is: imagine someone emails you a document asking for some edits. Which scenario would be easier: (1) sending a Microsoft Word document, or (2) sending a cell phone picture of the document. Both of them are human readable, but clearly there is an important difference between the two.

MBDVidia is one of the premier MBD translation tools on the market. It uses the best-in-class Spatial 3DInterOp toolkit to open proprietary CAD models, and can heal non-semantic PMI using its MBD Ready Check technology. With these two subsystems, MBDVidia is the best choice to overcome the obstacles of CAD and PMI data translation.

Pundit CMM

Pundit CMM uses Monte Carlo simulation to calculate task-specific measurement uncertainties for a CMM measurement. By “task-specific”, we mean that each tolerance on the part is assessed for measurement uncertainty, given the specific conditions under which it is measured: CMM type, probe type, thermal environment, etc.

Pundit is used in this app chain to ensure that not only is a CMM program generated; but that the CMM program generated will satisfy the requirements put forth by the GD&T, while at the same time not spending excessive time or resources on the measurement. The measurement optimization carried out by Pundit will choose the minimum number of points possible to achieve the required precision, thereby reducing the time spent to measure the part. Additionally, it will choose the lowest cost CMM system to operate, lowering overall cost of measurement.

The output from Pundit is a “high level” measurement plan; that is to say, the CMM measurement system to be used, and the sampling point locations to measure on the part.

CheckMate

Origin International’s CheckMate application carries out offline CMM programming. There are two special characteristics of CheckMate which make it particularly suited for this project: its automation capabilities, and its ability to generate programs for a variety of CMM systems.

Given a “high level” measurement plan, as previously described, CheckMate will automatically generate a full CMM program. Aside from sampling point locations, several tasks must be carried out to drive a CMM: part placement on the CMM, intra-feature probe path, inter-feature probe path, probe angles, collision avoidance, collision detection, etc. These tasks are automated in this DMC app.

Once a CMM program is created, it must be outputted to a file format which can be interpreted by CMM software in order to be executed. Various CMM software packages support a number of different data formats, and CheckMate is able to output CMM programs which cover almost all of these formats.

Once the CMM program file is generated, it is made available for download by the end user. The end user is then instructed to load the CMM program in their software and check it for completeness. In many cases, and nominal amount of manual “finishing” of the CMM program will be required of the user.

III. KPI'S & METRICS

Metric	Baseline	Goal	Results	Validation Method
Time spent to create CMM program	Typically, 6-12 hours ++	Automate 80% of this	First ~80% of program within minutes; final 20% manual	Run auto-generated CMM programs to show correctness
CMM programs with known measurement uncertainties	Measurement uncertainties unknown	Calculate task-specific measurement uncertainties with minimal overhead	Measurement uncertainties calculated by Pundit with minimal additional effort (just minutes)	Pundit previously validated in many commercial and public projects

IV. TECHNOLOGY OUTCOMES

System Notes

Software packages used in App Chain:

- MBDVidia (Capvidia): translate and manage CAD and PMI data
- Pundit CMM (Metrosage): calculate task specific measurement uncertainties for CMM measurements
- CheckMate (Origin): offline CMM programming and point cloud analysis software

The purpose of this App Chain is to automatically create optimized CMM programs on-demand for a given part using the specified available CMMs. This will be a “black-box” system, taking only simple inputs from the DMC interface at the front end, and outputting a DMIS file on the back end.

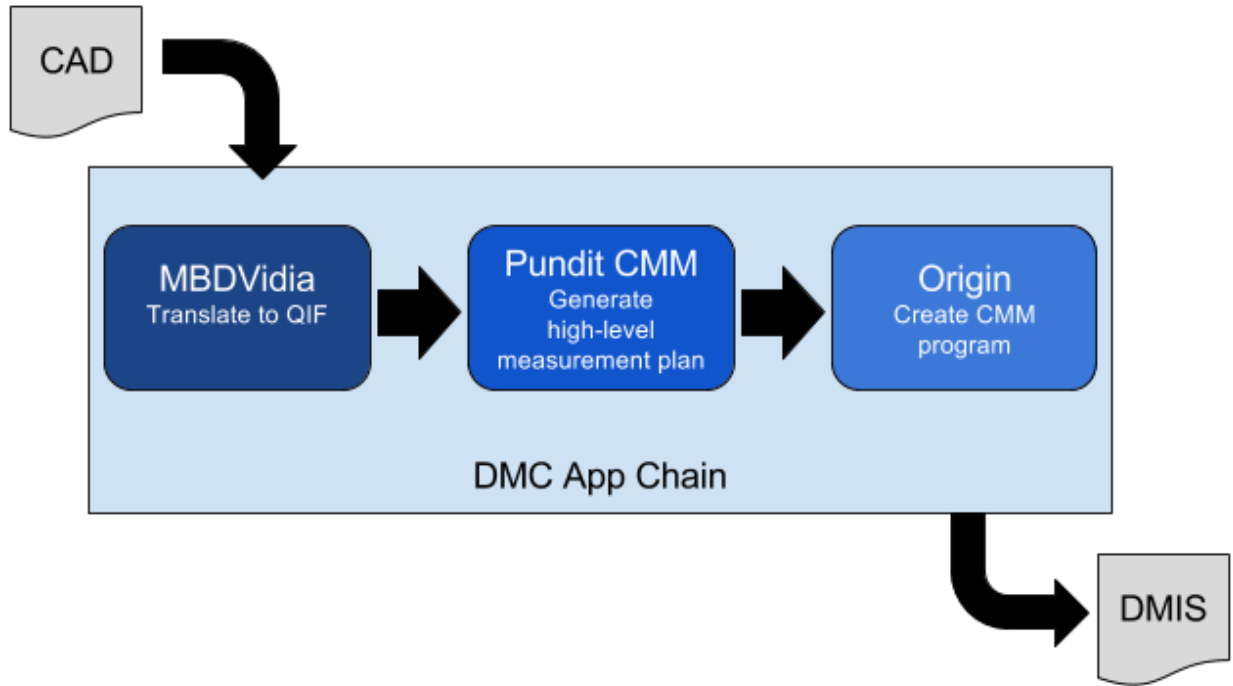


Figure 1: System overview

The DMIS program will be specifically tailored to work with the user's CMM software system.

The CAD model input should contain PMI data with all GD&T requirement, and should be in one of the following formats:

- QIF
- STEP AP242
- Creo
- NX
- SolidWorks
- CATIA

The system architecture is generally made up of the DMC front end, a back-end Microsoft Windows server with a DOME installation, and an Amazon S3 data storage container. The DMC handles all user interaction. When a CAD model is upload to the DMC, it is passed to the Amazon S3 storage. The DMC then signals to the DOME system to begin the app chain execution. The app chain is managed by a python script which executes one application after another, until the final CMM part program is generated. The python script grabs the user inputs from the Amazon S3 container, and when the app chain has completed its execution, it uploads the resulting CMM program to the Amazon S3 container, and notifies the DMC front end that execution has completed. Then, the DMC front end provides the resulting CMM program to the end user for download.

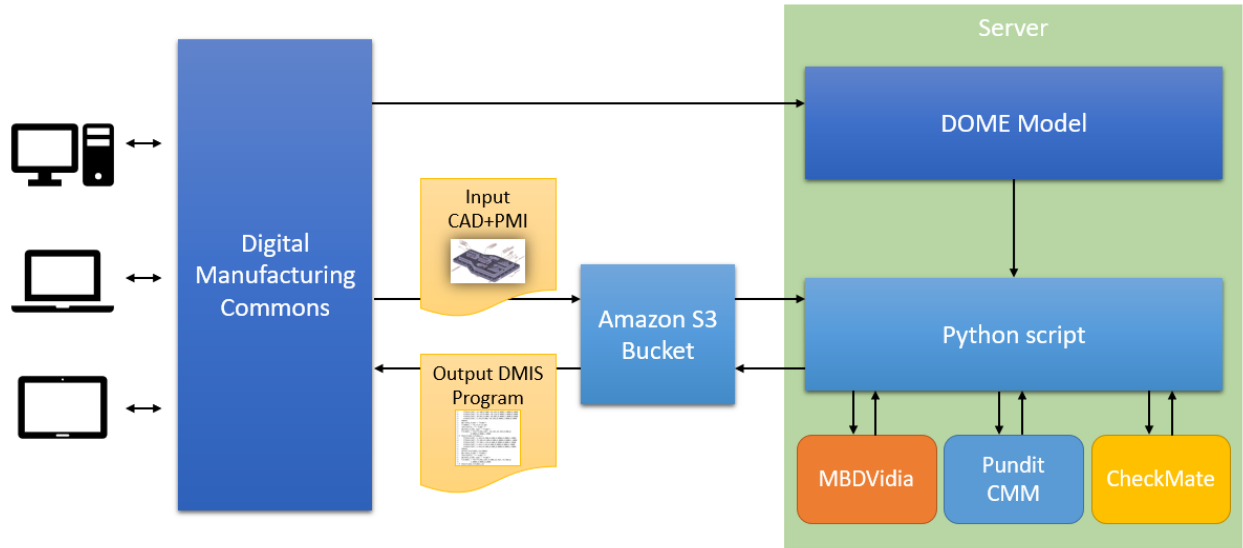


Figure 2: System architecture

Features and Modes of Operation

The initial translation of the CAD data can carry out some basic healing of the PMI, to help make it machine-readable (if it isn't initially).

The CMM program optimization takes the follow error influence quantities into account: CMM error, probing error, environmental conditions, CMM software algorithms, manufacturing feature form errors, and sampling strategies.

The system current provides CMM programs for traditional, 3 axis CMM systems, run by most major software packages on the market. These include:

- DMIS (ISO and various “flavors” of DMIS implemented by common CMM platforms)
- Proprietary programs for the following CMM software programs:
 - AVAIL
 - MMIV
 - MeasureMax
 - DirectInspect
- Partial support for the following proprietary CMM software programs:
 - CMES
 - UMESS-300
 - Tarus

There is a single mode of operation for this tool; the inputs and outputs are fixed.

Use Cases

The single use case supported is diagrammed in Figure 1. The user is required to input CAD data with PMI, and the output will be a CMM program targeted for the user's CMM.

In some cases, the CMM program will be 100% complete and ready for execution. In other cases, there will need to be some manual finishing work to be carried out by the end user.

In either case, the user should perform a verification of the CMM program. In particular, this include: (1) ensuring that all GD&T tolerances have been encoded in the CMM program (issues with CAD translation can, in some cases, cause tolerances to be lost), and (2) running the CMM program in simulation mode for collision detection, to avoid damage to the user's CMM or probing system.

V. ACCESSING THE TECHNOLOGY

This software app chain uses 3 commercially available proprietary applications: (1) MBDVidia, (2) Pundit CMM, and (3) CheckMate. While the intellectual property contained within these applications is proprietary to their respective owners, the tools are commercially available.

The DOME wrapper and python code to string the applications together into a chain are open and freely available.

The DMC is required to run this tool, and the DMC has since been decommissioned.

If the public would like to see this app chain, it would be possible for Metrosage, Capvidia, or Origin International to arrange such a demonstration.

VI. INDUSTRY IMPACT & POTENTIAL

The effect of automatically generated CMM programs will have significant impact on the manufacturing industry. Currently, an enormous amount of effort is expended carrying out tasks which are an inefficient use of a skilled engineer's time. This new paradigm will allow for these prized engineers adding value to the operation in ways that only human reasoning can. It will also significantly increase throughput of quality departments, as a typical bottleneck is found at the CMM programming stage. We are seeing this trend of automated CMM programs continue with tools like Zeiss Calypso and Mitutoyo MiCAT planner, but never at this ambitious level of automation.

Additionally, industry is beginning to awaken to the importance of a concrete, quantified awareness of measurement uncertainty requirements. As tolerances continue to get tighter and the competitiveness driving decreasing costs continues apace, effective adequacy of a quality department can be improved by *a priori* consideration of measurement uncertainty, thereby avoiding costs associated with real-world measurement failures.

Finally, providing this technology on a SaaS platform is a forward-thinking step which anticipates the direction manufacturing industry is headed. With the popularization of digital manufacturing and additive manufacturing, there are already getting into the concept of *Distributed Manufacturing*. The drive towards this trend is lead by companies like SAP with [their current Distributed Manufacturing efforts](#). This software tool provides a similar service, albeit in digital form. Rather than purchasing a software tool, transactions can be made on an as-needed basis by the end user.

The next steps for this project are twofold: (1) continue to mature the automation technology until it is ready for production level use (higher-TRL), and (2) to continue exploring options for platforms on which to host such a service.

VII. TECH TRANSITION PLAN & COMMERCIALIZATION

Immediate plans for this app chain is to mature the core technology of the tools by finding pilot project within industry to provide real-world usage scenarios. With pilot projects on real data, and with real industry users, we can expect to gain the experience to overcome some of the hurdles encountered during this project, and during typical software deployment. These pilots would be run using the full 3D applications involved in the app chain – MBDVidia, Pundit CMM, and CheckMate. This will allow for the user to achieve a reasonable and value-added outcome while only partial automation is possible.

There are various gaps in the technology which will continue to progress in pilot projects. Here are some of these issues.

Poorly Defined PMI

In typical practice, PMI is very often not correctly defined in CAD systems. What this means is that the relationship between the annotation and the geometry which it corresponds to is not unambiguously communicated in a way that software can understand the complete design intent. Instead, if the PMI looks human-readable, then the task of adding PMI is considered complete. Industry is only now awakening to the important distinction between human-readable PMI and machine-readable PMI.

What are some examples of poorly defined PMI? Here are a few:

- Annotations referencing trimming edges on the CAD model (as opposed to the model surfaces which are intended to be controlled)
- Use of non-semantic mechanisms in CAD to create PMI — e.g., use of a text note to add a tolerance, rather than using the CAD system’s GD&T functionality to create the annotation
- GD&T or Datums referencing auxiliary geometry in the model, rather than the geometry of the physical model itself. (This is very common with Datums in the PTC Creo Parametric CAD system, for example)
- Creation of a tolerance without any reference to model geometry. For example, often general notes will have GD&T information which has no reference to the model
- Patterns of features (e.g., “5 X ...”) which in fact only reference 1 of the features of the pattern.

This issue is being overcome by Capvidia’s MBD Ready Check technology, which is capable of automatically healing the PMI to make sure that it is machine-readable. MBD Ready Check will continue to improve over time, as more real-world data is encountered.

Loading PMI into 3rd Party Software

In addition to poorly formatted PMI, as described above, there is often difficulty loading PMI into 3rd party software. This is because the data is stored in a proprietary format (NX, Creo, CATIA, SolidWorks, etc.), thus making access to the data unreliable. 3rd party translator libraries are used by software systems consuming this data, but the results are unpredictable.

This problem can be solved by using direct API access to the CAD system itself to access the data. These plugins are commercially available from Capvidia, and this technology will be used with future work.

Uncertainty Simulation

Two additional error models will be added to Pundit CMM in order to further support measurement uncertainty simulation on common metrological equipment:

- Contact scanning
- Non-contact sensing
- Articulated arm CMMs

These measurement techniques are becoming more common in industry, and must be supported under any viable future system.

CMM Program Generation

The following gaps are currently being addressed by the CheckMate planning software in order to support 100% automation of CMM program generation:

- Initial, manual alignment of the part is necessary: CheckMate creates coordinate systems for all 6-degree-of-freedom DRFs like ABC. Some manual alignment is required so measurement can begin, this is often the ABC alignment but with a reduced point set. It could be an alignment to convenient, non-datum features, or it could be a single point measurement if the part is aligned to CMM axes. This will exist in the DMIS program in the form of instructions to the user on where to manually sample a few points with the CMM.
- Automatic probe selection and collision avoidance for 6-side part measurement, 5-sided measurement currently supported, 6th side (bottom side) measurements are not supported
- Target point location adjustment for accessibility not yet in place
- User selection of probe geometry required (initially from a set of predefined geometries)
- User selection of probe angles required (initially from a set of predefined angle sets)
- Part placement on CMM required (work view), currently part is placed so part axes match CMM axes
- Target point selection on free-form surfaces for non-prismatic measurement. (This is supported only by the full CAD-based CheckMate application; CAD is required to define and query freeform surfaces).

Barriers to Adoption

As is the case with any new software rollout, change is difficult. This is particularly true when rolling out a non-commoditized paradigm such as this. However, in this case, the industry reaction to this technology has been tremendously positive: the benefits are extensive and obviously apparent.

To ease the deployment of pilots of this technology, extensive training and interaction with the end users is the norm. Issues and gaps are identified and addressed. Many can be addressed with further training. Others can be addressed with a rapid development cycle and issuing quick software patches to the customer to address the gap. A development response time of days is required to maintain momentum of these pilot projects. This has led to success with deployment of pilot projects.

VIII. WORKFORCE DEVELOPMENT

A change in mentality will be required on the part of engineers in the quality department, in that working with MBD data will more and more become the norm. The typical industry practice is to transmit GD&T data on a 2D PDF document, which is then manually transcribed by the CMM engineer. MBD will eliminate this shortcoming, but will require some education and experience from the CMM engineer to become accustomed to this technology. This is being done by providing extensive training to end users, and continually providing close support on all issues.

IX. CONCLUSIONS/RECOMMENDATIONS

The deployment of these software packages on the DMC faces the challenge of carrying out this complex series of automatic steps without any user interaction. Each of the tools used in this App Chain are robust, 3D desktop applications with complex user interfaces. Without the user interface, full automation of the process from end-to-end is required.

While this is accomplished with a remarkable level of success, there are some limitations that must be pointed out. Each of the individual tools in the App Chain are robust tools, commonly used in industry. The App Chain itself, however, is still at a relatively low TRL level for industrial use — somewhere near TRL 6.

Furthermore, with the decommissioning of the DMC, an alternative suitable platform will need to be found. Ideally, this platform would include support for 3D interactive visualization using HTML5 or other suitable technology.

X. LESSONS LEARNED

The issues with PMI semantics of real-world CAD data were larger than originally anticipated. For the data to be more reliable, an alternative approach would need to be taken. This alternative approach would be to use plugins inside of each of the supported native CAD systems. These plugins would be able to access proprietary CAD data by using the CAD systems themselves. This would provide access to high-fidelity translations of the CAD and PMI data.

Additionally, the barrier of generation of 100% of a CMM program automatically, without any 3D visualization or any user interaction proved to be too difficult for production level use. A high level of automation can currently be achieved with minimal assistance from the user, but an interactive 3D user interface is required to gather this assistance.

The project delivered on a medium-range TRL deployment of automated and optimized CMM program generation.

XI. DEFINITIONS

What follows are a set of definitions, terms, and acronyms used in this document. These definitions were gathered from various source including the internet, reference papers, standards organizations, and the authors of these document.

CAD – Computer Aided Design

PMI – Product and Manufacturing Information; the annotations and metadata found in a CAD model which transmit design requirement critical to manufacturing and quality inspection.

GD&T – Geometric Dimensioning and Tolerancing; design tolerances assigned to a part to ensure it conforms to fit and function.

MBD – Model Based Definition; the use of CAD models with PMI to intelligently store a fuller set of design requirements for a product.

CMM – Coordinate Measuring Machine

QIF – ANSI standard Quality Information Framework

DMIS – Dimensional Measuring Interface Standard; an ISO standard used to specify CMM part programs

XII. APPENDICES

A deliverable package is provided in the file “deliverable-package.zip”. This file contains the software code and instructions for the DOME system. The system required the now defunct DMC to function. Access to the git source code repository can also be provided upon request. A demo of the tool was carried out using the model “input.prt” (a PTC Creo model with PMI). A video of a demo being executed in included in the deliverable package: “DMDII_15-16-02_demo.mp4”. This demonstration was carried out when the project was partially completed, but prior to the decommissioning of the DMC.